

# Network Security Tutorial

Contact: [training@apnic.net](mailto:training@apnic.net)

# Overview

- Network Security Fundamentals
- Security on Different Layers and Attack Mitigation
- Cryptography and PKI
- Resource Registration (Whois Database)
- Virtual Private Networks and IPsec

# Network Security Fundamentals

Network Security Workshop

# APNIC

-

# Why Security?

- The Internet was initially designed for connectivity
  - Trust assumed
  - We do more with the Internet nowadays
  - Security protocols are added on top of the TCP/IP
- Fundamental aspects of information must be protected
  - Confidential data
  - Employee information
  - Business models
  - Protect identity and resources
- We can't keep ourselves isolated from the Internet
  - Most business communications are done online
  - We provide online services
  - We get services from third-party organizations online

# Internet Evolution



LAN connectivity



Application-specific  
More online content

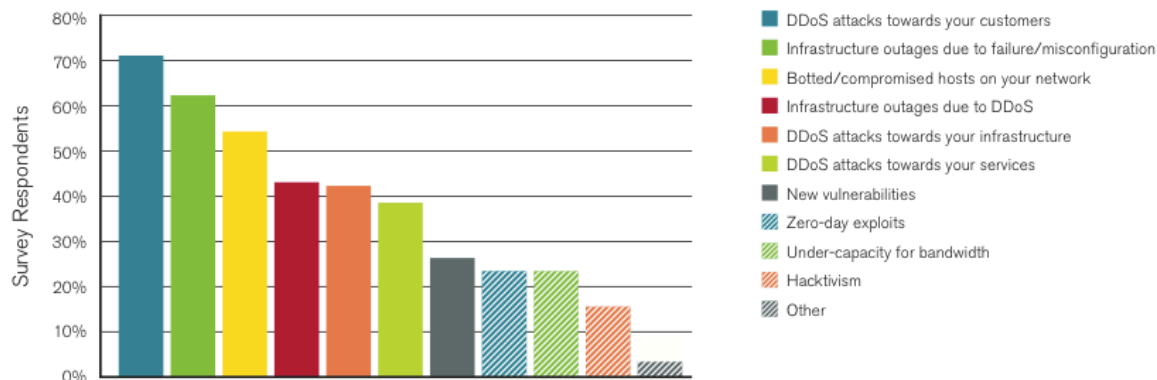


Cloud computing  
Application/data hosted  
in the cloud environment

- Different ways to handle security as the Internet evolves

# Why Security?

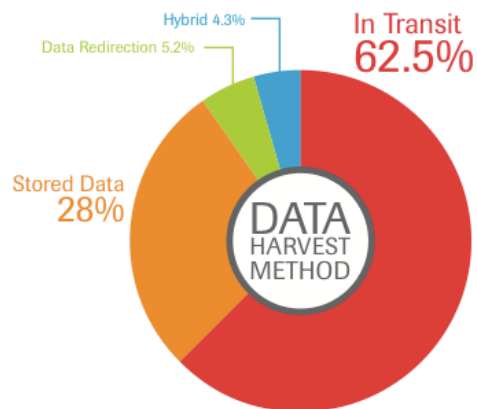
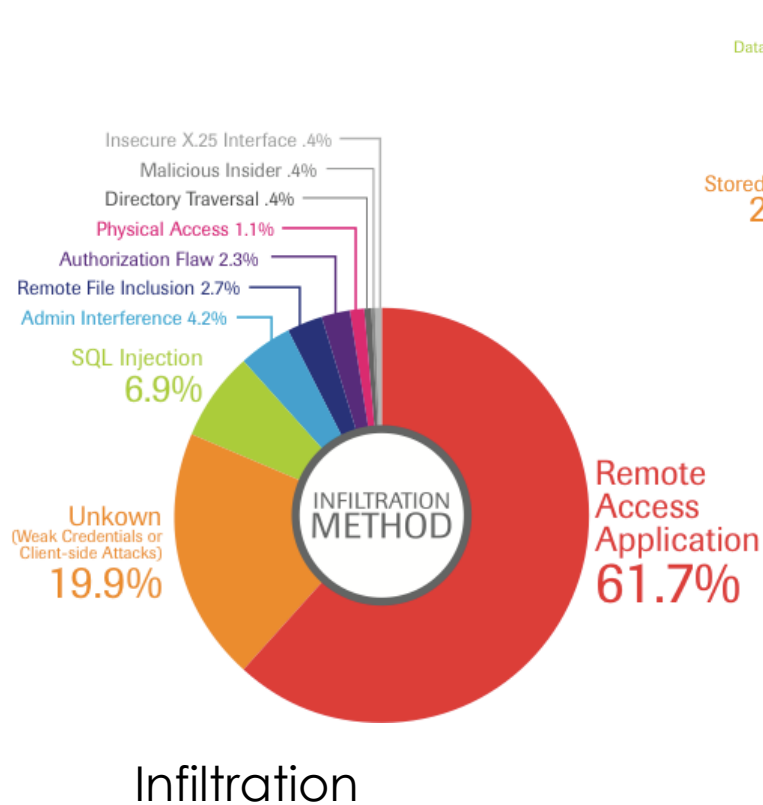
Most Significant Operational Threats



- Key findings:
  - Hacktivism and vandalism are the common DDoS attack motivation
  - High-bandwidth DDoS attacks are the ‘new normal’
  - First-ever IPv6 DDoS attacks are reported
  - Trust issues across geographic boundaries

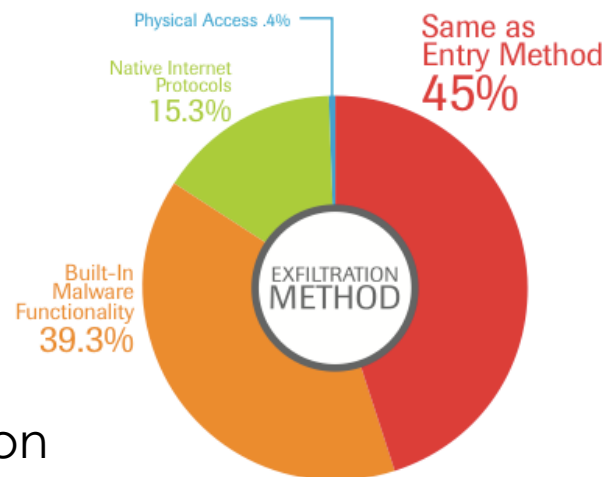
Source: Arbor Networks Worldwide Infrastructure Security Report Volume VII

# Breach Sources



Aggregation

Exfiltration



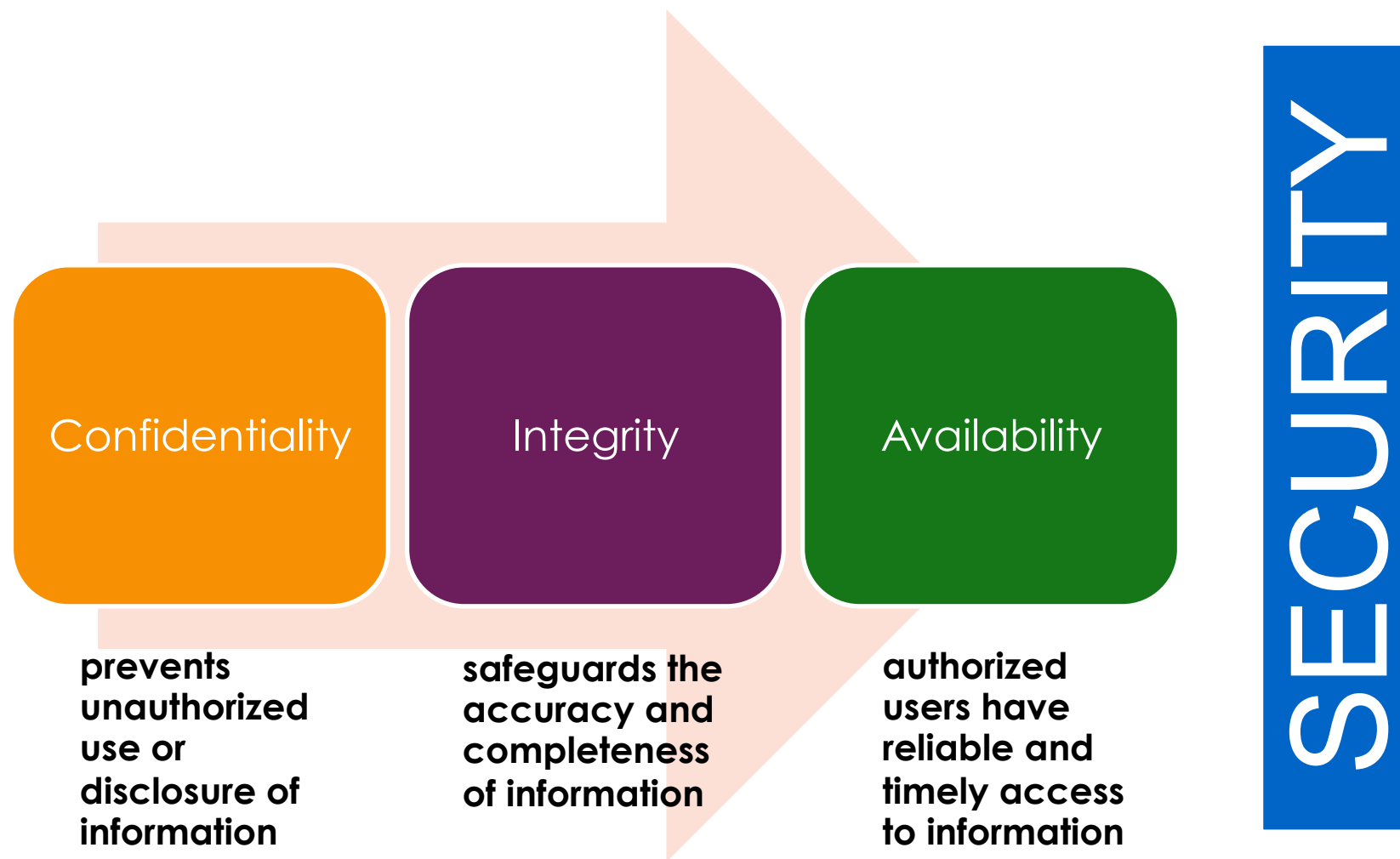
Source: Trustwave 2012 Global Security Report



# Types of Security

- Computer Security
  - generic name for the collection of tools designed to protect data and to thwart hackers
- Network Security
  - measures to protect data during their transmission
- Internet Security
  - measures to protect data during their transmission over a collection of interconnected networks

# Goals of Information Security



# Access Control

- The ability to permit or deny the use of an object by a subject.
- It provides 3 essential services:
  - Authentication (who can login)
  - Authorization (what authorized users can do)
  - Accountability (identifies what a user did)

# Authentication

- A means to verify or prove a user's identity
- The term “user” may refer to:
  - Person
  - Application or process
  - Machine or device
- Identification comes before authentication
  - Provide username to establish user's identity
- To prove identity, a user must present either of the following:
  - What you know (passwords, passphrase, PIN)
  - What you have (token, smart cards, passcodes, RFID)
  - Who you are (biometrics such as fingerprints and iris scan, signature or voice)

# Examples of Tokens



eToken



RFID cards



Smart Cards



Fingerprint scanner

# Trusted Network

- Standard defensive-oriented technologies
  - Firewall
  - Intrusion Detection
- Build TRUST on top of the TCP/IP infrastructure
  - Strong authentication
  - Public Key Infrastructure (PKI)

# Strong Authentication

- An absolute requirement
- Two-factor authentication
  - Passwords (something you know)
  - Tokens (something you have)
- Examples:
  - Passwords
  - Tokens
  - Tickets
  - Restricted access
  - PINs
  - Biometrics
  - Certificates

# Two-factor Authentication

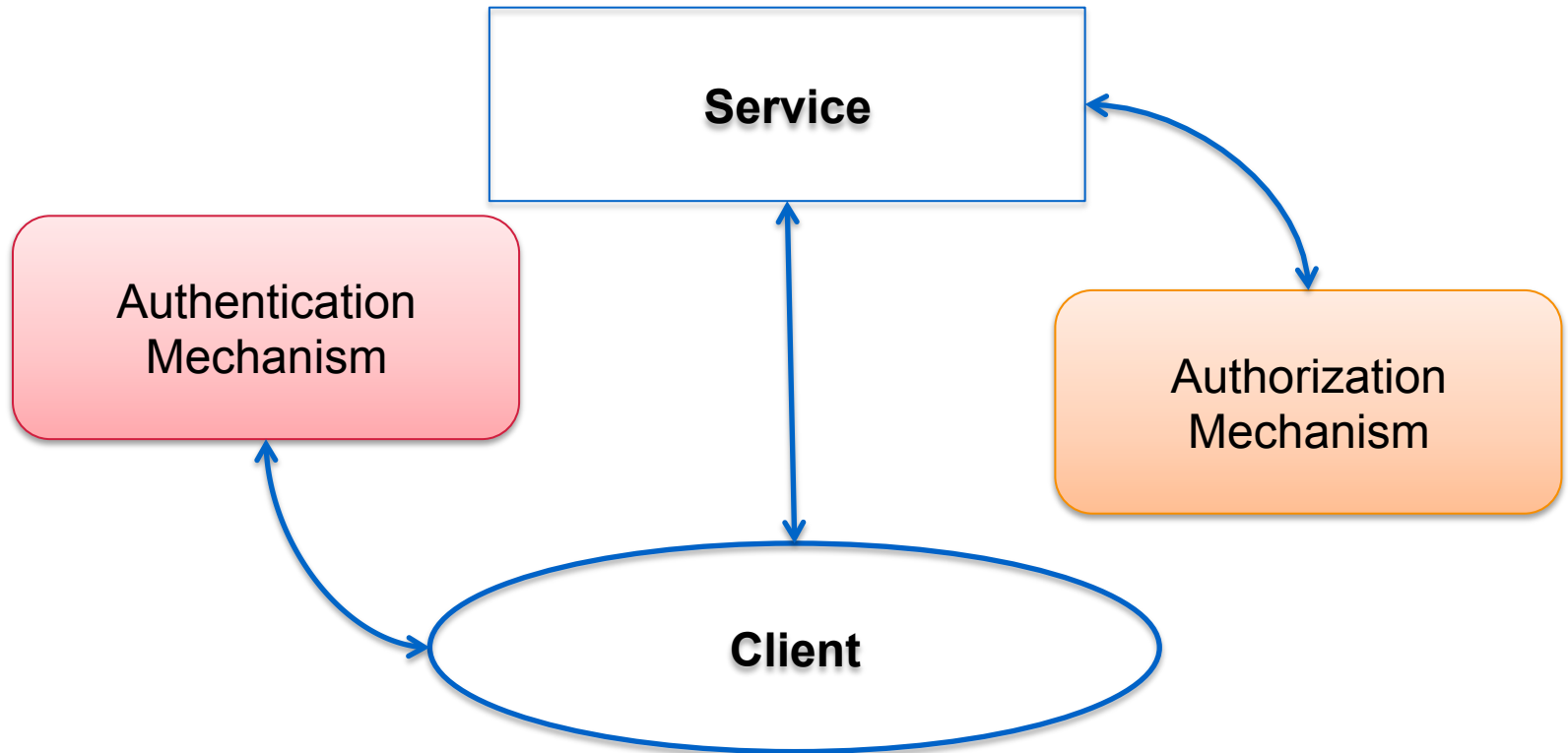
- Requires a user to provide at least two authentication 'factors' to prove his identity
  - something you know  
Username/userID and password
  - something you have  
Token using a one-time password (OTP)
- The OTP is generated using a small electronic device in physical possession of the user
  - Different OTP generated each time and expires after some time
  - An alternative way is through applications installed on your mobile device
- Multi-factor authentication is also common



# Authorization

- Defines the user's rights and permissions on a system
- Typically done after user has been authenticated
- Grants a user access to a particular resource and what actions he is permitted to perform on that resource
- Access criteria based on the level of trust:
  - Roles
  - Groups
  - Location
  - Time
  - Transaction type

# Authentication vs. Authorization



“Authentication simply identifies a party, authorization defines whether they can perform certain action” – RFC 3552

# Authorization Concepts

- Authorization creep
  - When users may possess unnecessarily high access privileges within an organization
- Default to Zero
  - Start with zero access and build on top of that
- Need to Know Principle
  - Least privilege; give access only to information that the user absolutely need
- Access Control Lists
  - List of users allowed to perform particular access to an object (read, write, execute, modify)

# Single Sign On

- Property of access control where a user logs in only once and gains access to all authorized resources within a system.
- Benefits:
  - Ease of use
  - Reduces logon cycle (time spent re-entering passwords for the same identity)
- Common SSO technologies:
  - Kerberos, RADIUS
  - Smart card based
  - OTP Token
- Disadvantage: Single point of attack

# Types of Access Control

- Centralized Access Control
  - Radius
  - TACACS+
  - Diameter
- Decentralized Access Control
  - Control of access by people who are closer to the resources
  - No method for consistent control

# Accountability

- The security goal that generates the requirement for actions of an entity to be traced uniquely to that entity
  - Senders cannot deny sending information
  - Receivers cannot deny receiving it
  - Users cannot deny performing a certain action
- Supports nonrepudiation, deterrence, fault isolation, intrusion detection and prevention and after-action recovery and legal action

Source: NIST Risk Management Guide for  
Information Technology Systems

# Integrity

- Security goal that generates the requirement for protection against either intentional or accidental attempts to violate data integrity
- Data integrity
  - The property that data has when it has not been altered in an unauthorized manner
- System integrity
  - The quality that a system has when it performs its intended function in an unimpaired manner, free from unauthorized manipulation

Source: NIST Risk Management Guide for Information Technology Systems

# Risk, Threat and Vulnerability

- Vulnerability - weakness in a system
- Risk - likelihood that a particular threat using a particular attack will exploit a particular vulnerability
- Exploit - taking advantage of a vulnerability
- Non-repudiation—assurance that both parties are involved in the transaction



# Vulnerability

- A weakness in security procedures, network design, or implementation that can be exploited to violate a corporate security policy
  - Software bugs
  - Configuration mistakes
  - Network design flaw
  - Lack of encryption
- Exploit
  - Taking advantage of a vulnerability

# Threat

- Any circumstance or event with the potential to cause harm to a networked system.
- These are some example of threats:
  - Denial of service
    - Attacks make computer resources (e.g., bandwidth, disk space, or CPU time) unavailable to its intended users
  - Unauthorised access
    - Access without permission issues by a rightful owner of devices or networks
  - Impersonation
  - Worms
  - Viruses

# Risk

- The possibility that a particular vulnerability will be exploited
- IT-related risks arise from:
  - Unauthorized (malicious or accidental) disclosure, modification, or destruction of information
  - Unintentional errors or omissions
  - IT disruptions due to natural or man-made disasters
  - Failure to exercise due care and diligence in implementation and operation of the IT system

$$\text{Risk} = \text{Threat} * \text{Vulnerability} \\ (* \text{ Impact})$$

# Risk Analysis

- Identification, assessment and reduction of risks to an acceptable level
- the process of identifying security risks and probability of occurrence, determining their impact, and identifying areas that require protection
- Three parts:
  - Risk assessment – determine the possible risks
  - Risk management – evaluating alternatives for mitigating the risk
  - Risk communication – presenting this material in an understandable way to decision makers and/or the public

# Risk Management vs. Cost of Security

- Risk mitigation
  - The process of selecting appropriate controls to reduce risk to an acceptable level
- The level of acceptable risk
  - Determined by comparing the risk of security hole exposure to the cost of implementing and enforcing the security policy
- Trade-offs between safety, cost, and availability

# Attack Sources

- Active vs. passive
  - Active involves writing data to the network. It is common to disguise one's address and conceal the identity of the traffic sender
  - Passive involves only reading data on the network. Its purpose is breach of confidentiality. This is possible if:
    - Attacker has gained control of a host in the communication path between two victim machines
    - Attacker has compromised the routing infrastructure to arrange the traffic pass through a compromised machine

---

## Active Attacks

Denial of Service attacks  
Spoofing  
Man in the Middle  
ARP poisoning  
Smurf attacks  
Buffer overflow  
SQL Injection

## Passive Attacks

Reconnaissance  
Eavesdropping  
Port scanning

Source: RFC 4778

# Attack Sources

- On-path vs. Off-path
  - On-path routers (transmitting datagrams) can read, modify, or remove any datagram transmitted along the path
  - Off-path hosts can transmit datagrams that appear to come from any hosts but cannot necessarily receive datagrams intended for other hosts
    - If attackers want to receive data, they have to put themselves on-path
  - How easy is it to subvert network topology?
    - It is not easy thing to do but, it is not impossible
- Insider vs. outsider
  - What is definition of perimeter/border?
- Deliberate attack vs. unintentional event
  - Configuration errors and software bugs are as harmful as a deliberate malicious network attack

Source: RFC 4778

# General Threats

- Masquerade
  - An entity claims to be another entity
- Eavesdropping
  - An entity reads information it is not intended to read
- Authorization violation
  - An entity uses a service or resource it is not intended to use
- Loss or modification of information
  - Data is being altered or destroyed
- Denial of communication acts (repudiation)
  - An entity falsely denies its participation in a communication act
- Forgery of information
  - An entity creates new information in the name of another entity
- Sabotage
  - Any action that aims to reduce the availability and/or correct functioning of services or systems



# Reconnaissance Attack

- Unauthorised users to gather information about the network or system before launching other more serious types of attacks
- Also called eavesdropping
- Information gained from this attack is used in subsequent attacks (DoS or DDoS type)
- Examples of relevant information:
  - Names, email address
    - Common practice to use a person's first initial and last name for accounts
  - Practically anything

# Man-in-the-Middle Attack

- Active eavesdropping
- Attacker makes independent connections with victims and relays messages between them, making them believe that they are talking directly to each other over a private connection, when in fact the entire conversation is controlled by the attacker
- Usually a result of lack of end-to-end authentication
- Masquerading - an entity claims to be another entity

# Session Hijacking

- Exploitation of a valid computer session, to gain unauthorized access to information or services in a computer system.
- Theft of a “magic cookie” used to authenticate a user to a remote server (for web developers)
- Four methods:
  - Session fixation – attacker sets a user’s session id to one known to him, for example by sending the user an email with a link that contains a particular session id.
  - Session sidejacking – attacker uses packet sniffing to read network traffic between two parties to steal the session cookie.

# Denial of Service (DoS) Attack

- Attempt to make a machine or network resource unavailable to its intended users.
- Purpose is to temporarily or indefinitely interrupt or suspend services of a host connected to the Internet
- Methods to carry out this attack may vary
  - Saturating the target with external communications requests (such that it can't respond to legitimate traffic) – SERVER OVERLOAD
  - May include malware to max out target resources (such as CPU), trigger errors, or crash the operating system
- DDoS attacks are more dynamic and comes from a broader range of attackers
- Examples: SYN flooding, Smurf attacks, Starvation
- Can be used as a redirection and reconnaissance technique

# Questions?

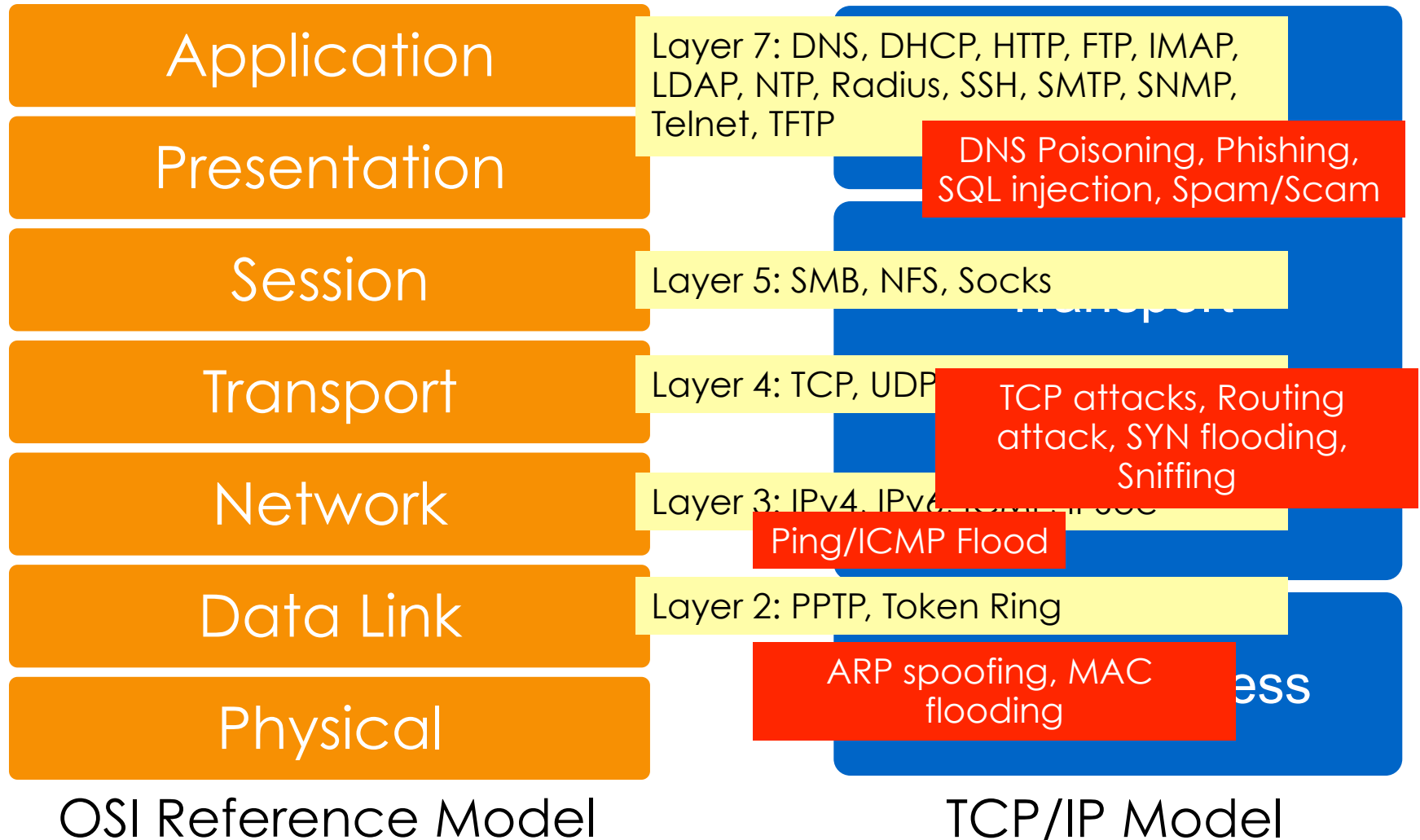
# Layered Security & Attack Mitigation

Network Security Workshop

# Overview

- Attacks in Different Layers
- Security Technologies
- Link-Layer Security
- Network Layer Security
- Transport Layer Security
- Application Layer Security

