

Internet (continuation)

Lecture 3.4

Module 3. Networking Fundamentals

Serhii Zakharchenko

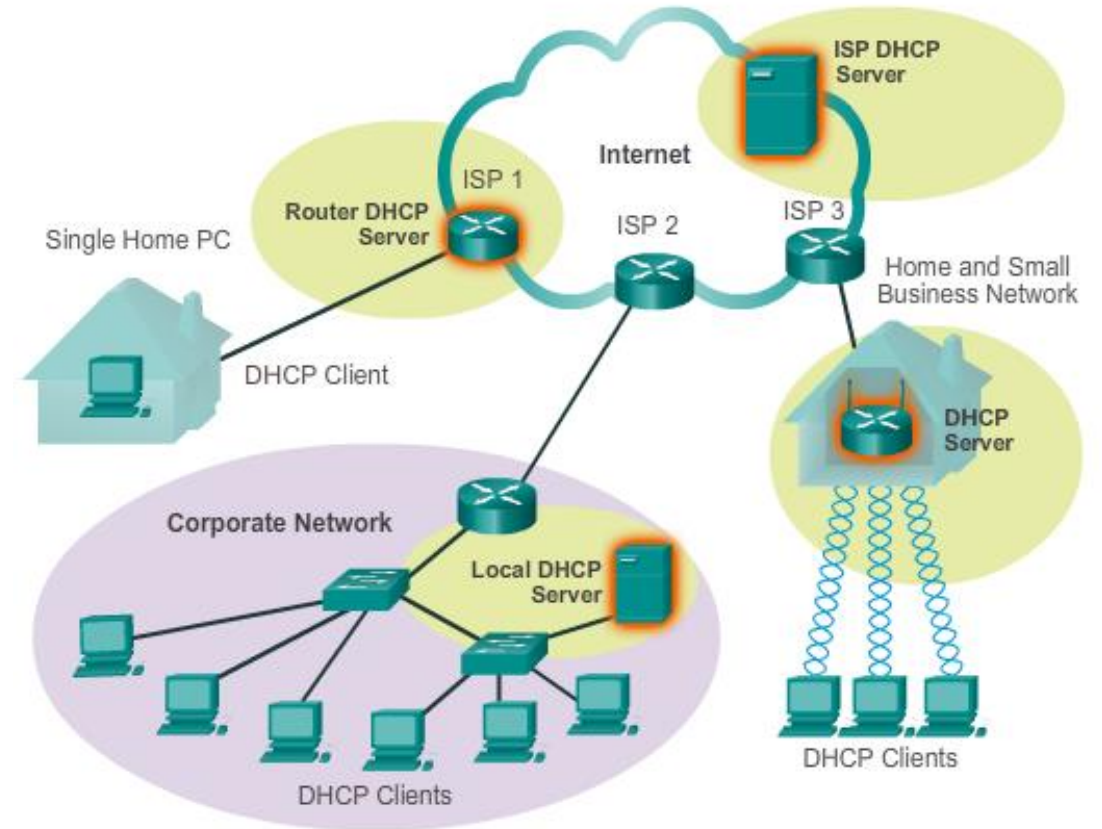
Agenda

- DHCP
- DNS
- NAT
- Q&A

DHCP

DHCP Introduction

- DHCP service enables devices on a network to obtain IP addresses, subnet masks, gateway, and other IP networking parameters **dynamically** from a DHCP server.
- DHCP server is contacted and address requested - chooses address from a configured range of addresses called a **pool** and “leases” it to the host for a **set period**
- DHCP used for general purpose hosts such as **end user devices**, and static addressing is used for network devices such as gateways, switches, servers and printers
- DHCP can pose a security risk because any device connected to the network can receive an address.



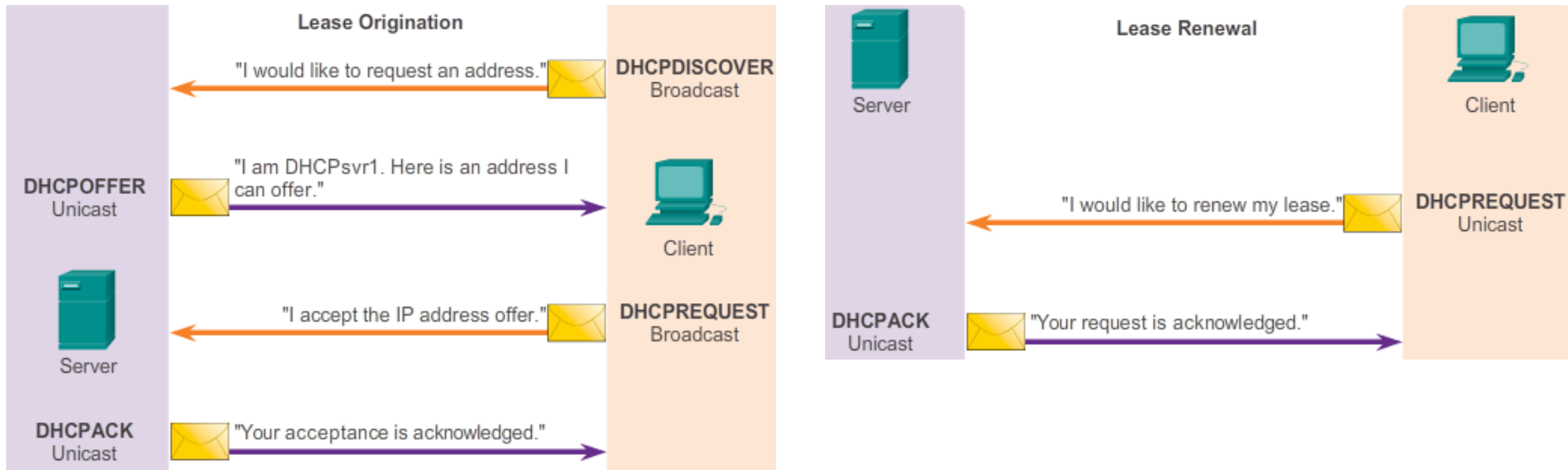
DHCP Operation

DHCPv4 uses three different address allocation methods

- **Manual Allocation** - The administrator assigns a pre-allocated IPv4 address to the client, and DHCPv4 communicates only the IPv4 address to the device.
- **Automatic Allocation** - DHCPv4 automatically assigns a static IPv4 address permanently to a device, selecting it from a pool of available addresses. No lease.
- **Dynamic Allocation** - DHCPv4 dynamically assigns, or leases, an IPv4 address from a pool of addresses for a limited period of time chosen by the server, or until the client no longer needs the address. Most commonly used.



DHCPv4 Lease Origination and Renew



DHCPv4 Message Format

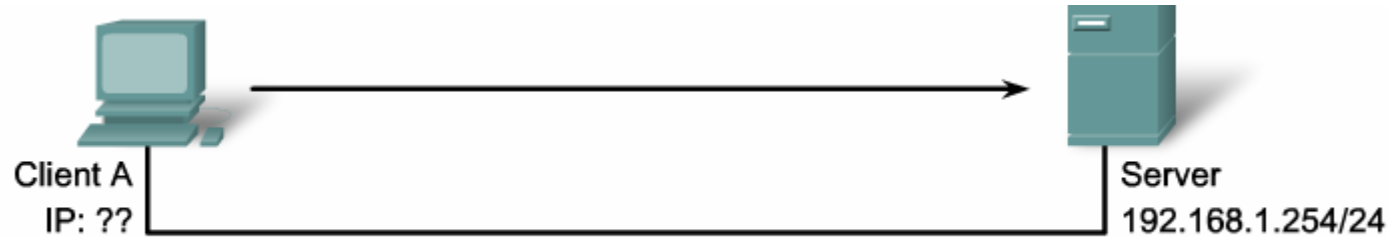
- **Operation (OP) Code** - Specifies the general type of message. A value of **1 - request**; a value of **2 - reply**.
- **Hardware Type** - Identifies the type of hardware used in the network. For example, 1 is Ethernet, 15 is Frame Relay, and 20 is a serial line.
- **Hardware Address Length** - Specifies the length of the address.
- **Hops** - Controls the forwarding of messages. Set to 0 by a client before transmitting a request.
- **Transaction Identifier** - Used by the client to match the request with replies received from DHCPv4 servers.

8	16	24	32
OP Code (1)	Hardware Type (1)	Hardware Address Length (1)	Hops (1)
Transaction Identifier			
Seconds - 2 bytes		Flags - 2 bytes	
Client IP Address (CIADDR) - 4 bytes			
Your IP Address (YIADDR) - 4 bytes			
Server IP Address (SIADDR) - 4 bytes			
Gateway IP Address (GIADDR) - 4 bytes			
Client Hardware Address (CHADDR) - 16 bytes			
Server Name (SNAME) - 64 bytes			
Boot Filename - 128 bytes			
DHCP Options - variable			

DHCPv4 Message Format

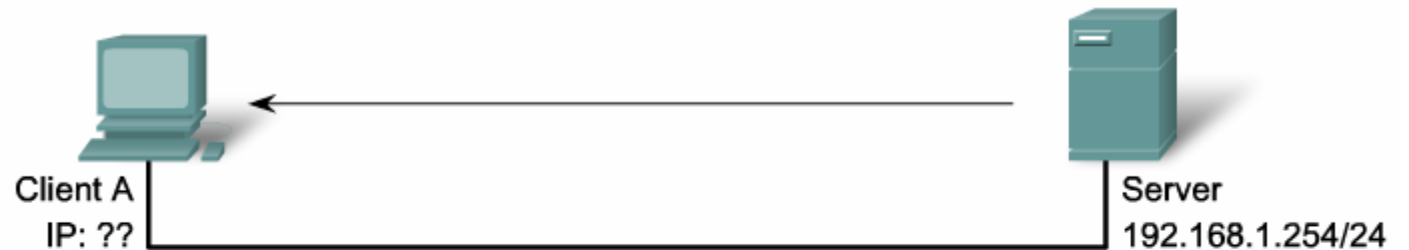
- **Seconds** - Identifies the number of seconds elapsed since a client began attempting to acquire or renew a lease. Used by DHCPv4 servers to prioritize replies when multiple client requests are outstanding.
- **Flags** - Used by a client that does not know its IPv4 address when it sends a request. Only one of the 16 bits is used, which is the broadcast flag.
- **Client IP Address** - Used by a client during lease renewal when the address of the client is valid and usable
- **Your IP Address** - Used by the server to assign an IPv4 address to the client.
- **Server IP Address** - Used by the server to identify the address of the server that the client should use for the next step in the bootstrap process.
- **Gateway IP Address** - Routes DHCPv4 messages when DHCPv4 relay agents are involved. The gateway address facilitates communications of DHCPv4 requests and replies between the client and a server that are on different subnets or networks.
- **Client Hardware Address** - Specifies the physical layer of the client.

DHCP Discovery and Offer packets



Ethernet Frame	IP	UDP	DHCPDISCOVER	
SRC MAC: MAC A	IP SRC: 0.0.0.0	UDP	CIADDR: ?	GIADDR: ?
DST MAC: FF:FF:FF:FF:FF:FF	IP DST: 255.255.255.255	67	Mask: ?	CHADDR: MAC A

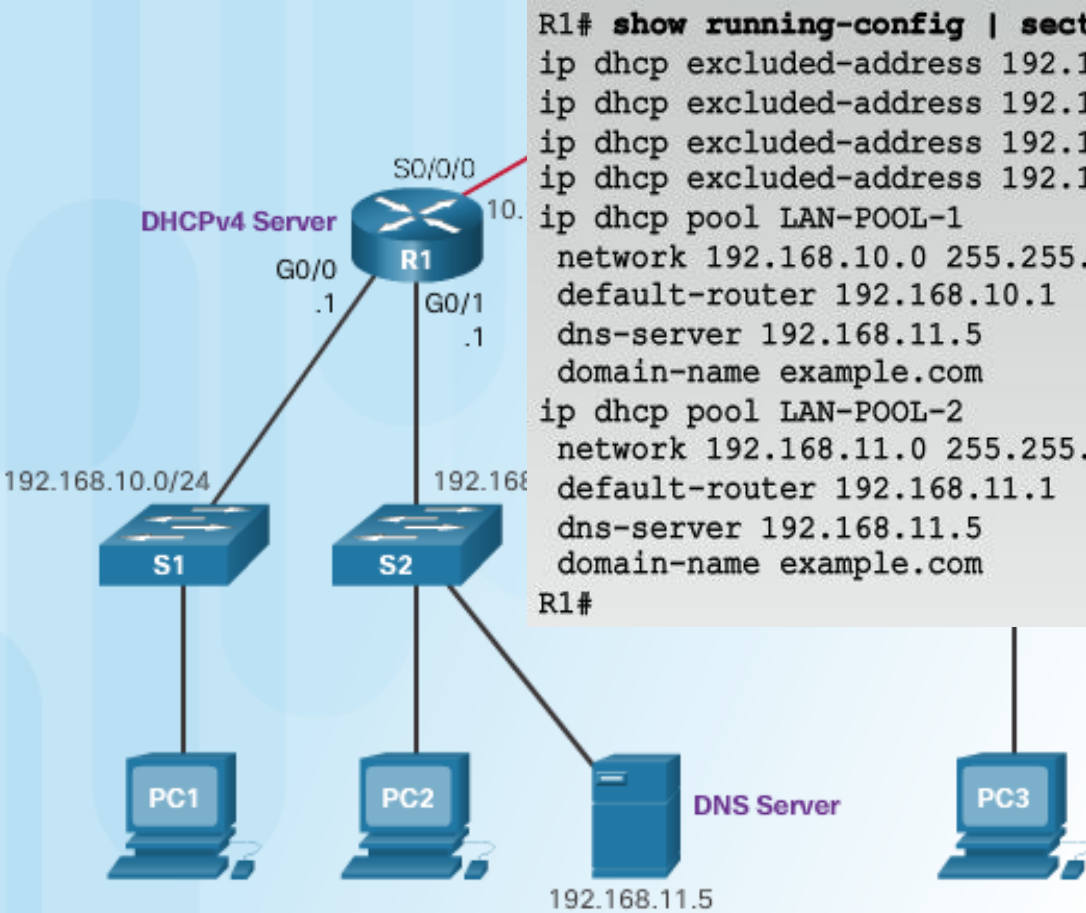
MAC: Media Access Control Address
CIADDR: Client IP Address
GIADDR: Gateway IP Address
CHADDR: Client Hardware Address



Ethernet Frame	IP	UDP	DHCP Offer	
SRC MAC: MAC Serv	IP SRC: 192.168.1.254	UDP	CIADDR: 192.168.1.10	GIADDR: ?
DST MAC: MAC A	IP DST: 192.168.1.10	68	Mask: 255.255.255.0	CHADDR: MAC A

MAC: Media Access Control Address
CIADDR: Client IP Address
GIADDR: Gateway IP Address
CHADDR: Client Hardware Address

Configuring and monitoring a DHCP Server



```
R1# show running-config | section dhcp
ip dhcp excluded-address 192.168.10.1 192.168.10.9
ip dhcp excluded-address 192.168.10.254
ip dhcp excluded-address 192.168.11.1 192.168.11.9
ip dhcp excluded-address 192.168.11.254
ip dhcp pool LAN-POOL-1
  network 192.168.10.0 255.255.255.0
  default-router 192.168.10.1
  dns-server 192.168.11.5
  domain-name example.com
ip dhcp pool LAN-POOL-2
  network 192.168.11.0 255.255.255.0
  default-router 192.168.11.1
  dns-server 192.168.11.5
  domain-name example.com
R1#
```

