Threat Pragmatics & Cryptography Basics

PacNOG19

28th November - 2nd December 2016

Nadi, Fiji

Issue Date: [31-12-2015]

Revision: [V.1]

APNIC

Why Security?

- The Internet was initially designed for connectivity
  - Trust is assumed, no security
  - Security protocols added on top of the TCP/IP
- Fundamental aspects of information must be protected
  - Confidential data
  - Employee information
  - Business models
  - Protect identity and resources
- The Internet has become fundamental to our daily activities (business, work, and personal)

Internet Evolution

LAN connectivity

Application-specific
More online content

Application/data
hosted in the “cloud”

Different ways to handle security as the Internet evolves

Goals of Information Security

Confidentiality
prevents unauthorized use or disclosure of information

Integrity
safeguards the accuracy and completeness of information

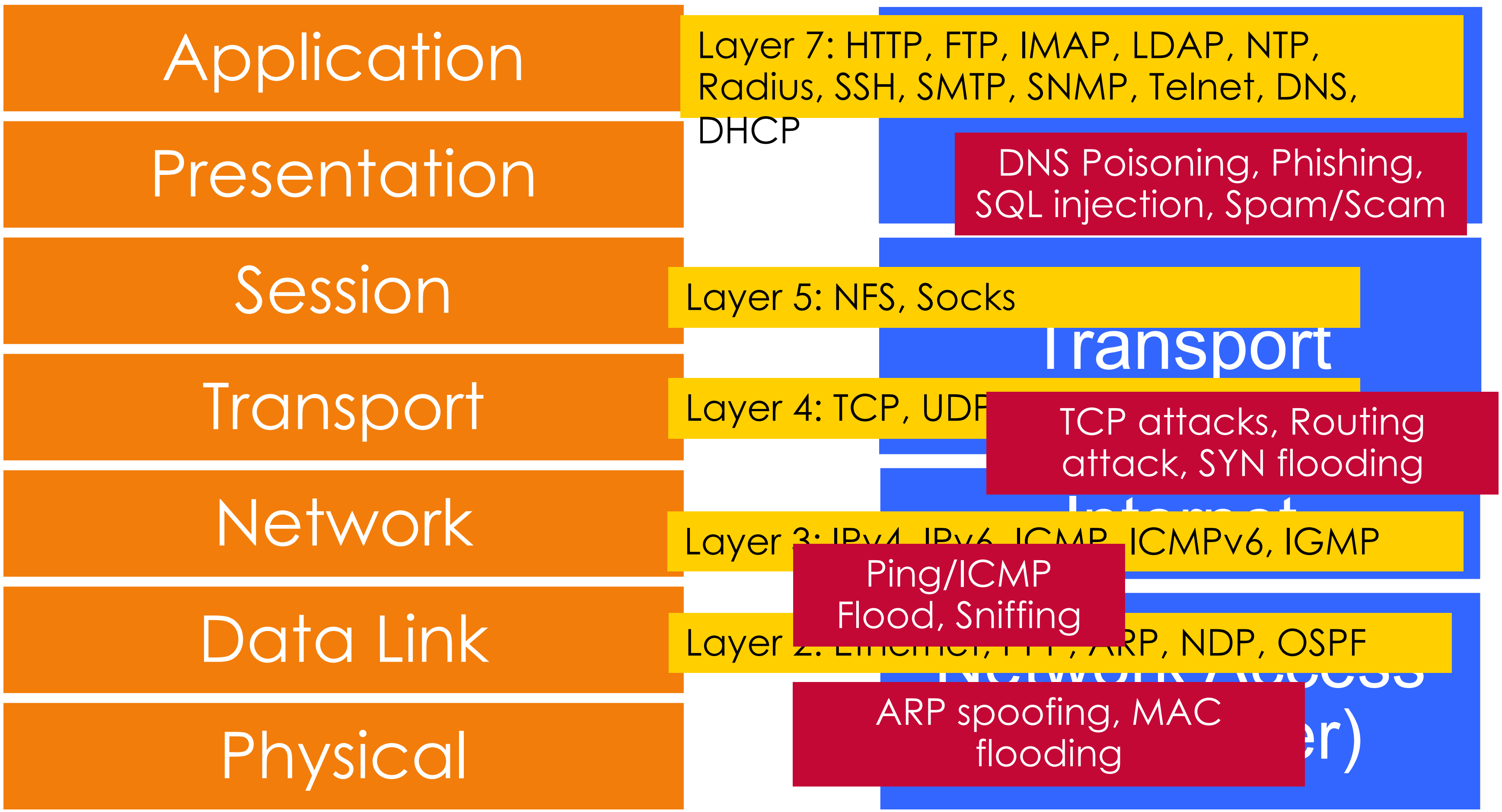
Availability
authorized users have reliable and timely access to information

SECURITY

Target

- Many sorts of targets:
  - Network infrastructure
  - Network services
  - Application services
  - User machines
- What’s at risk?