# Attacks on Different Layers

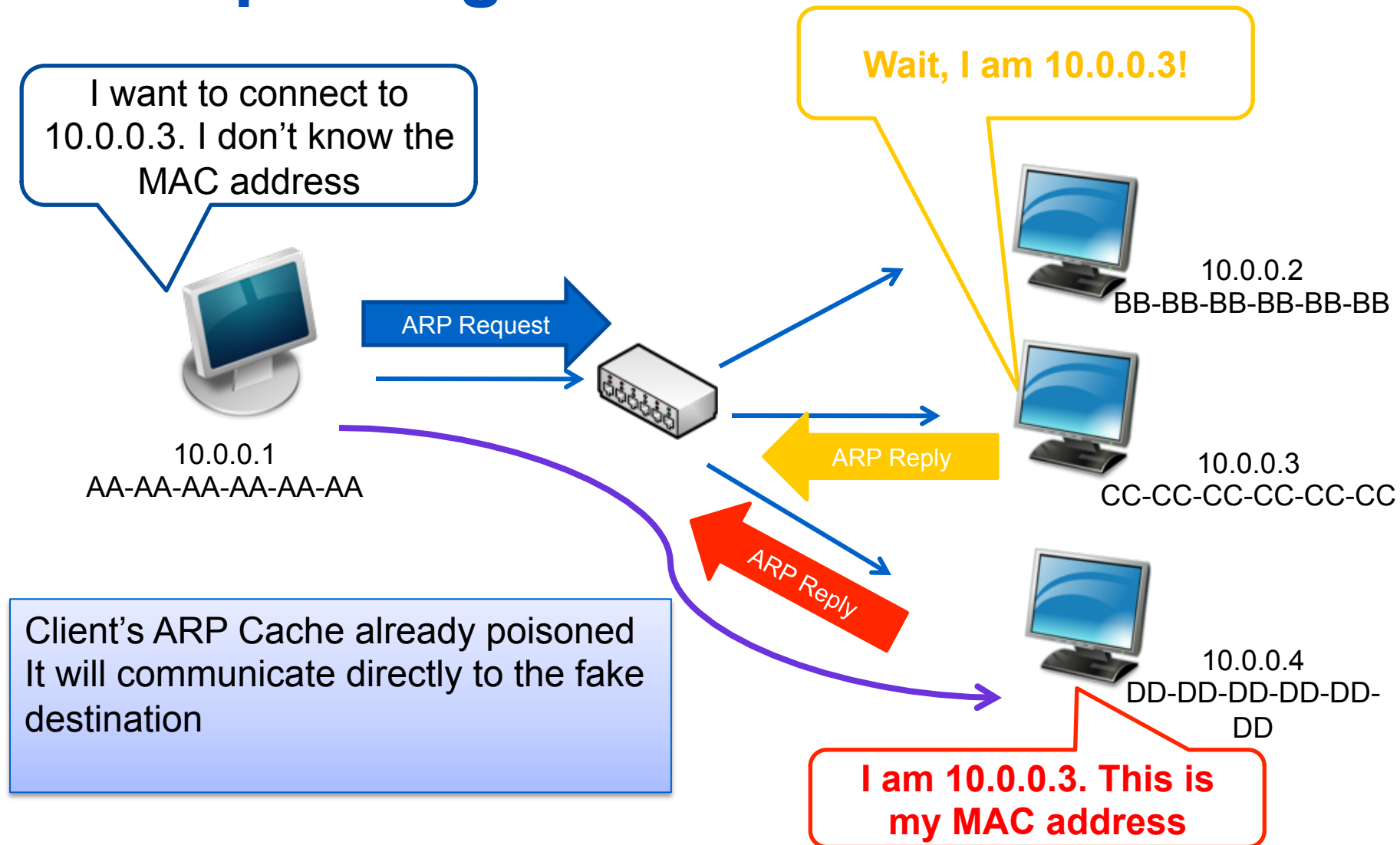




# Layer 2 Attacks

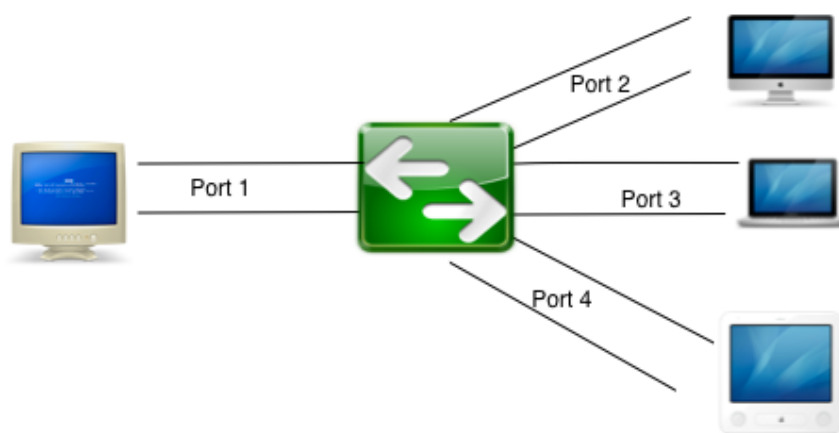
- ARP Spoofing
- MAC attacks
- DHCP attacks
- VLAN hopping

# ARP Spoofing



# MAC Flooding

- Exploits the limitation of all switches – fixed CAM table size
- CAM = Content Addressable memory = stores info on the mapping of individual MAC addresses to physical ports on the switch.

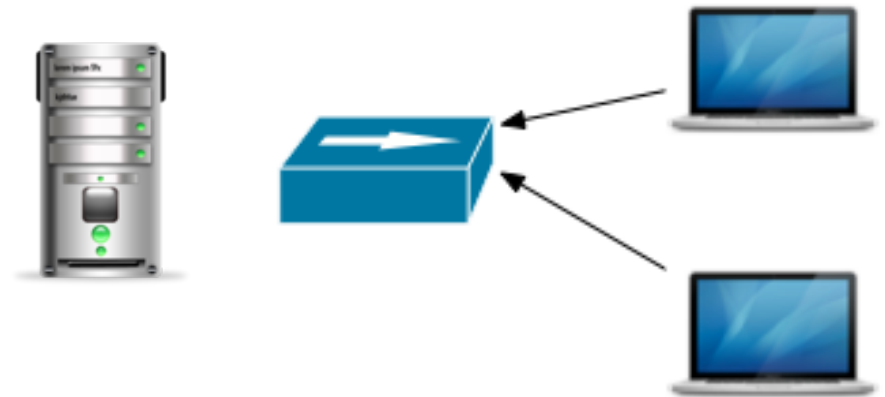


|                   | Port 1 | Port 2 | Port 3 | Port 4 |
|-------------------|--------|--------|--------|--------|
| 00:01:23:45:67:A1 | x      |        |        |        |
| 00:01:23:45:67:B2 |        | x      |        |        |
| 00:01:23:45:67:C3 |        |        | x      |        |
| 00:01:23:45:67:D4 |        |        |        | x      |

# DHCP Attacks

- DHCP Starvation Attack
  - Broadcasting vast number of DHCP requests with spoofed MAC address simultaneously.
  - DoS attack using DHCP leases
- Rogue DHCP Server Attacks

Server runs out of IP addresses to allocate to valid users



Attacker sends many different DHCP requests with many spoofed addresses.

# DHCP Attack Types

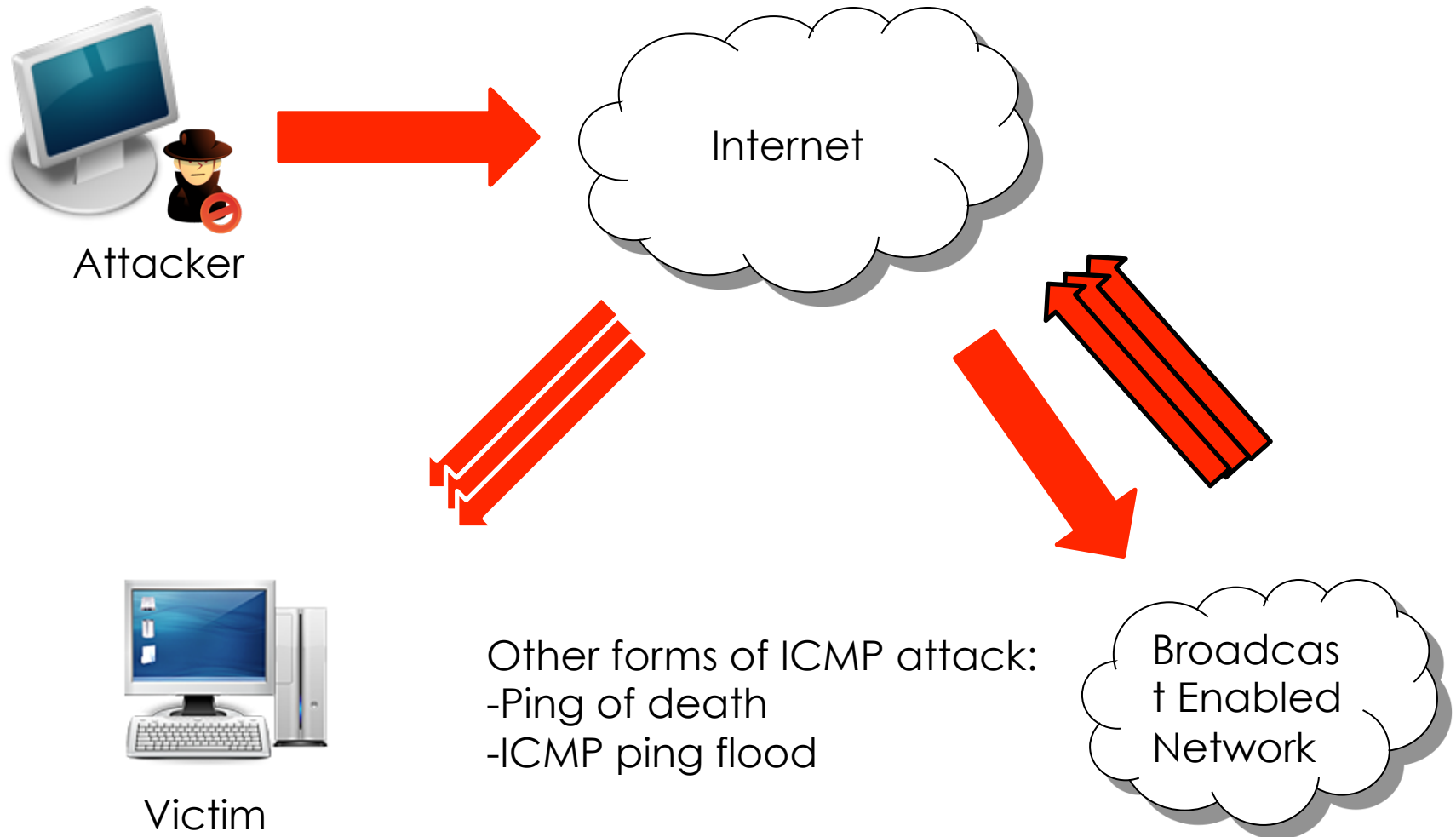
- Solution: enable DHCP snooping

```
ip dhcp snooping (enable dhcp snooping globally)
ip dhcp snooping vlan <vlan-id> (for specific
vlands)
ip dhcp snooping trust
ip dhcp snooping limit rate <rate>
```

# Layer 3 Attacks

- ICMP Ping Flood
- ICMP Smurf
- Ping of death

# Ping Flood



# Mitigating Sniffing Attacks

- Avoid using insecure protocols like basic HTTP authentication and telnet.
- If you have to use an insecure protocol, try tunneling it through something to encrypt the sensitive data.
- Run ARPwatch.
- Try running tools like sniffdet and Sentinel to detect network cards in promiscuous mode that may be running sniffing software.



# Routing Attacks

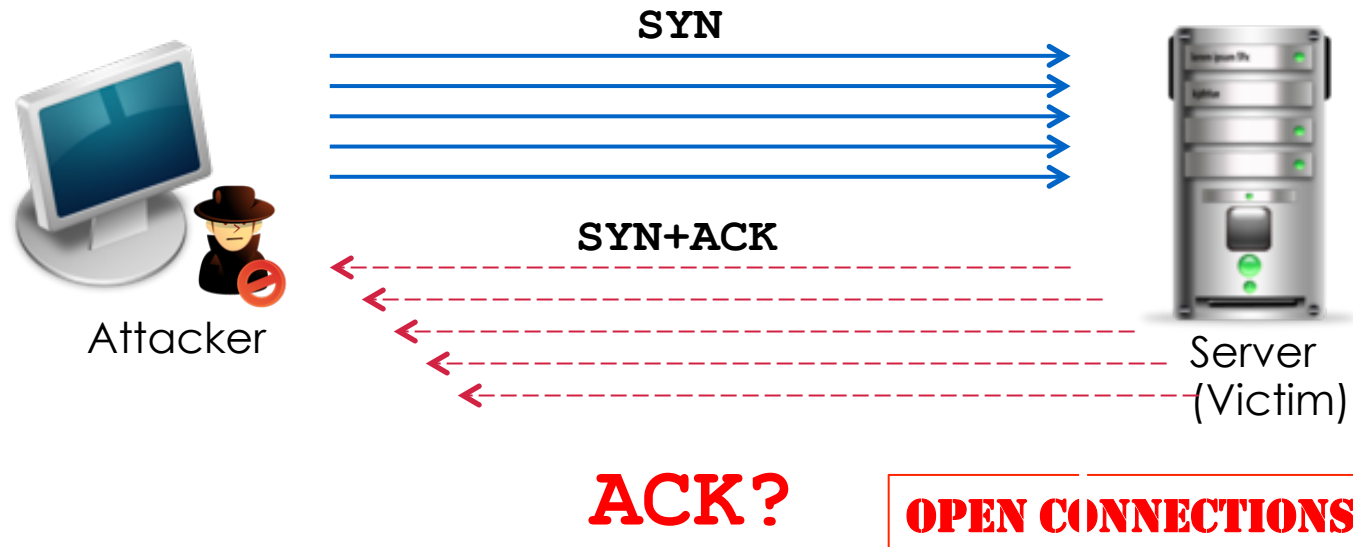
- Attempt to poison the routing information
- Distance Vector Routing
  - Announce 0 distance to all other nodes
    - Blackhole traffic
    - Eavesdrop
- Link State Routing
  - Can drop links randomly
  - Can claim direct link to any other routers
  - A bit harder to attack than DV
- BGP attacks
  - ASes can announce arbitrary prefix
  - ASes can alter path

# TCP Attacks

- SYN Flood – occurs when an attacker sends SYN requests in succession to a target.
- Causes a host to retain enough state for bogus half-connections such that there are no resources left to establish new legitimate connections.

# TCP Attacks

- Exploits the TCP 3-way handshake
- Attacker sends a series of SYN packets without replying with the ACK packet
- Finite queue size for incomplete connections



# Application Layer Attacks

- Applications don't authenticate properly
- Authentication information in clear
  - FTP, Telnet, POP
- DNS insecurity
  - DNS poisoning
  - DNS zone transfer

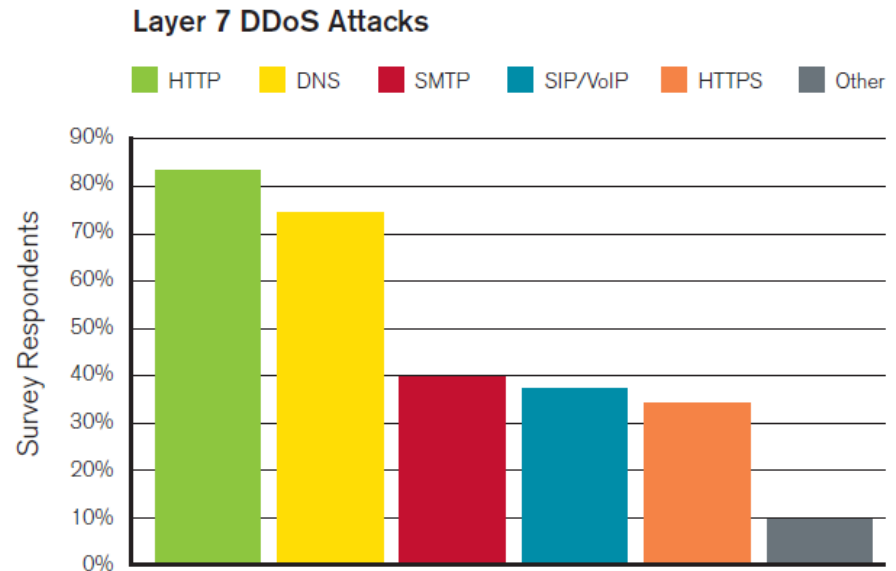


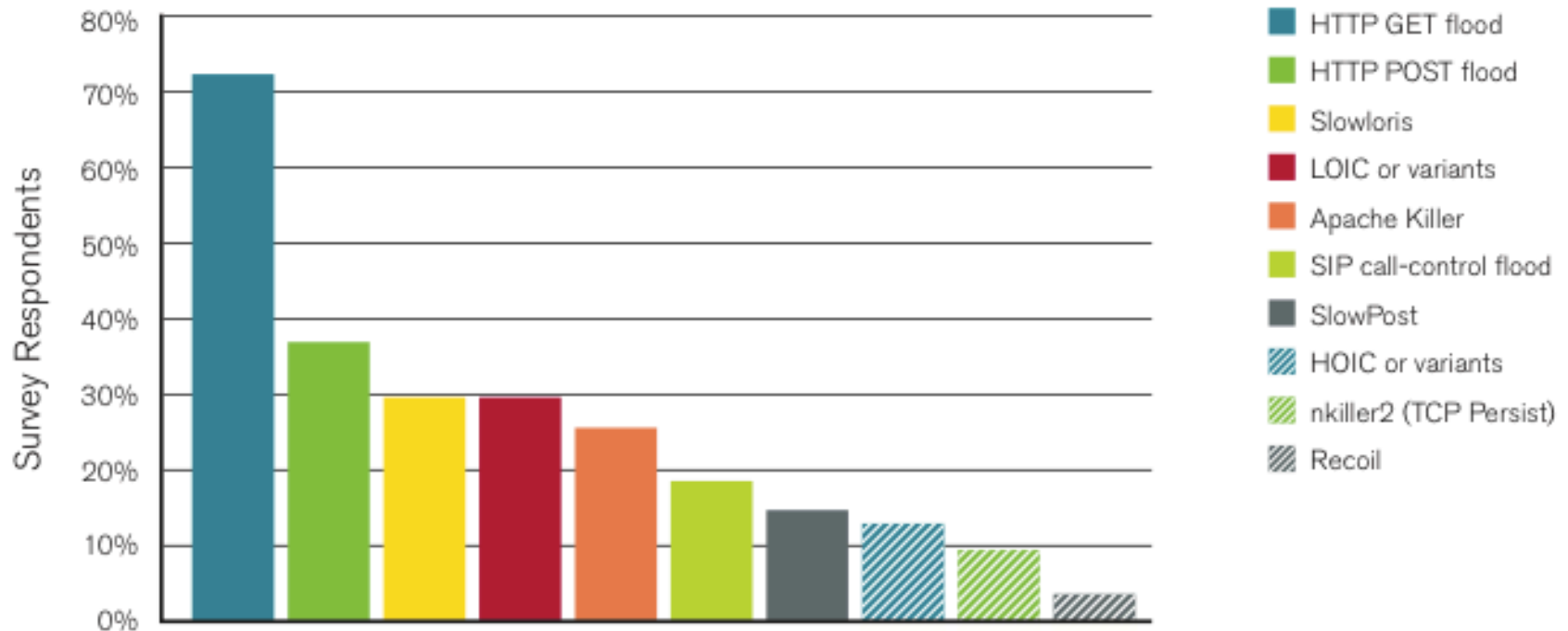
Figure 8  
Source: Arbor Networks, Inc.

# Application Layer Attacks

- Scripting vulnerabilities
- Cookie poisoning
- Buffer overflow
- Hidden field manipulation
- Parameter tampering
- Cross-site scripting
- SQL injection

# Application-Layer Attacks

## Application-Layer DDoS Attack Methodologies



Source: Arbor Networks Worldwide Infrastructure Security Report Volume VII

# Application Layer DDoS: Slowloris

- Incomplete HTTP requests
- Properties
  - Low bandwidth
  - Keep sockets alive
  - Only affects certain web servers
  - Doesn't work through load balancers
  - Managed to work around accf\_http

# Web Application Security Risks

- Injection
- Cross-Site Scripting
- Broken authentication and Session Management
- Insecure Direct Object References
- Cross-site Request Forgery (CSRF)
- Insecure Cryptographic Storage
- Failure to Restrict URL Access
- Insufficient Transport Layer Protection
- Unvalidated Redirects and Forwards

Source: OWASP Top 10 Application Security Risks, 2010



# DNS Changer

- “Criminals have learned that if they can control a user’s DNS servers, they can control what sites the user connects to the Internet.”
- How: infect computers with a malicious software (malware)
- This malware changes the user’s DNS settings with that of the attacker’s DNS servers
- Points the DNS configuration to DNS resolvers in specific address blocks and use it for their criminal enterprise
- For more: see the NANOG presentation by Merike

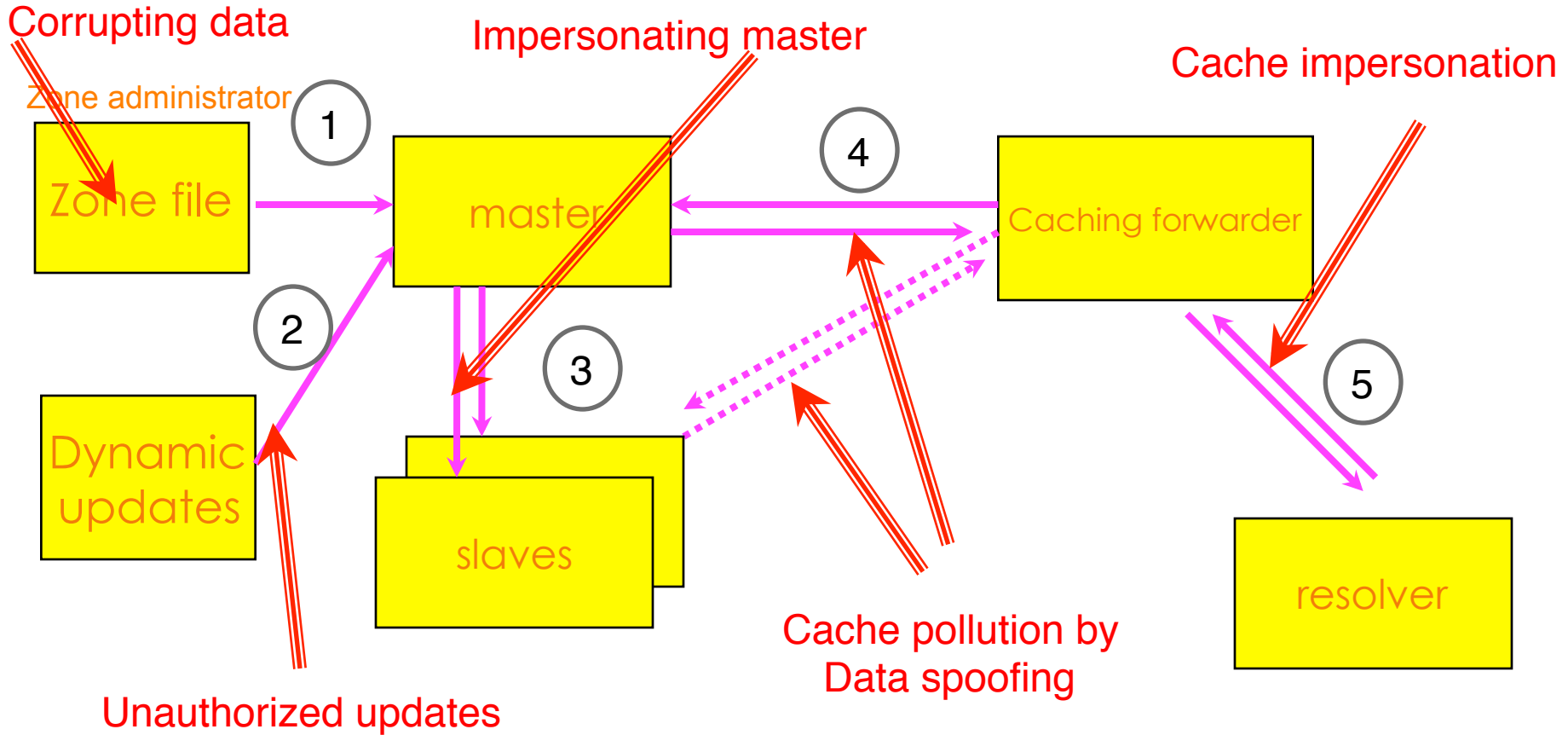
# Rogue DNS Servers

- 85.225.112.0 through 85.255.127.255
  - 67.210.0.0 through 67.210.15.255
  - 93.188.160.0 through 93.188.167.255
  - 77.67.83.0 through 77.67.83.255
  - 213.109.64.0 through 213.109.79.255
  - 64.28.176.0 through 64.28.191.255
- 
- If your computer is configured with one of these DNS servers, it is most likely infected with DNSChanger malware

# Top DNS Changer Infections

- By country (as of 11 June, 2012):
  - USA - 69517
  - IT – 26494
  - IN – 21302
  - GB – 19589
  - DE – 18427
- By ASNs
  - AS9829 (India) – 15568
  - AS3269 () – 13406
  - AS7922 () – 11964
  - AS3320 () – 9250
  - AS7132 () – 6743
- More info at <http://dcwg.org/>