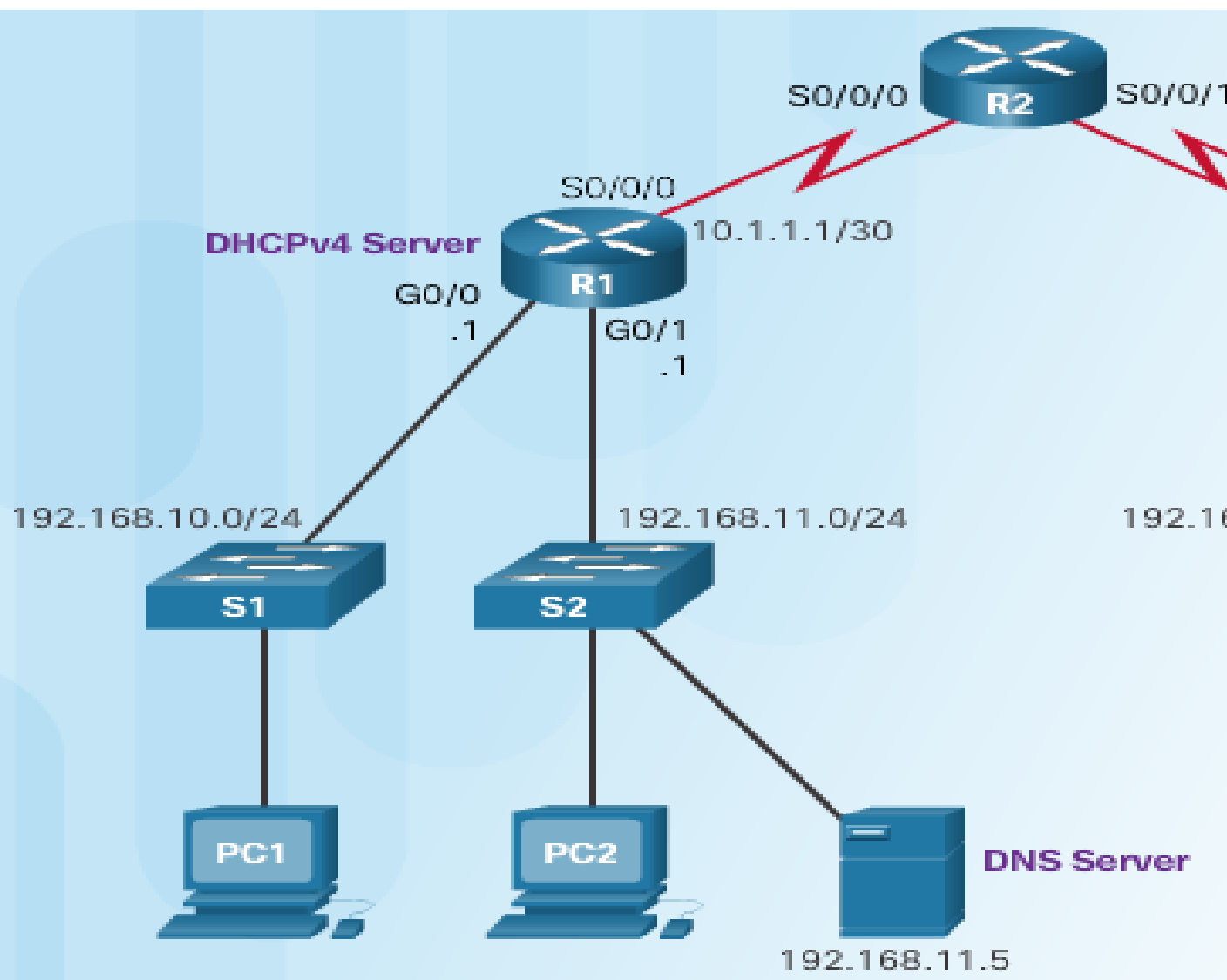
```
R1# show ip dhcp binding
Bindings from all pools not associated
IP address      Client-ID/      L
                Hardware address/
                User name
192.168.10.10    0100.e018.5bdd.35 M
192.168.11.10    0100.b0d0.d817.e6 M

R1# show ip dhcp server statistics
Memory usage      25307
Address pools      2
Database agents    0
Automatic bindings 2
Manual bindings    0
Expired bindings   0
Malformed messages 0
Secure arp entries 0

Message           Received
BOOTREQUEST        0
DHCPDISCOVER        8
DHCPREQUEST        3
DHCPDECLINE        0
DHCPRELEASE        0
DHCPINFORM         0

Message           Sent
```

DHCP Configuring



```
R1# show ip dhcp binding
Bindings from all pools not associa
IP address      Client-ID/      L
                Hardware address/
                User name
192.168.10.10   0100.e018.5bdd.35 M
192.168.11.10   0100.b0d0.d817.e6 M
```

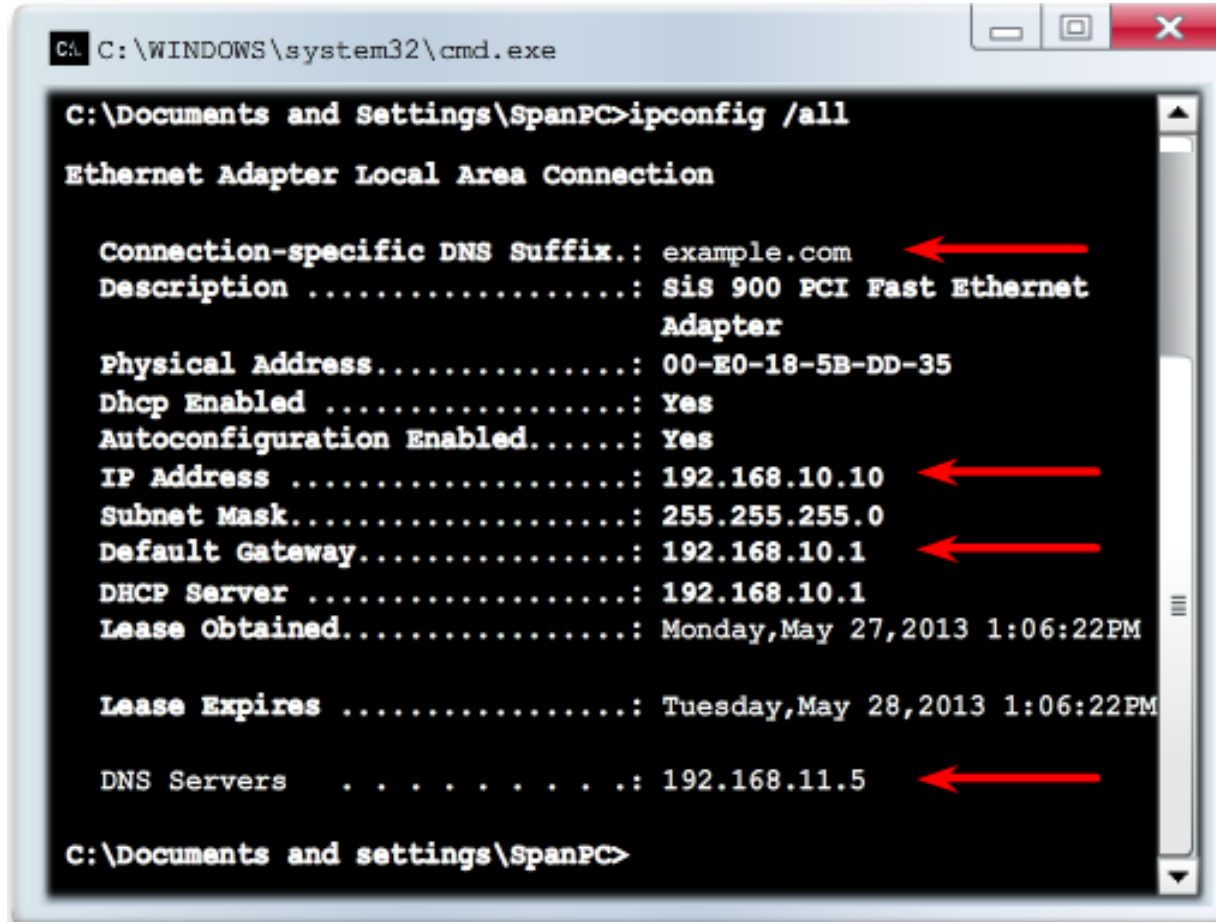
```
R1# show ip dhcp server statistics
Memory usage      25307
Address pools      2
Database agents    0
Automatic bindings 2
Manual bindings    0
Expired bindings   0
Malformed messages 0
Secure arp entries 0

Message           Received
BOOTREQUEST        0
DHCPDISCOVER        8
DHCPREQUEST         3
DHCPDECLINE         0
DHCPRELEASE         0
DHCPIFORM           0
```

```
Message           Sent
```



Verifying and renewing IP address



```
C:\WINDOWS\system32\cmd.exe
C:\Documents and Settings\SpanPC>ipconfig /all

Ethernet Adapter Local Area Connection

Connection-specific DNS Suffix.: example.com
Description .....: Sis 900 PCI Fast Ethernet Adapter
Physical Address.....: 00-E0-18-5B-DD-35
Dhcp Enabled .....: Yes
Autoconfiguration Enabled.....: Yes
IP Address .....: 192.168.10.10
Subnet Mask.....: 255.255.255.0
Default Gateway.....: 192.168.10.1
DHCP Server .....: 192.168.10.1
Lease Obtained.....: Monday, May 27, 2013 1:06:22PM

Lease Expires .....: Tuesday, May 28, 2013 1:06:22PM

DNS Servers . . . . .: 192.168.11.5

C:\Documents and settings\SpanPC>
```

```
C:\Documents and Settings\Administrator>ipconfig /release
```

Windows IP Configuration

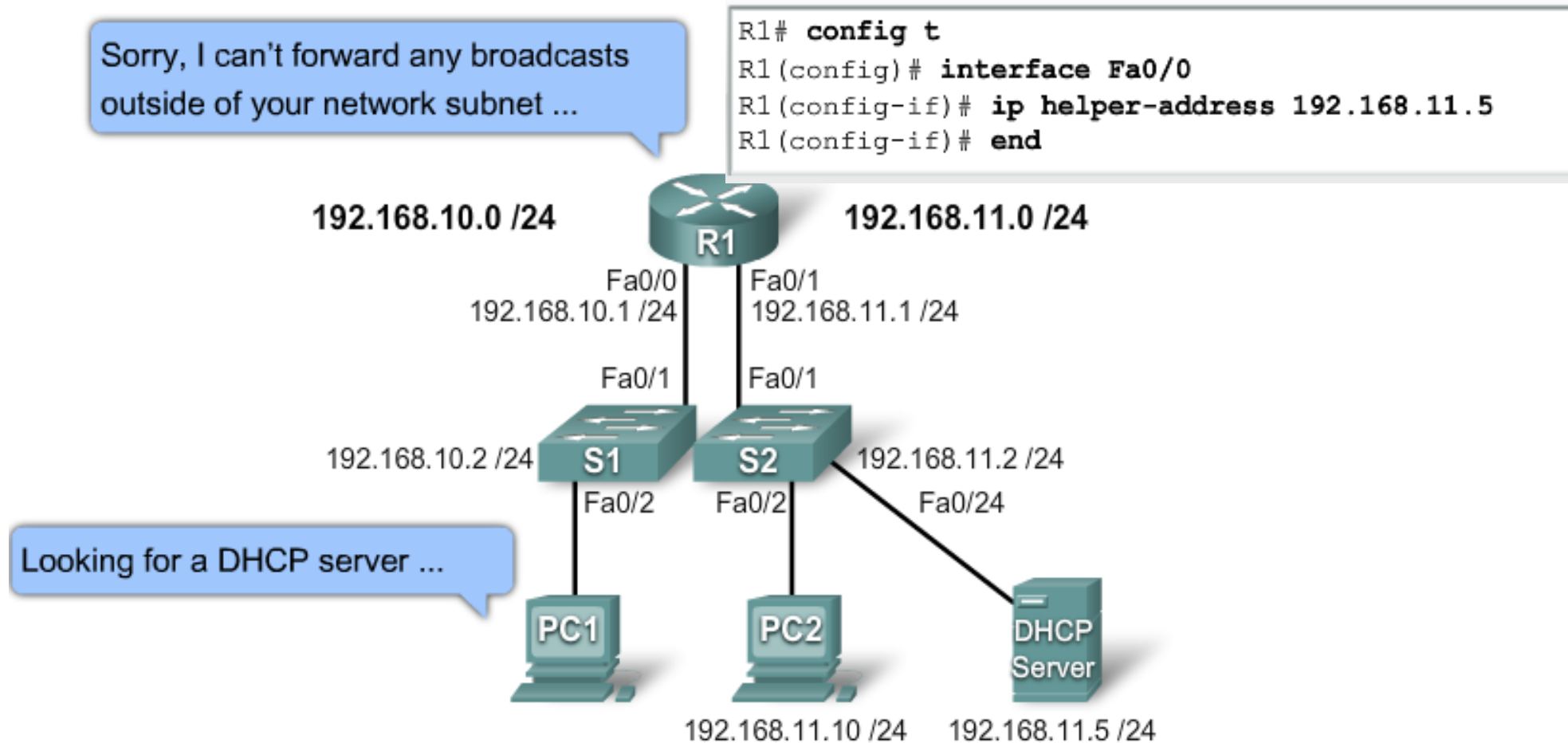
Ethernet adapter Local Area Connection:

```
Connection-specific DNS Suffix.:
IP Address .....: 0.0.0.0
Subnet Mask.....: 0.0.0.0
Default Gateway.....:
```

```
C:\Documents and Settings\Administrator>ipconfig /renew
```

```
Connection-specific DNS Suffix.:
IP Address .....: 192.168.10.11
Subnet Mask.....: 255.255.255.0
Default Gateway.....: 192.168.10.1
```