Attacks on Different Layers



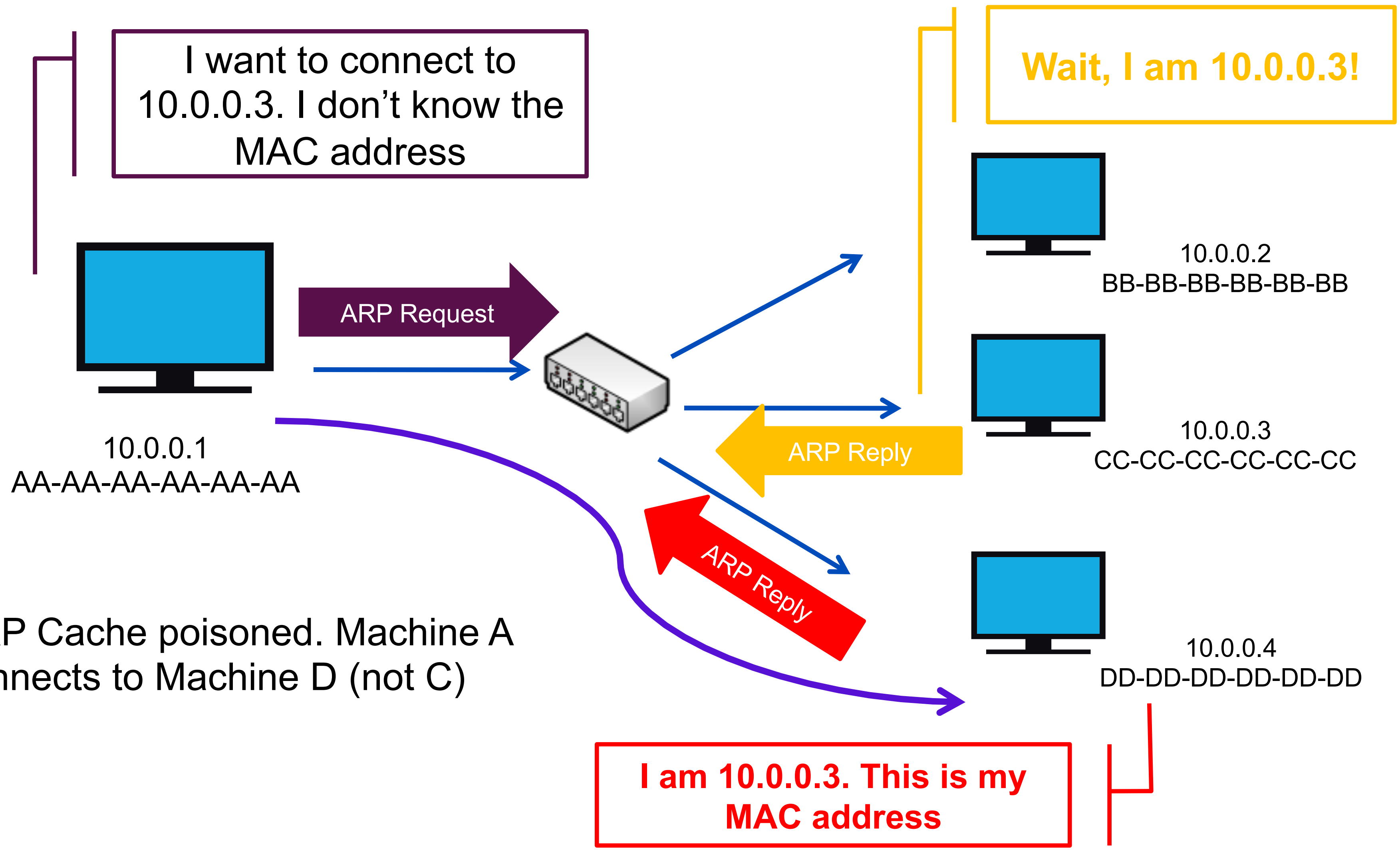
OSI Reference Model

TCP/IP Model

Layer 2 Attacks

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

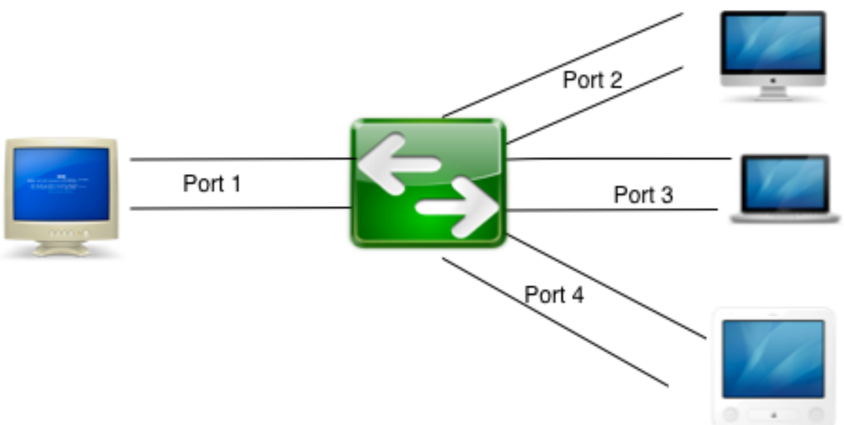
ARP Spoofing



ARP Cache poisoned. Machine A connects to Machine D (not C)

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.

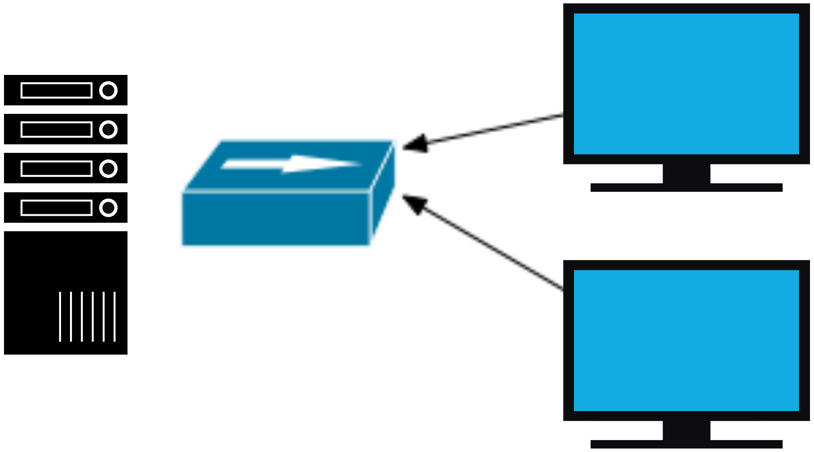


	Port 1	Port 2	Port 3	Port 4
00:01:23:45:67:A1	x			
00:01:23:45:67:B2		x		
00:01:23:45:67:C3			x	
00:01:23:45:67:D4				x

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

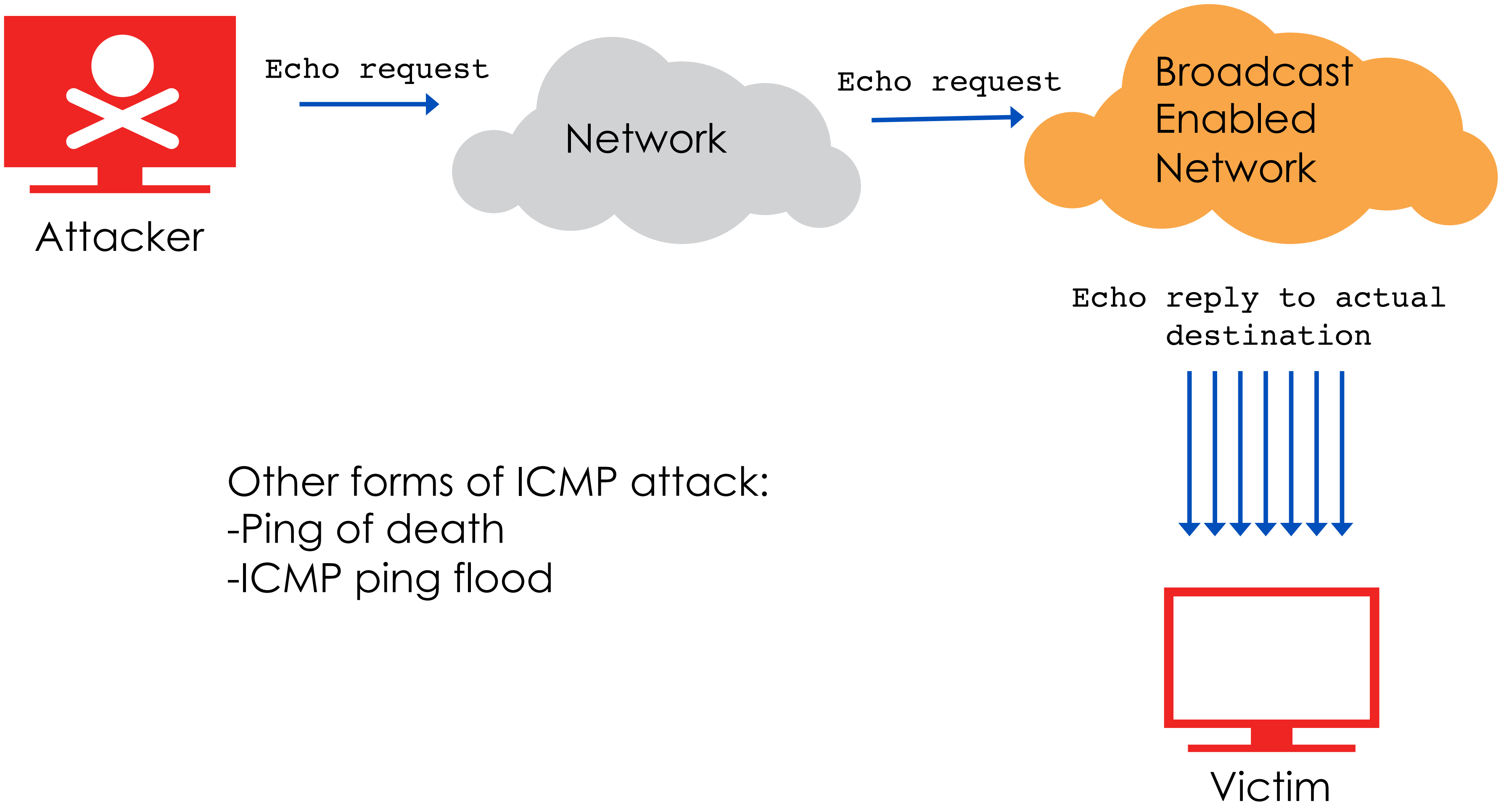


Attacker sends many different DHCP requests with many spoofed addresses.