# DNS Vulnerabilities



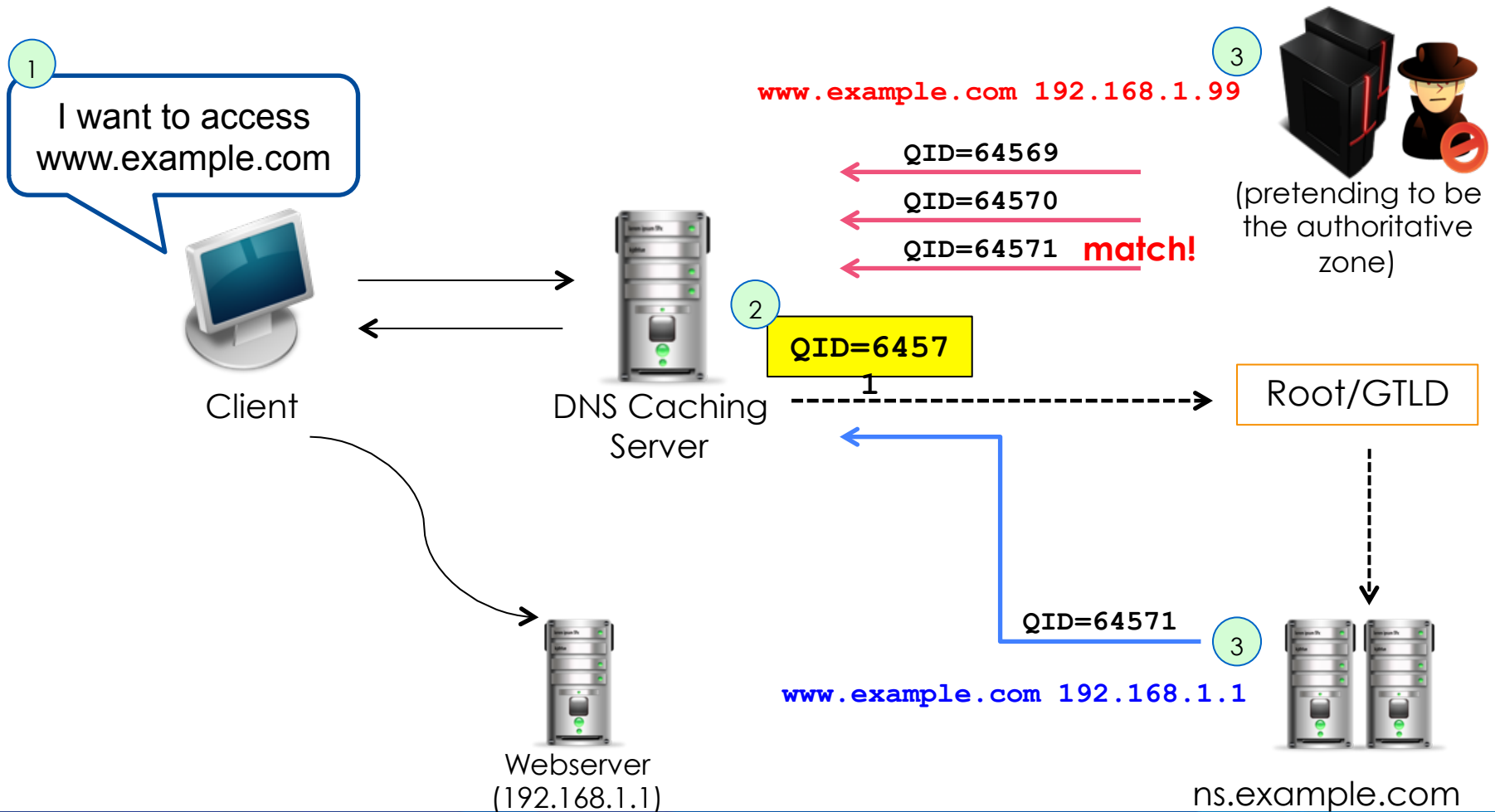
Server protection

Data protection

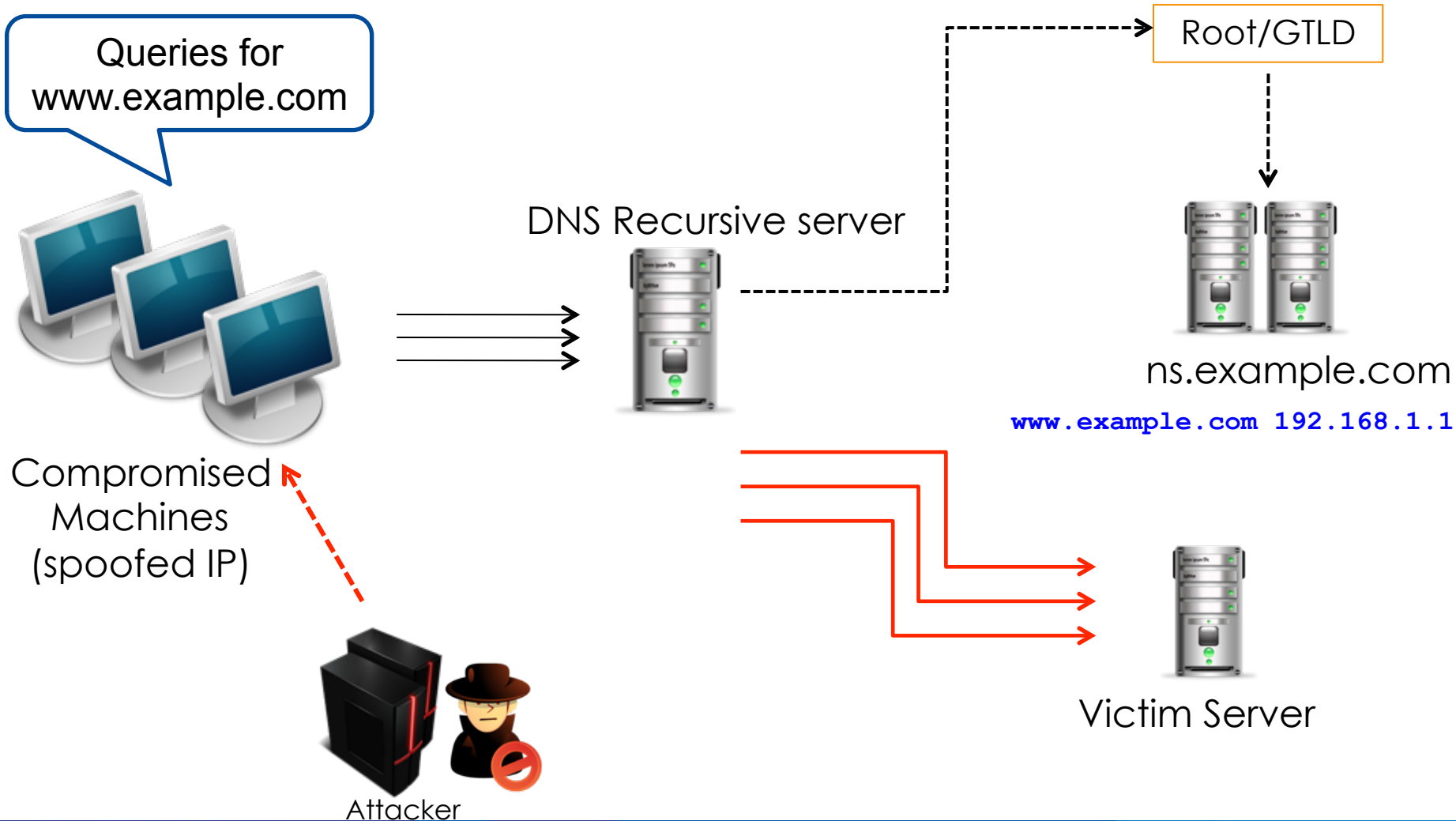
# DNS Cache Poisoning

- Caching incorrect resource record that did not originate from authoritative DNS sources.
- Result: connection (web, email, network) is redirected to another target (controlled by the attacker)

# DNS Cache Poisoning



# DNS Amplification



# Common Types of Attack

- Ping sweeps and port scans - reconnaissance
- Sniffing – capture packet as they travel through the network
- Man-in-the-middle attack – intercepts messages that are intended for a valid device
- Spoofing - sets up a fake device and trick others to send messages to it
- Hijacking – take control of a session
- Denial of Service (DoS) and Distributed DoS (DDoS)



# Wireless Attacks

- WEP – first security mechanism for 802.11 wireless networks
- Weaknesses in this protocol were discovered by Fluhrer, Mantin and Shamir, whose attacks became known as “FMS attacks”
- Tools were developed to automate WEP cracking
- Chopping attack were released to crack WEP more effectively and faster
- Cloud-based WPA cracker
  - <https://www.wpacracker.com/>

# Man in the Middle Attacks (Wireless)

- Creates a fake access point and have clients authenticate to it instead of a legitimate one.
- Capture traffic to see usernames, passwords, etc that are sent in clear text.

# Botnet

- Collection of compromised computers (or 'bot')
- Computers are targeted by malware (malicious software)
- Once controlled, an attacker can use the compromised computer via standards-based network protocol such as IRC and HTTP
- How to become a bot:
  - Drive-by downloads (malware)
  - Go to malicious websites (exploits web browser vulnerabilities)
  - Run malicious programs (Trojan) from websites or as email attachment

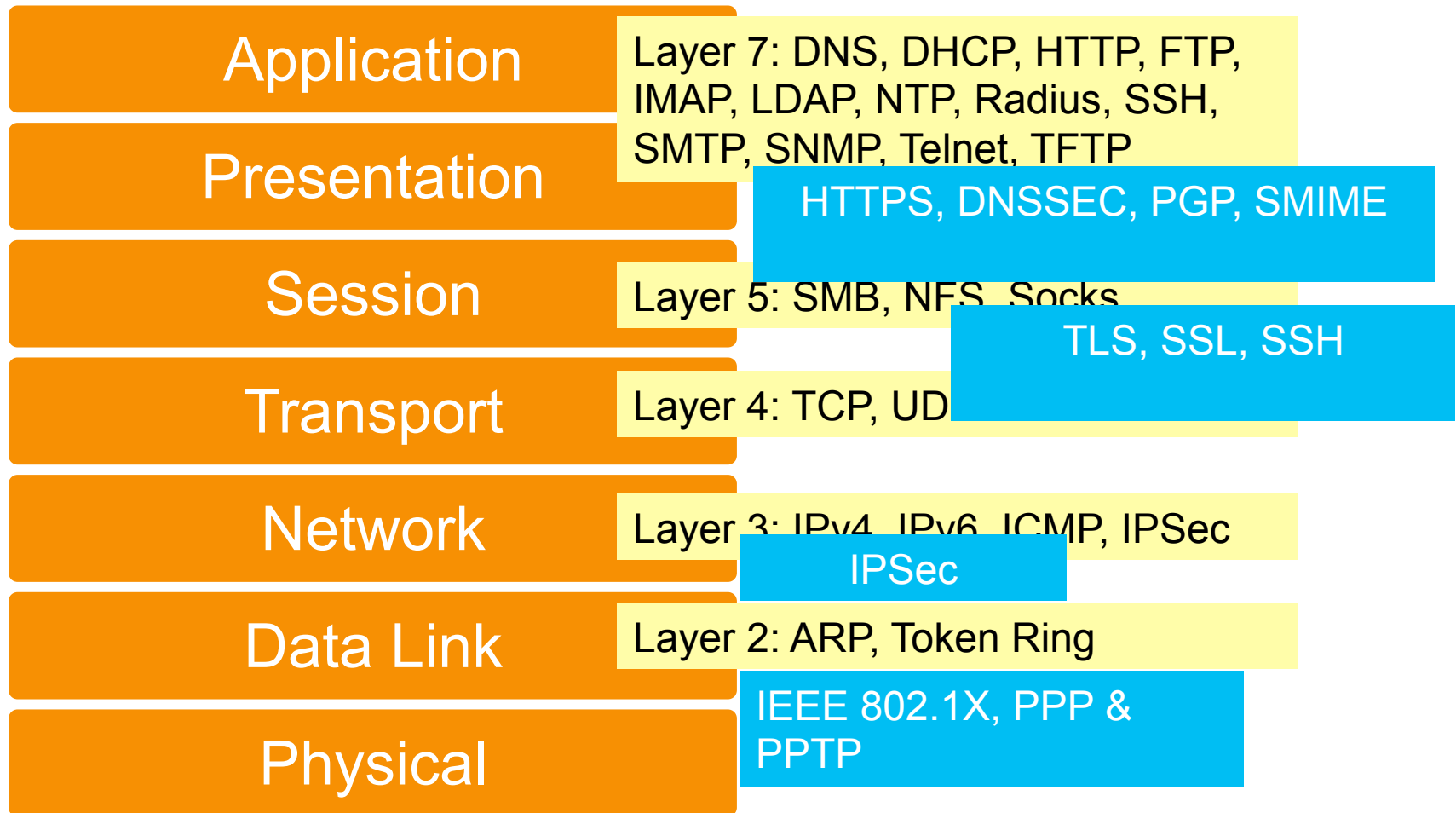
# Password Cracking

- Dictionary attacks
  - Guessing passwords using a file of 1M possible password values
    - Ordinary words and people's names
  - Offline dictionary attack when the entire password file has been attacked
  - Use random characters as password with varying upper and lower case, numbers, and symbols
- Brute-force attacks
  - Checking all possible values until it has been found
  - The resource needed to perform this attack grows exponentially while increasing the key size
- Social engineering

# Pharming and Phishing

- Phishing – victims are redirected to a fake website that looks genuine. When the victim supplies his account and password, this can be used by the attacker to the target site
  - Typically uses fraud emails with clickable links to fake websites
- Pharming – redirect a website's traffic to another fake site by changing the victim's DNS settings or hosts file

# Security on Different Layers

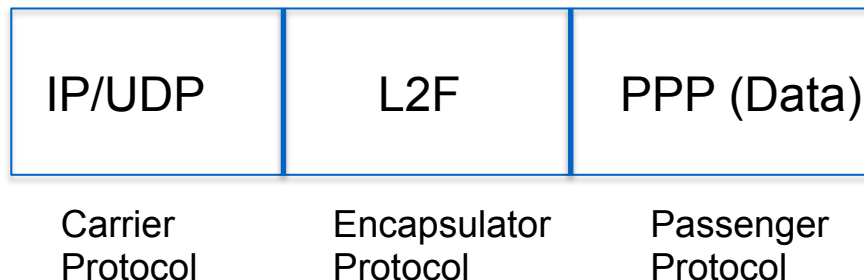


# Link-Layer Security

- Layer 2 Forwarding (L2F)
- Point-to-Point Tunneling Protocol (PPTP)
- Layer 2 Tunneling Protocol (L2TP)

# Layer 2 Forwarding Protocol

- Created by Cisco Systems and replaced by L2TP
- Permits the tunneling of the link layer – High-level Data Link Control (HDLC), async HDLC, or Serial Line Internet Protocol (SLIP) frames – of higher-level protocols





# Point to Point Tunneling Protocol

- Initiated by Microsoft but later became an informational standard in the IETF (RFC 2637)
- Client/server architecture that allows PPP to be tunneled through an IP network and decouples functions that exist in current NAS.
- Connection-oriented

# Layer 2 Tunneling Protocol

- Combination of L2F and PPTP
- Published as RFC 2661 and known as L2TPv2
- L2TPv3 provides additional security features and the ability to carry data links other than PPP
- The two end-points are L2TP Access Concentrator (LAC) or L2TP Network Server (LNS)

# PPPoE

- PPP over Ethernet
- Defined in RFC 2516
- A means to encapsulate PPP packets over the Ethernet link layer
- Mostly used in ADSL environments to provide access control, billing, and type of service on a per-user rather than a per-site basis

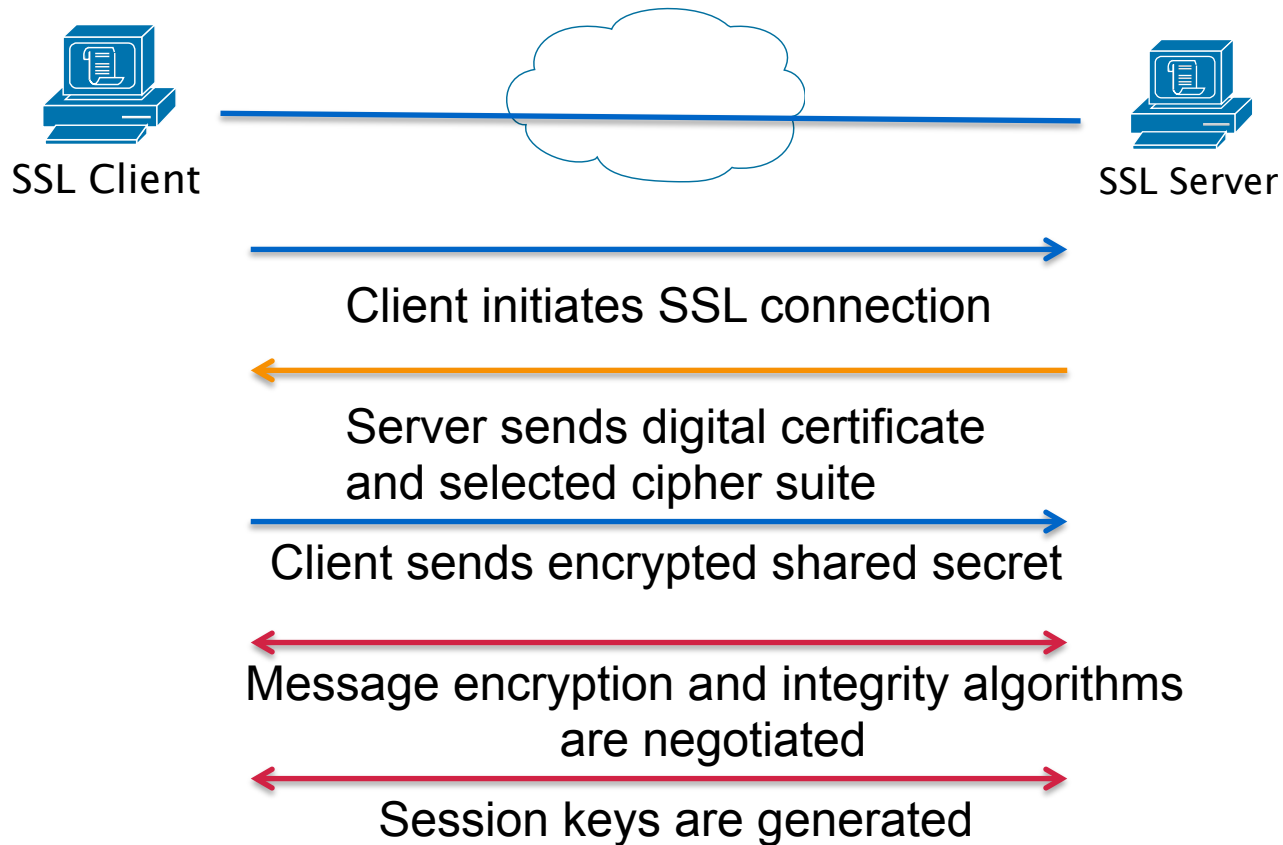
# Transport Layer Security

- Secure Socket Layer (SSL)
- Secure Shell Protocol
- SOCKS Protocol

# SSL/TLS

- TLS and SSL encrypts the segments of network connections above the Transport Layer.
- Versions:
  - SSLv1 – designed by Netscape
  - SSLv2 – publicly released in 1994; has a number of security flaws; uses RC4 for encryption and MD5 for authentication
  - SSLv3 – added support for DSS for authentication and DH for key agreement
  - TLS – based on SSLv3; uses DSS for authentication, DH for key agreement, and 3DES for encryption
- TLS is the IETF standard which succeeded SSL.

# SSL Handshake



# Advantages of SSL

- The connection is private
  - Encryption is used after initial handshake to define a secret key
  - Encryption uses symmetric cryptography (DES or RC4)
- Peer's identity can be authenticated using asymmetric cryptography (RSA or DSS)
- The connection is reliable
  - Message transport includes message integrity check using a keyed MAC. Secure hash functions (SHA or MD5) are used for MAC computation.

# Applications Using SSL/TLS

| Protocol    | Defined Port Number | SSL/TLS Port Number |
|-------------|---------------------|---------------------|
| HTTP        | 80                  | 443                 |
| NNTP        | 119                 | 563                 |
| LDAP        | 389                 | 636                 |
| FTP-data    | 20                  | 989                 |
| FTP-control | 21                  | 990                 |
| Telnet      | 23                  | 992                 |
| IMAP        | 143                 | 993                 |
| POP3        | 110                 | 994                 |
| SMTP        | 25                  | 995                 |



# Secure Shell Protocol (SSH)

- Protocol for secure remote login
- Provides support for secure remote login, secure file transfer, and secure forwarding of TCP/IP and X Window System traffic
- Consists of 3 major components:
  - Transport layer protocol (server authentication, confidentiality, integrity)
  - User authentication protocol (authenticates client to the server)
  - Connection protocol (multiplexes the encrypted tunnel into several logical channels)

# Application Layer Security

- HTTPS
- PGP (Pretty Good Privacy)
- SMIME (Secure Multipurpose Internet Mail Extensions)
- TSIG and DNSSEC
- Wireless Encryption - WEP, WPA, WPA2

# HTTPS

- Hypertext Transfer Protocol Secure
- Widely-used, message-oriented communications protocol
- Connectionless oriented protocol
- Technically not a protocol in itself, but simply layering HTTP on top of the SSL/TLS protocol
- Encapsulates data after security properties of the session
- Not to be confused with S-HTTP

Note: A website must use HTTPS everywhere, otherwise it is still vulnerable to some attacks

# Pretty Good Privacy (PGP)

- Stands for Pretty Good Privacy, developed by Phil Zimmerman in 1995
- PGP is a hybrid cryptosystem
  - combines some of the best features of both conventional and public key cryptography
- Assumptions:
  - All users are using public key cryptography and have generated private/public key pairs (using RSA or El Gamal)
  - All users also use symmetric key system (DES or Rijndael)
- Offers authentication, confidentiality, compression, e-mail compatibility and segmentation

# S/MIME

- Secure Multipurpose Internet Mail Extensions
- Uses public key certificates conforming to standard X.509
- Very similar to PGP

# Securing the Nameserver

- Run the most recent version of the DNS software
  - Bind 9.9.1 or Unbound 1.4.16
  - Apply the latest patches
- Hide version
- Restrict queries
  - `Allow-query { acl_match_list; };`
- Prevent unauthorized zone transfers
  - `Allow-transfer { acl_match_list; };`
- Run BIND with the least privilege (use `chroot`)
- Randomize source ports
  - don't use `query-source` option
- Secure the box
- Use TSIG and DNSSEC

# DNSSEC

- DNSSEC – Domain Name Security Extensions
- A set of extensions to DNS that provides
  - Origin authentication of DNS data
  - Data integrity
  - Authenticated denial of existence
- designed to protect against attacks such as DNS cache poisoning.
- Adds four new resource record types:
  - RRSIG (Resource Record Signature)
  - DNSKEY (DNS Public Key)
  - DS (Delegation Signer)
  - NSEC (Next Secure)

# Questions?



# Cryptography

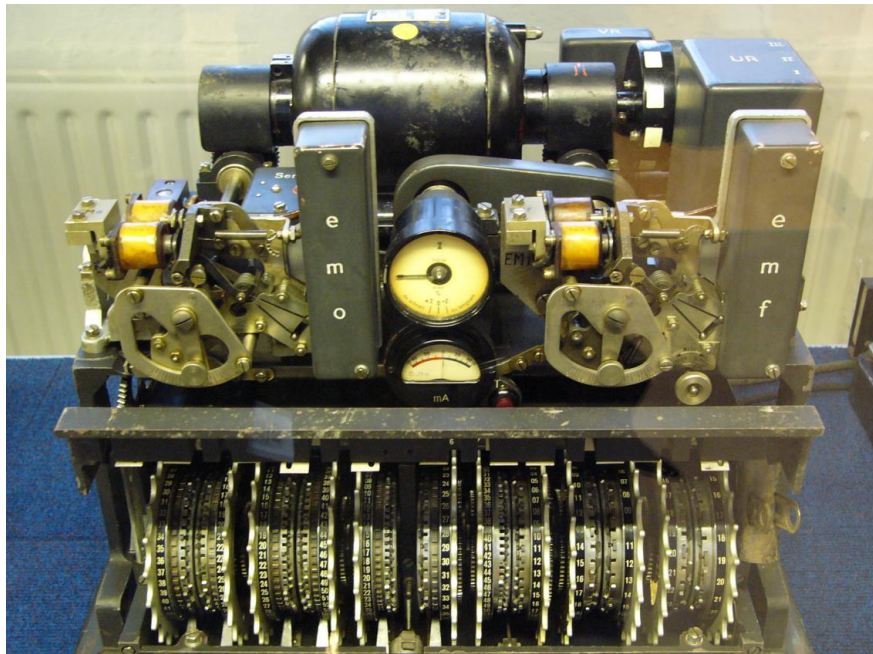
Network Security Workshop

# Overview

- What is Cryptography?
- Symmetric Key Cryptography
- Asymmetric Key Cryptography
- Block and Stream Cipher
- Digital Signature and Message Digest

# Cryptography

- Cryptography is everywhere



German Lorenz cipher machine

# Cryptography

- Cryptography deals with creating documents that can be shared secretly over public communication channels
- Other terms closely associated
  - Cryptanalysis = code breaking
  - Cryptology
    - Kryptos (hidden or secret) and Logos (description) = secret speech / communication
    - combination of cryptography and cryptanalysis
- Cryptography is a function of plaintext and a cryptographic key

$$C = F(P, k)$$

Notation:

Plaintext (P)

Ciphertext (C)

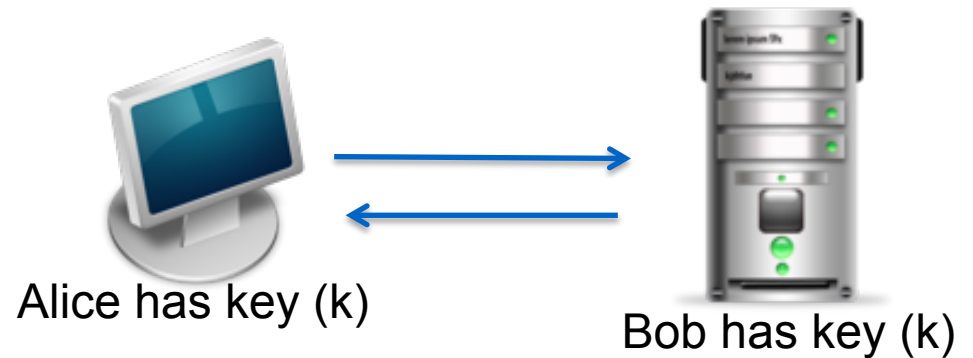
Cryptographic Key (k)

# Typical Scenario

- Alice wants to send a “secret” message to Bob
- What are the possible problems?
  - Data can be intercepted
- What are the ways to intercept this message?
- How to conceal the message?
  - Encryption