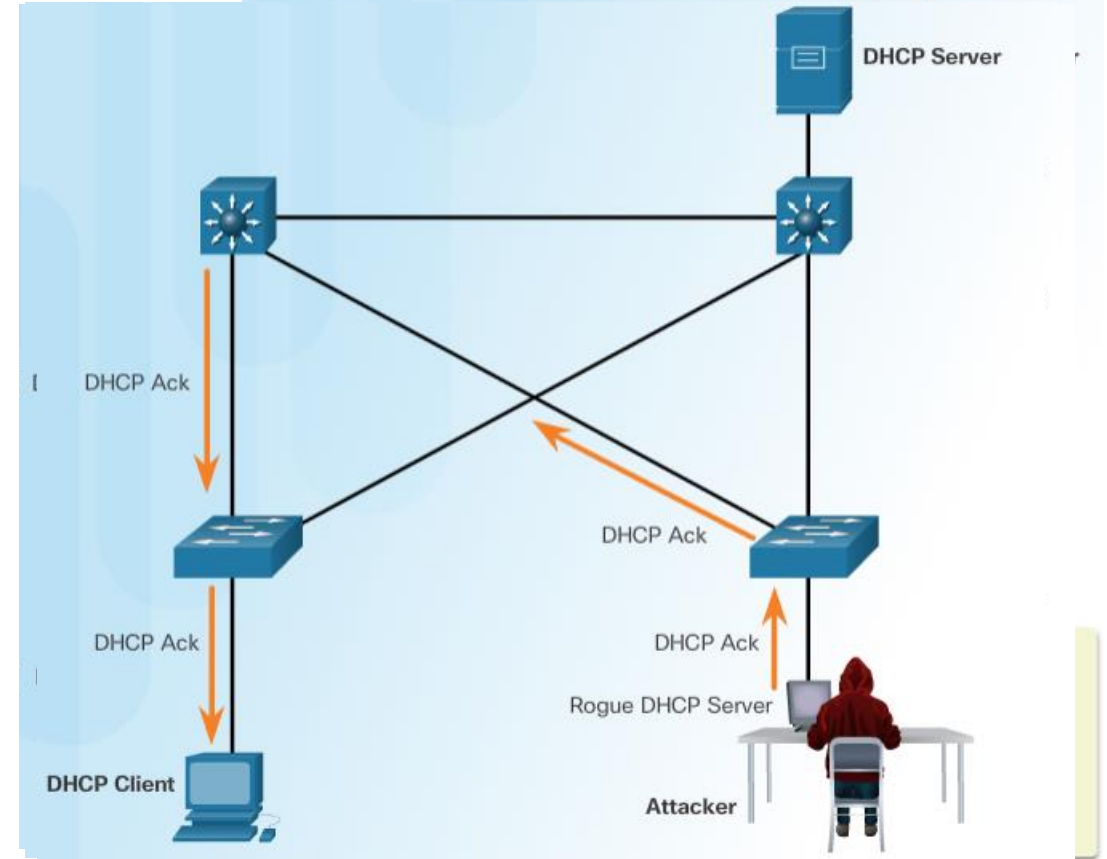
DHCP Relay



DHCP Spoofing Attack

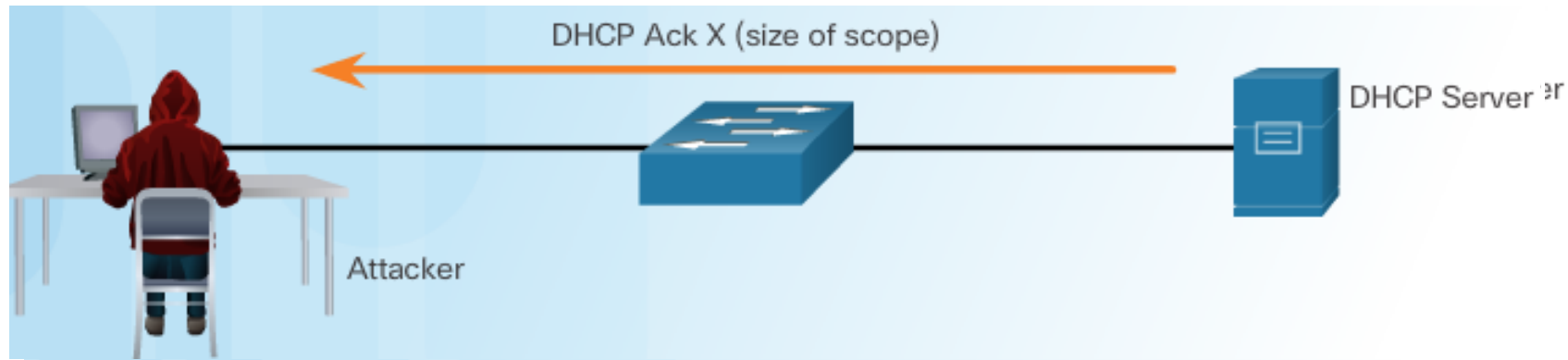
A DHCP spoofing attack occurs when a rogue DHCP server is connected to the network and provides false IP configuration parameters to legitimate clients. A rogue server can provide a variety of misleading information:

- **Wrong default gateway** - Attacker provides an invalid gateway or the IP address of its host to create a man-in-the-middle attack. This may go entirely undetected as the intruder intercepts the data flow through the network.
- **Wrong DNS server** - Attacker provides an incorrect DNS server address pointing the user to a nefarious website.
- **Wrong IP address** - Attacker provides an invalid default gateway IP address and creates a DoS attack on the DHCP client.



DHCP Starvation Attack

- The goal of this attack is to **create a DoS for connecting clients**. DHCP starvation attacks require an attack tool such as **Gobbler**.
- Gobbler has the ability to **look at the entire scope of leasable IP addresses** and tries to lease them all. Specifically, it creates DHCP discovery messages with bogus MAC addresses.

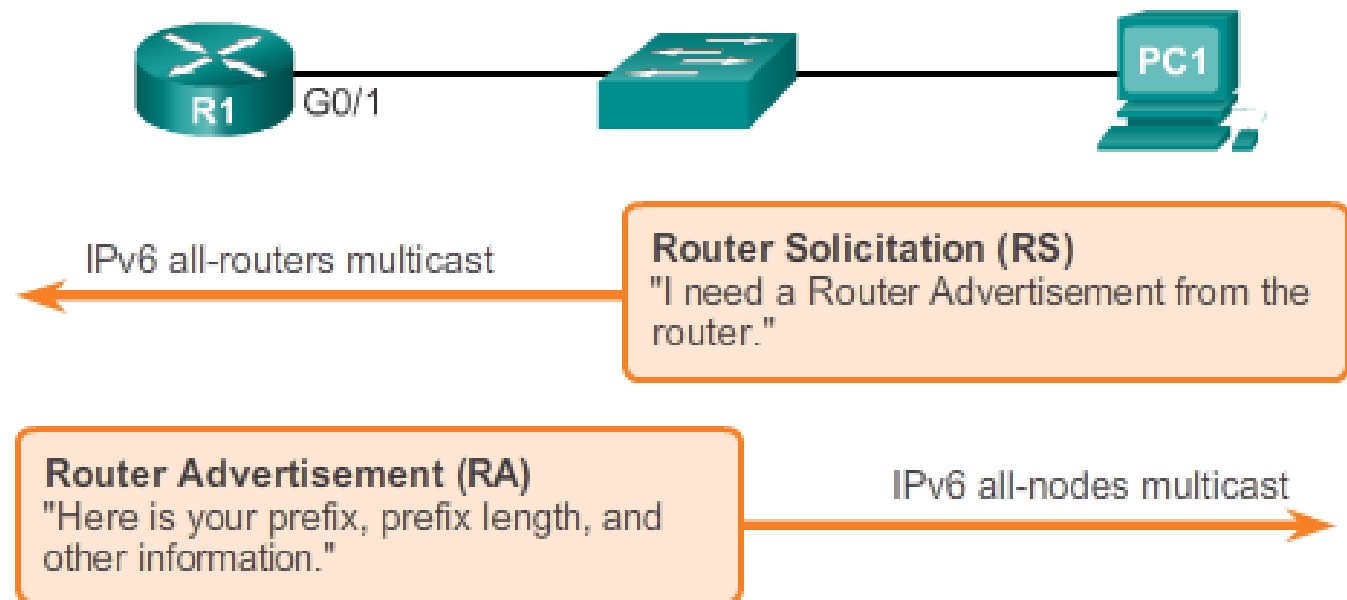


IPv6 dynamic global unicast addresses configuration

There are two methods in which IPv6 global unicast addresses can be assigned dynamically:

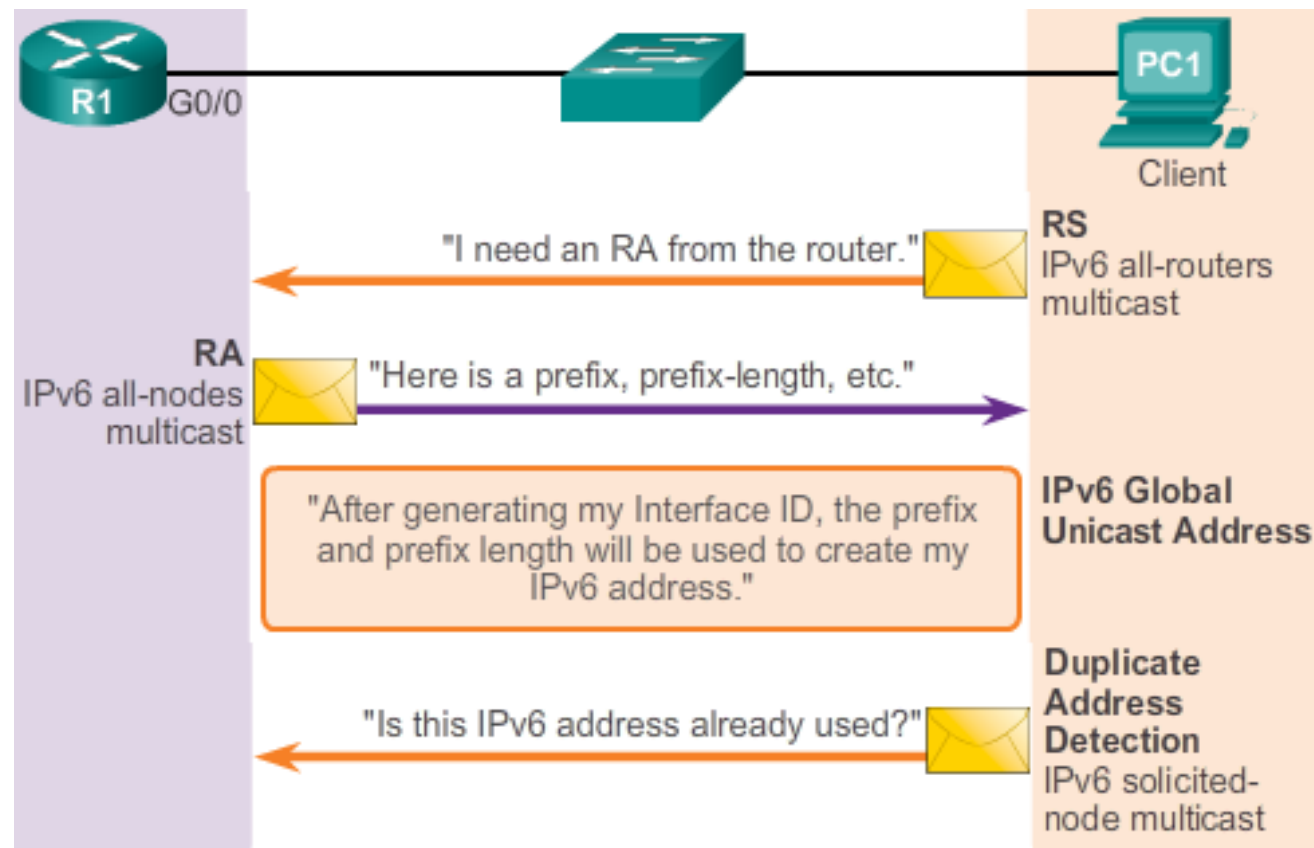
- **Dynamic Host Configuration Protocol** for IPv6 (Stateful DHCPv6)
- **Stateless Address Autoconfiguration** (SLAAC)

SLAAC is a method in which a device can obtain an IPv6 global unicast address **without** the services of a DHCPv6 server.



Router Solicitation and Advertisement

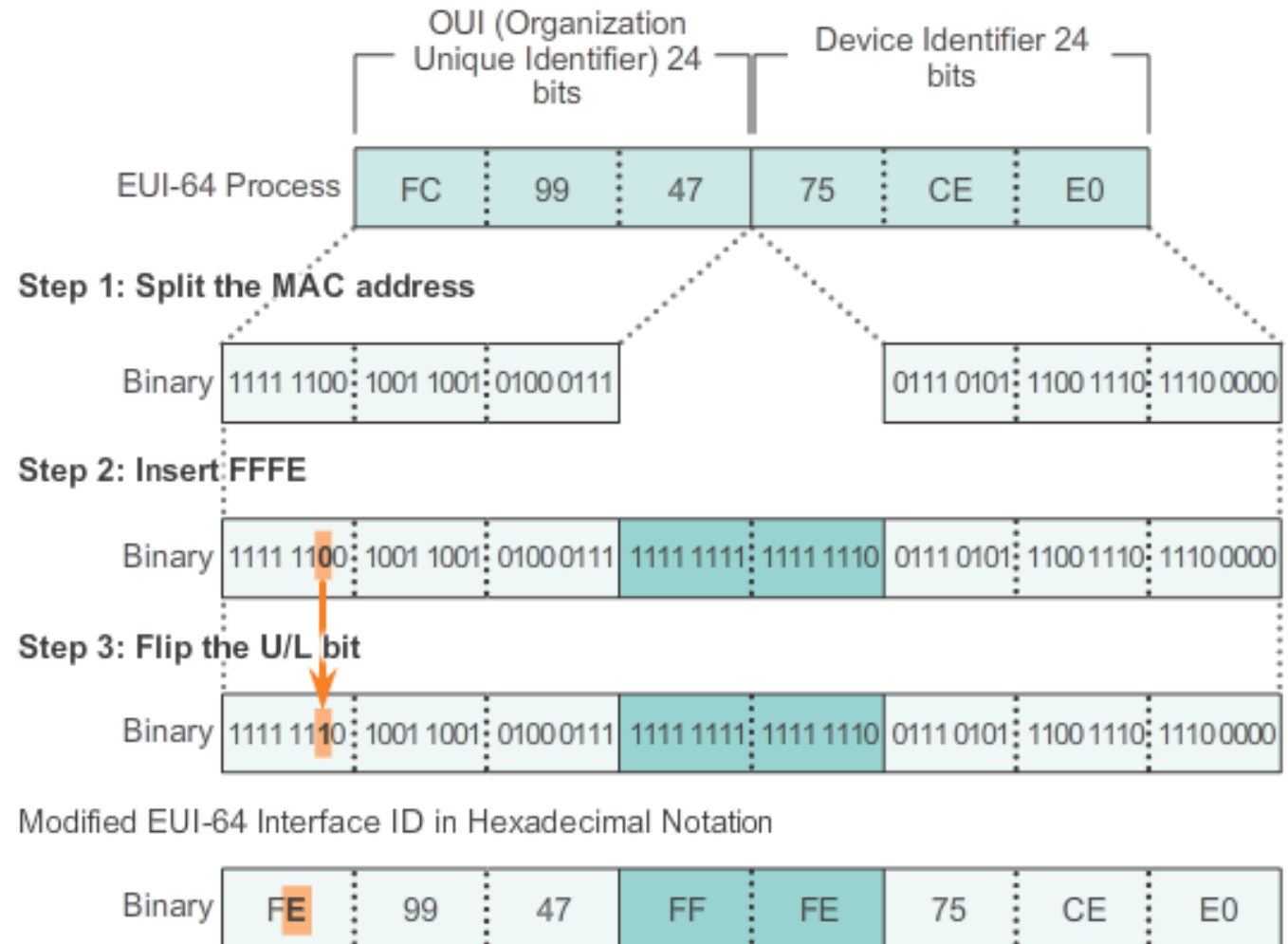
- **Router Solicitation (запит) (RS) message** - When a client is configured to obtain its addressing information automatically using SLAAC, the **client sends** an RS message to the all-routers multicast address **FF02::2**.
- **Router Advertisement (RA) message** - RA messages are sent by routers to provide addressing information to clients configured to obtain their IPv6 addresses automatically. The RA message includes the **prefix and prefix length of the local segment**. A router sends an RA message periodically, or in response to an RS message. By default, Cisco routers send RA messages every **200 seconds**. RA messages are always sent to the IPv6 all-nodes multicast address **FF02::1**.