Layer 3 Attacks

- ICMP Ping Flood
- ICMP Smurf
- Ping of death

Ping Flood



Other forms of ICMP attack:
-Ping of death
-ICMP ping flood

Routing Attacks

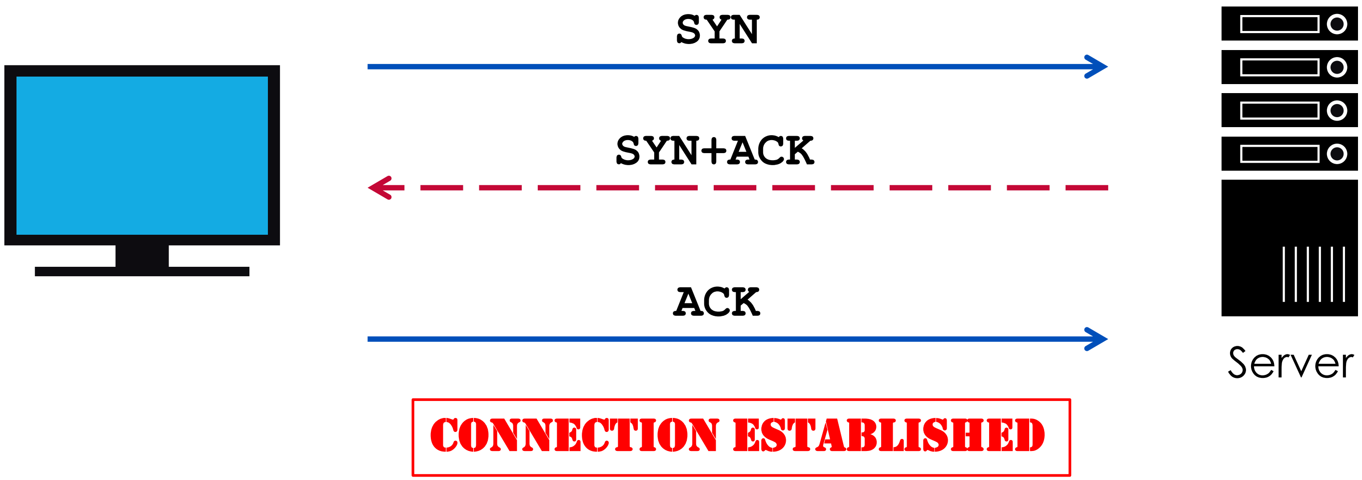
- Attempt to poison the routing information
- Distance Vector Routing
  – Announce 0 distance to all other nodes
    - Blackhole traffic
    - Eavesdrop
- Link State Routing
  – Can drop links randomly
  – Can claim direct link to any other routers
  – A bit harder to attack than DV
- BGP attacks
  –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

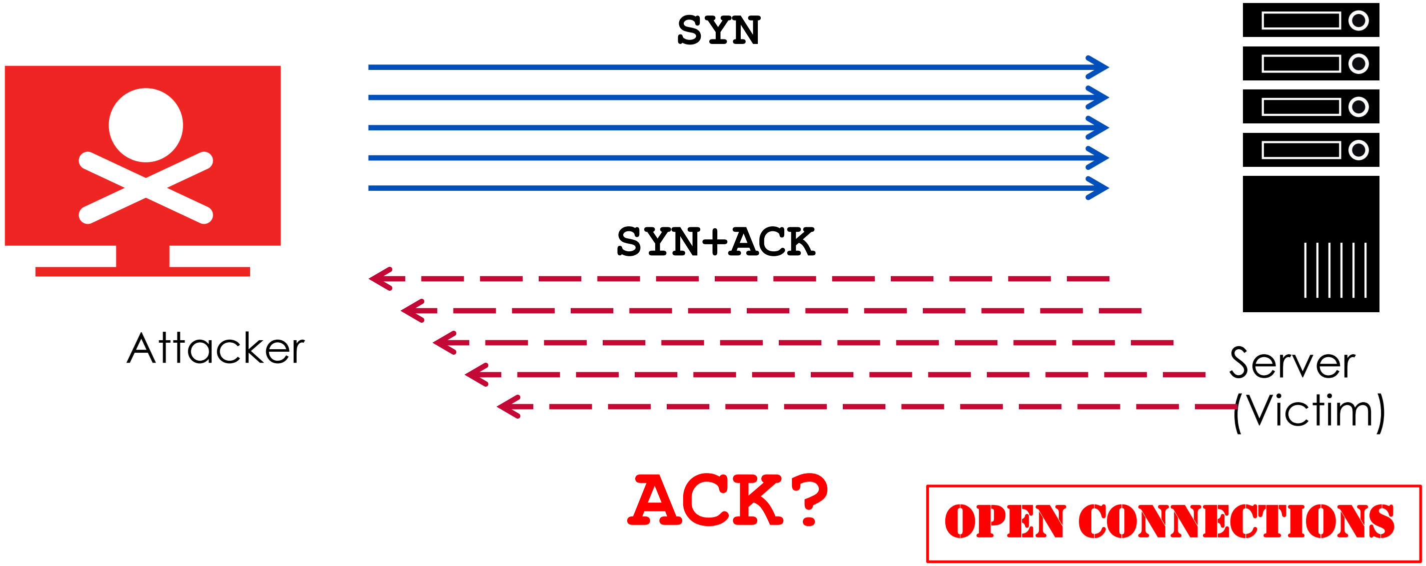
TCP Attacks

- 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



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

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

Layer 7 DDoS Attack

- Traditional DoS attacks focus on Layer 3 and Layer 4
- In Layer 7, a DoS attack is targeted towards the applications disguised as legitimate packets
- The aim is to exhaust application resources (bandwidth, ports, protocol weakness) rendering it unusable
- Includes:
  – HTTP GET
  – HTTP POST
  – Slowloris
  – LOIC / HOIC
  – RUDY (R-U-Dead Yet)