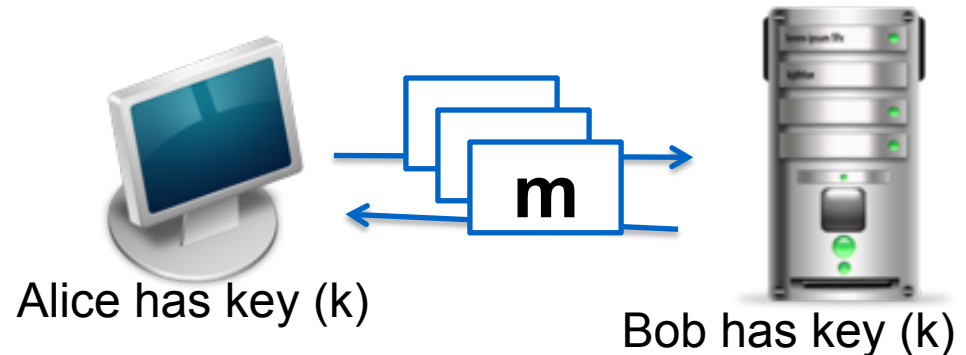
# Crypto Core

- Secure key establishment



- Secure communication

**Confidentiality and integrity**



Source: Dan Boneh, Stanford

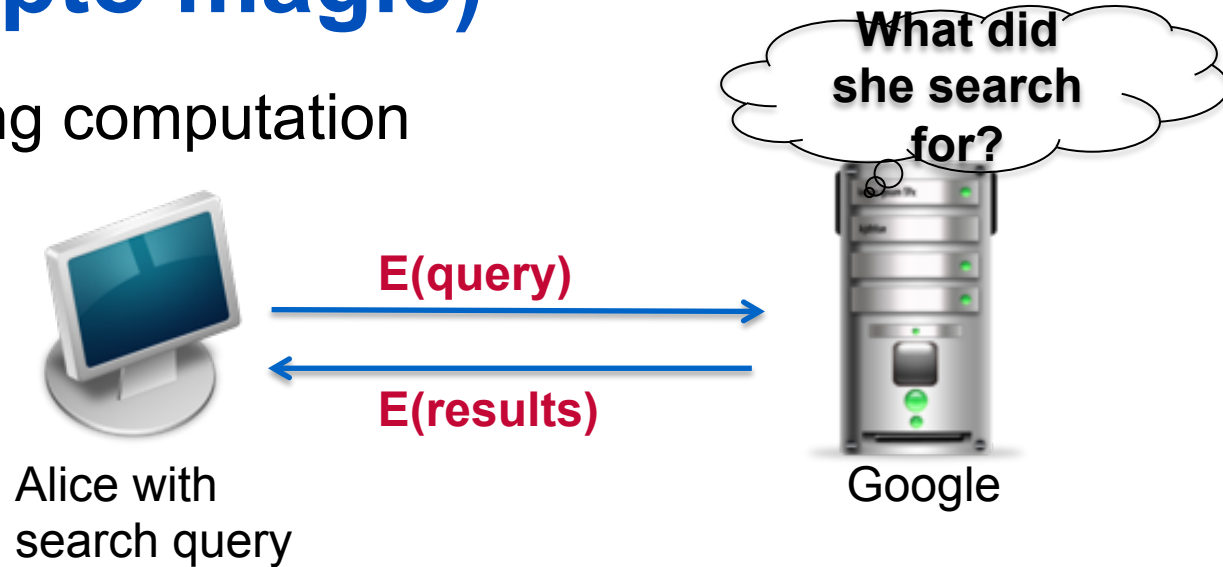
# It can do much more

- Digital Signatures
- Anonymous communication
- Anonymous digital cash
  - Spending a digital coin without anyone knowing my identity
  - Buy online anonymously?
- Elections and private auctions
  - Finding the winner without actually knowing individual votes (privacy)

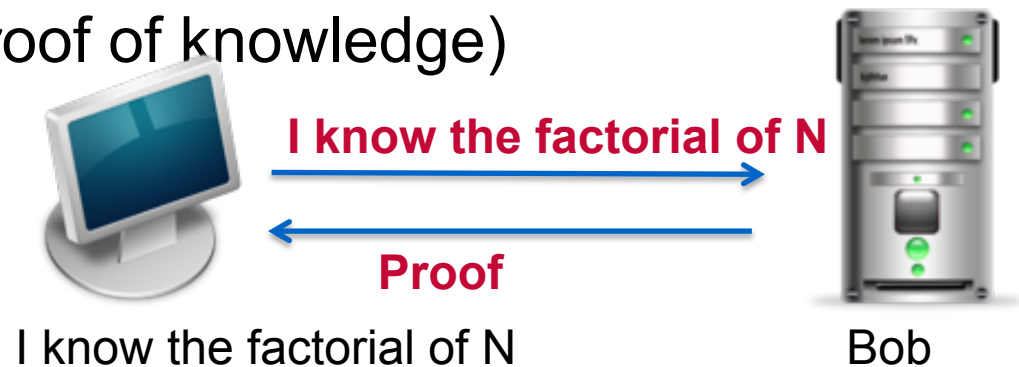
Source: Dan Boneh, Stanford

# Other uses are also theoretically possible (Crypto magic)

- Privately outsourcing computation



- Zero knowledge (proof of knowledge)



Source: Dan Boneh, Stanford



# History: Ciphers

- Substitution cipher
  - involves replacing an alphabet with another character of the same alphabet set
  - Can be mono-alphabetic (single set for substitution) or poly-alphabetic system (multiple alphabetic sets)
- Example:
  - Caesar cipher, a mono-alphabetic system in which each character is replaced by the third character in succession
  - Vigenere cipher, a poly-alphabetic cipher that uses a 26x26 table of characters

# How to Break a Substitution Cipher

UKBYBIPOUZBCUFEEBORUKBYBHOBBERFESPVKBWFOFERVNBCVBZPRUBOFERVNBCVBPCYYFVUFO  
FEIKNWFRFIKJNUPWRFIPOUNVNIPUBRNCUKBEFWWFDNCHXCBOHOPYXPUBNCUBOYNRVNIWN  
CPOJIOFHOPZRVFZIXUBORJRUBZRBCHNCBBONCHRJZSFWNVRJRUBZRPCYZPUKBZPUNVPWPCYVF  
ZIXUPUNFCPWRVNBCVBRPYYNUNFCPWWJUKBYBIPOUZBCUIPOUNVNIPUBRNCHOPYXPUBNCUB  
OYNRVNIWNCPOJIOFHOPZRNCRVNBCUNENVVFZIXUNCHPCYVFZIXUPUNFCPWZPUKBZPUNVR

(1) Use frequency of the English letters

e = 12.7%

t = 9.1 %

a = 8.1%

(2) Use frequency of pairs of letters

he, in, an, th

In the example,

**B** appeared 36 times, **U** 33 times, and **P** 32 times

**NC** appeared 11 times, **PU** 10 times

**UKB** appeared 6 times

Source: Dan Boneh, Stanford

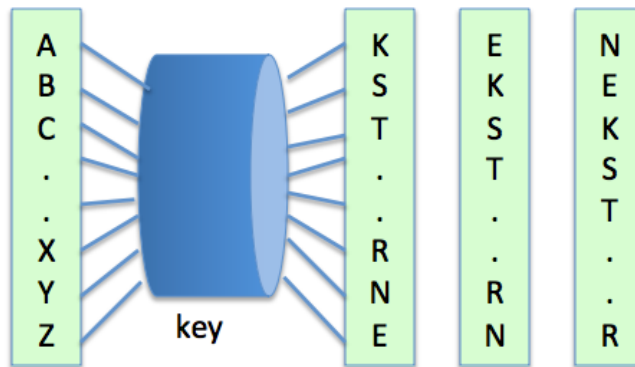
# Transposition Cipher

- No letters are replaced, they are just rearranged.
- Rail Fence Cipher – another kind of transposition cipher in which the words are spelled out as if they were a rail fence.

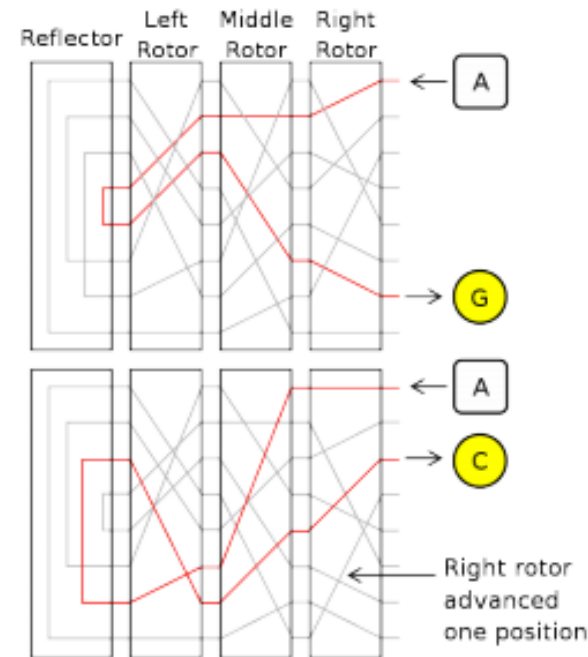
```
T...U...B...N...J...E...E...E...Y..  
.H.Q.I.K.R.W.F.X.U.P.D.V.R.H.L.Z.D.G..  
..E...C...O...O...M...O...T...A...O
```

# History: Rotor Machines (1870-1943)

- Hebern machine – single rotor



- Enigma - 3-5 rotors



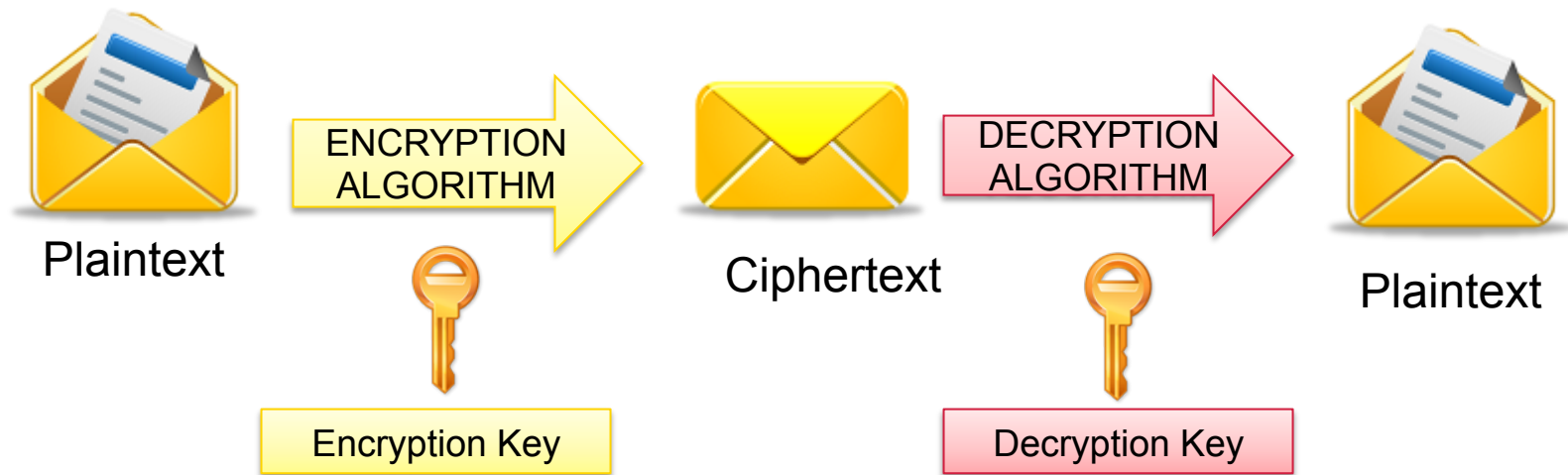
# Modern Crypto Algorithms

- specifies the mathematical transformation that is performed on data to encrypt/decrypt
- Crypto algorithm is NOT proprietary
- Analyzed by public community to show that there are no serious weaknesses
- Explicitly designed for encryption

# Encryption

- process of transforming plaintext to ciphertext using a cryptographic key
- Used all around us
  - In Application Layer – used in secure email, database sessions, and messaging
  - In session layer – using Secure Socket Layer (SSL) or Transport Layer Security (TLS)
  - In the Network Layer – using protocols such as IPSec
- Benefits of good encryption algorithm:
  - Resistant to cryptographic attack
  - They support variable and long key lengths and scalability
  - They create an avalanche effect
  - No export or import restrictions

# Encryption and Decryption

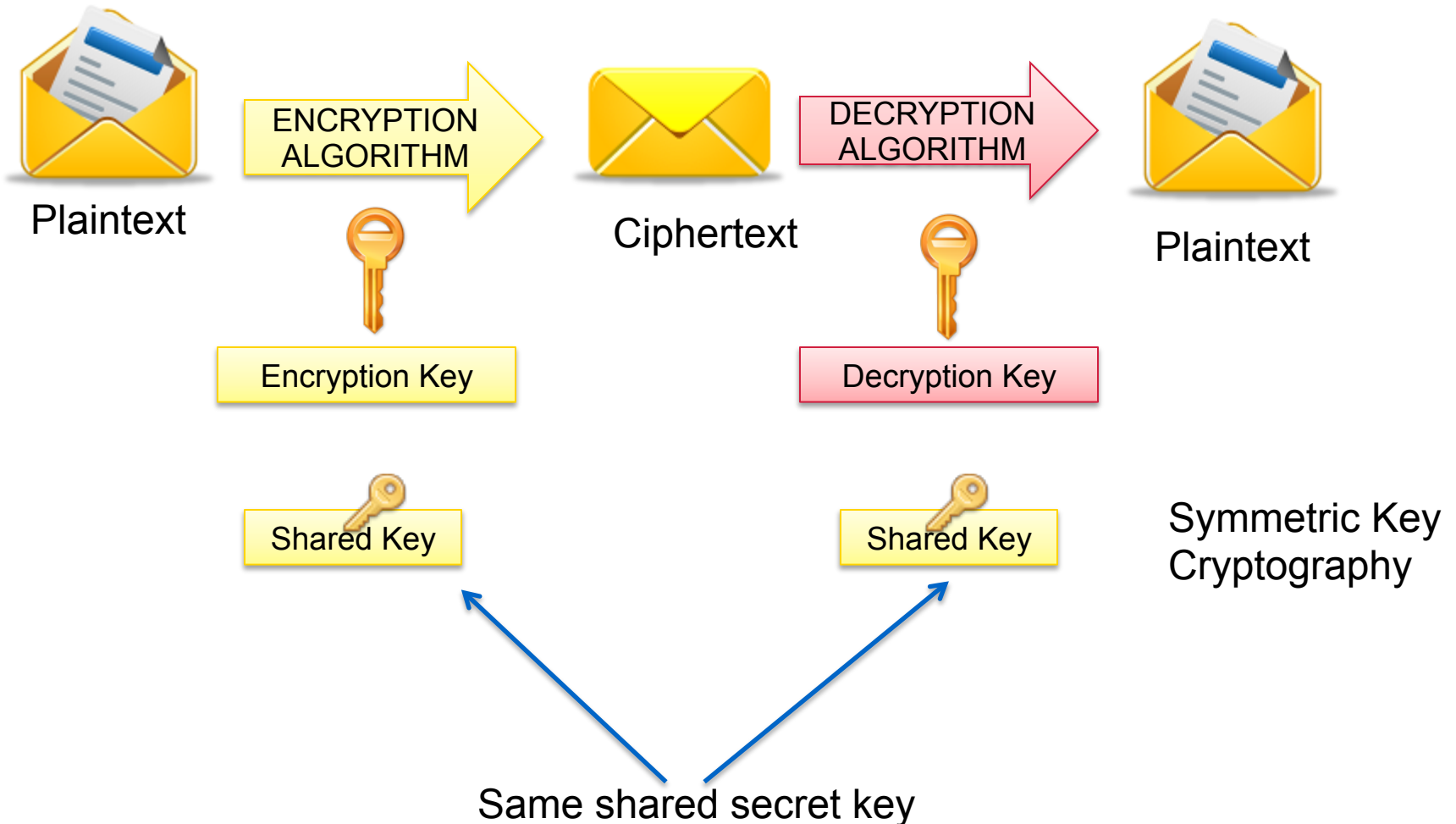


# Symmetric Key Algorithm

- Uses a single key to both encrypt and decrypt information
- Also known as a secret-key algorithm
  - The key must be kept a “secret” to maintain security
  - This key is also known as a private key
- Follows the more traditional form of cryptography with key lengths ranging from 40 to 256 bits.



# Symmetric Encryption



# Symmetric Key Algorithm

- DES – block cipher using shared key encryption, 56-bit
- 3DES (Triple DES) – a block cipher that applies DES three times to each data block
- AES – replacement for DES; it is the current standard
- RC4 – variable-length key, “stream cipher” (generate stream from key, XOR with data)
- RC6
- Blowfish

# Symmetric Key Algorithm

| Symmetric Algorithm | Key Size                   |
|---------------------|----------------------------|
| DES                 | 56-bit keys                |
| Triple DES (3DES)   | 112-bit and 168-bit keys   |
| AES                 | 128, 192, and 256-bit keys |
| IDEA                | 128-bit keys               |
| RC2                 | 40 and 64-bit keys         |
| RC4                 | 1 to 256-bit keys          |
| RC5                 | 0 to 2040-bit keys         |
| RC6                 | 128, 192, and 256-bit keys |
| Blowfish            | 32 to 448-bit keys         |

Note:

Longer keys are more difficult to crack, but more computationally expensive.

# Block and Stream Cipher

- Block cipher
  - takes a block of bits and encrypts them as a single unit
  - operate on a pre-determined block of bits (one byte, one word, 512 bytes, so forth), mixing key data in with the message data in a variety of different ways.
- Stream cipher
  - encrypts bits of the message at a time
  - typically bit-wise.
  - They either have a very long key (that eventually repeats) or a reusable key that generates a repeatable but seemingly random string of bits.
  - They perform some operation (typically an exclusive OR) with one of these key bits and one of the message bits.

# Block Cipher

- Transforms a fixed-length block of plain text into a block of ciphertext
- Works with data per block
- Common block ciphers:
  - DES and 3DES (in ECB and CBC mode)
  - Skipjack
  - Blowfish
  - RSA
  - AES
  - IDEA
  - Secure and Fast Encryption Routing (SAFER)

# Stream Cipher

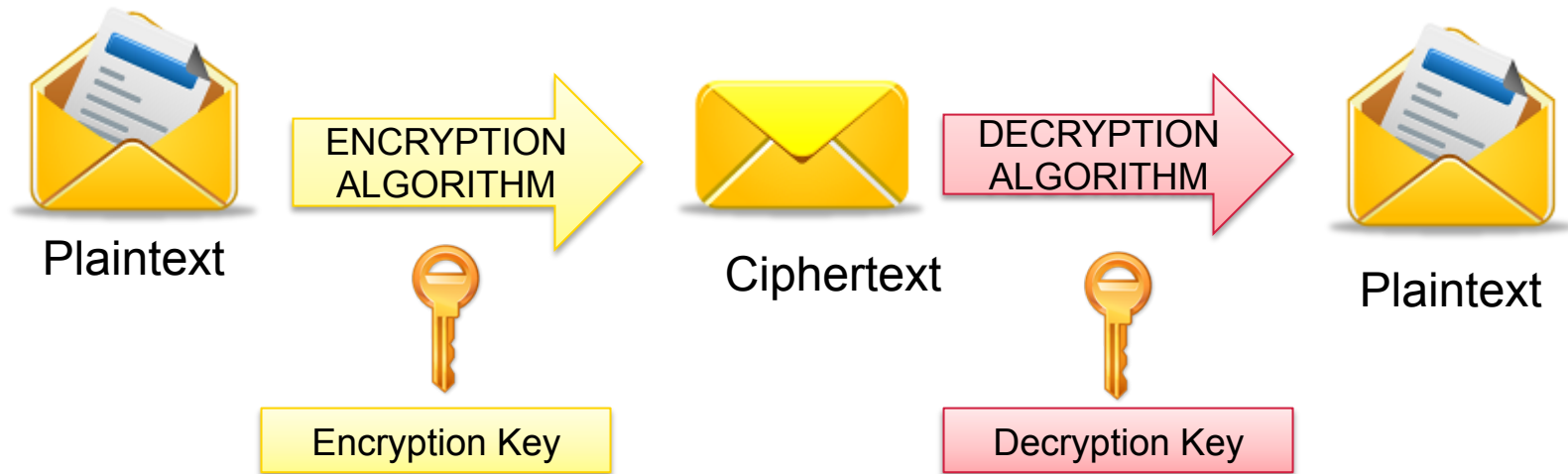
- Use smaller units of plaintext than what are used with block ciphers.
- Typically work with bits
- Common stream ciphers:
  - RC4
  - DES and 3DES (running OFB or CFB mode)
  - Software encryption algorithm (SEAL)

# Data Encryption Standard (DES)

- Developed by IBM for the US government in 1973-1974, and approved in Nov 1976.
- Based on Horst Feistel's Lucifer cipher
- block cipher using shared key encryption, 56-bit key length
- Block size: 64 bits

# DES: Illustration

64-bit blocks of input text



56-bit keys +  
8 bits parity



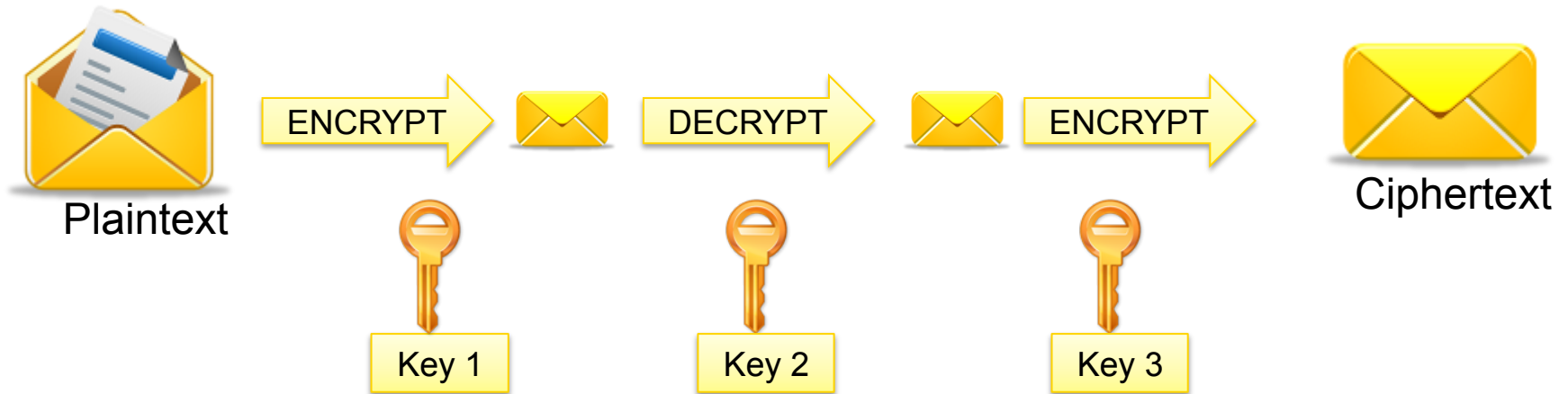
# Triple DES

- 3DES (Triple DES) – a block cipher that applies DES three times to each data block
- Uses a key bundle comprising of three DES keys (K1, K2, K3), each with 56 bits excluding parity.
- DES encrypts with K1, decrypts with K2, then encrypts with K3

$$C_i = E_{K3}(D_{K2}(E_{K1}(P_i)))$$

- Disadvantage: very slow

# 3DES: Illustration



- Note:
  - If  $\text{Key1} = \text{Key2} = \text{Key3}$ , this is similar to DES
  - Usually,  $\text{Key1} = \text{Key3}$

# Advanced Encryption Standard (AES)

- Published in November 2001
- Symmetric block cipher
- Has a fixed block size of 128 bits
- Has a key size of 128, 192, or 256 bits
- Based on Rijndael cipher which was developed by Joan Daemen and Vincent Rijmen
- Better suited for high-throughput, low latency environments

# Rivest Cipher

| RC Algorithm | Description   |
|--------------|---|
| RC2          | Variable key-sized cipher used as a drop in replacement for DES                                 |
| RC4          | Variable key sized stream cipher; Often used in file encryption and secure communications (SSL) |
| RC5          | Variable block size and variable key length; uses 64-bit block size; Fast, replacement for DES  |
| RC6          | Block cipher based on RC5, meets AES requirement  |

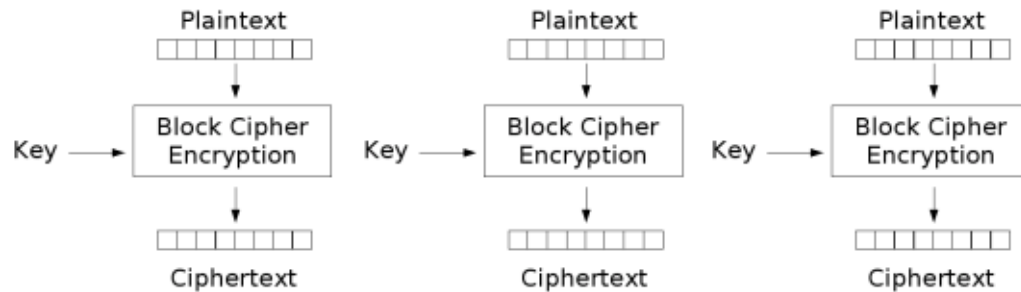
# RC4

- Most widely used stream cipher
- Popularly used in Secure Socket Layer (SSL) and Wired Equivalent Privacy (WEP) protocols
- Although simple and fast, it is vulnerable and can lead to insecure systems

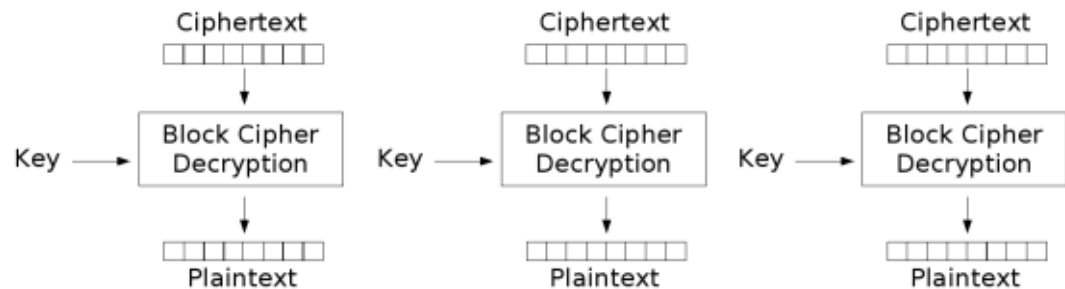
# Block Cipher Modes

- Defines how the block cipher algorithm is applied to the data stream
- Four Basic Modes
  - Electronic Code Book (ECB)
  - Cipher Block Chaining (CBC)
  - Cipher Feedback (CFB)
  - Output Feedback (OFB)