The ways for creating PC own unique IID

There are two ways PC1 can create its own unique IID:

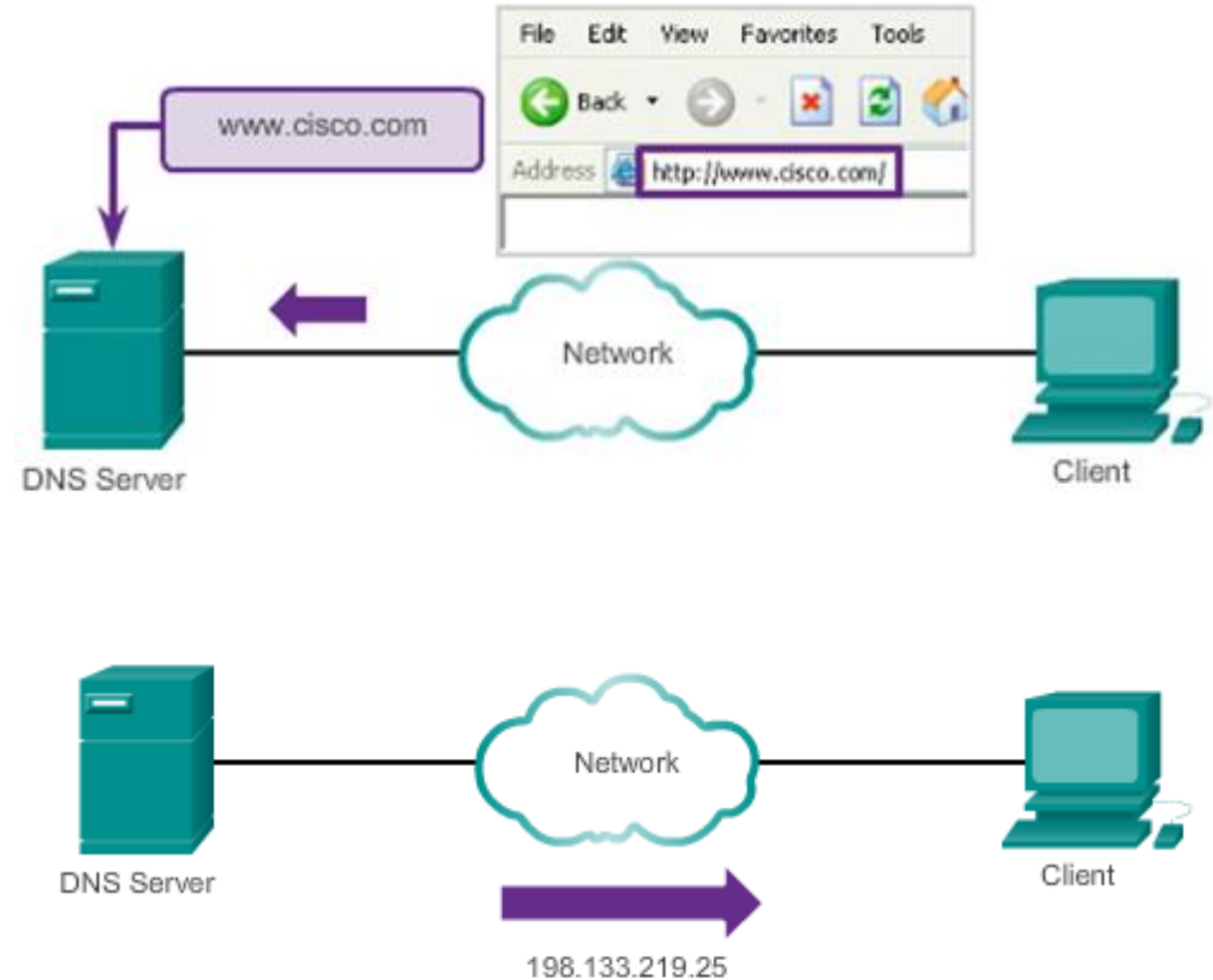
- **EUI-64** - Using the EUI-64 process, PC1 will create an IID using its 48-bit MAC address.
- **Randomly generated** - The 64-bit IID can be a random number generated by the client operating system.



DNS

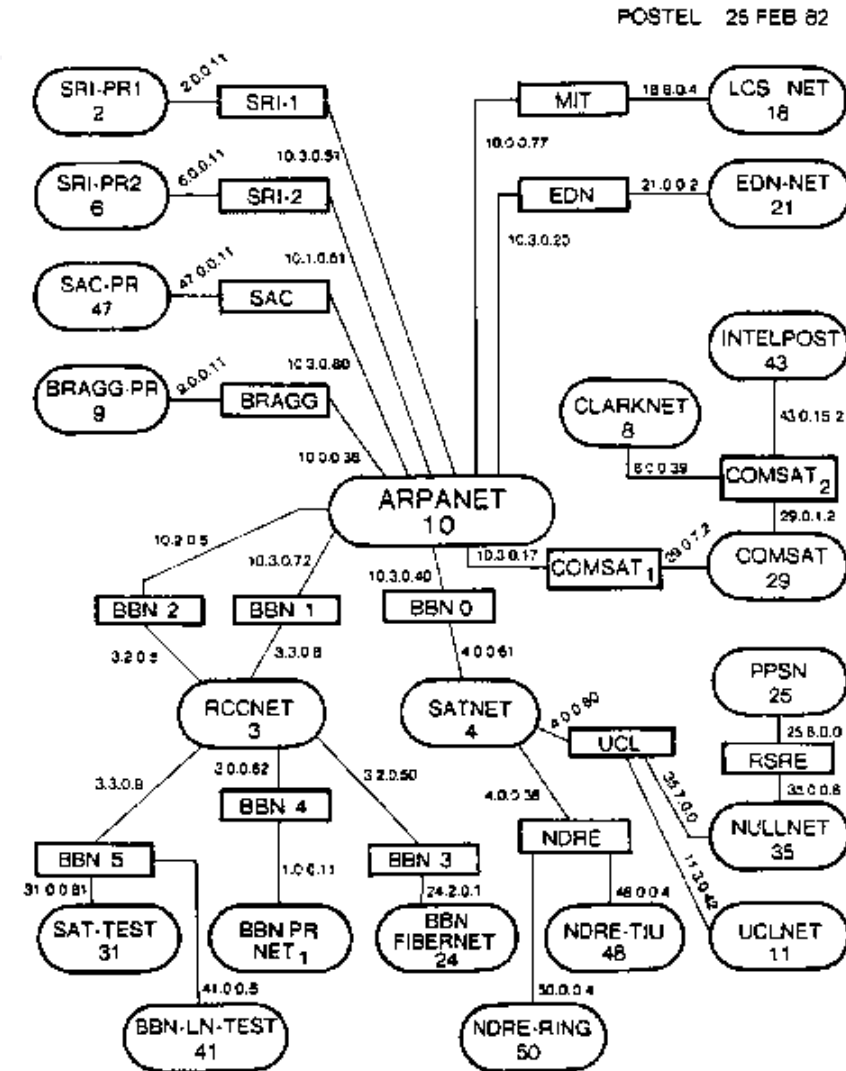
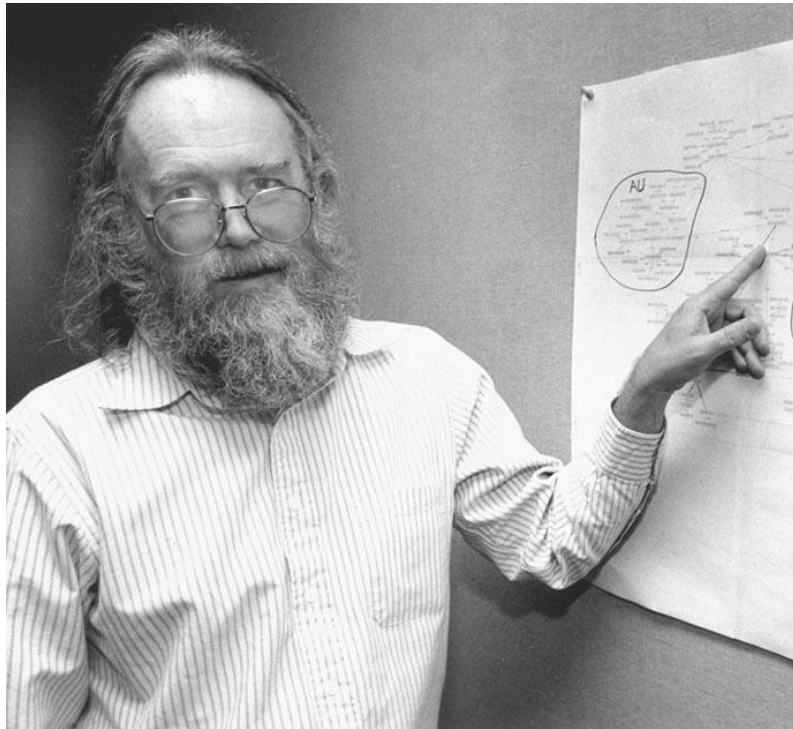
Domain Name Service

- The Domain Name System (DNS) was created for **domain name to address resolution** for these networks.
- DNS uses a **distributed set of servers** to resolve the names associated with these numbered addresses.
- The DNS protocol defines an automated service that matches resource names with the required numeric network address. It includes the format for queries, responses, and data.



DNS History

Jonathan Bruce Postel



File hosts

%SystemRoot%\system32\drivers\etc\hosts

```
# Copyright (c) 1993-2009 Microsoft Corp.
#
# This is a sample HOSTS file used by Microsoft TCP/IP for Windows.
#
# This file contains the mappings of IP addresses to host names. Each
# entry should be kept on an individual line. The IP address should
# be placed in the first column followed by the corresponding host name.
# The IP address and the host name should be separated by at least one
# space.
#
# Additionally, comments (such as these) may be inserted on individual
# lines or following the machine name denoted by a '#' symbol.
#
# For example:
#
#       102.54.94.97      rhino.acme.com          # source server
#       38.25.63.10      x.acme.com              # x client host

# localhost name resolution is handled within DNS itself.
#       127.0.0.1        localhost
#       ::1              localhost
```

DNS Hierarchy

Examples top-level domains:

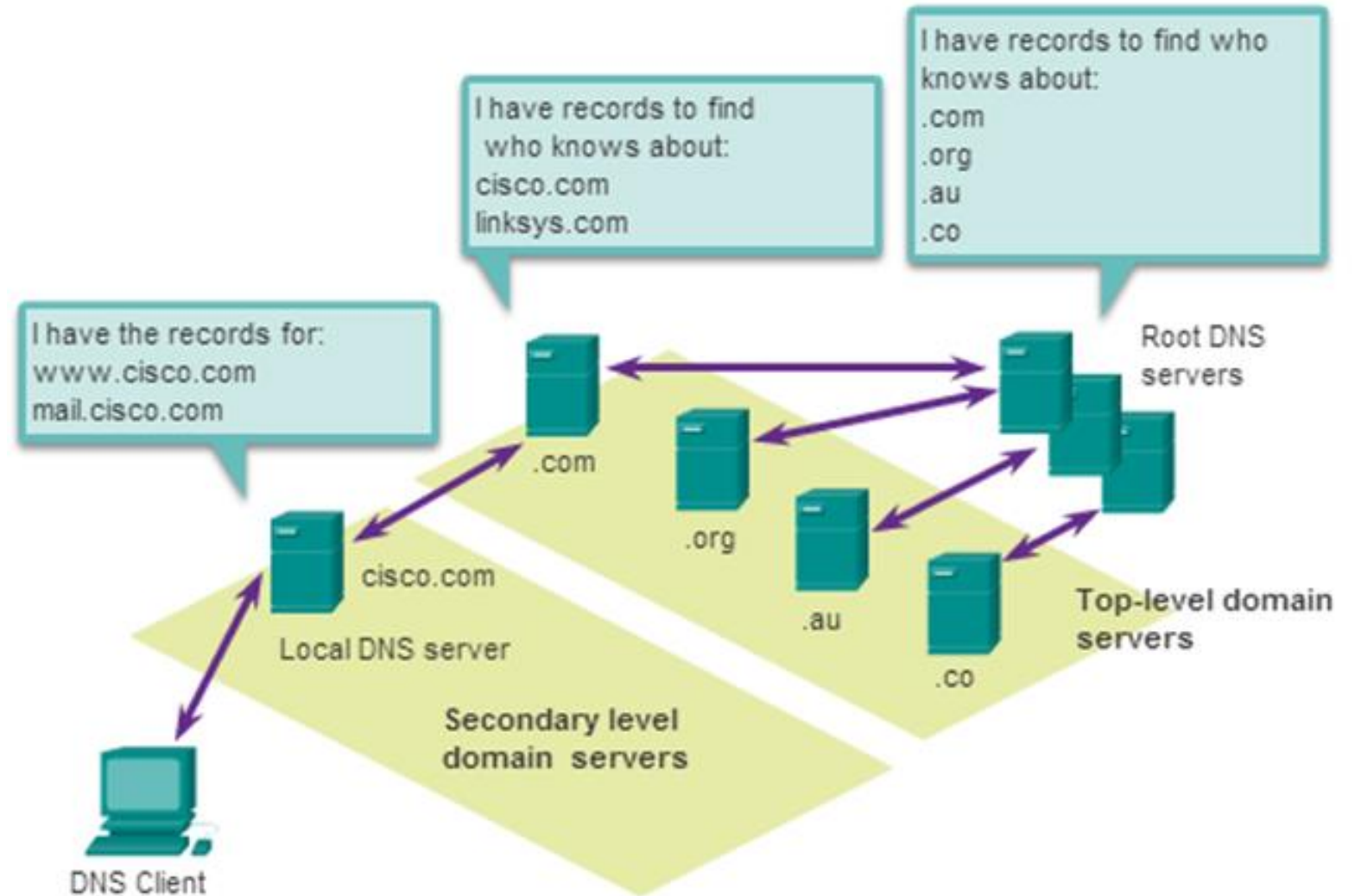
.au - Australia

.co - Colombia

.com - business or industry

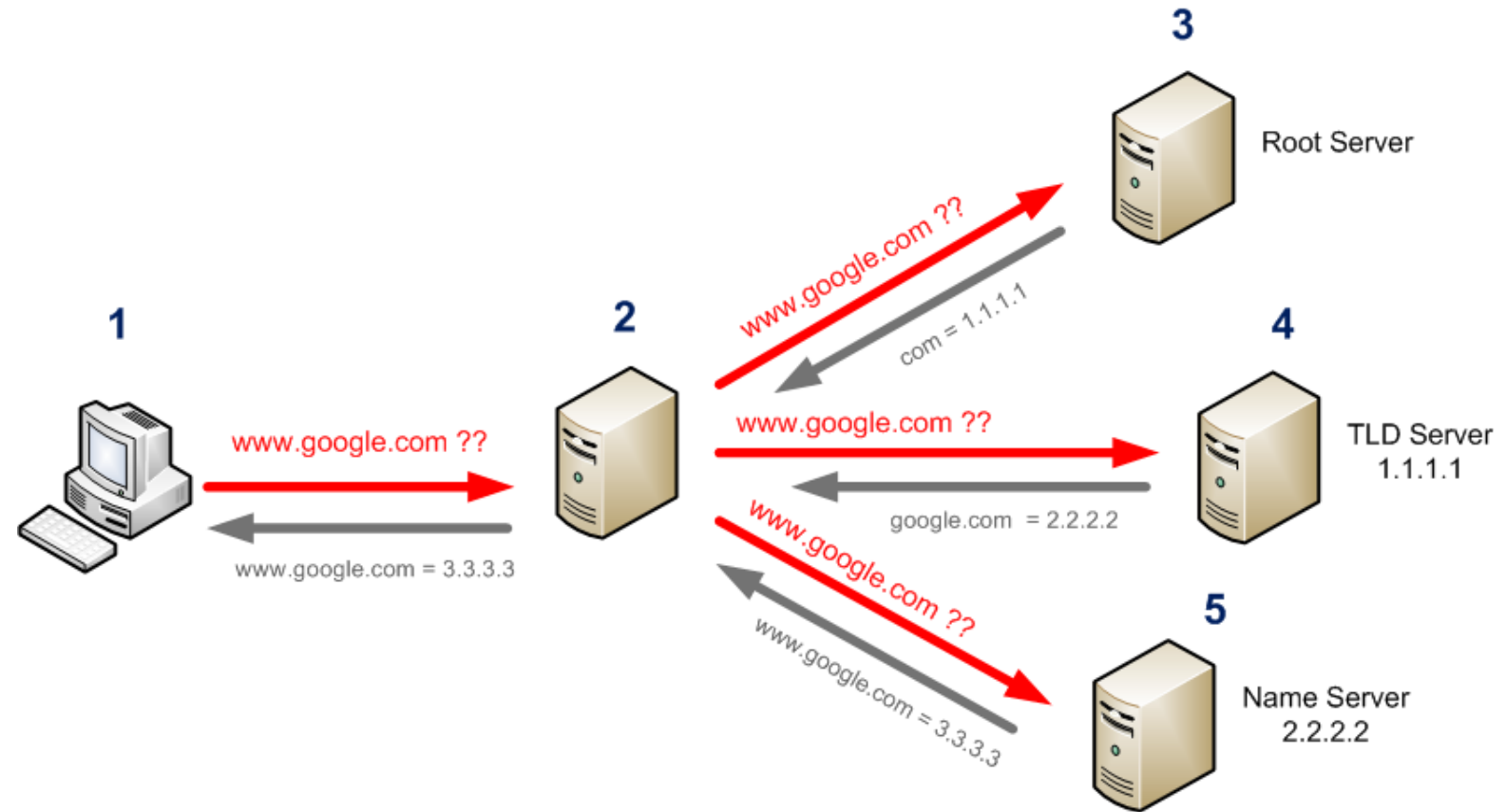
.jp - Japan

.org - non-profit organization



DNS Server Types

- Recursive resolver
- Root nameserver
- TLD nameserver (top-level domain)
- Authoritative nameserver