Layer 7 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

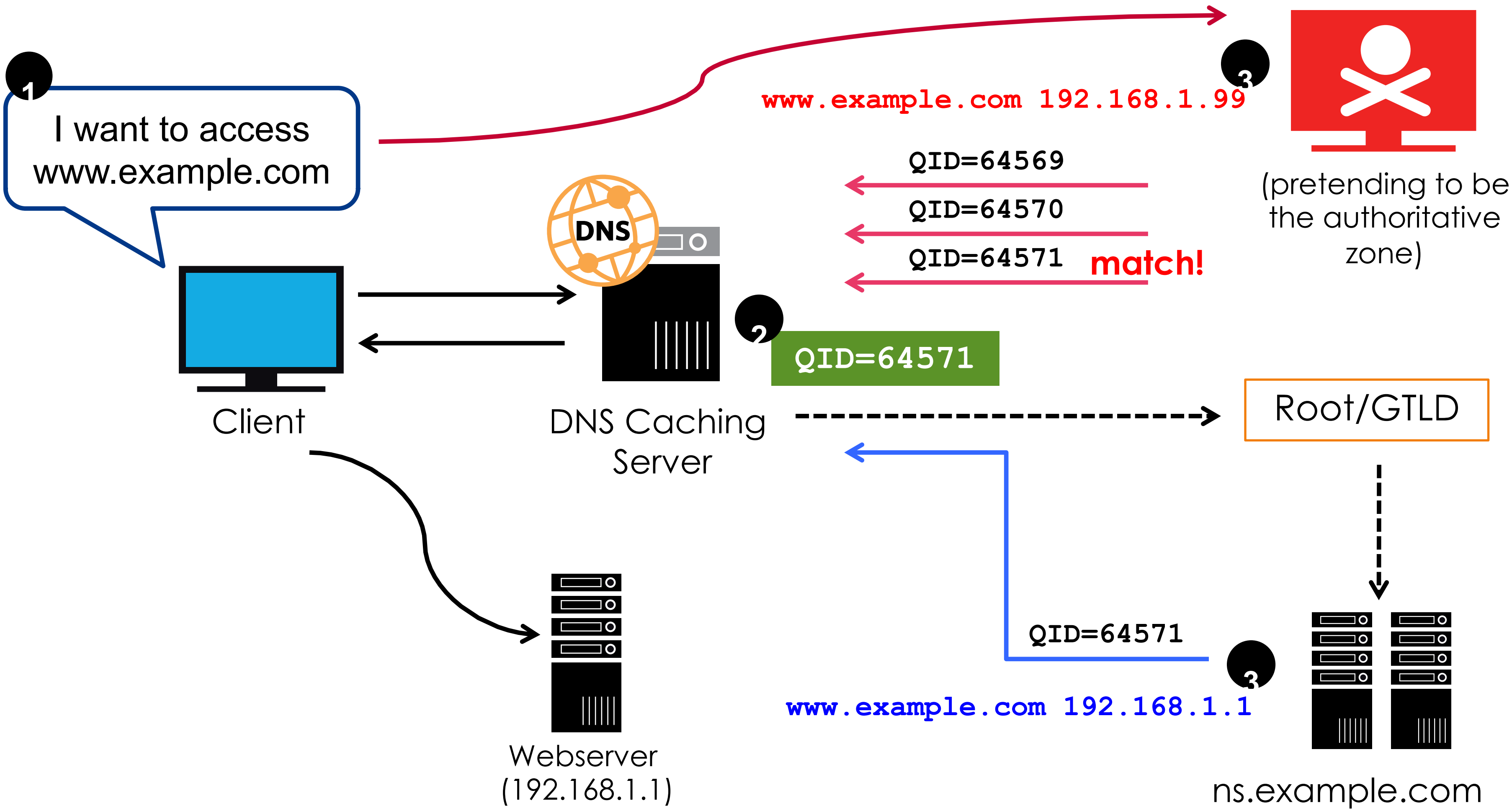
DNS Changer

- "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

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



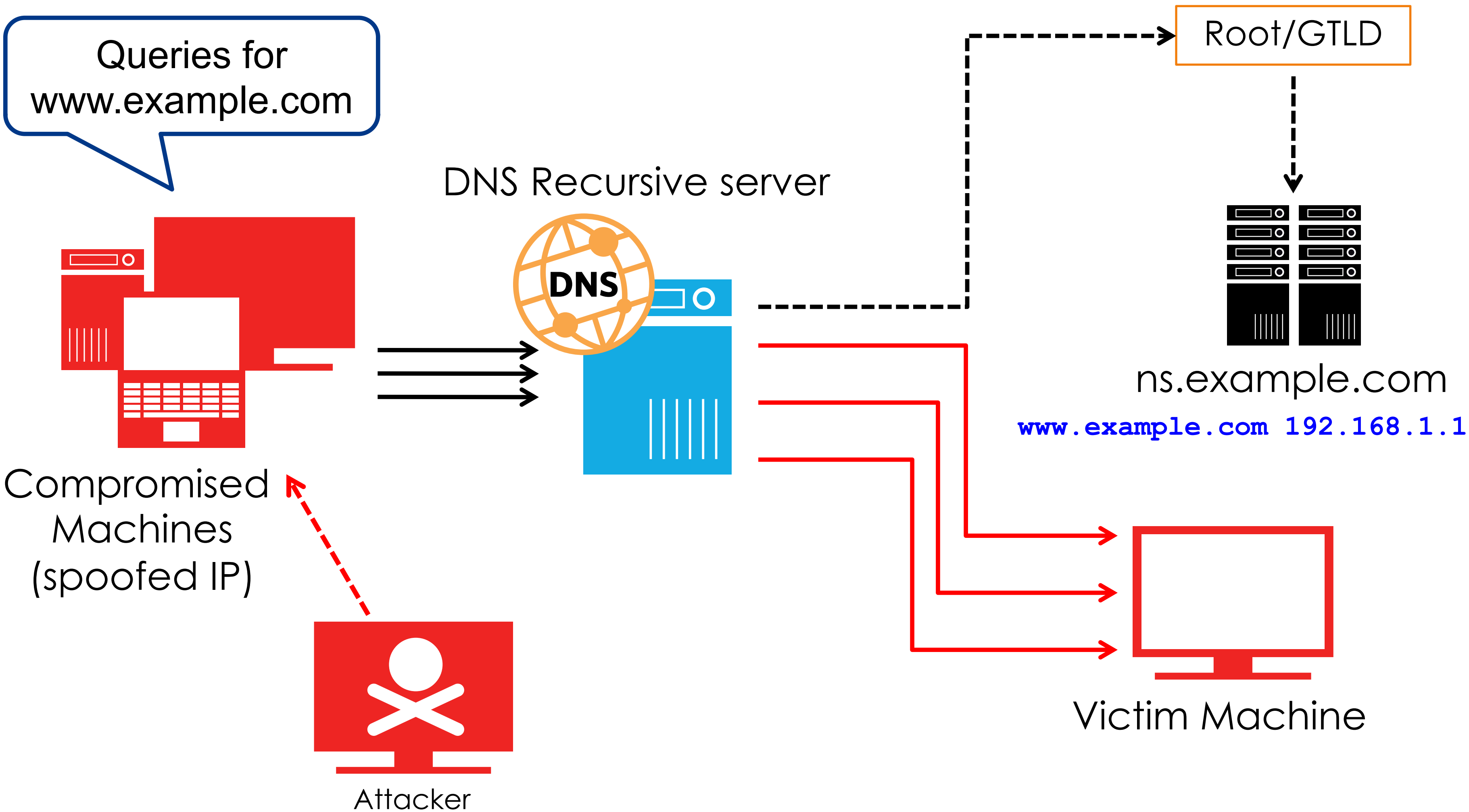
Amplification Attacks

- Exploiting UDP protocol to return large amplified amounts of traffic / data
- Small request, large reply
- Examples:
  - DNS
  - NTP
  - SMTP
  - SSDP