# Electronic Codebook (ECB)

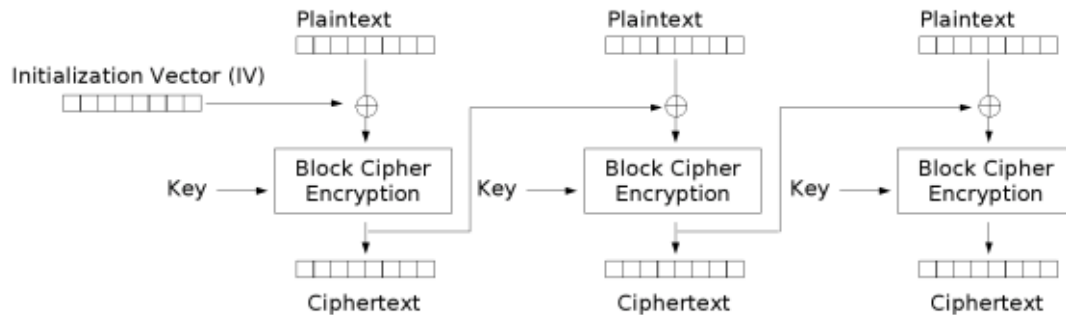


Electronic Codebook (ECB) mode encryption



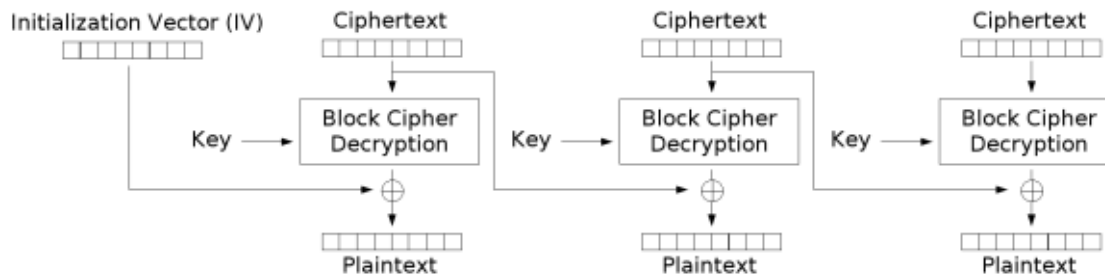
Electronic Codebook (ECB) mode decryption

# Ciphertext Block Chaining (CBC)



Cipher Block Chaining (CBC) mode encryption

$$C_i = E_k(P_i \oplus C_{i-1}), C_0 = IV$$

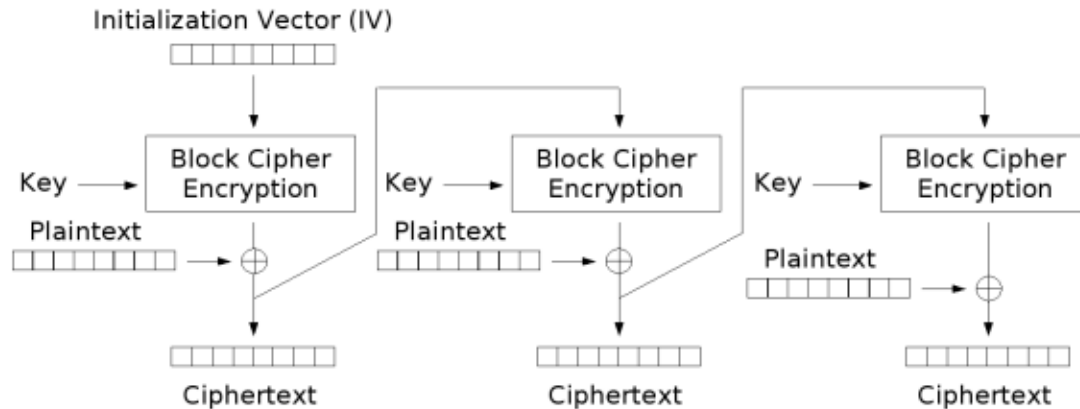


Cipher Block Chaining (CBC) mode decryption

$$P_i = D_k(C_i) \oplus C_{i-1}, C_0 = IV$$



# Cipher Feedback (CFB)

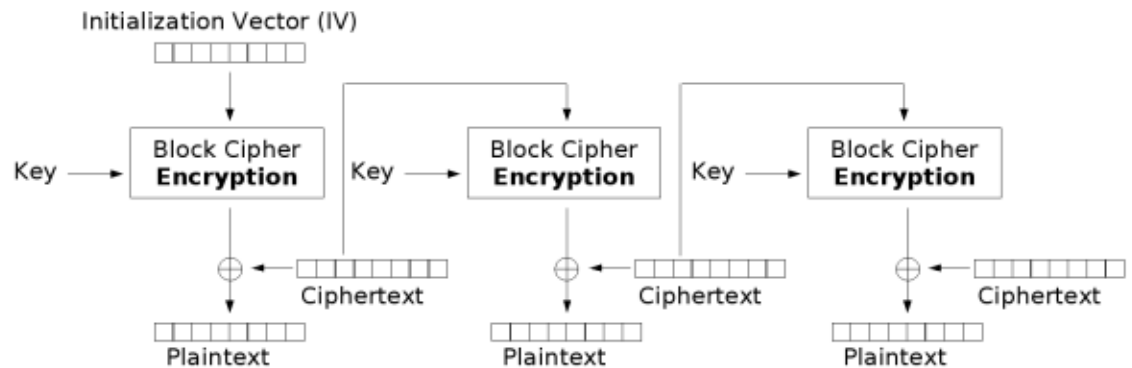


Cipher Feedback (CFB) mode encryption

$$C_i = E_k(C_{i-1}) \oplus P_i$$

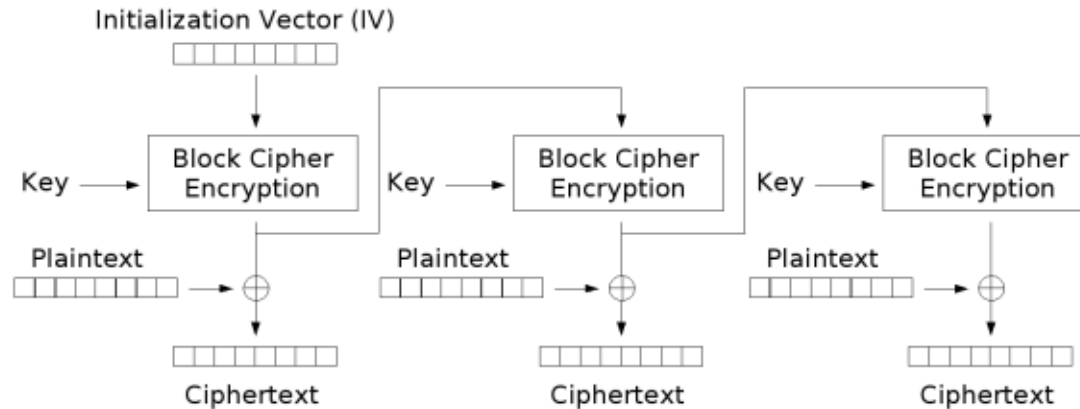
$$P_i = E_k(C_{i-1}) \oplus C_i$$

$$C_o = IV$$

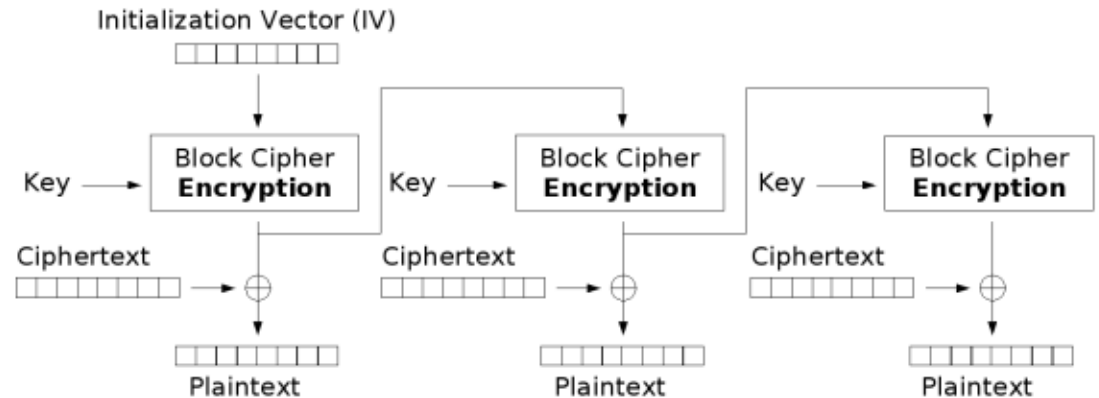


Cipher Feedback (CFB) mode decryption

# Output Feedback (OFB)



Output Feedback (OFB) mode encryption



Output Feedback (OFB) mode decryption

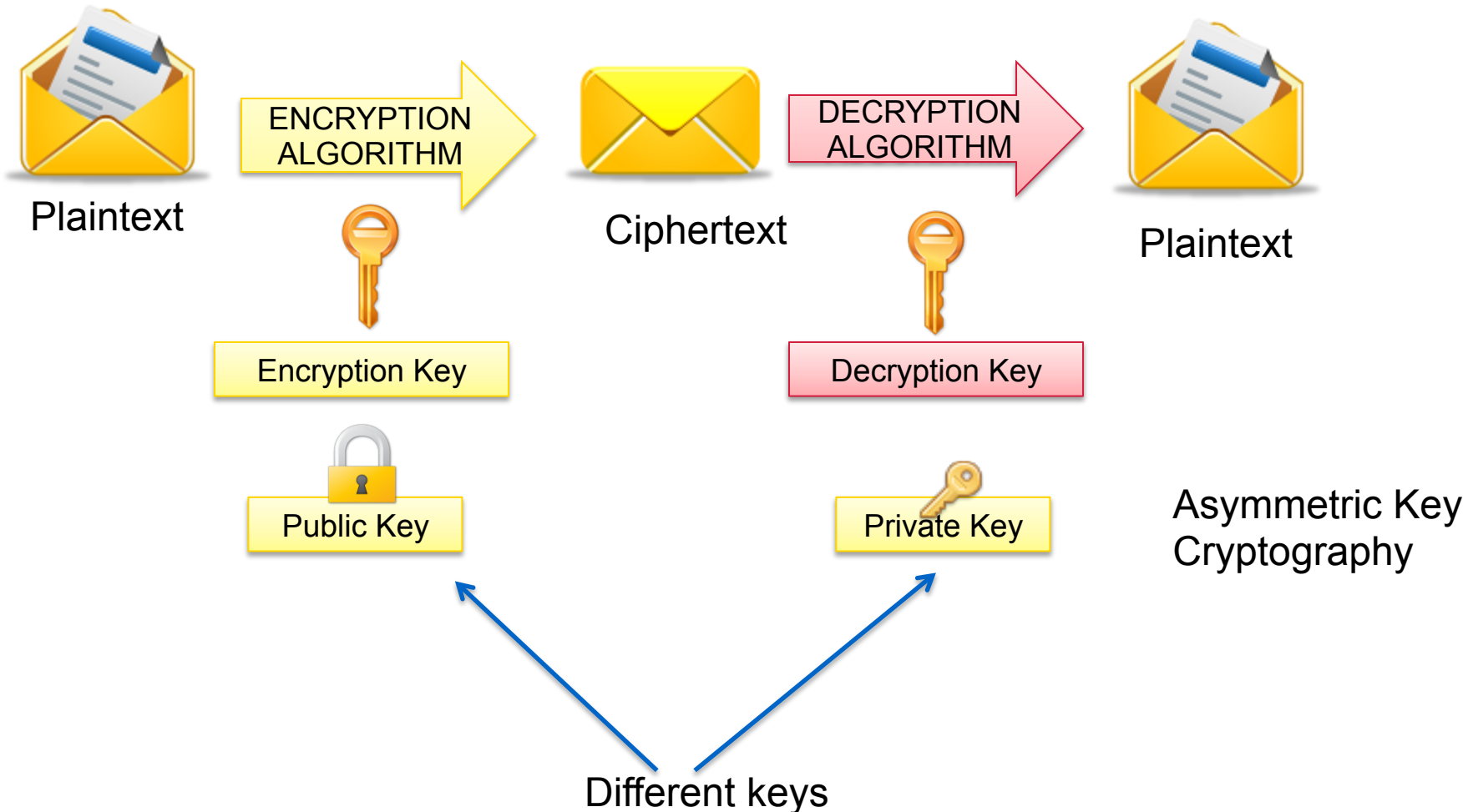
# Selecting a Block Cipher Mode

- Small amounts of truly random data: ECB
  - Example: randomly generated keying material
  - Other modes can be used but ECB is most efficient
- Protocols with crypto integrity protection: CBC, CFB, OFB
- Arbitrary communications with arbitrary data: CBC, CFB
  - Repeated plaintext data is obscured
  - Constantly changing encryption keys defeat differential cryptanalysis attacks

# Asymmetric Key Algorithm

- Also called public-key cryptography
  - Keep private key private
  - Anyone can see public key
- separate keys for encryption and decryption (public and private key pairs)
- Examples of asymmetric key algorithms:
  - RSA, DSA, Diffie-Hellman, El Gamal, Elliptic Curve and PKCS

# Asymmetric Encryption



# Asymmetric Key Algorithm

- RSA – the first and still most common implementation
- DSA – specified in NIST's Digital Signature Standard (DSS), provides digital signature capability for authentication of messages
- Diffie-Hellman – used for secret key exchange only, and not for authentication or digital signature
- ElGamal – similar to Diffie-Hellman and used for key exchange
- PKCS – set of interoperable standards and guidelines

# Symmetric vs. Asymmetric Key

## Symmetric

generally fast  
Same key for both encryption and decryption

## Asymmetric

Can be 1000 times slower  
Uses two different keys (public and private)  
Decryption key cannot be calculated from the encryption key  
Key lengths: 512 to 4096 bits  
Used in low-volume

# Hash Functions

- produces a condensed representation of a message (hashing)
- The fixed-length output is called the hash or message digest
- A hash function takes an input message of arbitrary length and outputs fixed-length code. The fixed-length output is called the hash, or the message digest, of the original input message.
- A form of signature that uniquely represents the data
- Uses:
  - Verifying file integrity - if the hash changes, it means the data is either compromised or altered in transit.
  - Digitally signing documents
  - Hashing passwords



# Hash Functions

- Message Digest (MD) Algorithm
  - Outputs a 128-bit fingerprint of an arbitrary-length input
- Secure Hash Algorithm (SHA)
  - SHA-1 produces a 160-bit message digest similar to MD5
  - Widely-used on security applications (TLS, SSL, PGP, SSH, S/MIME, IPsec)
  - SHA-256, SHA-384, SHA-512 are also commonly used, which can produce hash values that are 256, 384, and 512-bits respectively
- RIPEMD

# Digital Signature

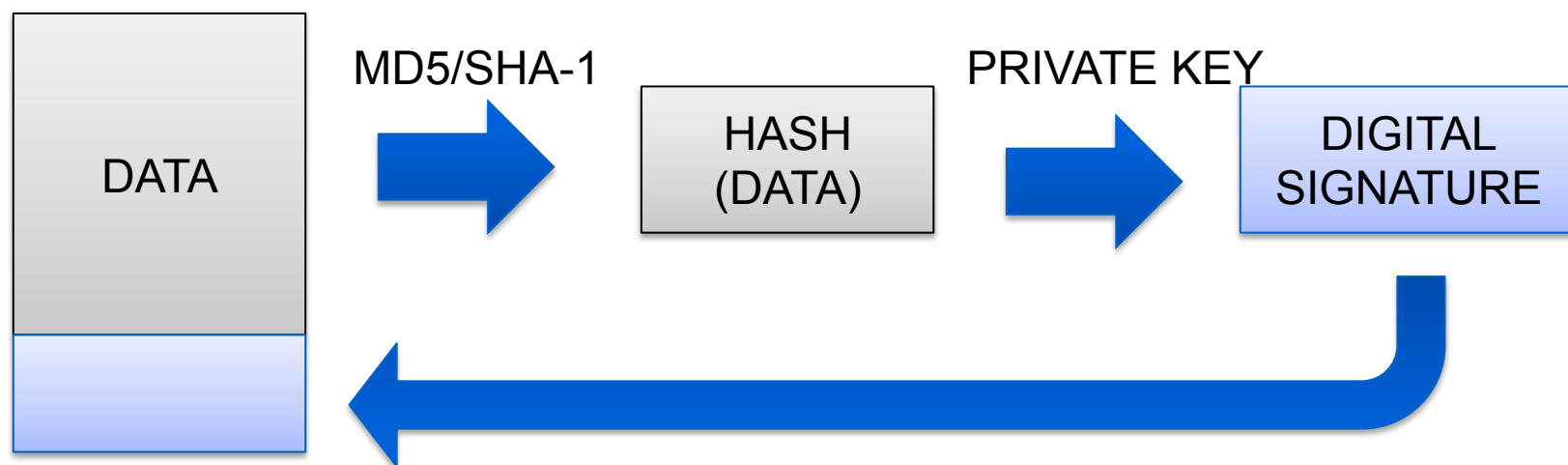
- A digital signature is a message appended to a packet
- The sender encrypts message with own private key instead of encrypting with intended receiver's public key
- The receiver of the packet uses the sender's public key to verify the signature.
- Used to prove the identity of the sender and the integrity of the packet

# Digital Signature

- Two common public-key digital signature techniques:
  - RSA (Rivest, Shamir, Adelman)
  - DSS (Digital Signature Standard)
- Successful verification assures:
  - The packet has not been altered
  - The identity of the sender

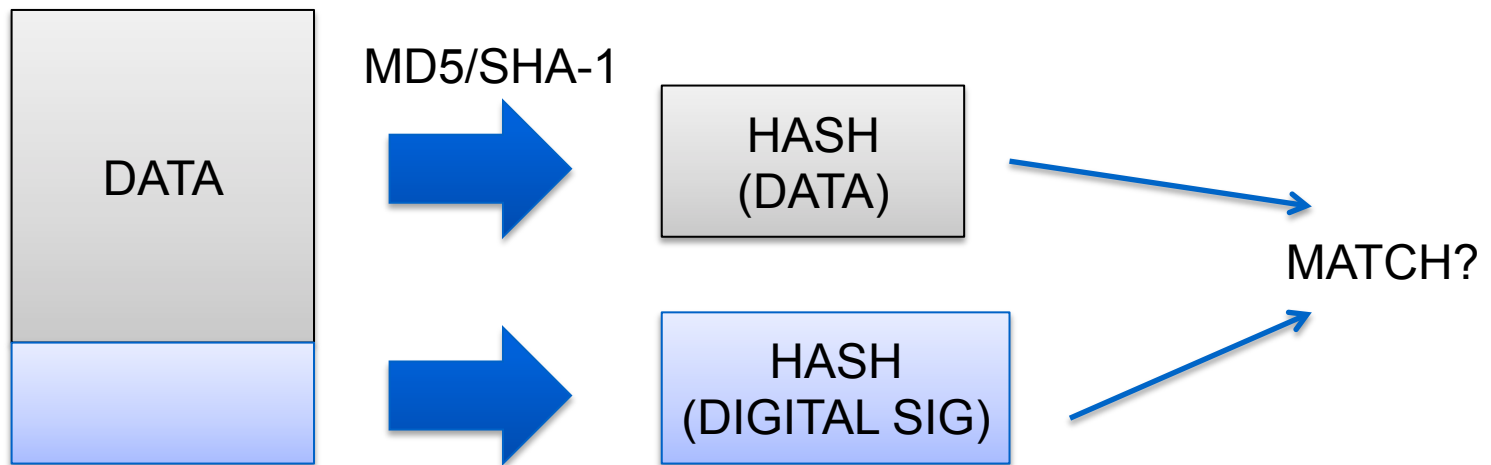
# Digital Signature Process

1. Hash the data using one of the supported hashing algorithms (MD5, SHA-1, SHA-256)
2. Encrypt the hashed data using the sender's private key
3. Append the signature (and a copy of the sender's public key) to the end of the data that was signed)



# Signature Verification Process

1. Hash the original data using the same hashing algorithm
2. Decrypt the digital signature using the sender's public key. All digital signatures contain a copy of the signer's public key
3. Compare the results of the hashing and the decryption. If the values match then the signature is verified. If the values do not match, then the data or signature was probably modified.



# Questions?

# Public Key Infrastructure

Network Security Workshop

# Overview

- Public Key Infrastructure
- Digital Certificates
- Certificate Authority
- RPKI Introduction



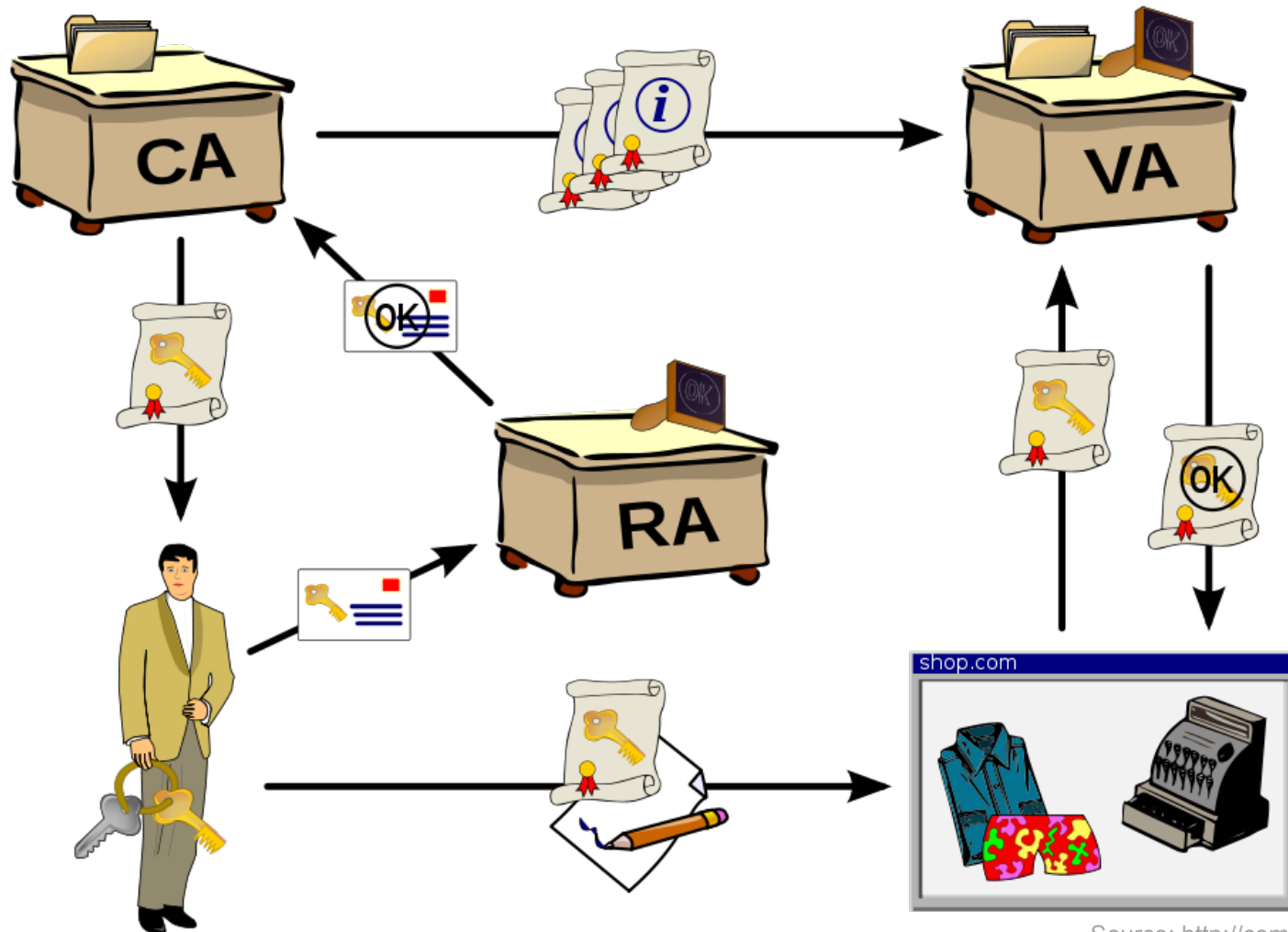
# Public Key Infrastructure

- Framework that builds the network of trust
- Combines public key cryptography, digital signatures, to ensure confidentiality, integrity, authentication, nonrepudiation, and access control
- Protects applications that require high level of security

# Functions of a PKI

- Registration
- Initialization
- Certification
- Key pair recovery
- Key generation
- Key update
- Cross-certification
- Revocation

# Public Key Infrastructure



Source: <http://commons.wikimedia.org>

# Components of a PKI

- Certificate authority
  - The trusted third party
  - Trusted by both the owner of the certificate and the party relying upon the certificate.
- Validation authority
- Registration authority
  - For big CAs, a separate RA might be necessary to take some work off the CA
  - Identity verification and registration of the entity applying for a certificate
- Central directory

# Certificates

- Public key certificates bind public key values to subjects
- A trusted certificate authority (CA) verifies the subject's identity and digitally sign each certificate
  - Validates
- Has a limited valid lifetime
- Can be used using untrusted communications and can be cached in unsecured storage
  - Because client can independently check the certificate's signature
- Certificate is NOT equal to signature
  - It is implemented using signature
- Certificates are static
  - If there are changes, it has to be re-issued

# Digital Certificate

- Digital certificate – basic element of PKI; secure credential that identifies the owner
- Also called public key certificate



# Digital Certificate

- deals with the problem of
  - Binding a public key to an entity
  - A major legal issue related to eCommerce
- A digital certificate contains:
  - User's public key
  - User's ID
  - Other information e.g. validity period
- Certificate examples:
  - X509 (standard)
  - PGP (Pretty Good Privacy)
  - Certificate Authority (CA) creates and digitally signs certificates

# Digital Certificate

- To obtain a digital certificate, Alice must:
  - Make a certificate signing request to the CA
  - Alice sends to CA:
    - Her identifier  $IdA$
    - Her public key  $KA\_PUB$
    - Additional information
- CA returns Alice's digital certificate, cryptographically binding her identity to public key:
  - $CertA = \{IDA, KA\_PUB, info, SigCA(IDA,KA\_PUB,info)\}$

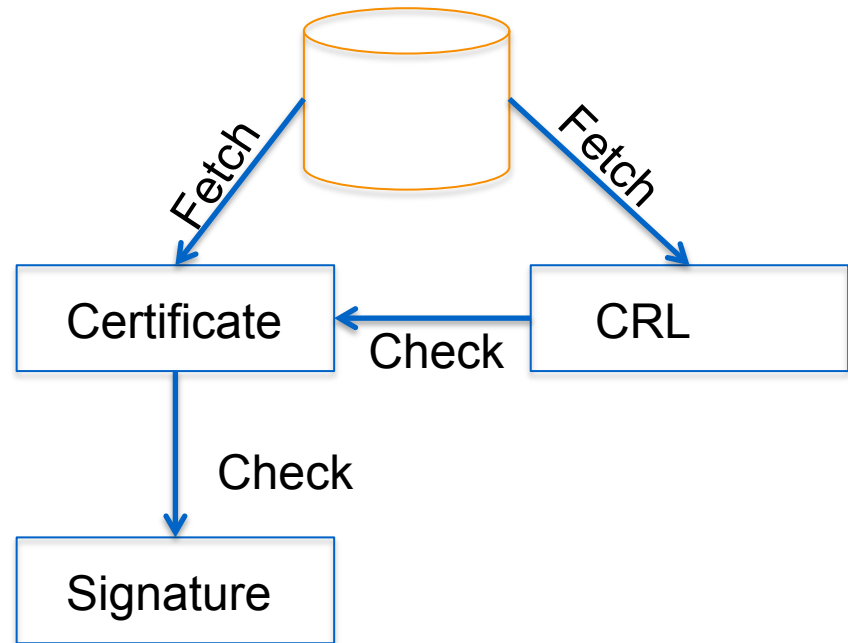


# X.509

- An ITU-T standard for a public key infrastructure (PKI) and Privilege Management Infrastructure (PMI)
- Assumes a strict hierarchical system of Certificate Authorities (CAs)
- RFC 1422 – basis of X.509-based PKI
- Current version X.509v3 provides a common baseline for the Internet
- Structure of a Certificate, certificate revocation (CRLs)

# X.509 Certificate Usage

- Fetch certificate
- Fetch certificate revocation list (CRL)
- Check the certificate against the CRL
- Check signature using the certificate



# Every certificate contains...

- Body of the certificate
  - Version number, serial number, names of the issuer and subject
  - Public key associated with the subject
  - Expiration date (not before, not after)
  - Extensions for additional tributes
- Signature algorithm
  - Used by the CA to sign the certificate
- Signature
  - Created by applying the certificate body as input to a one-way hash function. The output value is encrypted with the CA's private key to form the signature value

# Certificate Authority

- Issuer and signer of the certificate
- Trusted (Third) Party
  - Based on trust model
  - Who to trust?
- Types:
  - Enterprise CA
  - Individual CA (PGP)
  - Global CA (such as VeriSign)
- Functions:
  - Enrolls and Validates Subscribers
  - Issues and Manages Certificates
  - Manages Revocation and Renewal of Certificates
  - Establishes Policies & Procedures

# Certificate Revocation Lists

- CA periodically publishes a data structure called a certificate revocation list (CRL).
- Described in X.509 standard.
- Each revoked certificate is identified in a CRL by its serial number.
- CRL might be distributed by posting at known Web URL or from CA's own X.500 directory entry.