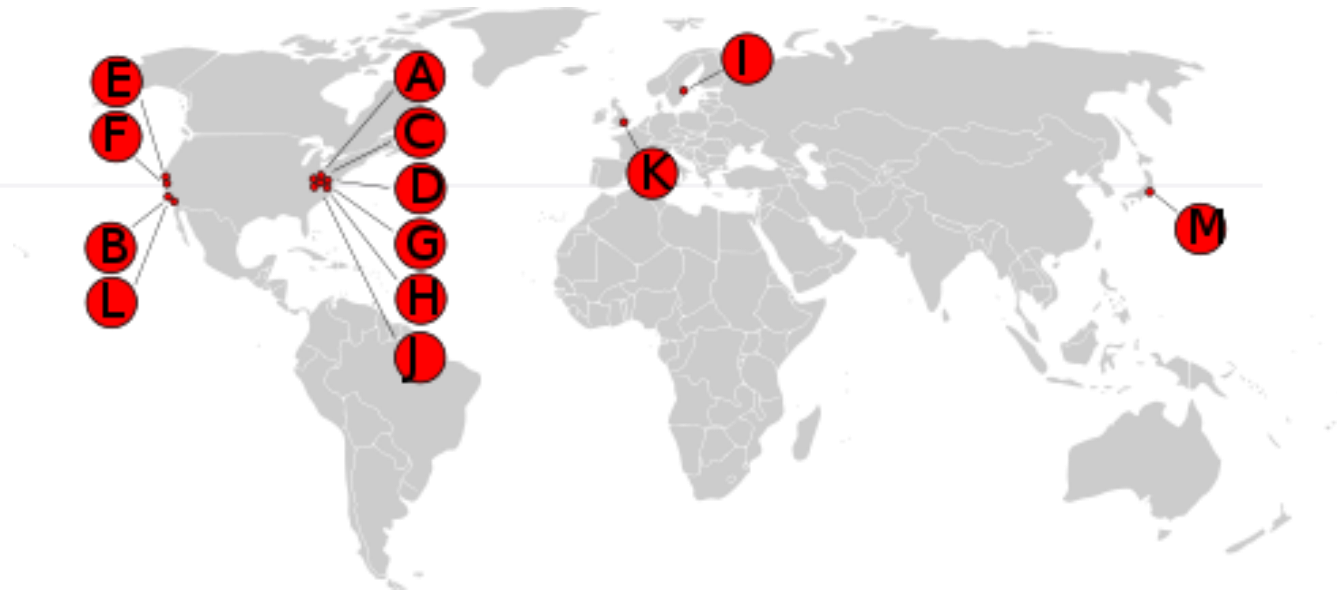


Recursive Resolver

- A recursive resolver (also known as a DNS recursor) is the first step in a DNS query.
- The recursive resolver acts as a **middleman** between a client and a DNS nameserver.
- After receiving a DNS query from a web client, a recursive resolver will either respond with cached data, or send a request to a root nameserver, followed by another request to a TLD nameserver, and then one last request to an authoritative nameserver.
- After receiving a response from the authoritative nameserver containing the requested IP address, the recursive resolver then sends a response to the client.

Root nameserver

- The 13 DNS root nameservers are known to every recursive resolver, and they are the first step in a recursive resolver's quest for DNS records.
- A root server accepts a recursive resolver's query which includes a domain name, and the root nameserver responds by directing the recursive resolver to a TLD nameserver, based on the extension of that domain (.com, .net, .org, etc.).
- The root nameservers are overseen by a nonprofit called the Internet Corporation for Assigned Names and Numbers (ICANN).
- <https://root-servers.org/>



HOSTNAME	IP ADDRESSES	MANAGER
a.root-servers.net	198.41.0.4, 2001:503:ba3e::2:30	VeriSign, Inc.
b.root-servers.net	199.9.14.201, 2001:500:200::b	University of Southern California (ISI)
c.root-servers.net	192.33.4.12, 2001:500:2::c	Cogent Communications
d.root-servers.net	199.7.91.13, 2001:500:2d::d	University of Maryland
e.root-servers.net	192.203.230.10, 2001:500:a8::e	NASA (Ames Research Center)
f.root-servers.net	192.5.5.241, 2001:500:2f::f	Internet Systems Consortium, Inc.
g.root-servers.net	192.112.36.4, 2001:500:12::d0d	US Department of Defense (NIC)
h.root-servers.net	198.97.190.53, 2001:500:1::53	US Army (Research Lab)
i.root-servers.net	192.36.148.17, 2001:7fe::53	Netnod
j.root-servers.net	192.58.128.30, 2001:503:c27::2:30	VeriSign, Inc.
k.root-servers.net	193.0.14.129, 2001:7fd::1	RIPE NCC
l.root-servers.net	199.7.83.42, 2001:500:9f::42	ICANN
m.root-servers.net	202.12.27.33, 2001:dc3::35	WIDE Project

TLD and Authoritative nameserver

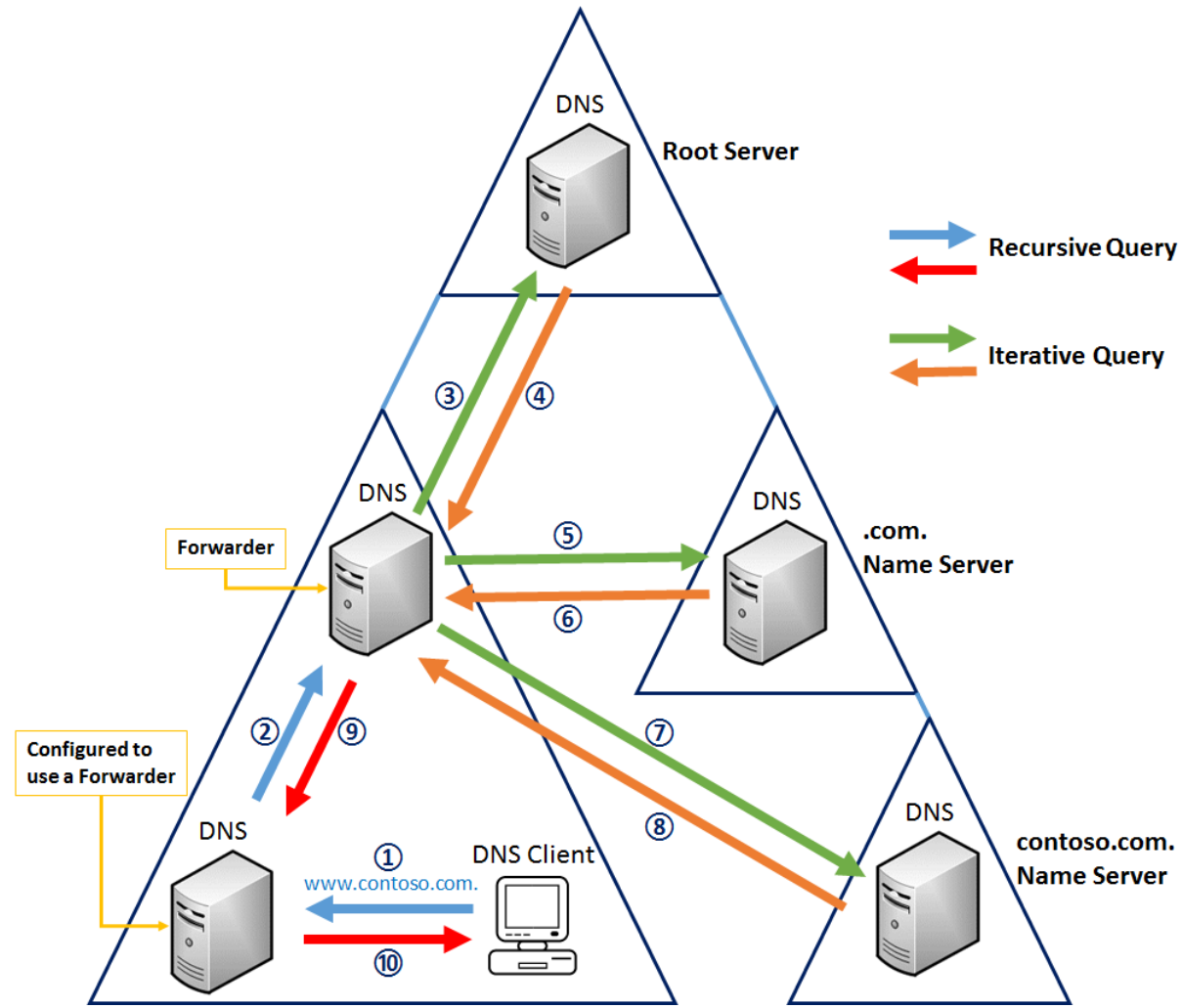
TLD nameserver

- A TLD nameserver maintains information for all the domain names that share a common domain extension, such as .com, .net, or whatever comes after the last dot in a url.
- Management of TLD nameservers is handled by the Internet Assigned Numbers Authority (IANA), which is a branch of ICANN. The IANA breaks up the TLD servers into two main groups:
 - **Generic top-level domains:** These are domains that are not country specific, some of the best-known generic TLDs include .com, .org, .net, .edu, and .gov.
 - **Country code top-level domains:** These include any domains that are specific to a country or state. Examples include .uk, .us, .ru, and .jp.

Authoritative nameserver

- The authoritative nameserver is usually the resolver's last step in the journey for an IP address.
- The authoritative nameserver **contains information specific to the domain name it serves** (e.g. google.com) and it can provide a recursive resolver with the IP address of that server found in the [DNS A record](#).

DNS server lookup process



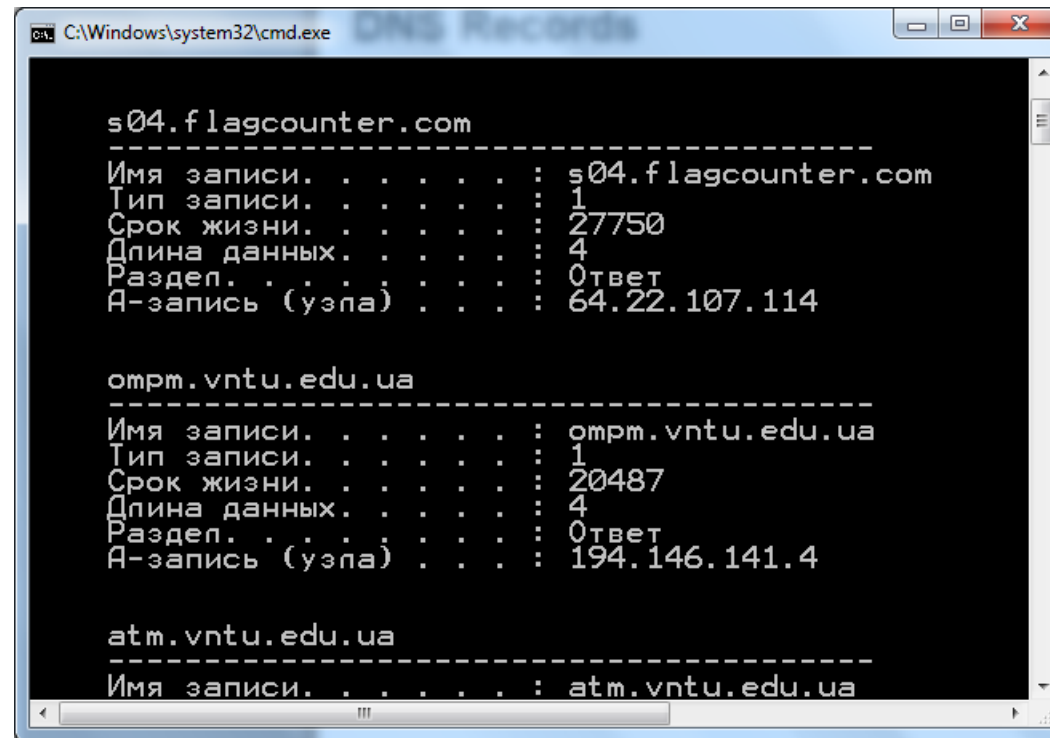
DNS Records

- DNS server stores different types of resource records used to resolve names
- Any record contain the **name**, **address**, and **type**
- Some record types:
 - **A** - an end device address
 - **NS** - an authoritative name server
 - **CNAME** - the canonical name for an alias; used when multiple services have the single network address but each service has its own entry in DNS
 - **MX** - mail exchange record; maps a domain name to a list of mail exchange servers
 - **PTR** - used as "reverse records" - to map IP addresses to domain names

Common DNS Record Types	
Record	Description
A	Address record (IPv4)
AAAA	Address record (IPv6)
CNAME	Canonical Name record
MX	Mail Exchanger record
NS	Nameserver record
PTR	Pointer record
SOA	Start of Authority record
SRV	Service Location record
TXT	Text record