DNS Amplification Attack

- A type of reflection attack combined with amplification
  - Source of attack is reflected off another machine
  - Traffic received is bigger (amplified) than the traffic sent by the attacker
- UDP packet's source address is spoofed

DNS Amplification



NTP Amplification

- Network Time Protocol (NTP)
- Port 123/UDP
- Exploits NTP versions older than v4.2.7
  - monlist
- Several incidents in 2014

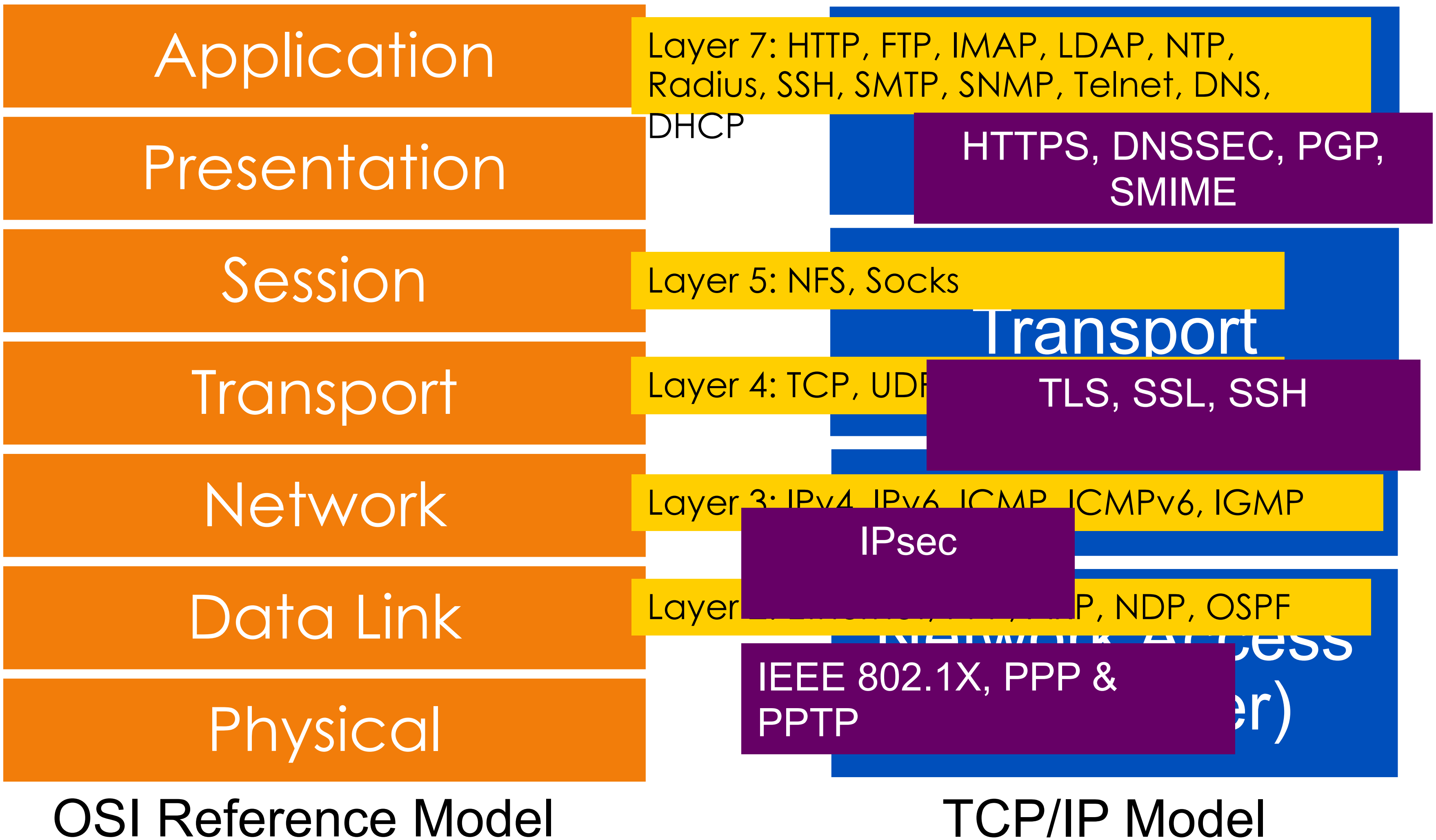
Wireless Attacks

- 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 crackers might speed it up

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.

Attacks on Different Layers



Link-Layer Security

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

Transport Layer Security

- Secure Socket Layer (SSL)
- Secure Shell Protocol

Application Layer Security

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

Cryptography

Cryptography

- ?

Cryptography

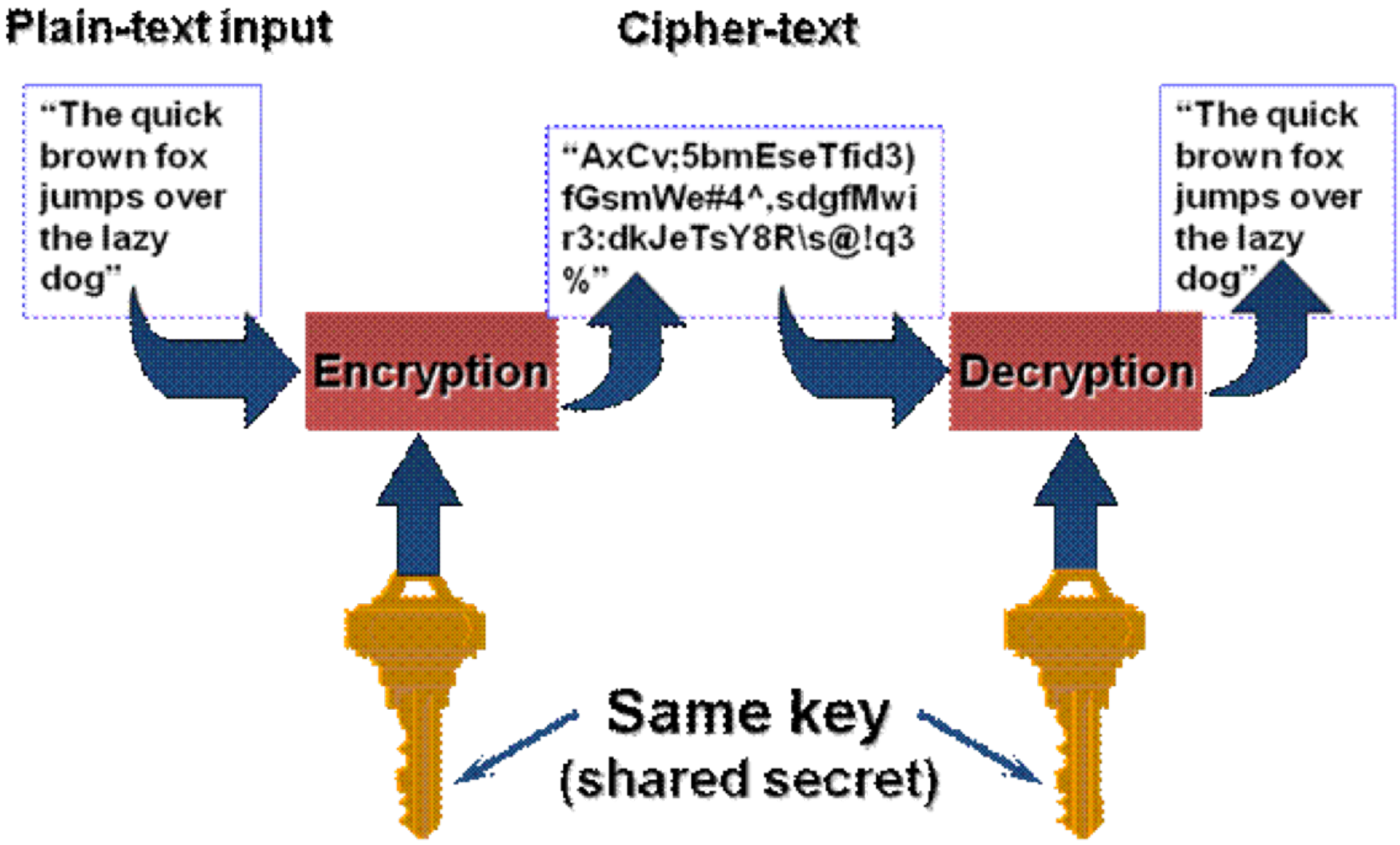
- Cryptography deals with creating documents that can be shared secretly over public communication channels
- Other terms closely associated
  - Cryptanalysis = code breaking
  - Cryptology
    - Kryptos (hidden or secret) and Logos (description) = secret speech / communication
    - combination of cryptography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