# Questions?

# Resource Registration

Network Security Workshop

# Resource Registration

- As part of your membership agreement with APNIC, all Members are required to register their resources in the APNIC database.
  - First allocation/assignment, APNIC will create:
    - Inetnum or inet6num object
    - Autnum object (if you received an ASN)
    - Maintainer object (to protect your data)
    - Role object
- Members must keep records up to date:
  - Whenever there is a change in contacts
  - When new resources are received
  - When resources are sub-allocated or assigned



# What is the APNIC Database?

- Public network management database
  - Operated by Internet Registries
    - Public data only
    - (For private data, please see “Privacy of customer assignment” module)
- Tracks network resources
  - IP addresses, ASNs, Reverse Domains, Routing policies
- Records administrative information
  - Contact information (persons/roles)
  - Authorization

# Whois Database Query - Clients

- Standard whois client
  - Included with many Unix distributions
  - RIPE extended whois client
  - <http://ftp.apnic.net/apnic/dbase/tools/ripe-dbase-client.tar.gz>
- Query via the APNIC website
  - <http://www.apnic.net/apnic-bin/whois2.pl>
- Query clients – MS Windows etc

# Object Types

## OBJECT

- person
- role
- inetnum
- Inet6num
- aut-num
- domain
- route
- mntner
- mnt-irt

## PURPOSE

contact persons

contact groups/roles

IPv4 addresses

IPv6 addresses

Autonomous System number

reverse domains

prefixes being announced

(maintainer) data protection

Incident Response Team

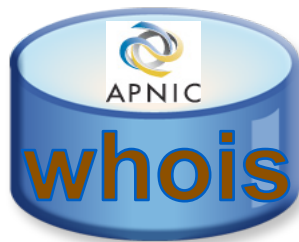


<http://www.apnic.net/db/>

# Database Object

- An object is a set of attributes and values
- Each attribute of an object...
  - Has a value
  - Has a specific syntax
  - Is mandatory or optional
  - Is single or multi-valued
- Some attributes are ...
  - Primary (unique) keys
  - Lookup keys for queries
  - Inverse keys for queries
- Object templates illustrate this structure

# Inter-Related Objects



inetnum:

202.64.10.0 – 202.64.10.255

...

admin-c: EC196-AP

tech-c: ZU3-AP

...

mnt-by: MAINT-WF-EX  
*IPv4 addresses*

...

mntner:  
MAINT-WF-EX

...

...

*Data protection*

person/role:

...

nic-hdl: EC196-AP

... *Contact info*

person/role:

...

nic-hdl: ZU3-AP

... *Contact info*

# New Members

- If you are receiving your first allocation or assignment, APNIC will create the following objects for you:
  - role object
  - inetnum or inet6num object
  - maintainer object (to protect your data)
  - aut-num object (if you received an ASN)
- Information is taken from your application for resources and membership

# Inetnum / Inet6num Objects

- Contains IP allocation and assignment information
- APNIC creates an inetnum (or inet6num) object for each allocation or assignment they make to the Member
- All members must create inetnum (or inet6num) objects for each sub-allocation or assignment they make to customers

# Whois – Inet6num Example

```
inet6num:      2001:0DF0:000A::/48
netname:       APNIC-TRAININGIPv6-DC-20080424
descr:        APNIC Training IPv6 Address for data centre
country:      AU
admin-c:      AT480-AP
tech-c:       AT480-AP
status:       ASSIGNED PORTABLE
mnt-by:       MAINT-AU-APNICTRAINING
mnt-routes:   MAINT-AU-APNICTRAINING
remarks:      -+-+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
remarks:      This object can only be updated by APNIC hostmasters.
remarks:      To update this object, please contact APNIC
remarks:      hostmasters and include your organisation's account
remarks:      name in the subject line.
remarks:      -+-+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
changed:      hm-changed@apnic.net 20080424
changed:      hm-changed@apnic.net 20100818
source:       APNIC
```

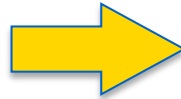


# Person Object

- Represents a contact person for an organization
  - Every Member must have at least one contact person registered
  - Large organizations often have several contacts for different purposes
- Is referenced in other objects
- Has a nic-hdl
  - Eg. EC17-AP

# What is a 'nic-hdl' ?

- Unique identifier for a person or role
- Represents a person or role object
  - Referenced in objects for contact details
    - (inetnum, aut-num, domain...)
  - format: <XXXX-AP>
    - Eg: EC196-AP



## **Person: Eric Chu**

address: ExampleNet Service Provider  
address: Level 1 33 Park Road Milton  
address: Wallis and Futuna Islands  
country: WF  
phone: +680-368-0844  
fax-no: +680-367-1797  
e-mail: echu@example.com

## **nic-hdl: EC196-AP**

mnt-by: MAINT-WF-EX  
changed: echu@example.com 20020731  
source: APNIC

# Role Object

- Represents a group of contact persons for an organization
  - Eases administration
  - Can be referenced in other objects instead of the person objects for individuals
- Also has a nic-hdl
  - Eg. HM20-AP

NOC Role



Admin Role



# How a Role Object Works

- Role Object is used instead of a Person Object as a reference in other objects
- If a contact leaves the organization:
  - New Person Object is created
  - The nic-hdl of the new contact replaces nic-hdl of the old person in the Role Object
  - Old Person Object is deleted
- This means only a single replacement is required instead of many

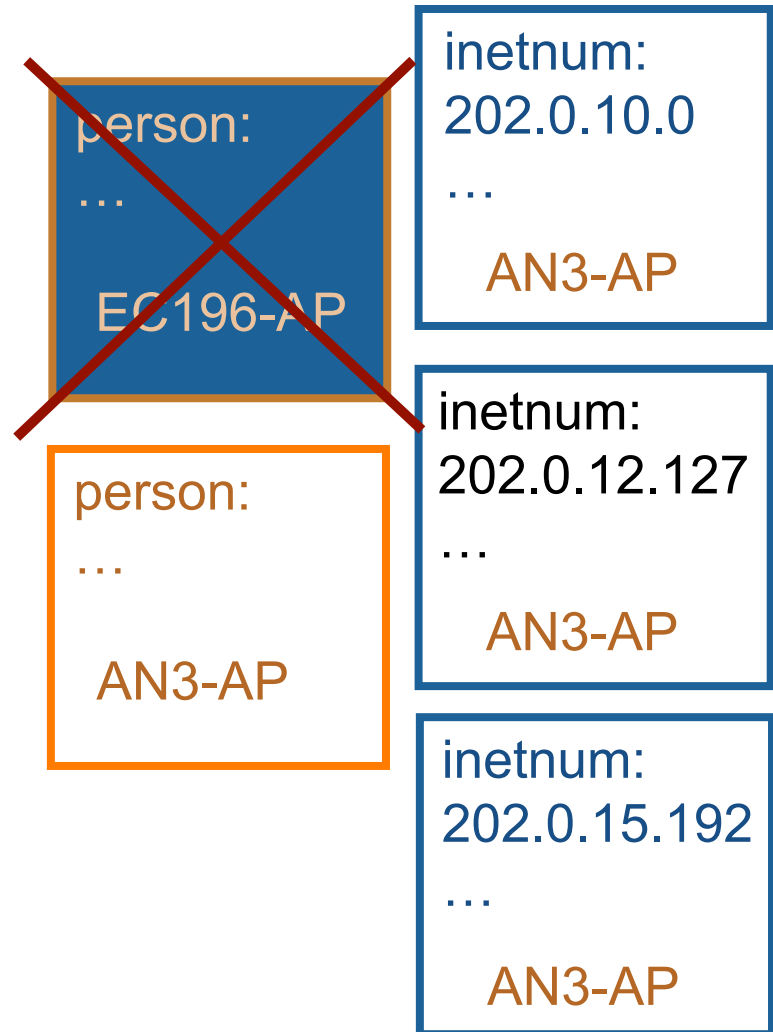
# Replacing Contacts in the DB

## - Using Person Objects

*E. Chu is leaving my organization.*

*A. Nagali is replacing him.*

1. Create a Person Object for new contact (**E. Chu**)
2. Find all objects containing old contact (**E. Chu**)
3. Update all objects, replacing old contact (EC196-AP) with new contact (AN3-AP)
4. Delete old contact's (EC196-AP) Person Object



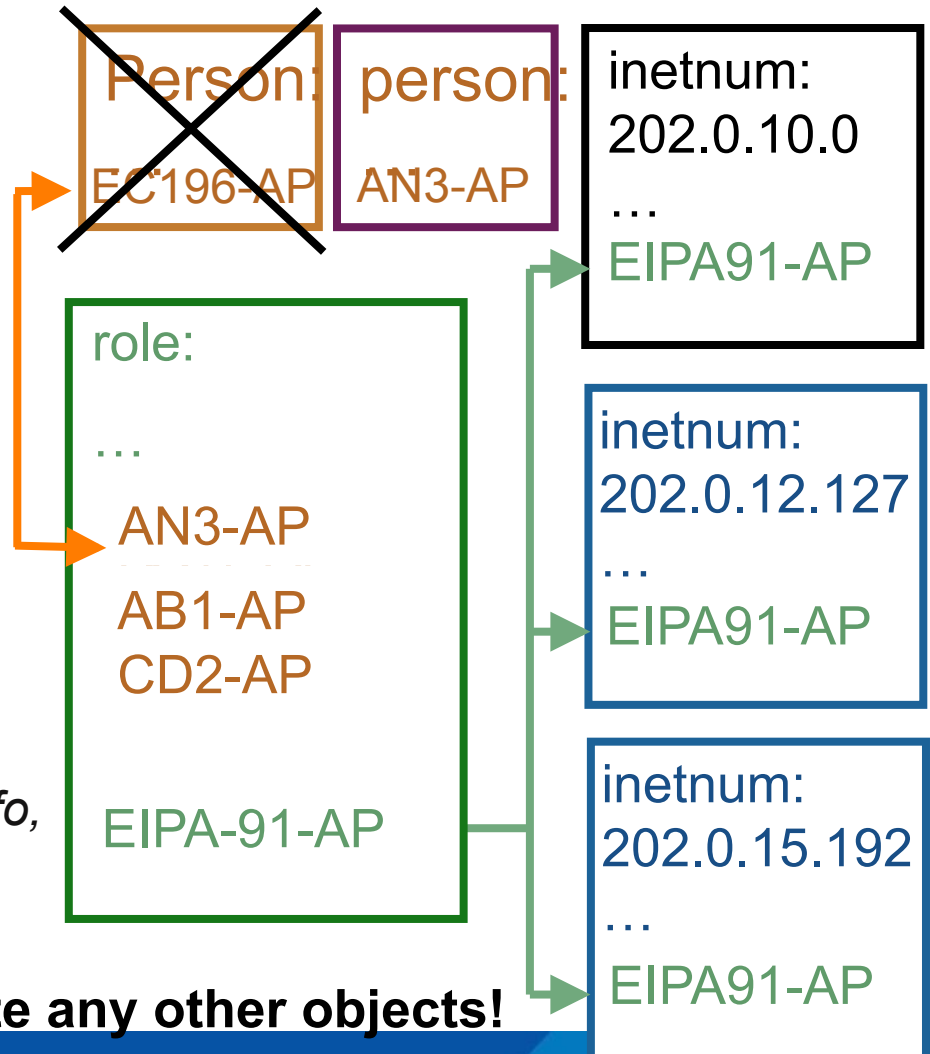
# Replacing Contacts in the DB – Using a Role Object

*E. Chu is leaving my organization.*

*A. Nagali is replacing him.*

1. Create a Person Object for new contact (A. Nagali)
2. Replace old contact (EC196-AP) with new contact (AN3-AP) in Role Object
3. Delete old contact's Person Object.

*My Role Object contains all contact info, that is referenced in all my objects.*



**No need to update any other objects!**

# Whois - Role vs Person Objects

```
% APNIC found the following authoritative answer from: whois.apnic.net

% [whois.apnic.net node-1]
% Whois data copyright terms    http://www.apnic.net/db/dbcopyright.html

role:                APNIC Training
address:           Level 1 33 Park Rd. 4064 Milton, Brisbane
country:           AU
phone:             +617 38583100
fax-no:            +617 38583199
e-mail:            training@apnic.net
admin-c:           AA196-AP
tech-c:            AA196-AP
nic-hdl:           AT480-AP
mnt-by:            MAINT-AU-APNICTRAINING
changed:           hm-changed@apnic.net 20080424
source:            APNIC

person:            Amante Alvaran
nic-hdl:           AA196-AP
e-mail:            amante@apnic.net
address:           Level 1 33 Park Road Milton
address:           Brisbane QLD Australia
phone:             +617-3858-3100
fax-no:            +617-3858-3199
country:           AU
mnt-by:            MAINT-AU-APNICTRAINING
changed:           hm-changed@apnic.net 20051025
changed:           hm-changed@apnic.net 20080424
source:            APNIC
```

# IRT Object

- Incident Response Team (IRT)
  - Dedicated abuse handling teams (not netops)
- Implemented in Nov 2010 through Prop-079
- Abuse contact information
- Mandatory object reference in inetnum, inet6num, and aut-num objects



# IRT Object

- Why provide abuse contact
  - Dedicated contacts or team that specifically resolve computer security incidents
  - Efficient and accurate response
  - Stops the tech-c and admin-c from getting abuse reports
  - Shared response to address abuse

# Database Protection - Maintainers

- protects other objects in the APNIC Whois Database
- used to prevent unauthorized persons from changing the details in whois
- Multiple levels of maintainers exist in a hierarchical manner
  - Maint-by
  - Maint-lower
- Applied to any object created directly below that maintainer object

# Database Protection

- **Authorisation**

- “mnt-by” references as maintainer object
  - Can be found in all database objects
  - “mnt-by” should be used with every object

- **Authentication**

- Updates to an object must pass the authentication rule specified by its maintainer
- Authentication methods (using ‘auth’ attribute)
  - Crypt-PW
  - PGP – GNUPG
  - MD5

# Database Protection

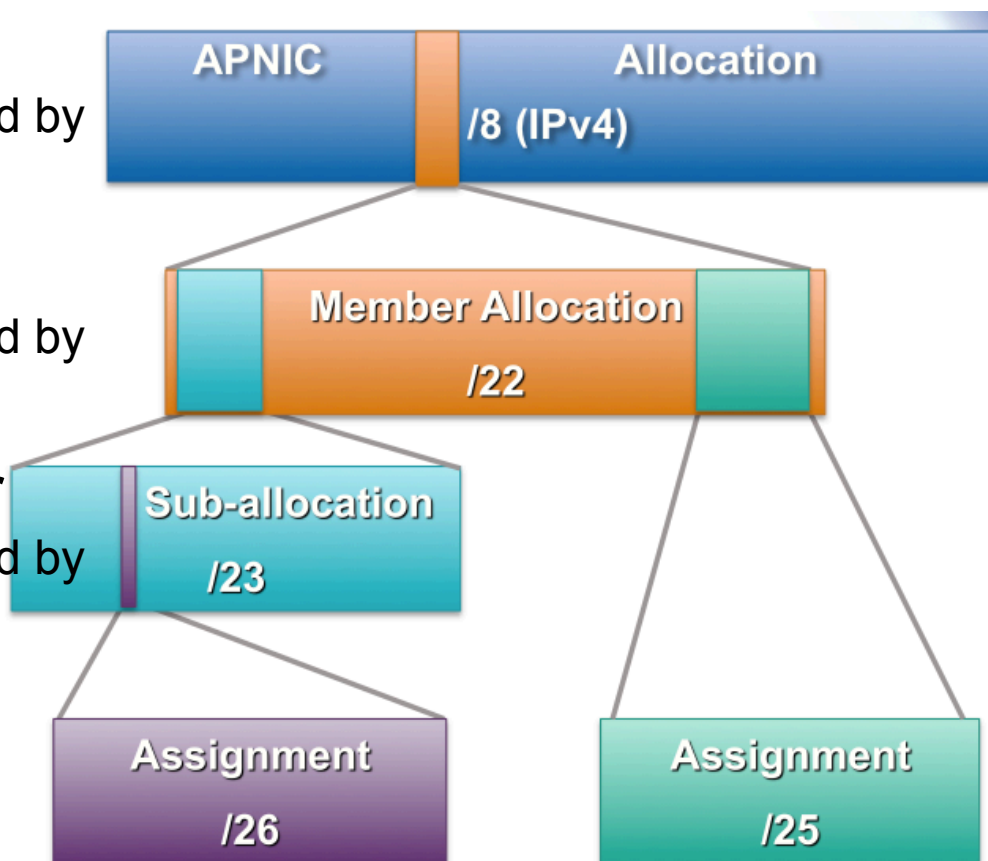
## Maintainer Object

```
mntner:          MAINT-AU-APNICTRAINING
descr:           APNIC Training
country:         AU
admin-c:         AA196-AP
tech-c:          AA196-AP
auth:            MD5-PW $1$FUrnj.4g$sIyzbkZj2XJoDanL/ndXN0
mnt-by:          MAINT-AU-APNICTRAINING
upd-to:          amante@apnic.net
referral-by:     APNIC-HM
changed:         hm-changed@apnic.net 20080424
changed:         hm-changed@apnic.net 20090325
changed:         hm-changed@apnic.net 20090403
changed:         hm-changed@apnic.net 20090702
changed:         hm-changed@apnic.net 20091111
changed:         hm-changed@apnic.net 20091217
changed:         hm-changed@apnic.net 20100528
source:          APNIC
```



# Maintainer Hierarchy Diagram

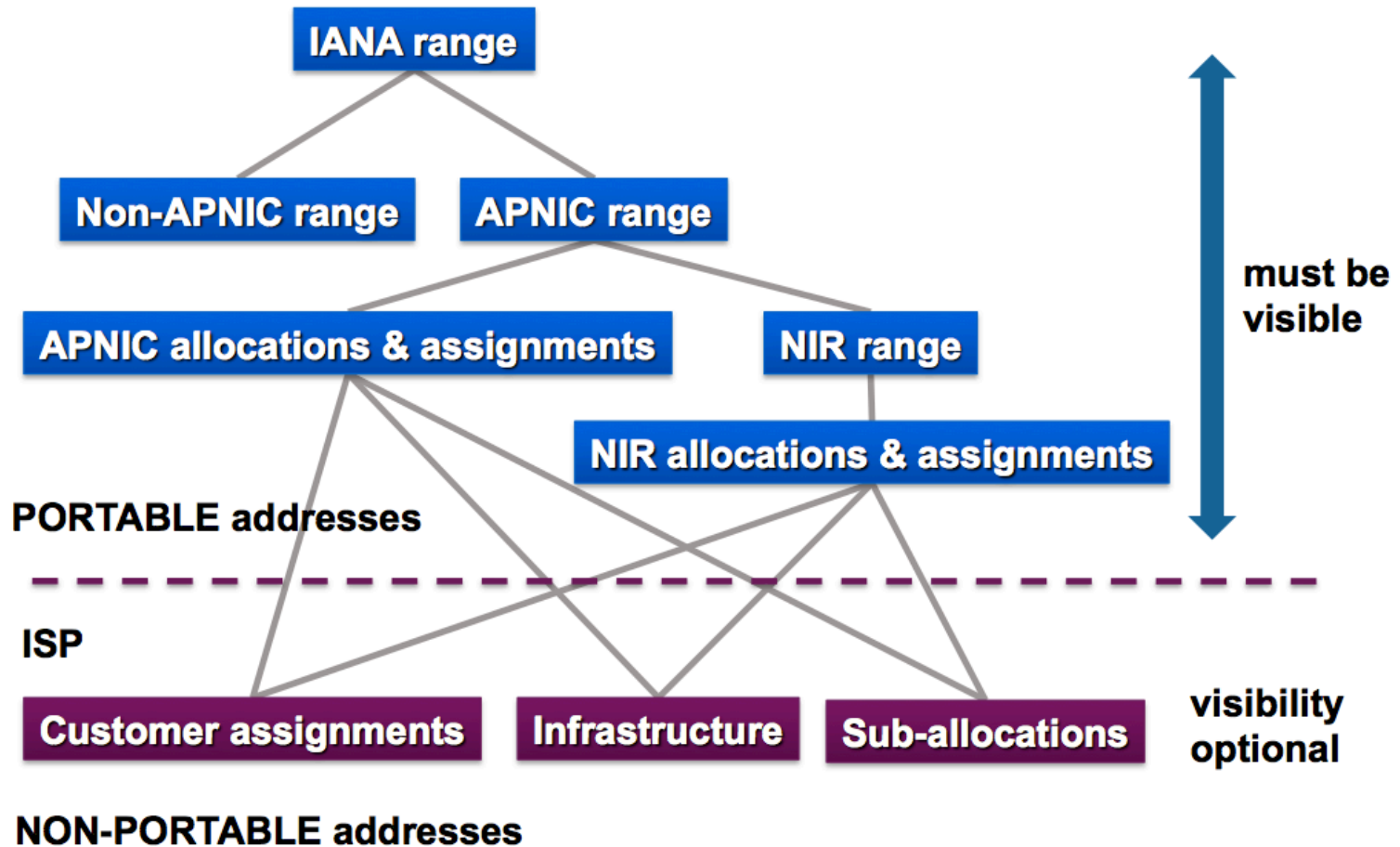
- **Allocated to APNIC**
  - Maint-by can only be changed by IANA
- **Allocated to Member**
  - Maint-by can only be changed by APNIC
- **Sub-allocated to Customer**
  - Maint-by can only be changed by Members



# Customer Privacy

- Privacy issues
  - Concerns about publication of customer information
  - Increasing government concern
- APNIC legal risk
  - Legal responsibility for accuracy and advice
  - Damages incurred by maintaining inaccurate personal data
- Customer data is hard to maintain
- Customer assignment registration is still mandatory

# What Needs to be Visible?

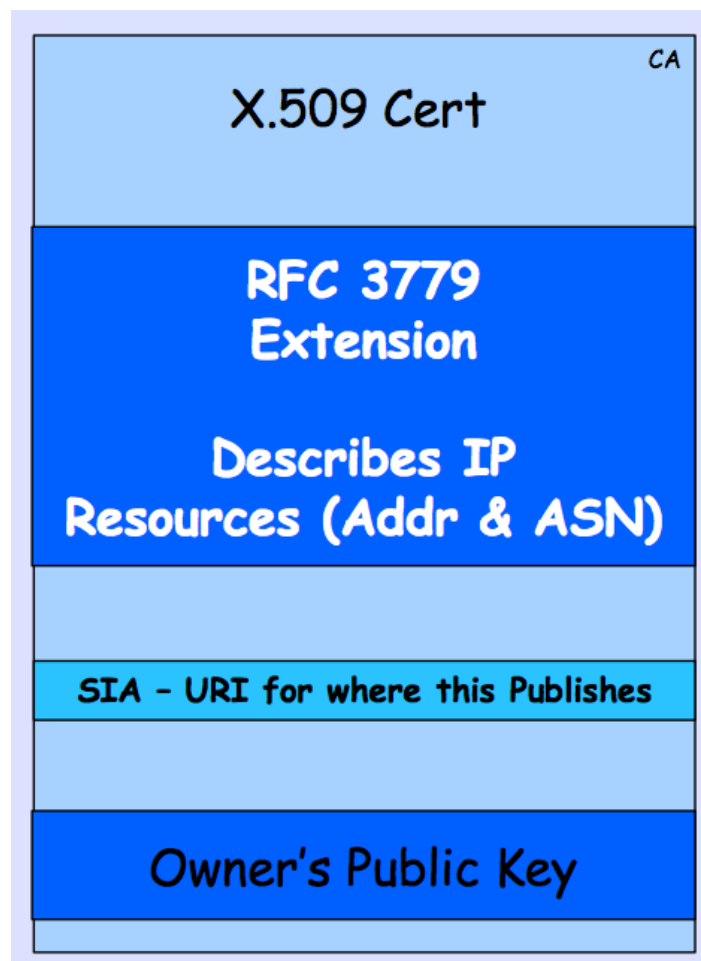


# RPKI

- Resource Public Key Infrastructure
- verify the authenticity of data that has been digitally signed by the originator of the data
- Based on the X.509 certificate format (RFC5280) and extended by RFC3779
- RPKI is in the process of standardization through the Secure Inter-Domain Routing (SIDR) working group.