DNS Records

- If server unable to resolve the name using its stored records, it contacts other servers
- Server temporarily stores the numbered address that matches the name in cache memory
- Windows **ipconfig /displaydns** displays all cached DNS



```
C:\Windows\system32\cmd.exe DNS Records

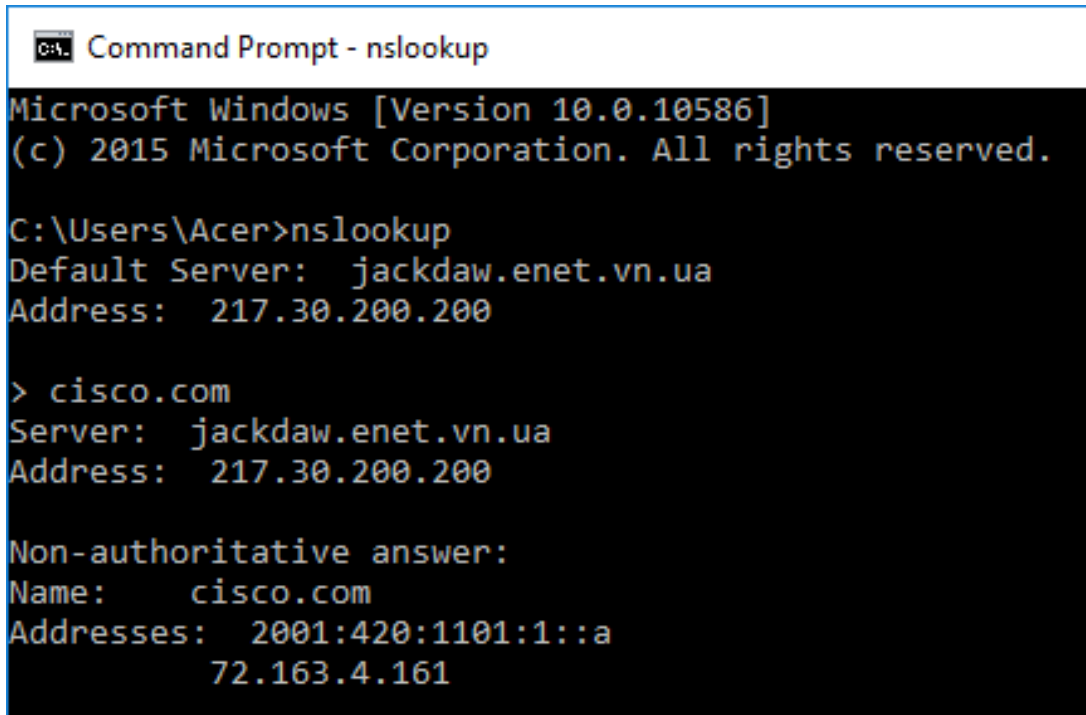
-----
s04.flagcounter.com
-----
Имя записи. . . . . : s04.flagcounter.com
Тип записи. . . . . : 1
Срок жизни. . . . . : 27750
Длина данных. . . . : 4
Раздел. . . . . : 0твет
А-запись (уэпа) . . . : 64.22.107.114

-----
ompm.vntu.edu.ua
-----
Имя записи. . . . . : ompm.vntu.edu.ua
Тип записи. . . . . : 1
Срок жизни. . . . . : 20487
Длина данных. . . . : 4
Раздел. . . . . : 0твет
А-запись (уэпа) . . . : 194.146.141.4

-----
atm.vntu.edu.ua
-----
Имя записи. . . . . : atm.vntu.edu.ua
```

Nslookup

- Operating system utility called **nslookup** allows the user to manually query the name servers to resolve a given host name
- Utility can be used to troubleshoot name resolution issues and to verify the current status of the name servers



```
Command Prompt - nslookup
Microsoft Windows [Version 10.0.10586]
(c) 2015 Microsoft Corporation. All rights reserved.

C:\Users\Acer>nslookup
Default Server:  jackdaw.enet.vn.ua
Address:  217.30.200.200

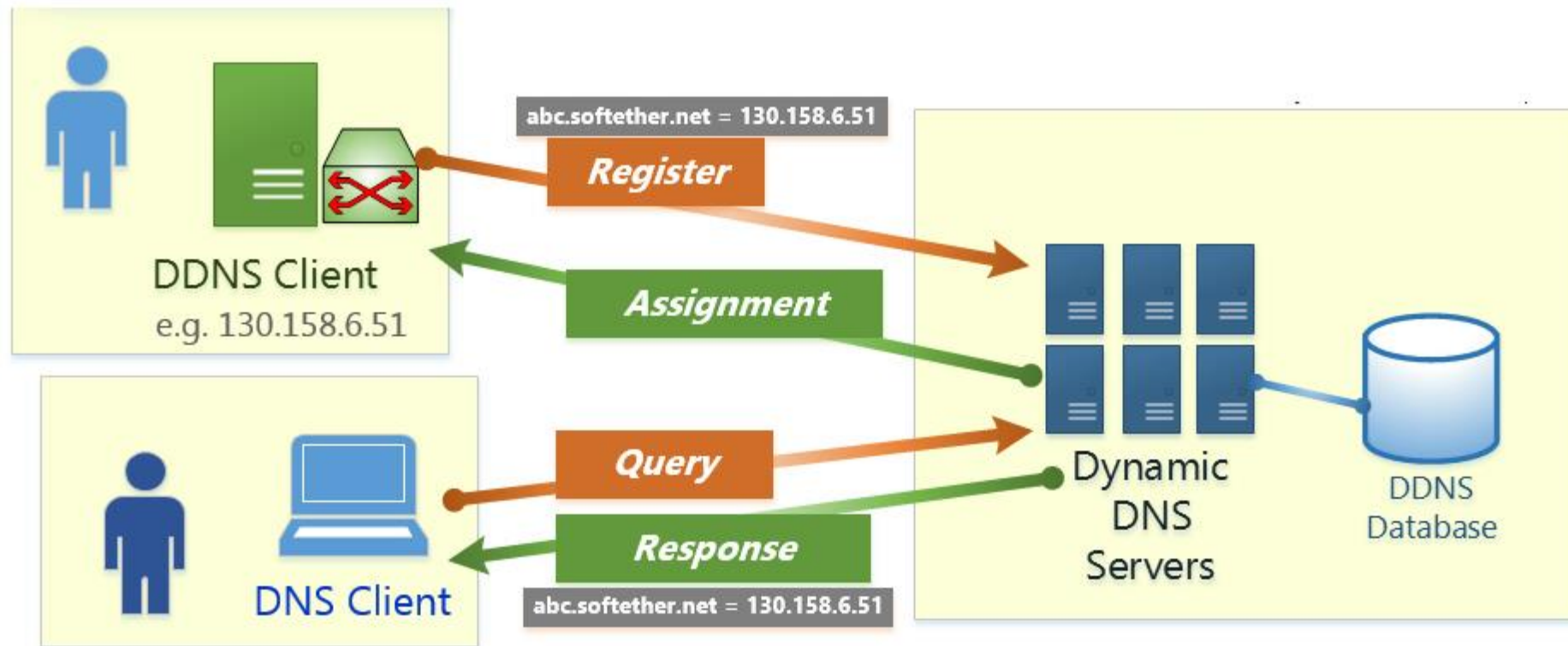
> cisco.com
Server:  jackdaw.enet.vn.ua
Address:  217.30.200.200

Non-authoritative answer:
Name:     cisco.com
Addresses: 2001:420:1101:1::a
          72.163.4.161
```

- **nslookup cisco.com** – request for default dns server for resolving domain name (cisco.com)
- **nslookup cisco.com 8.8.8.8** – request for explicit pointed dns server (8.8.8.8) for resolving domain name (cisco.com)
- **nslookup -type=mx cisco.com** – request ip address of the domain (cisco.com) mail servers
- **nslookup -type=any cisco.com 8.8.8.8** - request of any ip address for domain (cisco.com)
- **nslookup 91.206.200.104** - reverse lookup

Dynamic DNS

- Allows a user or organization to register an IP address with a domain name as in DNS.
- When the IP address of the mapping changes, the new mapping can be propagated through the DNS almost **instantaneously**.



OpenDNS

- **OpenDNS** (<https://www.opendns.com/>) is a company and service that extends the Domain Name System (DNS) by adding features such as phishing protection and optional content filtering in addition to DNS lookup, if its DNS servers are used.
- The OpenDNS Global Network processes an estimated **100 billion** DNS queries daily from **85 million users** through 25 data centers worldwide. DNS-servers addresses:
 - **208.67.222.222** (resolver1.opendns.com)
 - **208.67.220.220** (resolver2.opendns.com)
 - **208.67.222.220** (resolver3.opendns.com)
 - **208.67.220.222** (resolver4.opendns.com)¹



DNS Attacks

DNS servers resolve names to IP addresses and are a major target of attackers. Some DNS exploits are:

- **DNS Open Resolvers** (public name servers)
- **DNS Stealth Attacks**
- **DNS Shadowing Attacks** – hijacked domains are used to create subdomains which are used to resolve to malicious web sites
- **DNS Tunneling Attacks** - hides malicious instructions inside DNS queries and responses



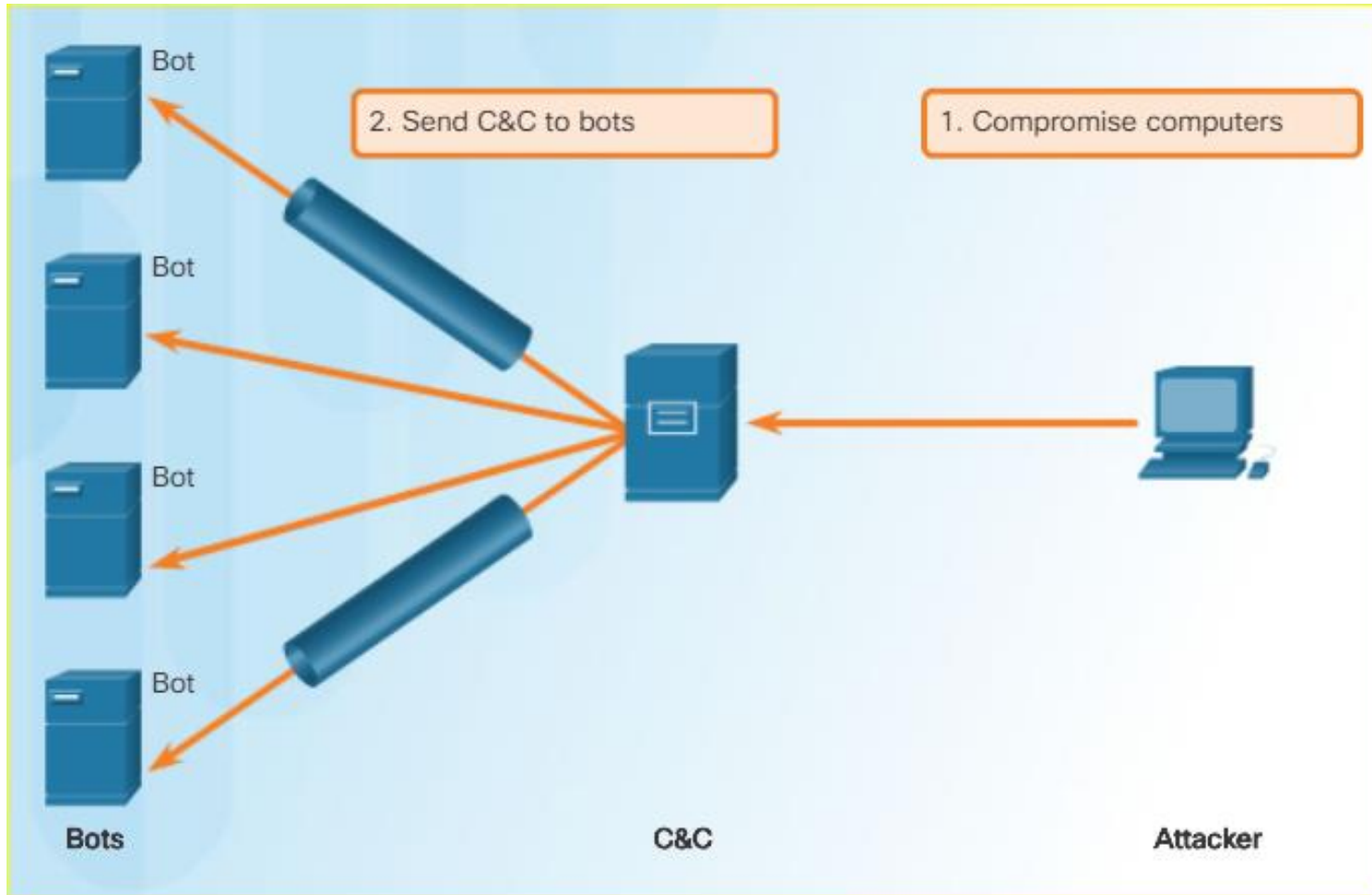
DNS Open Resolvers

- Many organizations use the services of publicly open DNS servers such as Google DNS (8.8.8.8) to provide responses to queries. This type of DNS server is called an **open resolver**.
- DNS open resolvers are vulnerable to multiple malicious activities, including:
 - **DNS cache poisoning attacks** - Threat actors send spoofed, falsified RR information to a DNS resolver to redirect users from legitimate sites to malicious sites.
 - **DNS amplification and reflection attacks** – Threat actors send DNS messages to the open resolvers using the IP address of a target host (victim).
 - **DNS resource utilization attacks** - A DoS attack that consumes the resources of the DNS open resolvers. Examples of such resources include CPU, memory, and socket buffers.

DNS stealth techniques

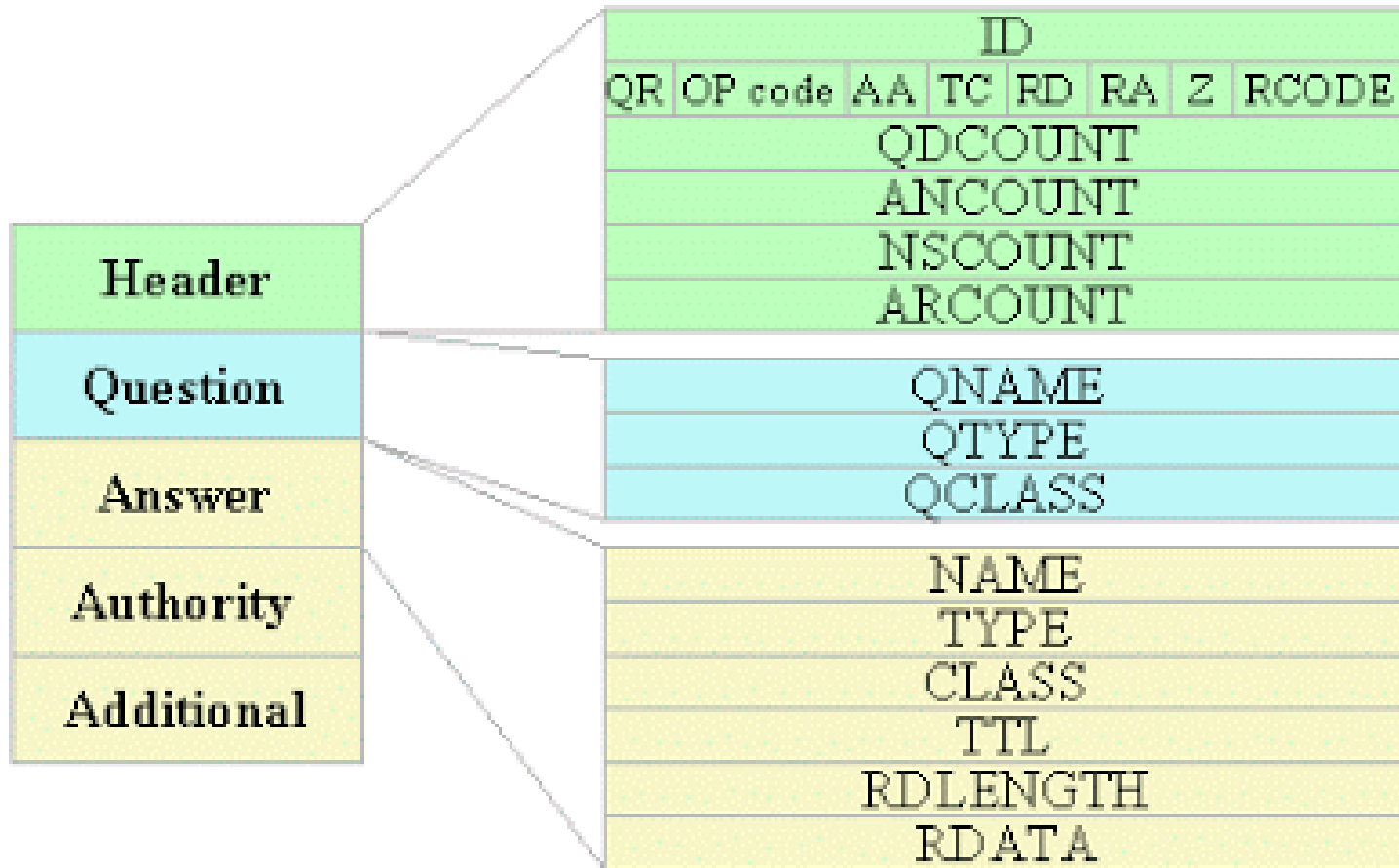
- **Fast flux** – Threat actors use this technique to hide their phishing and malware delivery sites behind a quickly-changing network of compromised DNS hosts. The DNS IP addresses are continuously changed within minutes. Botnets often employ Fast Flux techniques to effectively hide (i.e., cloak) malicious servers from being detected.
- **Double IP flux** - Threat actors use this technique to rapidly change the hostname to IP address mappings and to also change the authoritative name server. This increases the difficulty of identifying the source of the attack.
- **Domain generation algorithms** – Threat actors use this technique in malware to randomly generate domain names that can then be used as rendezvous points to their command and control (C&C) servers.