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

Notation:

Plaintext (P)
Ciphertext (C)
Cryptographic Key (k)

Encryption & Decryption



Terminology

- Cryptography : the practice and study of hiding information
- Cryptanalysis : to find some weakness or insecurity in a cryptographic scheme
- Encryption : the method of transforming data (plain text) into an unreadable format
- Plaintext - the “scrambled” format of data after being encrypted

Cryptosystem Terminology

- Decryption : the method of turning cipher text back into plaintext
- Encryption Algorithm : a set of rules or procedures that dictates how to encrypt and decrypt data, also called encryption cipher
- Key : (cryptovariable) a value used in the encryption process to encrypt and decrypt

Cryptosystem Terminology

- Key Space : the range of possible values used to construct keys
- Example :
  - key can be 4 digits (0-9)
  - key space = 10,000
  - key can be 6 digits
  - key space = 1,000,000
- Key Clustering : when two different key generate the same cipher text from the same plaintext
- Work Factor : estimated time and resources to break a cryptosystem

Cryptosystem Development

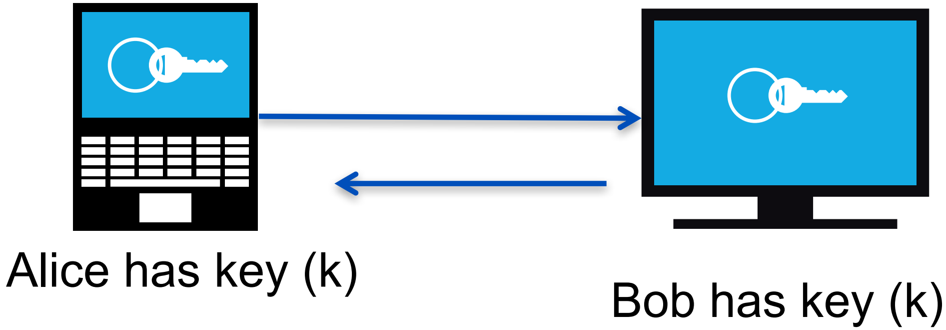
- Open algorithms to review
- Assume the attacker knows your encryption/decryption algorithm
- The only thing that should be secret in a cryptosystem is the key - Kirchhoff’s Principle

Work Factor

- The amount of processing power and time to break a crypto system
- No system is unbreakable
- Make it too expensive to break

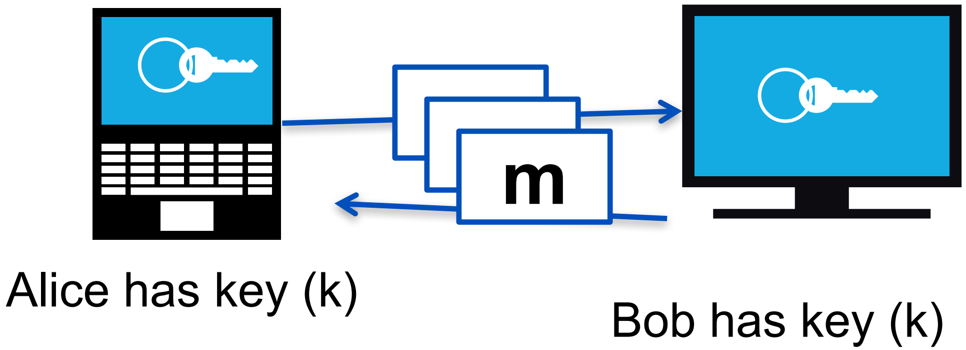
Crypto Core

- Secure key establishment



- Secure communication

Confidentiality and integrity



Source: Dan Boneh, Stanford

Kerckhoff's Law (1883)

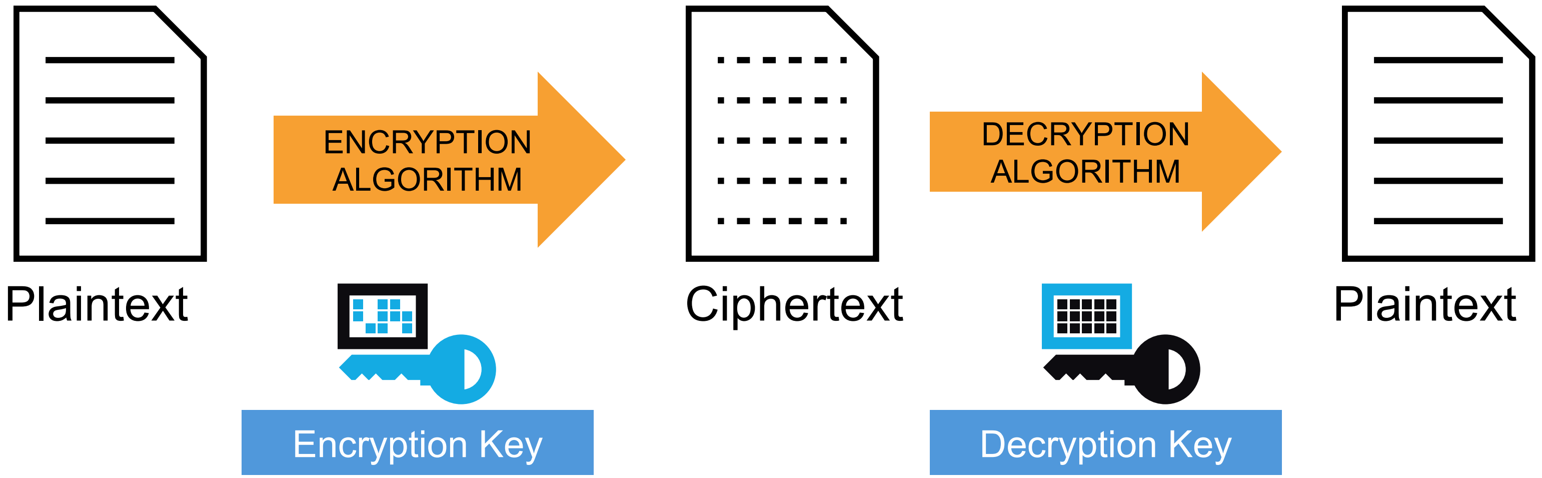
- The system must not be required to be secret, and it must be able to fall into the hands of the enemy without inconvenience.