# X.509 Certificate + 3779 Ext



# Resource Certification

- RIRs have been developing a new service for their members
- APNIC has now launched Resource Certification for the AP region
- The goal is to improve the security of inter-domain routing and augmenting the information published in the APNIC Whois Database

# Terminologies

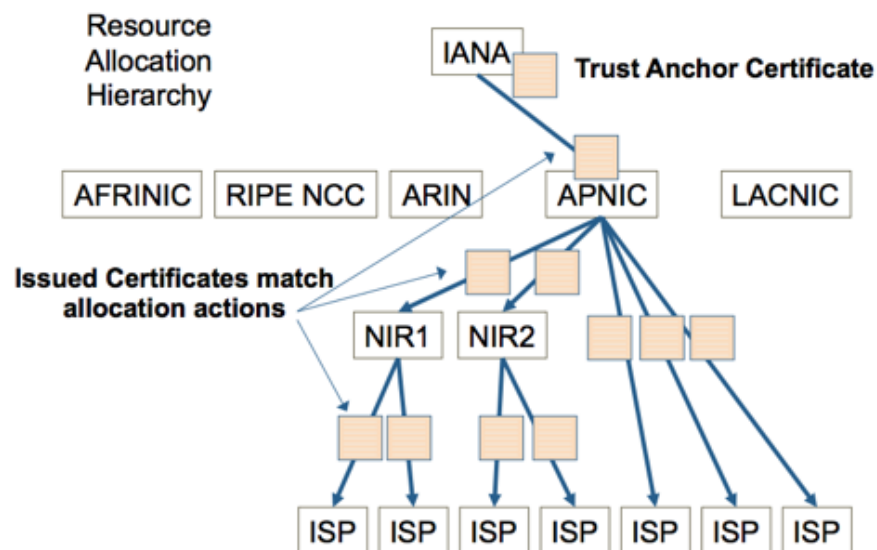
- Resource holders include:
  - Regional Internet Registries (RIRs)
  - Local Internet Registries (LIRs)
  - Internet Service Providers (ISPs)
  - End-user organizations
- Internet resources are:
  - IPv4 and IPv6 address blocks
  - Autonomous System (AS) numbers

# Resource Certification Benefits

- Routing information corresponds to properly delegated address resources
- Resource Certification gives resource holders proof that they hold certain resources
- Resource holders can attest to those resources when distributing them
- Resource Certification is a highly robust means of preventing the injection of false information into the Internet's routing system.

# Resource Public Key Infrastructure

- RPKI hierarchy is based on the administrative resource allocation hierarchy
  - IANA → RIRs → LIRs → end-users
- Main components:
  - Trust anchors
  - ROAs
  - validators



# Route Origin Attestations (ROAs)

- allow entities to verify that an autonomous system (AS) has been given permission by an IP address block holder to advertise routes to one or more prefixes within that block. We call this mechanism a Route Origin Attestation (ROA).
- The certificate holder uses their private key to sign an ROA for specific IP address blocks to be routed by a specific AS, and this can be tested and verified by the public key, and the certificate hierarchy.
  - Example: the ROA might state the following: "ISP 4 permits AS 65000 to originate a route for the prefix 192.2.200.0/24"

# More Info on RPKI

- RPKI Origin Validation, Randy Bush
- Securing BGP with BGPsec, Geoff Huston and Randy Bush

# Questions?



# IP Security (IPSec)

Network Security Workshop

# Overview

- Introduction to VPN
- IPSec Fundamentals
- Tunnel and Transport Mode IPSec
- Architecture and Components of IPSec
- Internet Key Exchange
- Configuring IPSec for IPv4 and IPv6

# Virtual Private Network

- Creates a secure tunnel over a public network
  - Client to firewall
  - Router to router
  - Firewall to firewall
- Uses the Internet as the public backbone to access a secure private network
  - Remote employees can access their office network
- Two types:
  - Remote access
  - Site-to-site VPN

# Virtual Private Network

- There are three basic types of VPN:
  - **Remote access VPNs** or virtual private dial-up networks (VPDNs)
  - **Site-to-site VPN**, where multiple fixed sites are connected over a public network i.e. Internet
  - **Point-to-Point VPN**, these are also referred to as "leased-line VPNs." Two or more networks are connected using a dedicated line from an ISP. These lines can be packet or circuit switched. For example, T1's, Metro Ethernet, DS3, ATM or something else

# VPN Implementations

- Hardware
  - Usually a VPN-type router
  - Pros: highest network throughput, plug and play, dual purpose
  - Cons: cost and lack of flexibility
- Software
  - Ideal for two end-points in different organisations
  - Pros: flexible, and low relative cost
  - Cons: lack of efficiency, more labor training required, lower productivity; higher labor costs
- Firewall
  - Pros: cost effective, tri-purpose, hardens the operating system
  - Cons: still relatively costly

# VPN Protocols

- PPTP (Point-to-Point tunneling Protocol)
  - Developed by Microsoft to secure dial-up connections
  - Operates in the data-link layer
- L2F (Layer 2 Forwarding Protocol)
  - Developed by Cisco
  - Similar as PPTP
- L2TP (Layer 2 Tunneling Protocol)
  - IETF standard
  - Combines the functionality of PPTP and L2F
- IPSec (Internet Protocol Security)
  - Open standard for VPN implementation
  - Operates on the network layer

# Advantages of VPN

- Cheaper connection
  - Use the Internet connection instead of a private lease line
- Scalability
  - Flexibility of growth
  - Efficiency with broadband technology
- Availability
  - Available everywhere there is an Internet connection

# Disadvantages of VPN

- VPNs require an in-depth understanding of public network security issues and proper deployment precautions
- Availability and performance depends on factors largely outside of their control
- VPNs need to accommodate protocols other than IP and existing internal network technology



# IPsec

- Provides Layer 3 security (RFC 2401)
  - Transparent to applications (no need for integrated IPsec support)
- A set of protocols and algorithms used to secure IP data at the network layer
- Combines different components:
  - Security associations (SA)
  - Authentication headers (AH)
  - Encapsulating security payload (ESP)
  - Internet Key Exchange (IKE)
- A security context for the VPN tunnel is established via the ISAKMP

# IPsec Standards

- RFC 4301 “The IP Security Architecture”
  - Defines the original IPsec architecture and elements common to both AH and ESP
- RFC 4302
  - Defines authentication headers (AH)
- RFC 4303
  - Defines the Encapsulating Security Payload (ESP)
- RFC 2408
  - ISAKMP
- RFC 5996
  - IKE v2 (Sept 2010)
- RFC 4835
  - Cryptographic algorithm implementation for ESP and AH

# Benefits of IPsec

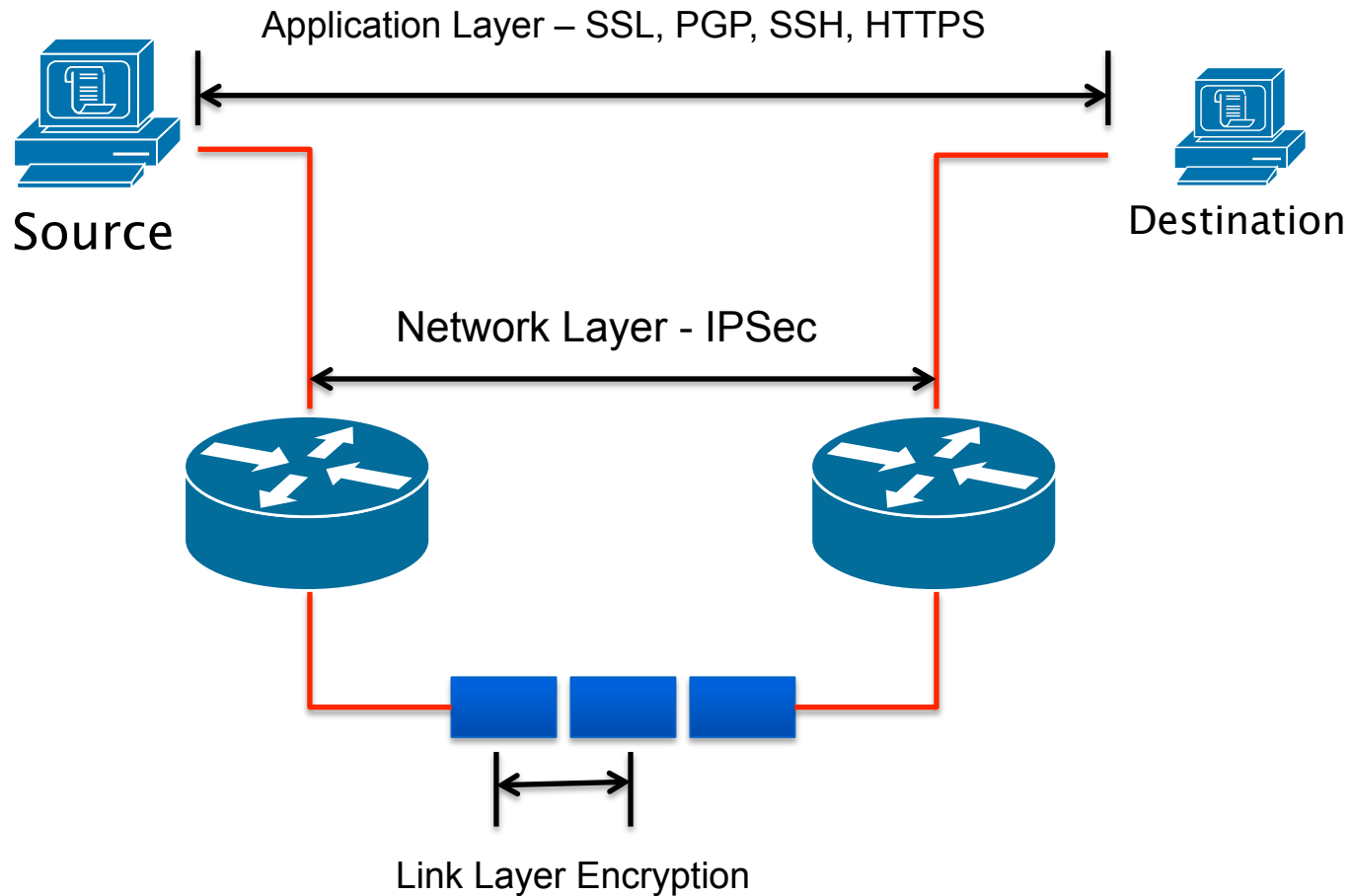
- Confidentiality
  - By encrypting data
- Integrity
  - Routers at each end of a tunnel calculates the checksum or hash value of the data
- Authentication
  - Signatures and certificates
  - All these while still maintaining the ability to route through existing IP networks

“IPsec is designed to provide interoperable, high quality, cryptographically-based security for IPv4 and IPv6” - (RFC 2401)

# Benefits of IPsec

- Data integrity and source authentication
  - Data “signed” by sender and “signature” is verified by the recipient
  - Modification of data can be detected by signature “verification”
  - Because “signature” is based on a shared secret, it gives source authentication
- Anti-replay protection
  - Optional; the sender must provide it but the recipient may ignore
- Key management
  - IKE – session negotiation and establishment
  - Sessions are rekeyed or deleted automatically
  - Secret keys are securely established and authenticated
  - Remote peer is authenticated through varying options

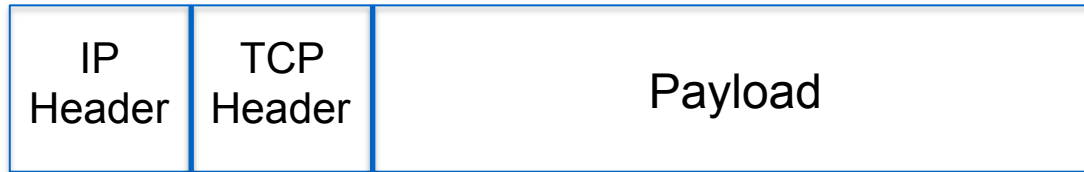
# Different Layers of Encryption



# IPsec Modes

- Tunnel Mode
  - Entire IP packet is encrypted and becomes the data component of a new (and larger) IP packet.
  - Frequently used in an IPsec site-to-site VPN
- Transport Mode
  - IPSec header is inserted into the IP packet
  - No new packet is created
  - Works well in networks where increasing a packet's size could cause an issue
  - Frequently used for remote-access VPNs

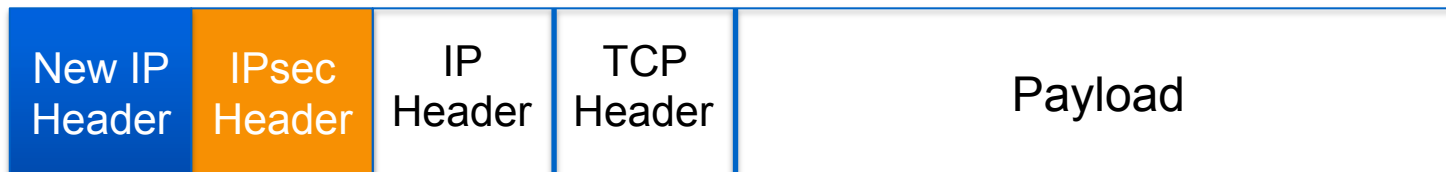
# Tunnel vs. Transport Mode IPsec



Without IPsec

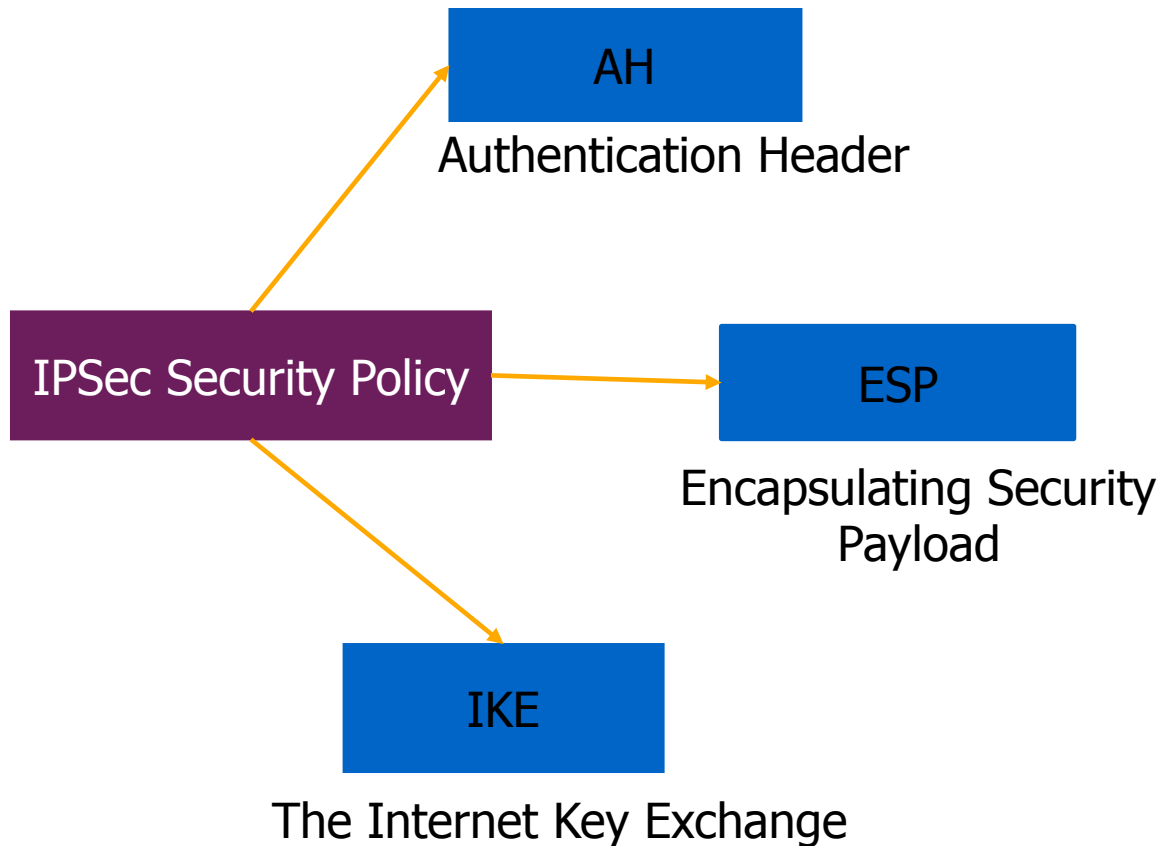


Transport Mode  
IPsec



Tunnel Mode  
IPsec

# IPsec Architecture

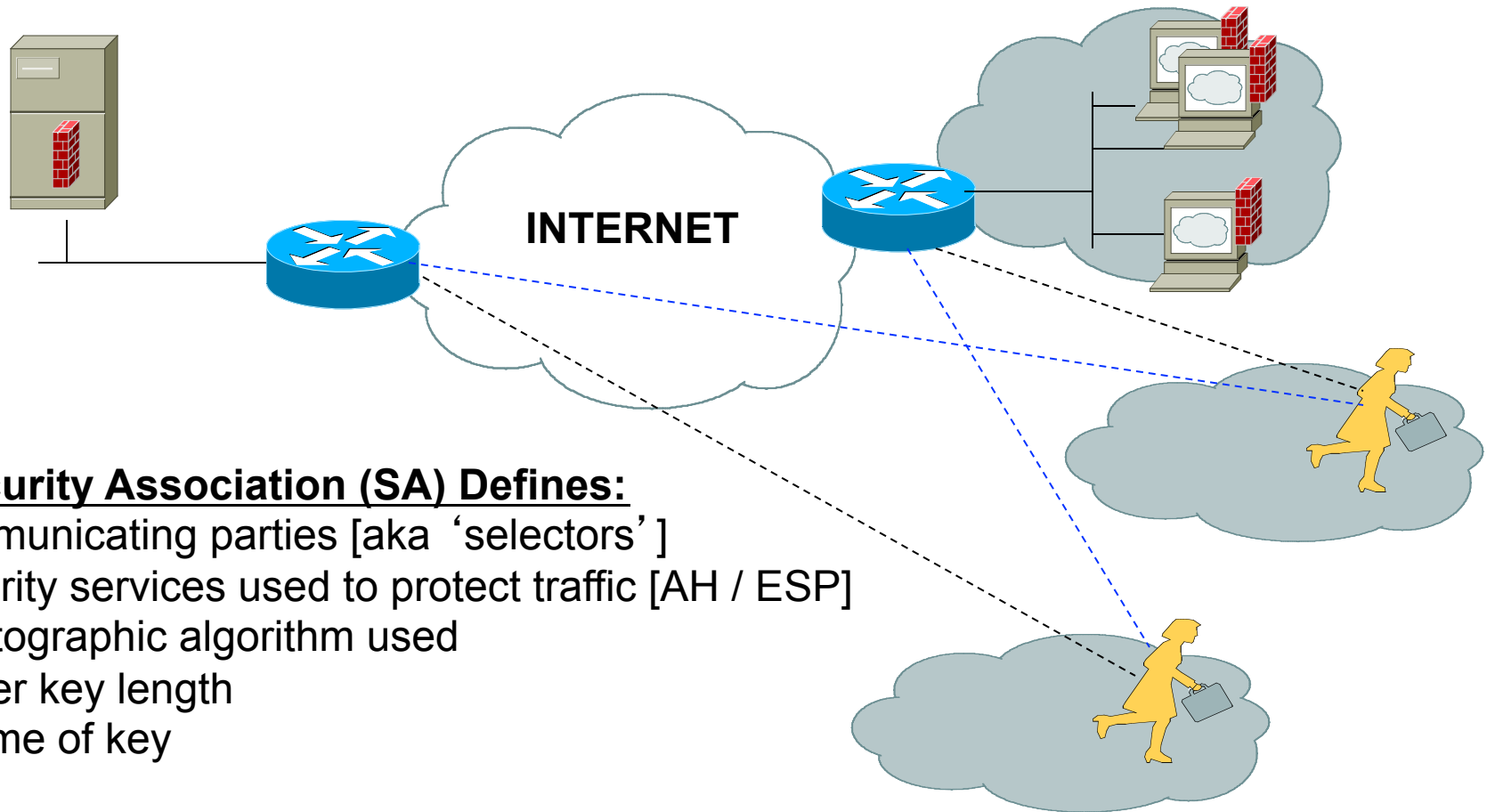




# Security Associations (SA)

- A collection of parameters required to establish a secure session
- Uniquely identified by three parameters consisting of
  - Security Parameter Index (SPI)
  - IP destination address
  - Security protocol (AH or ESP) identifier
- An SA is unidirectional
  - Two SAs required for a bidirectional communication
- A single SA can be used for AH or ESP, but not both
  - must create two (or more) SAs for each direction if using both AH and ESP

# Security Associations



## A Security Association (SA) Defines:

- communicating parties [aka 'selectors']
- security services used to protect traffic [AH / ESP]
- cryptographic algorithm used
- cipher key length
- lifetime of key

# Security Parameter Index (SPI)

- A unique 32-bit identification number that is part of the Security Association (SA)
- It enables the receiving system to select the SA under which a received packet will be processed.
- Has only local significance, defined by the creator of the SA.
- Carried in the ESP or AH header
- When an ESP/AH packet is received, the SPI is used to look up all of the crypto parameters

# How to Set Up SA

- Manually
  - Sometimes referred to as “manual keying”
  - You configure on each node:
    - Participating nodes (I.e. traffic selectors)
    - AH and/or ESP [tunnel or transport]
    - Cryptographic algorithm and key
- Automatically
  - Using IKE (Internet Key Exchange)

# ISAKMP

- Internet Security Association and Key Management Protocol
- Used for establishing Security Associations (SA) and cryptographic keys
- Only provides the framework for authentication and key exchange, but key exchange is independent
- Key exchange protocols
  - Internet Key Exchange (IKE) and Kerberized Internet Negotiation of Keys (KINK)

# Selectors

- Defines when to create an SA and what the SA will be used for
- Classifies the type of traffic requiring IPsec protection and the kind of protection to be applied.
- Elements of a selector:
  - Source IP address
  - Destination IP address
  - Protocol (TCP or UDP)
  - Upper layer protocol
    - Example: use ESP with NULL encryption and HMAC-SHA1 for routing updates, but use ESP with 3DES and SHA-1 for telnet and TFTP access for a router

# Authentication Header (AH)

- Provides source authentication and data integrity
  - Protection against source spoofing and replay attacks
- Authentication is applied to the entire packet, with the mutable fields in the IP header zeroed out
- If both AH and ESP are applied to a packet, AH follows ESP
- Operates on top of IP using protocol 51
- In IPv4, AH protects the payload and all header fields except mutable fields and IP options (such as IPSec option)

# AH Header Format

0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31

| Next Header  | Payload Length | Reserved |
|--|----------------|----------|
| Security Parameter Index (SPI)                         |                |          |
| Sequence Number  |                |          |
| Authentication Data<br>[ Integrity Check Value (ICV) ] |                |          |

**Next Header (8 bits):** indicates which upper layer protocol is protected (UDP, TCP, ESP)

**Payload Length (8 bits):** size of AH in 32-bit longwords, minus 2

**Reserved (16 bits):** for future use; must be set to all zeroes for now

**SPI (32 bits):** arbitrary 32-bit number that specifies to the receiving device which security association is being used (security protocols, algorithms, keys, times, addresses, etc)

**Sequence Number (32 bits):** start at 1 and must never repeat. It is always set but receiver may choose to ignore this field

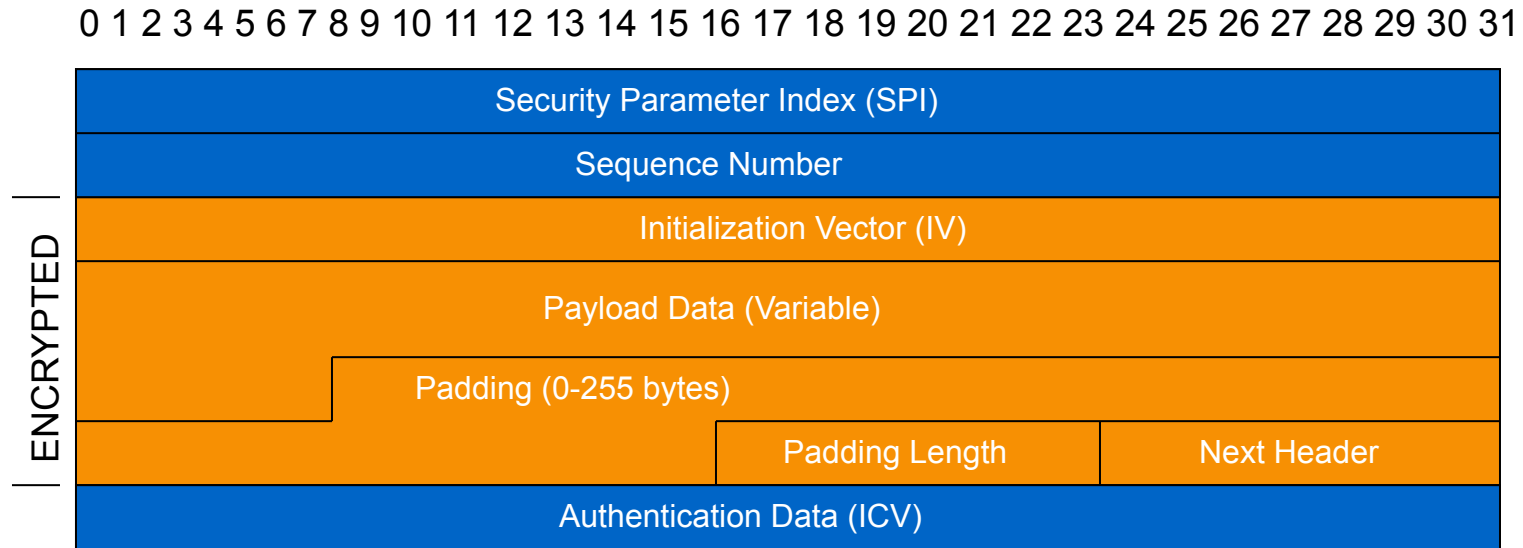
**Authentication Data:** ICV is a digital signature over the packet and it varies in length depending on the algorithm used (SHA-1, MD5)



# Encapsulating Security Payload (ESP)

- Uses IP protocol 50
- Provides all that is offered by AH, plus data confidentiality
  - It uses symmetric key encryption
- Must encrypt and/or authenticate in each packet
  - Encryption occurs before authentication
- Authentication is applied to data in the IPsec header as well as the data contained as payload

# ESP Header Format



**SPI:** arbitrary 32-bit number that specifies SA to the receiving device

**Seq #:** start at 1 and must never repeat; receiver may choose to ignore

**IV:** used to initialize CBC mode of an encryption algorithm

**Payload Data:** encrypted IP header, TCP or UDP header and data

**Padding:** used for encryption algorithms which operate in CBC mode

**Padding Length:** number of bytes added to the data stream (may be 0)

**Next Header:** the type of protocol from the original header which appears in the encrypted part of the packet

**Authentication Header:** ICV is a digital signature over the packet and it varies in length depending on the algorithm used (SHA-1, MD5)