DNS Tunneling



- Threat actors who use DNS tunneling place non-DNS traffic within DNS traffic.
- This method often circumvents security solutions. For the threat actor to use DNS tunneling, the different types of DNS records such as TXT, MX, SRV, NULL, A, or CNAME are altered.

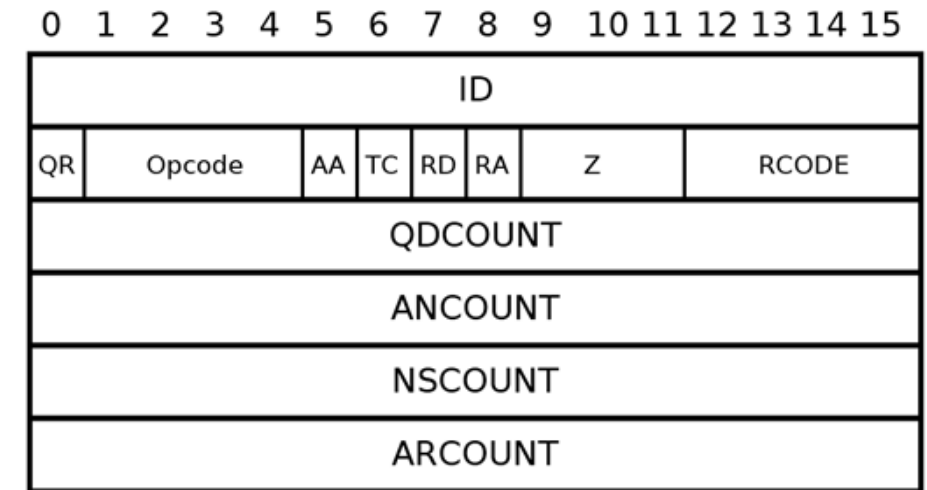
DNS Packet structure



- Header – 12 octets
- Question – domain name name for resolving
- Answer – IP address or addresses as a result of resolving
- Authority - information about authoritative servers were used to obtain information included in the answer section.
- Additional - additional entries that relate to the request, but are not strictly answers to the question

DNS Header structure

- ID - A 16 bit identifier assigned by the program that generates any kind of query. This identifier is copied the corresponding reply and can be used by the requester to match up replies to outstanding queries.
- QR - A one bit field that specifies whether this message is a query (0), or a response (1)
- OPCODE - A four bit field that specifies kind of query in this message (standard, reverse etc)
- AA - Authoritative Answer - this bit is only meaningful in responses, and specifies that the responding name server is an authority for the domain name in question section.
- RD - Recursion Desired - this bit directs the name server to pursue the query recursively.
- RA - Recursion Available - this be is set or cleared in a response, and denotes whether recursive query support is available in the name server.

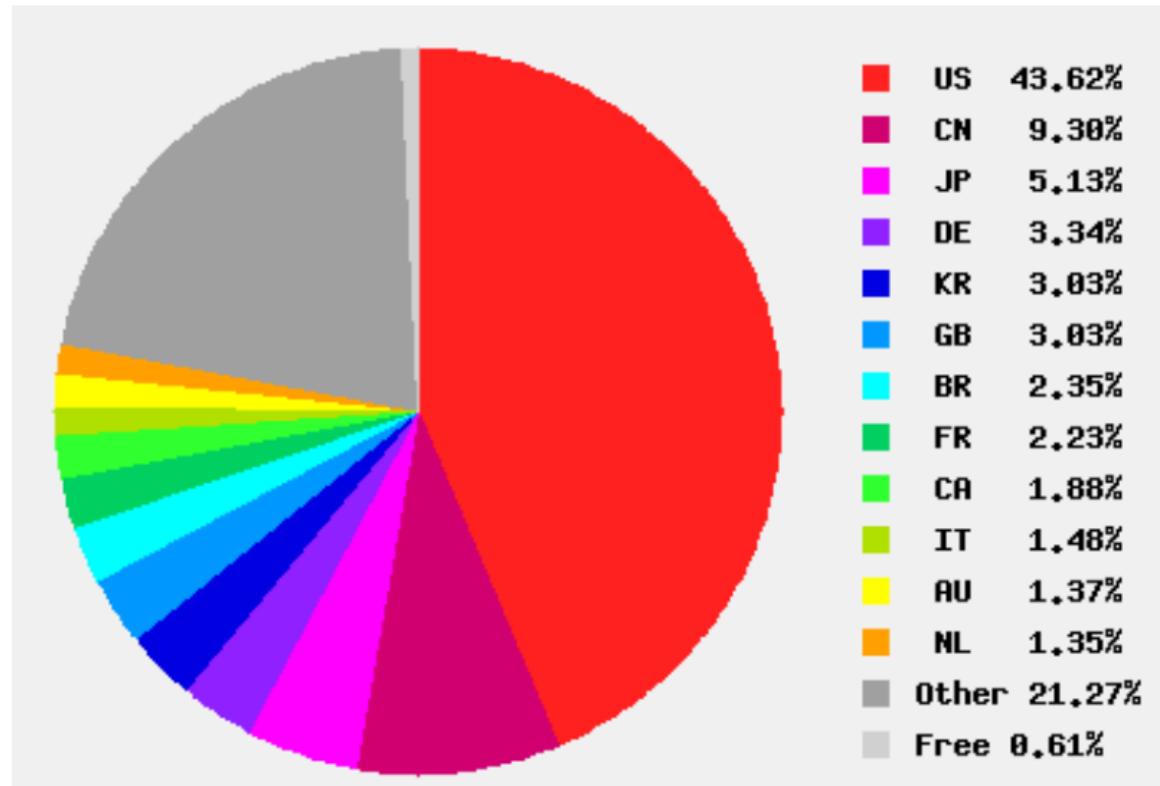


- RCODE - Response code - this 4 bit field is set as part of responses.
- XXCOUNT - the number of records in the corresponding section

NAT

IPv4 address space distribution

Usable: 3706650624 3706.65 million



Ukraine UA 10.71 million 0.21%

<http://www.bgpexpert.com/addressespercountry.php>

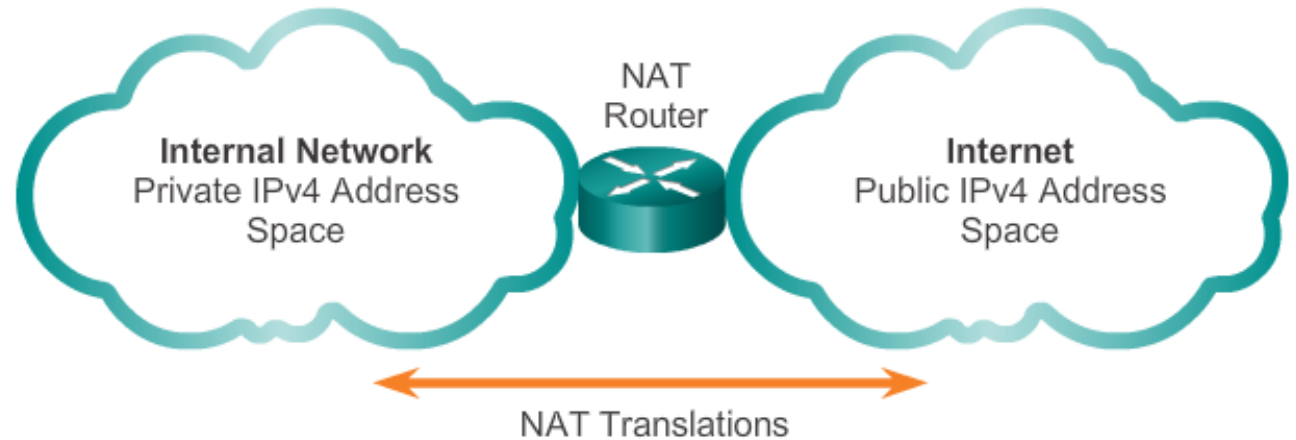
How many hosts are currently connected to computer networks?

According to Cisco IBSG

35-40 billions

IPv4 Private Address Space

- The IPv4 address space is **not big enough** to uniquely address all the devices that need to be connected to the Internet
- Network private addresses are described in RFC 1918 and are designed to be used within an organization or site only
- Private addresses are not routed by Internet routers while public addresses are
- Private addresses can alleviate IPv4 scarcity but since they aren't routed by Internet devices, they need to be translated first.
- NAT is process used to perform such translation



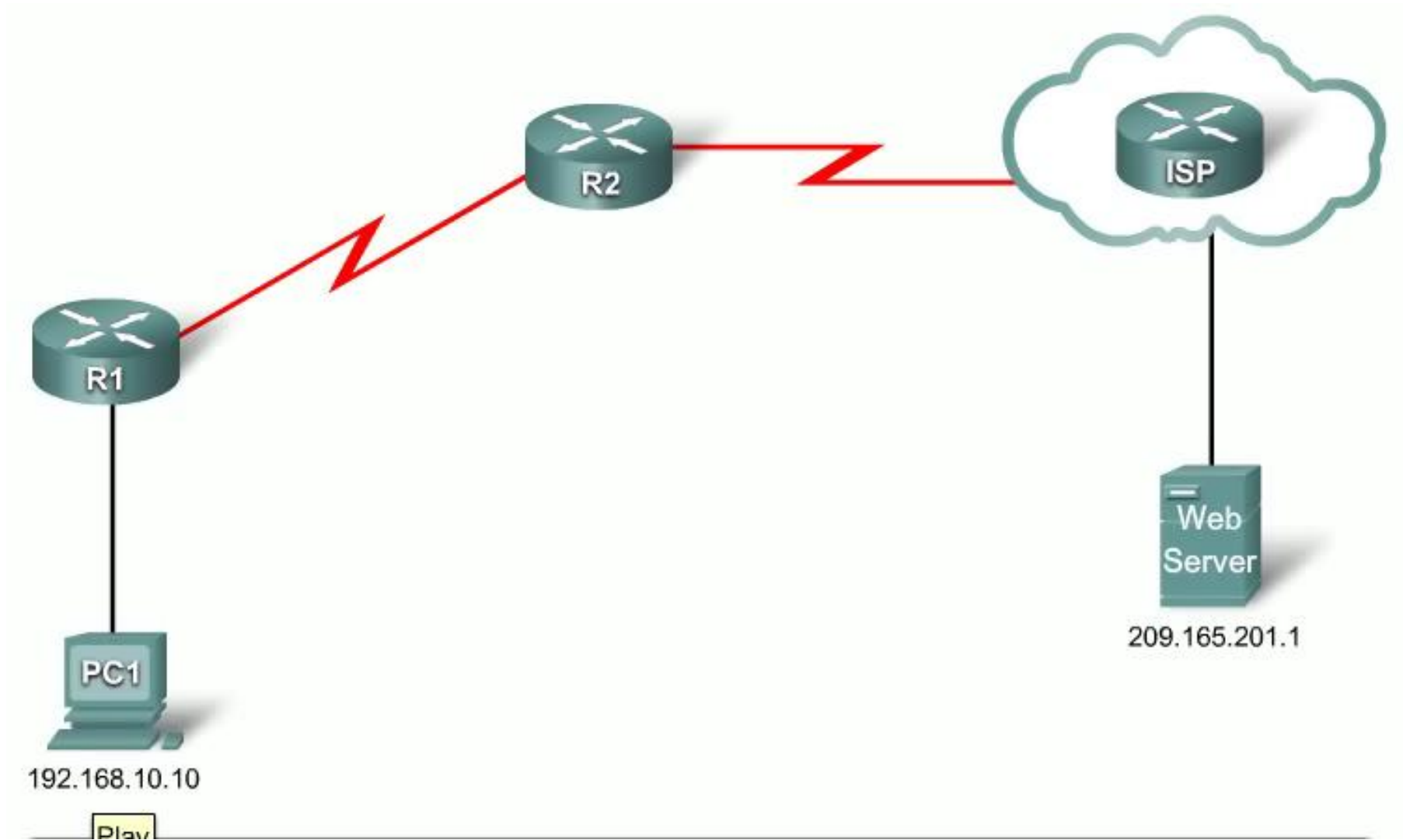
Private Internet addresses are defined in RFC 1918:

Class	RFC 1918 Internal Address Range	CIDR Prefix
A	10.0.0.0 - 10.255.255.255	10.0.0.0/8
B	172.16.0.0 - 172.31.255.255	172.16.0.0/12
C	192.168.0.0 - 192.168.255.255	192.168.0.0/16

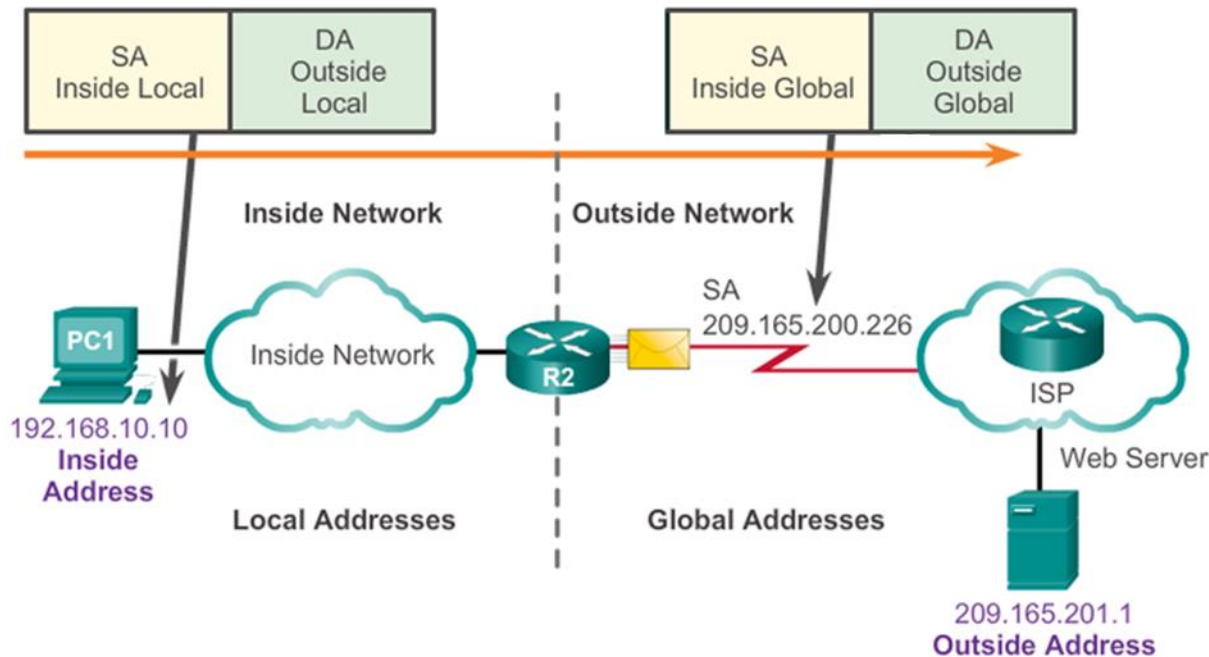
What is NAT?