- In other words, the security of the system must rest entirely on the secrecy of the key.

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

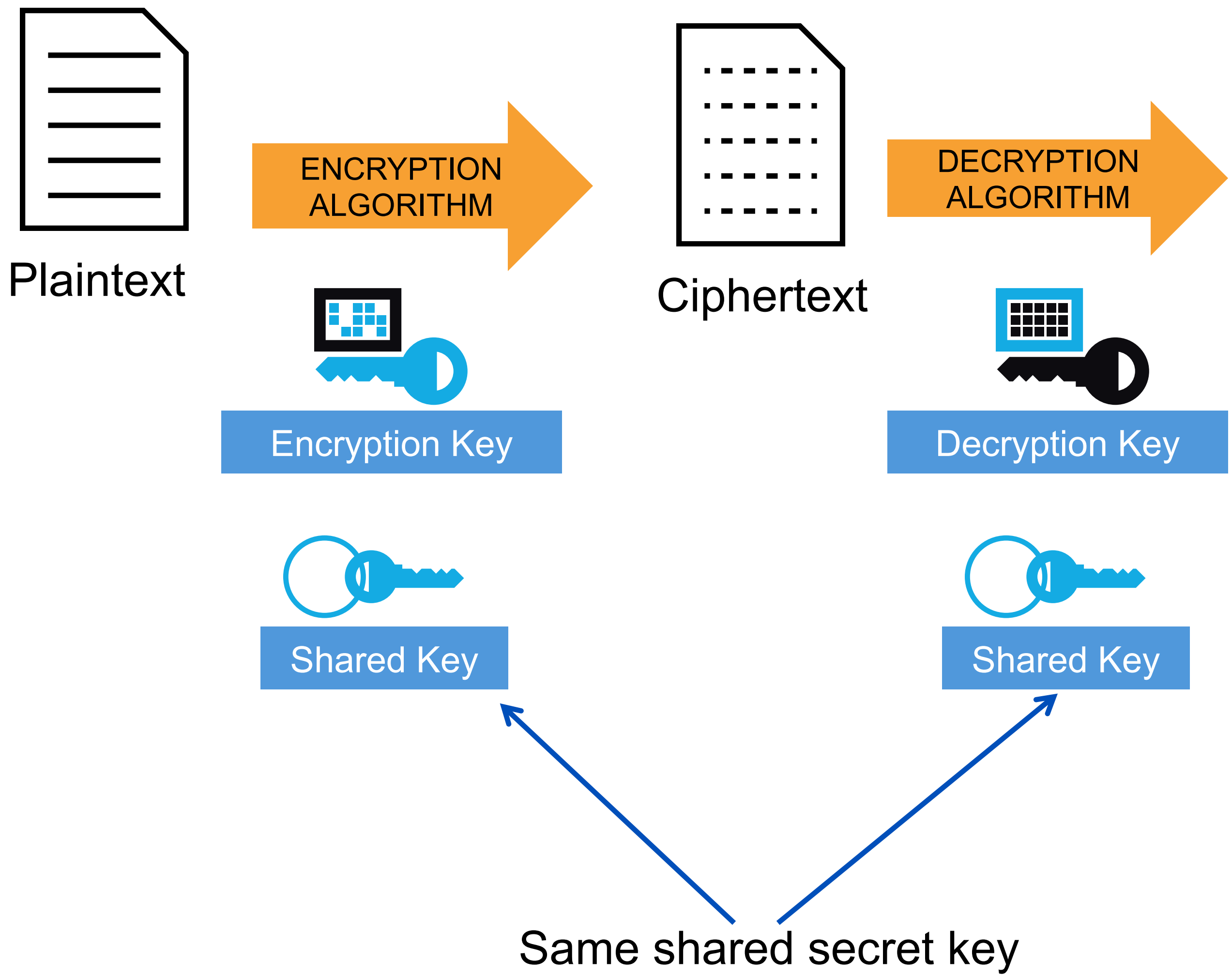
Encryption and Decryption



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.
- Examples:
  - DES, 3DES, AES, RC4, RC6, Blowfish

Symmetric Encryption



Symmetric Key Cryptography

Symmetric Key Algorithm

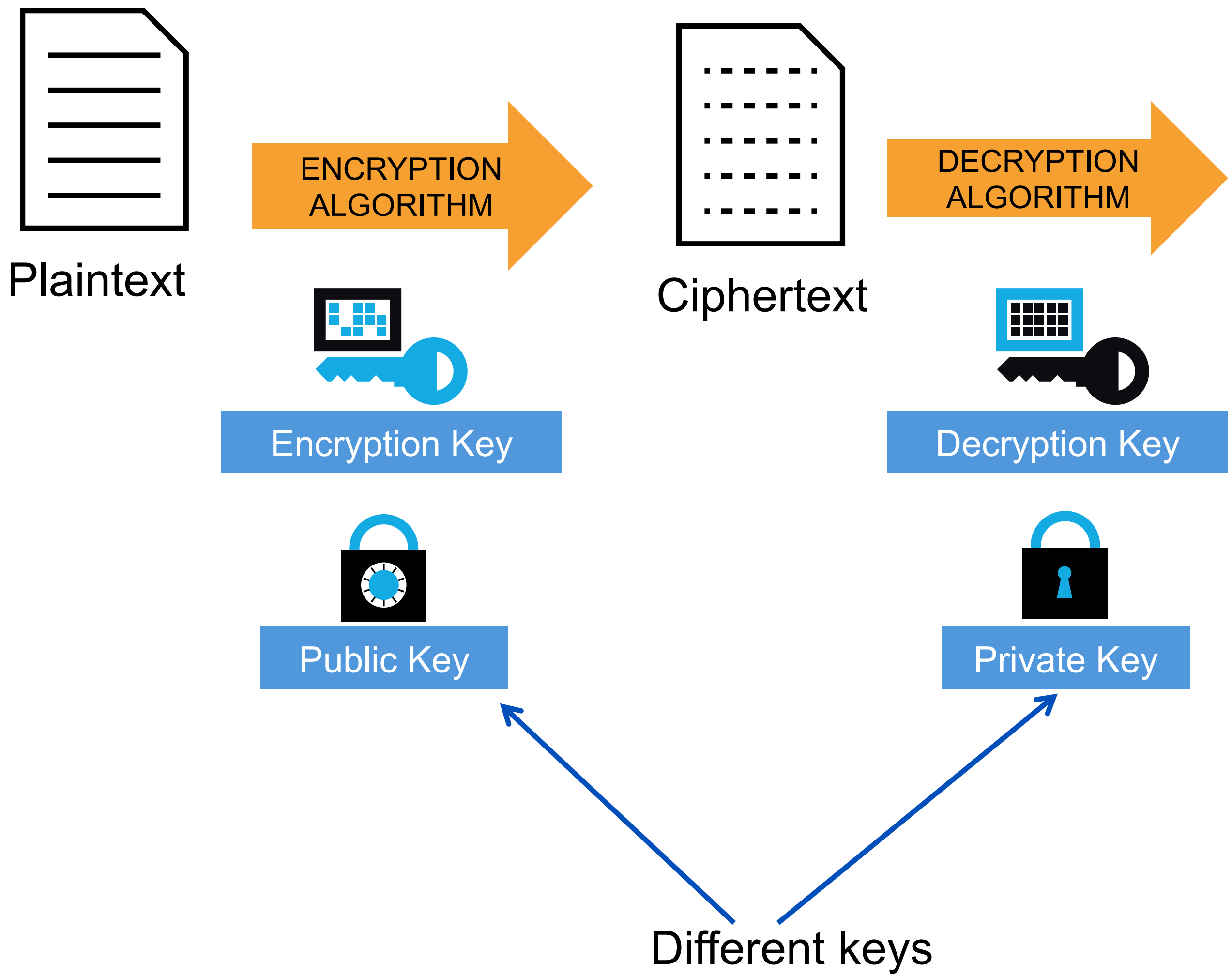
Symmetric Algorithm	Key Size
DES	56-bit keys
Triple DES (3DES)	112-bit and 168-bit keys
AES	128, 192, and 256-bit keys
IDEA	128-bit keys
RC2	40 and 64-bit keys
RC4	1 to 256-bit keys
RC5	0 to 2040-bit keys
RC6	128, 192, and 256-bit keys
Blowfish	32 to 448-bit keys