# Packet Format Alteration for AH Transport Mode

## Authentication Header

Without AH



With AH

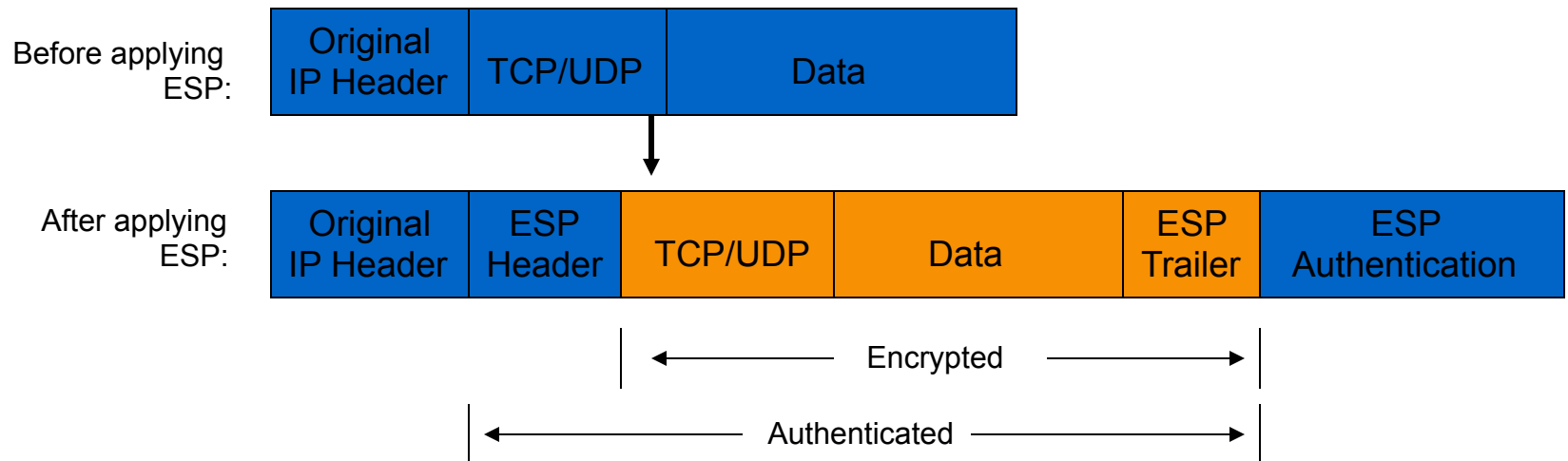


← Authenticated except for mutable fields in IP header →

- ToS
- TTL
- Header Checksum
- Offset
- Flags

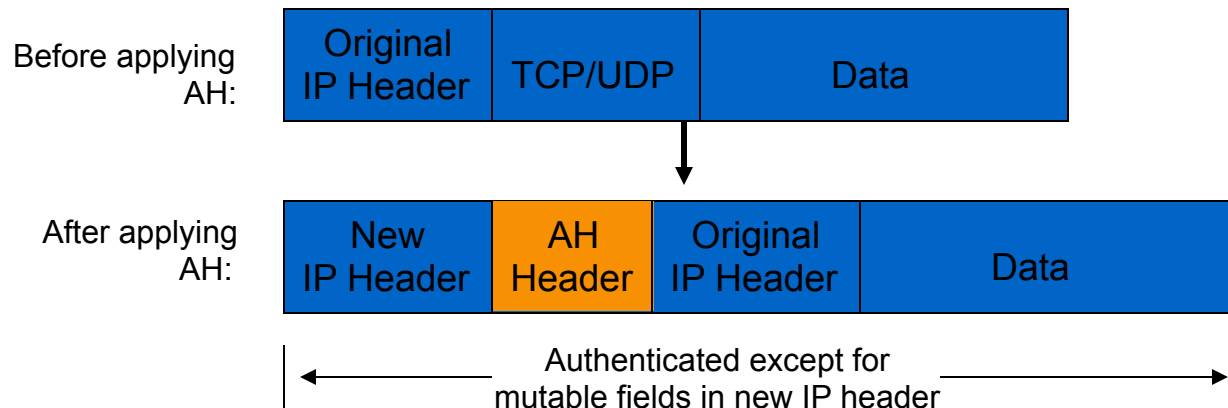
# Packet Format Alteration for ESP Transport Mode

## Encapsulating Security Payload



# Packet Format Alteration for AH Tunnel Mode

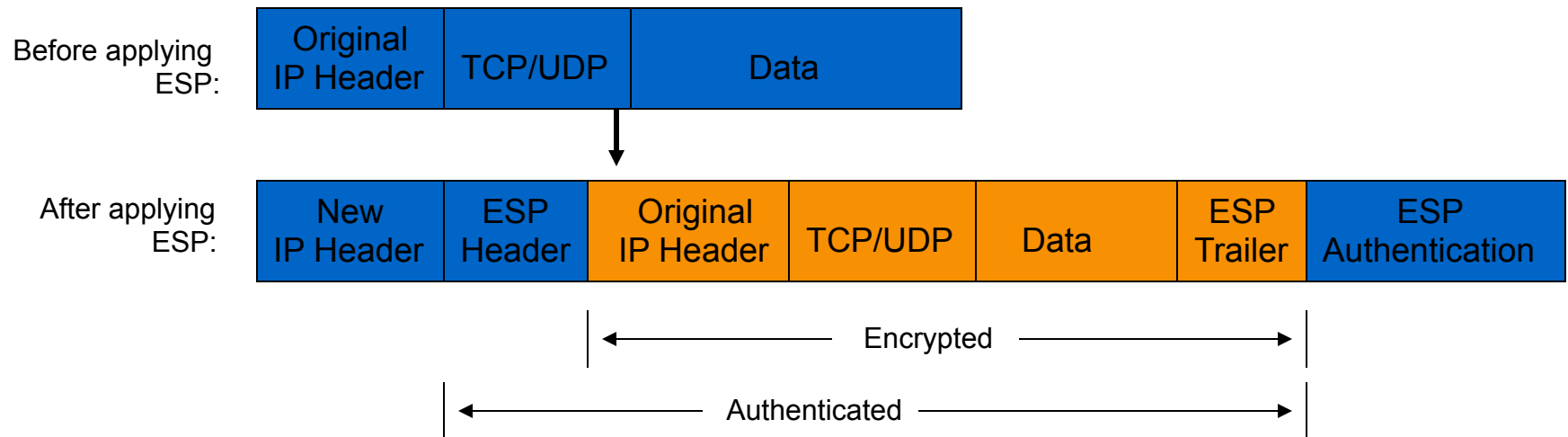
## Authentication Header



- ToS
- TTL
- Header Checksum
- Offset
- Flags

# Packet Format Alteration for ESP Tunnel Mode

## Encapsulating Security Payload



# Internet Key Exchange (IKE)

- “An IPSec component used for performing mutual authentication and establishing and maintaining Security Associations.” (RFC 5996)
- Typically used for establishing IPSec sessions
- A key exchange mechanism
- Five variations of an IKE negotiation:
  - Two modes (aggressive and main modes)
  - Three authentication methods (pre-shared, public key encryption, and public key signature)
- Uses UDP port 500

# IKE Modes

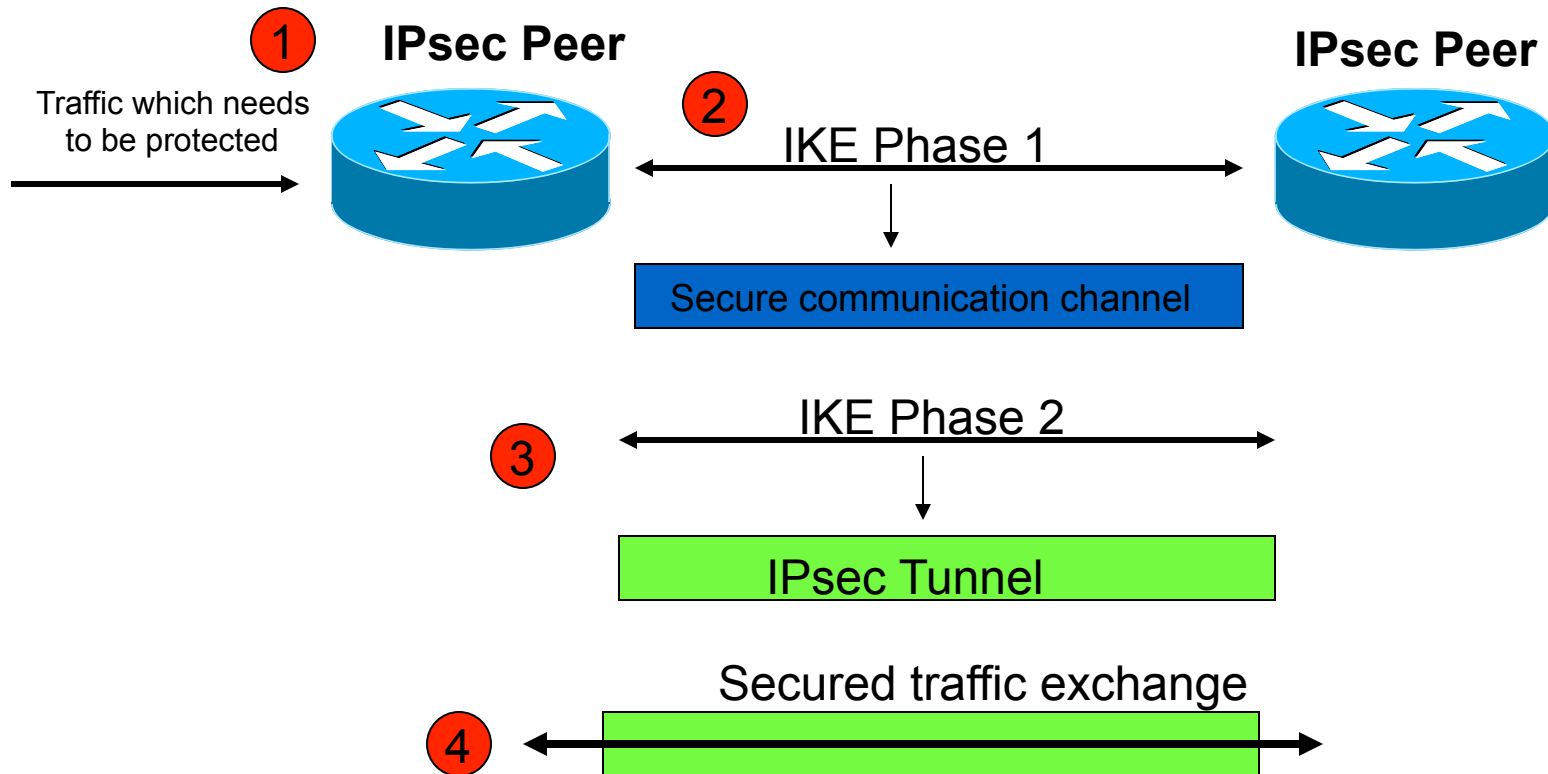
| Mode            | Description   |
|-----------------|---|
| Main mode       | Three exchanges of information between IPsec peers.<br>Initiator sends one or more proposals to the other peer (responder)<br>Responder selects a proposal  |
| Aggressive Mode | Achieves same result as main mode using only 3 packets<br>First packet sent by initiator containing all info to establish SA<br>Second packet by responder with all security parameters selected<br>Third packet finalizes authentication of the ISAKMP session |
| Quick Mode      | Negotiates the parameters for the IPsec session.<br>Entire negotiation occurs within the protection of ISAKMP session   |



# Internet Key Exchange (IKE)

- Phase I
  - Establish a secure channel (ISAKMP SA)
  - Using either main mode or aggressive mode
  - Authenticate computer identity using certificates or pre-shared secret
- Phase II
  - Establishes a secure channel between computers intended for the transmission of data (IPsec SA)
  - Using quick mode

# Overview of IKE



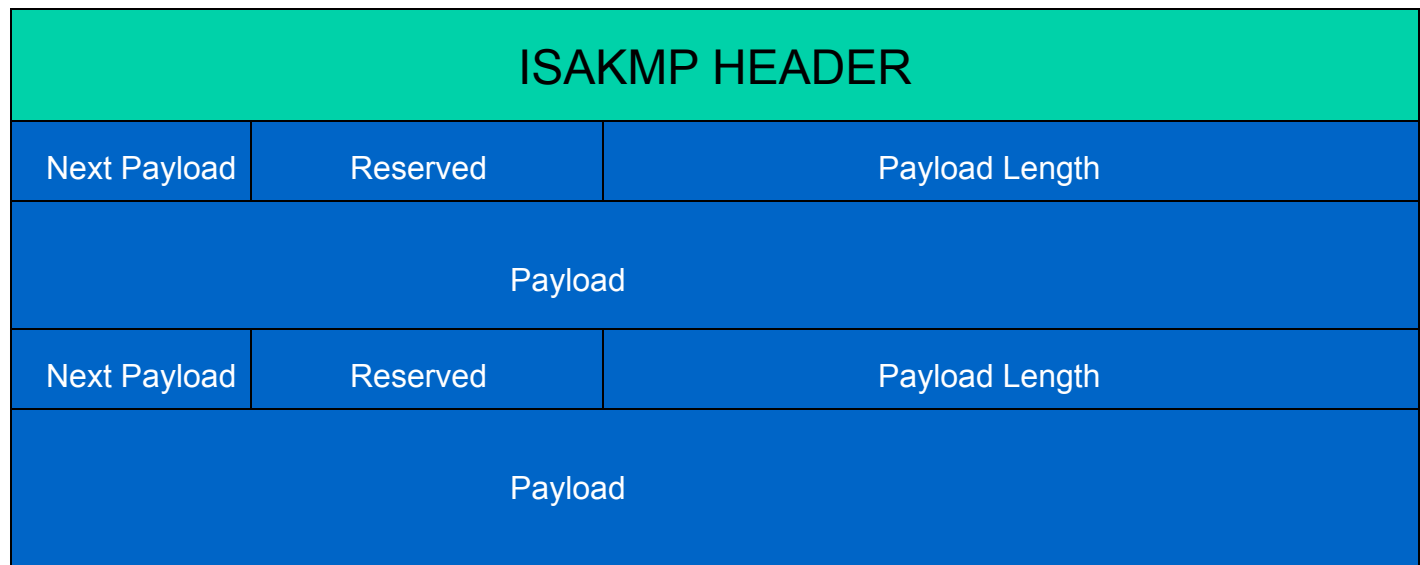
# ISAKMP Header Format

0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31

|                         |               |               |               |       |
|-------------------------|---------------|---------------|---------------|-------|
| Initiator Cookie        |               |               |               |       |
| Responder Cookie        |               |               |               |       |
| Next Payload            | Major Version | Minor Version | Exchange Type | Flags |
| Message ID              |               |               |               |       |
| Total Length of Message |               |               |               |       |

# ISAKMP Message Format

0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31



**Next Payload:** 1byte; identifier for next payload in message. If it is the last payload It will be set to 0

**Reserved:** 1byte; set to 0

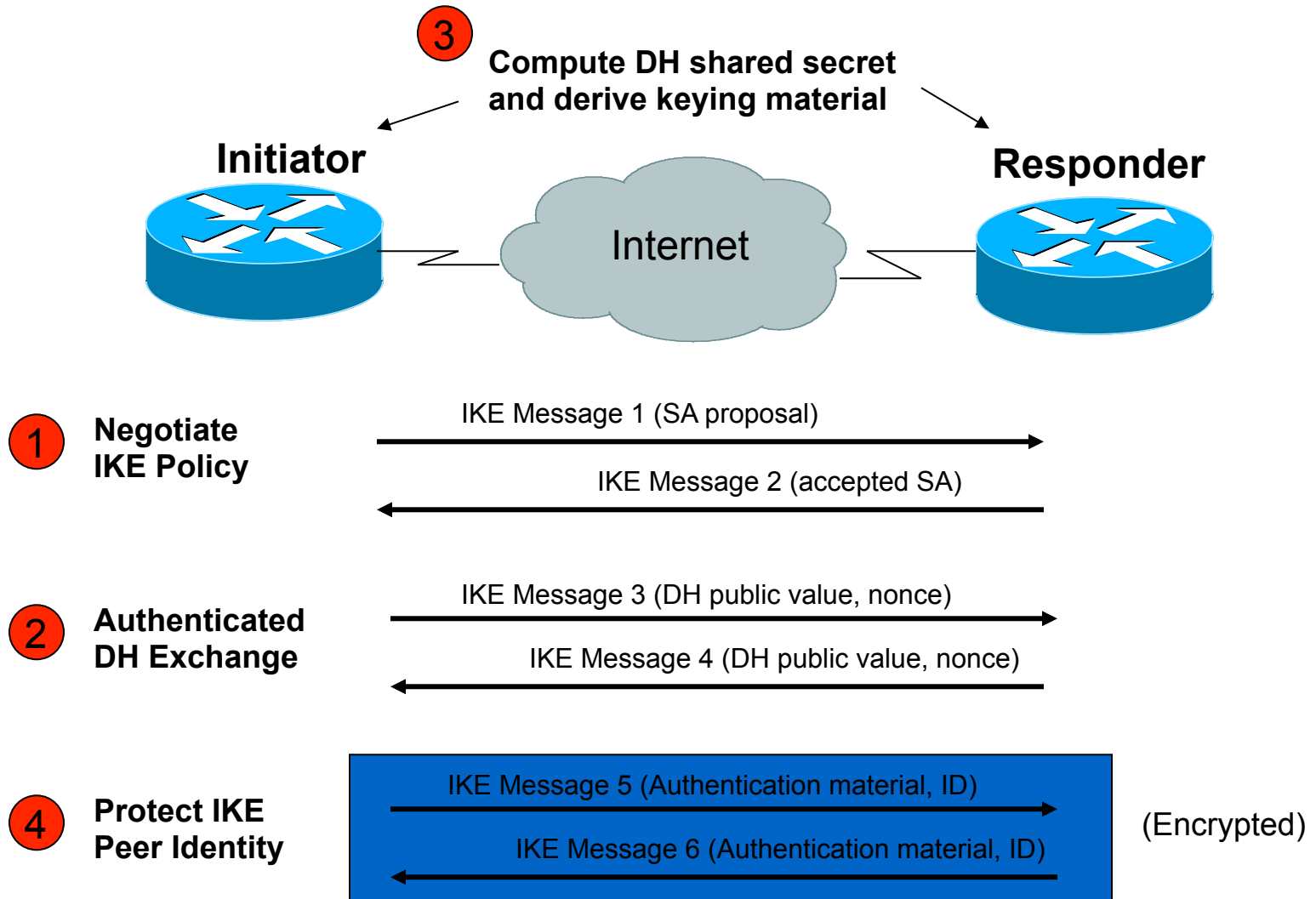
**Payload Length:** 2 bytes; length of payload (in bytes) including the header

**Payload:** The actual payload data

# IKE Phase 1 (Main Mode)

- Main mode negotiates an ISAKMP SA which will be used to create IPsec SAs
- Three steps
  - SA negotiation (encryption algorithm, hash algorithm, authentication method, which DH group to use)
  - Do a Diffie-Hellman exchange
  - Provide authentication information
  - Authenticate the peer

# IKE Phase 1 (Main Mode)



# IKE Phase 1 (Aggressive Mode)

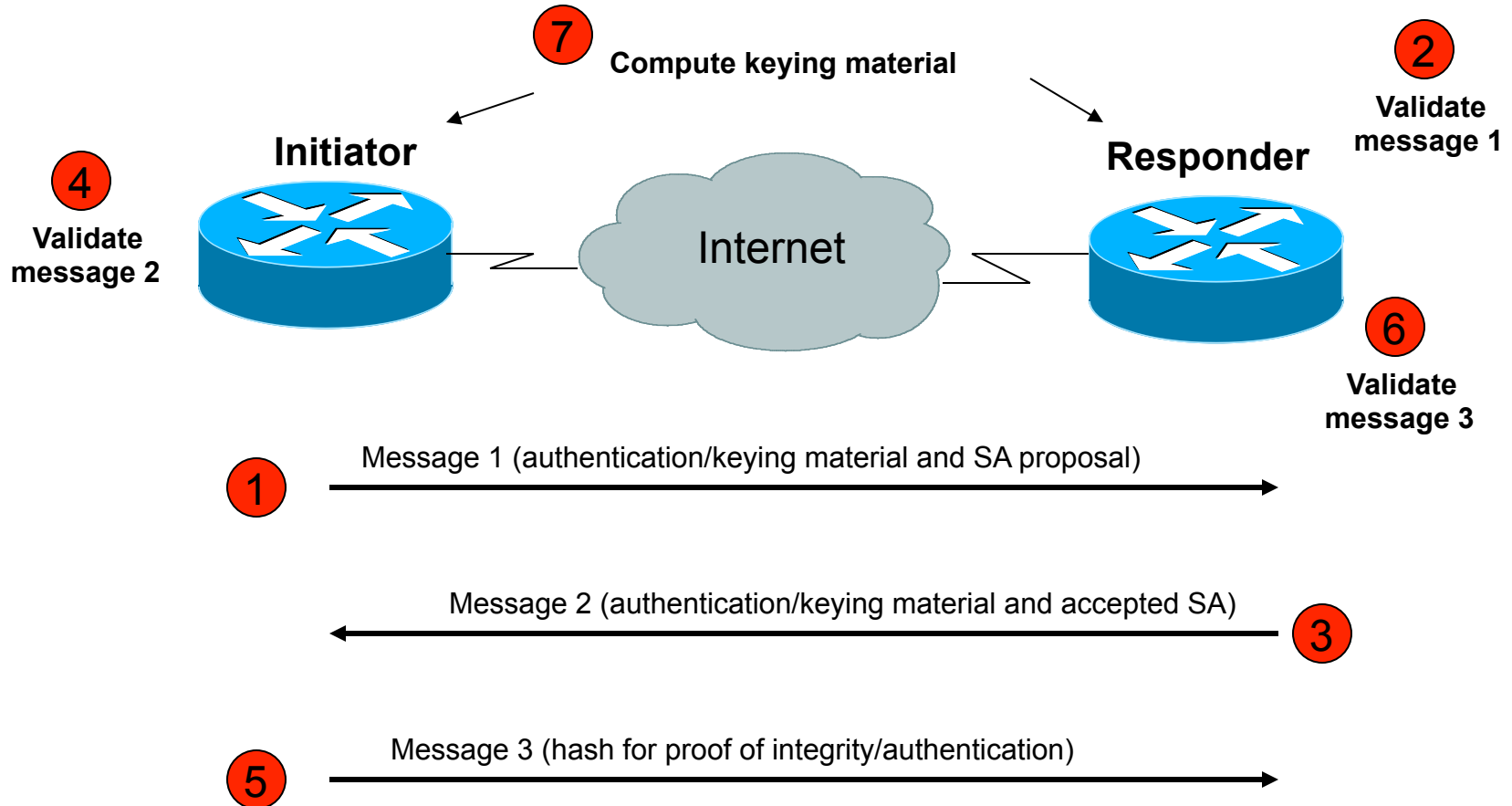
- Uses 3 (vs 6) messages to establish IKE SA
- No denial of service protection
- Does not have identity protection
- Optional exchange and not widely implemented

# IKE Phase 2 (Quick Mode)

- All traffic is encrypted using the ISAKMP Security Association
- Each quick mode negotiation results in two IPsec Security Associations (one inbound, one outbound)
- Creates/refreshes keys



# IKE Phase 2 (Quick Mode)



# IPSec Best Practices

- Use IPSec to provide integrity in addition to encryption
  - Use ESP option
- Use strong encryption algorithms
  - AES instead of DES
- Use a good hashing algorithm
  - SHA instead of MD5
- Reduce the lifetime of the Security Association (SA) by enabling Perfect Forward Secrecy (PFS)
  - Increases processor burden so do this only if data is highly sensitive

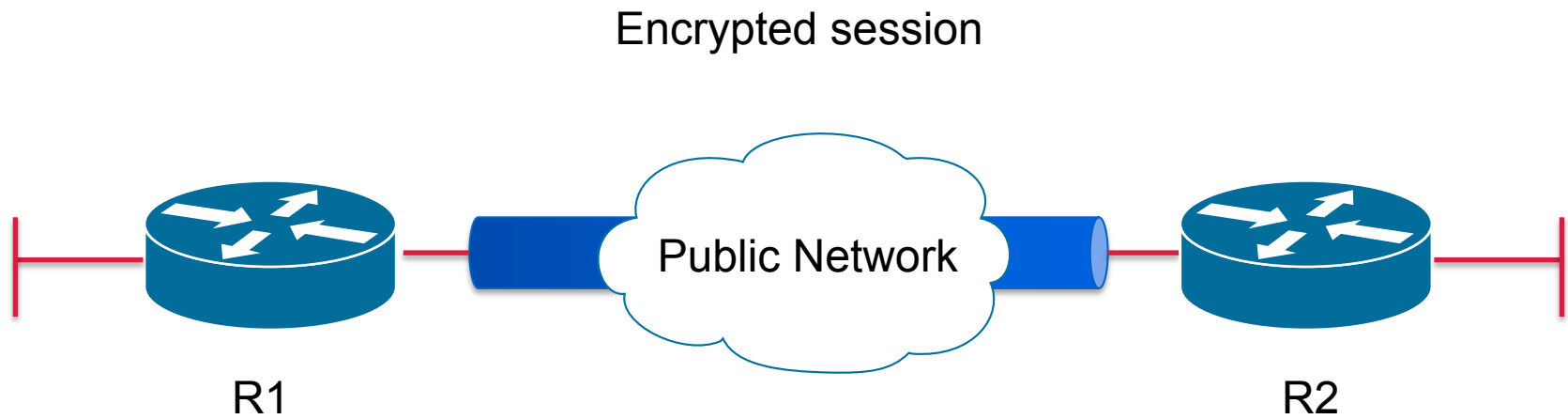
# Configuring IPSec

- Step 1: Configure the IKE Phase 1 Policy (ISAKMP Policy)
  - `crypto isakmp policy [priority]`
- Step 2: Set the ISAKMP Identity
  - `crypto isakmp identity {ipaddress|hostname}`
- Step 3: Configure the IPSec transfer set
  - `crypto ipsec transform-set transform-set-name  
<transform1> <transform2> mode [tunnel|transport]`
  - `crypto ipsec security-association lifetime seconds  
seconds`

# Configuring IPSec

- **Step 5: Creating map with name**
  - `Crypto map crypto-map-name seq-num ipsec-isakmp`
  - `Match address access-list-id`
  - `Set peer [ipaddress|hostname]`
  - `Set transform-set transform-set-name`
  - `Set security-association lifetime seconds seconds`
  - `Set pfs [group1|group2]`
- **Step 6: Apply the IPsec Policy to an Interface**
  - `Crypto map crypto-map-name local-address interface-id`

# IPSec Layout



# Router Configuration

```
crypto isakmp policy 1
  authentication pre-share
  encryption aes
  hash sha
  group 5
```

Phase 1 SA

Encryption and authentication

```
crypto isakmp key Training123 address 172.16.11.66
!
```

```
crypto ipsec transform-set ESP-AES-SHA esp-aes esp-sha-hmac
!
```

```
crypto map LAB-VPN 10 ipsec-isakmp
  match address 101
  set transform-set ESP-AES-SHA
  set peer 172.16.11.66
```

Phase 2 SA

# Router Configuration

```
int fa 0/1
```

```
crypto map LAB-VPN
```

```
Exit
```

```
!
```

```
access-list 101 permit ip 172.16.16.0  
0.0.0.255 172.16.20.0 0.0.0.255
```

Apply to an  
outbound interface

Define interesting  
VPN traffic

# Questions?



# Thank You!

End of Workshop