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

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:
  - RSA, DSA, Diffie-Hellman, ElGamal, PKCS

Asymmetric Encryption



Asymmetric Key Cryptography

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

Hash Functions

- produces a condensed representation of a message
- takes an input message of arbitrary length and outputs fixed-length code
  – The fixed-length output is called the hash or message digest
- A form of signature that uniquely represents the data
- Uses:
  – Verifying file integrity
  – Digitally signing documents
  – Hashing passwords

Hash Functions

- Message Digest (MD) Algorithm
  - Outputs a 128-bit fingerprint of an arbitrary-length input
  - MD4 is obsolete, MD5 is widely-used

- 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 can produce hash values that are 256, 384, and 512-bits respectively

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

- a message appended to a packet
- used to prove the identity of the sender and the integrity of the packet
- how it works:
  - sender signs the message with own private key
  - receiver uses the sender's public key to verify the signature

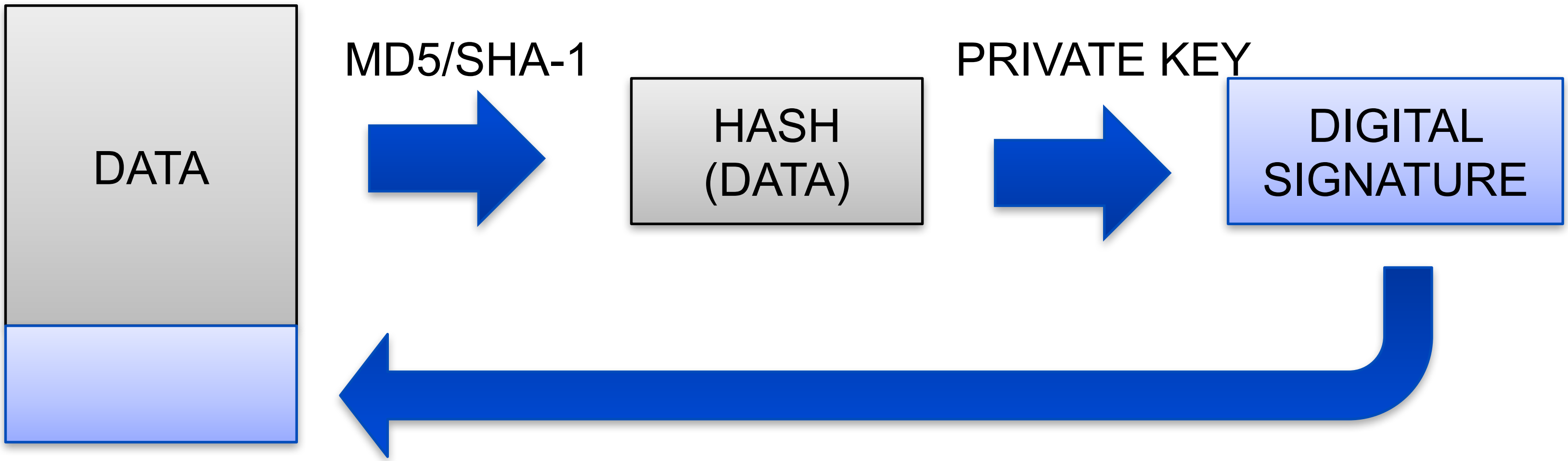
PKI / PGP Primer

- 🔑 Public Key
- 🗝 Private Key
- 📝 Message

- 📝+🔑 = 🔒✉️ Encrypted
- 🔒✉️+🗝 = 🔓📝 Decrypted
- 📝+🗝 = 🔏✉️ Signed
- 🔏✉️ + 🔑 = 👤 Authenticated

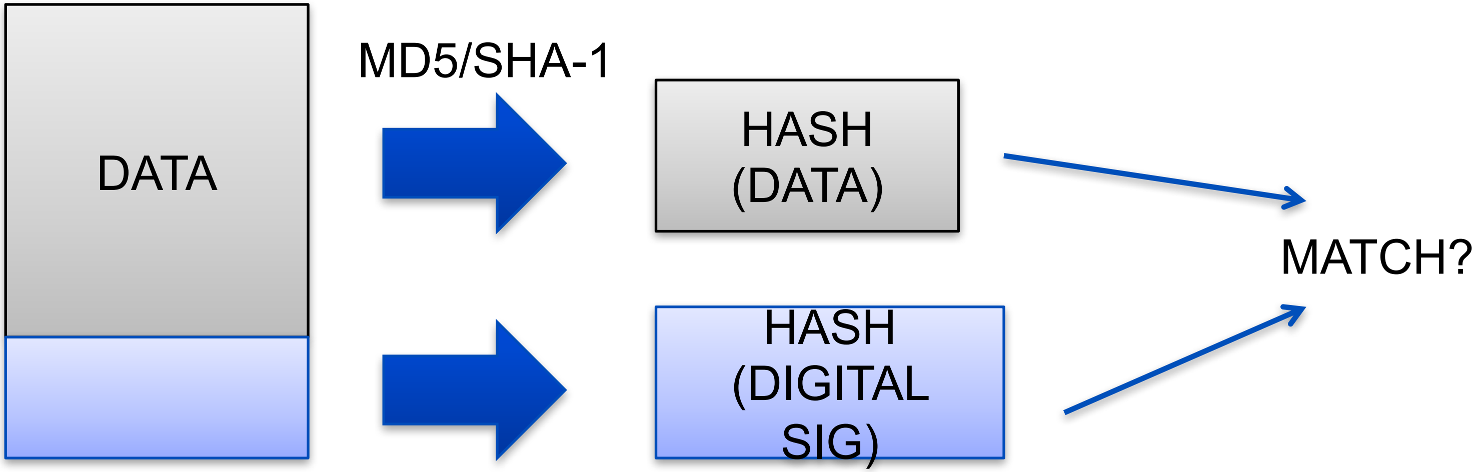
Digital Signature Process

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



Signature Verification Process

- Hash the original data using the same hashing algorithm
- Decrypt the digital signature using the sender's public key. All digital signatures contain a copy of the signer's public key
- 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!

APNIC