- NAT is a process used to **translate** network addresses
- NAT's primary use is to **conserve** public IPv4 addresses
- Usually implemented **at border network devices** such as firewalls or routers
- This allows the networks to use private addresses internally, only translating to public addresses when needed
- Devices within the organization can be assigned private addresses and operate with locally unique addresses.
- When traffic must be sent/received to/from other organizations or the Internet, the border router translates the addresses to a public and globally unique address

What is NAT?

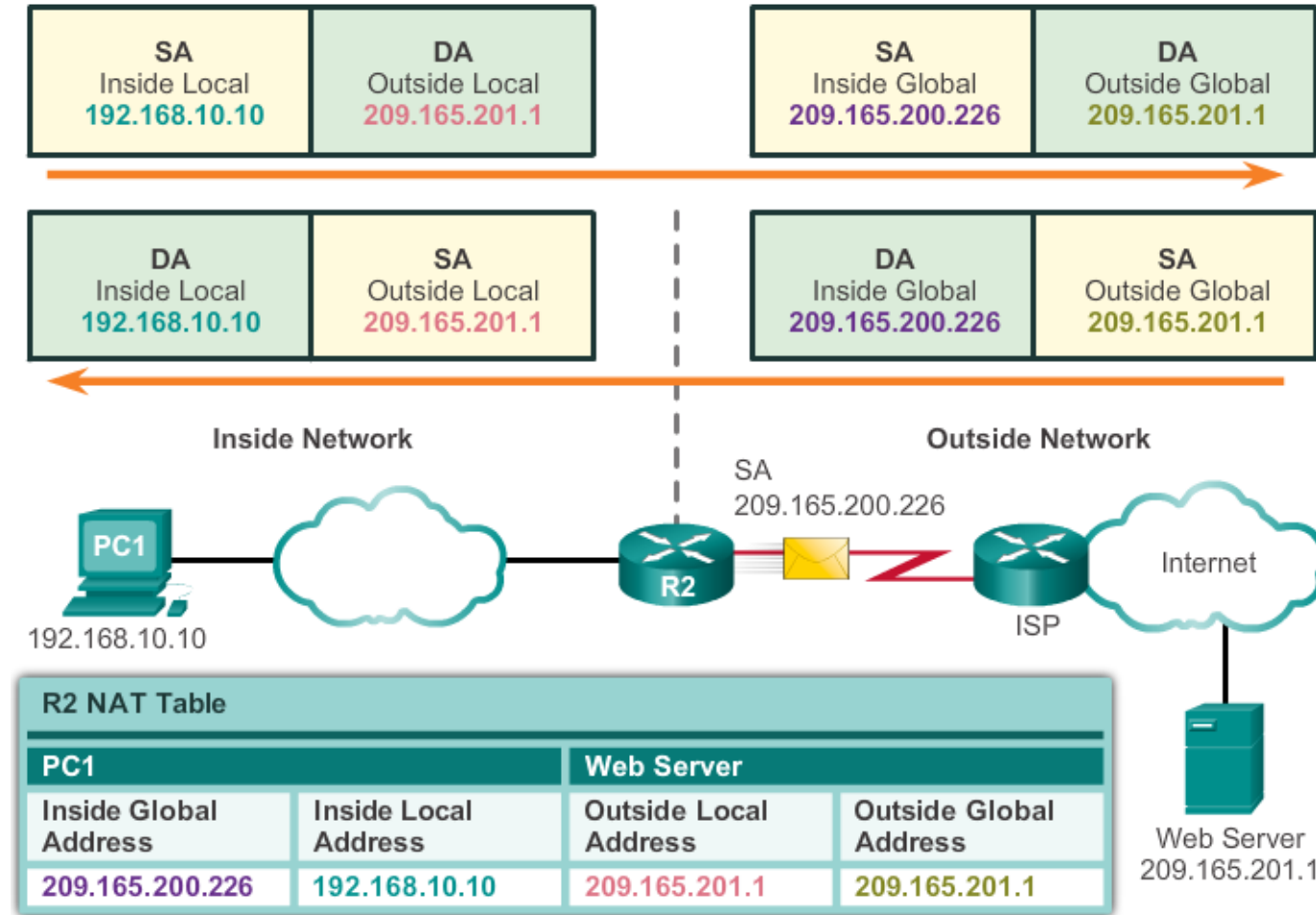


NAT Terminology



- **Inside local address** - Usually not an IP address assigned by a RIR or service provider and is most likely an RFC 1918 private address.
- **Inside global address** - Valid public address that the inside host is given when it exits the NAT router.
- **Outside global address** - Valid public IP address assigned to a host on the Internet.
- **Outside local address** - The local IP address assigned to a host on the outside network. In most situations, this address will be identical to the outside global address of that outside device.

How NAT Works

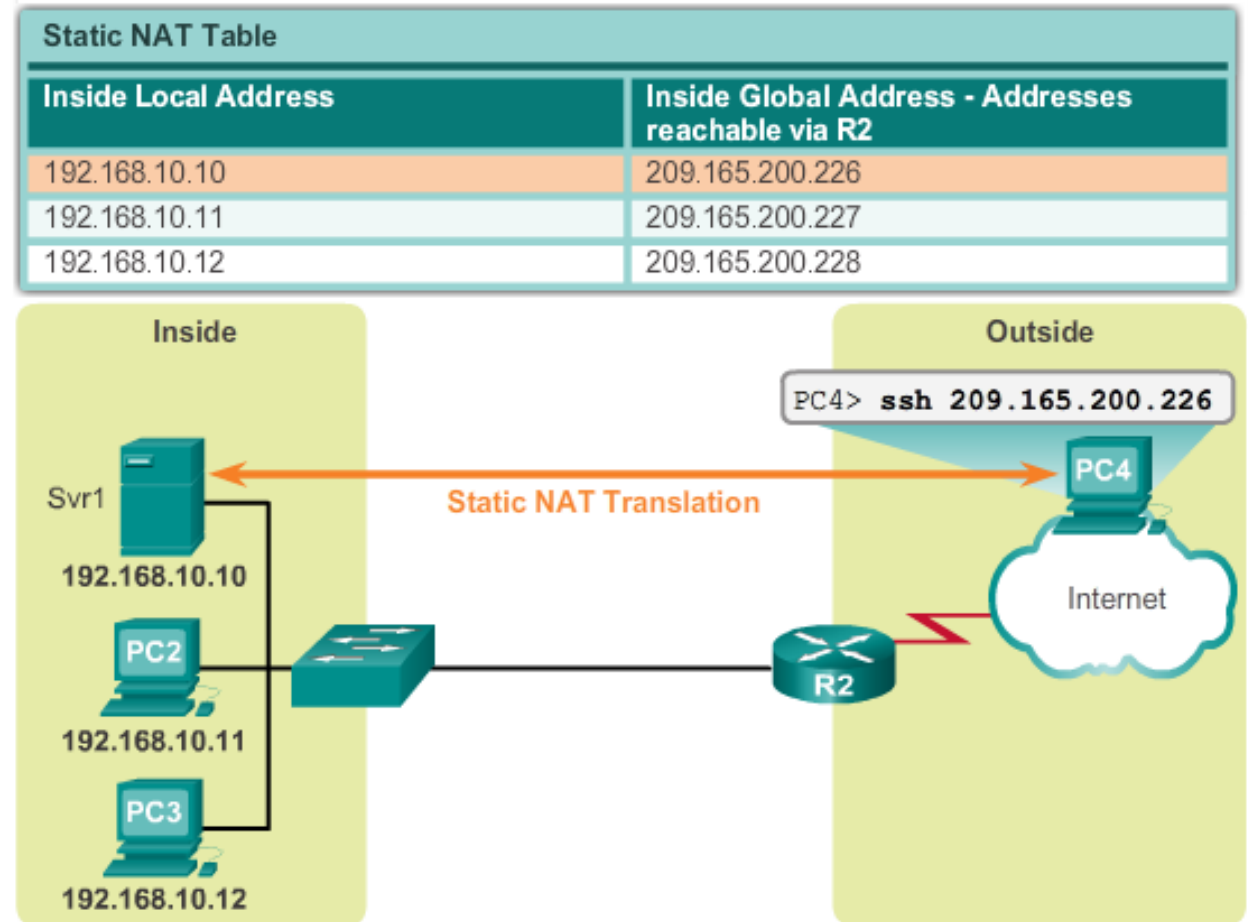


Types of NAT

- **Static address translation (static NAT)** - One-to-one address mapping between local and global addresses.
- **Dynamic address translation (dynamic NAT)** - Many-to-many address mapping between local and global addresses.
- **Port Address Translation (PAT)** - Many-to-one address mapping between local and global addresses. This method is also known as overloading (NAT overloading).
- **Port Forwarding** - Forwarding a network port from one network node to another

Static NAT

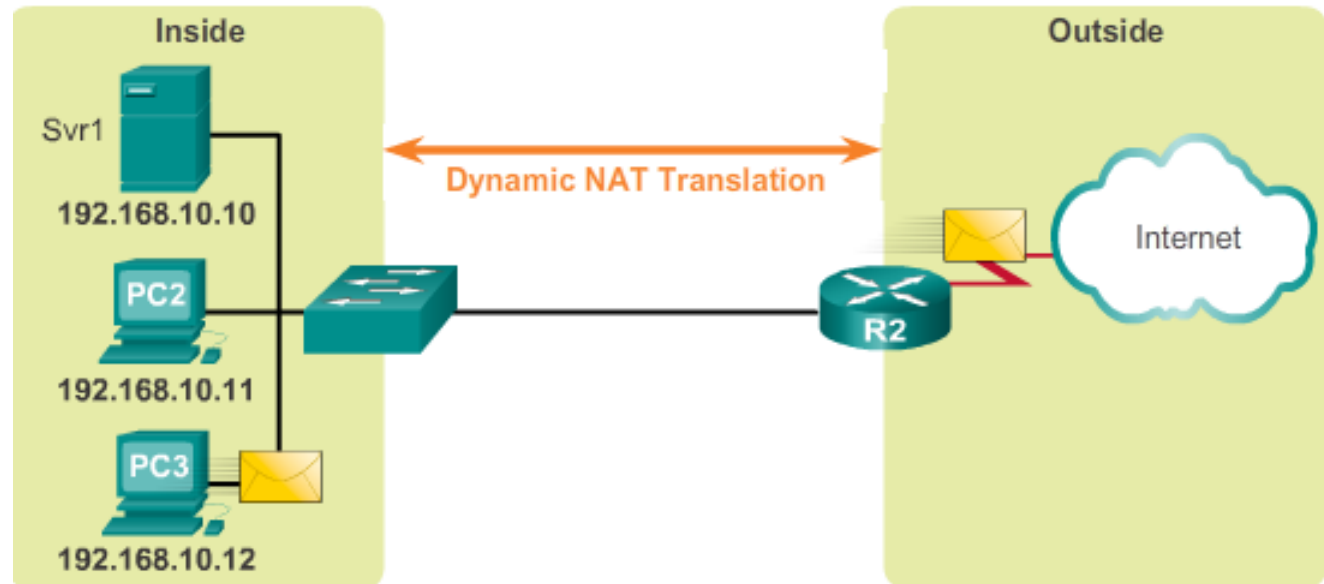
- Static NAT uses a one-to-one mapping of local and global addresses
- These mappings are configured by the network administrator and remain constant
- Static NAT is particularly useful when servers hosted in the inside network must be accessible from the outside network
- A network administrator can SSH to a server in the inside network by pointing his SSH client to the proper inside global address



Dynamic NAT

- Dynamic NAT uses a **pool of public** addresses and assigns them on a first-come, first-served basis
- When an inside device requests access to an outside network, dynamic NAT assigns an available public IPv4 address from the pool
- Dynamic NAT requires that enough public addresses are available to satisfy the total number of simultaneous user sessions

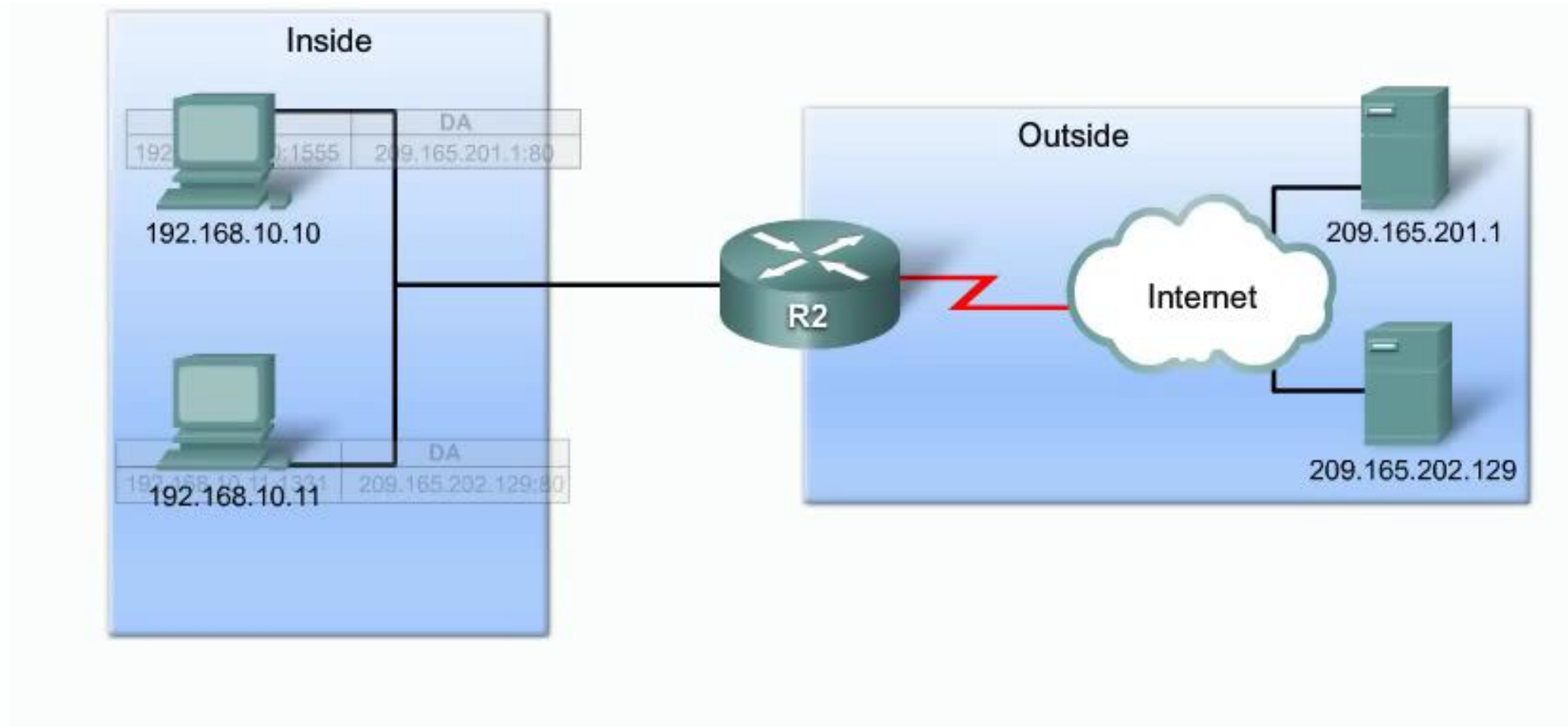
Inside Local Address	Inside Global Address Pool - Addresses reachable via R2
192.168.10.12	209.165.200.226
Available	209.165.200.227
Available	209.165.200.228
Available	209.165.200.229
Available	209.165.200.230



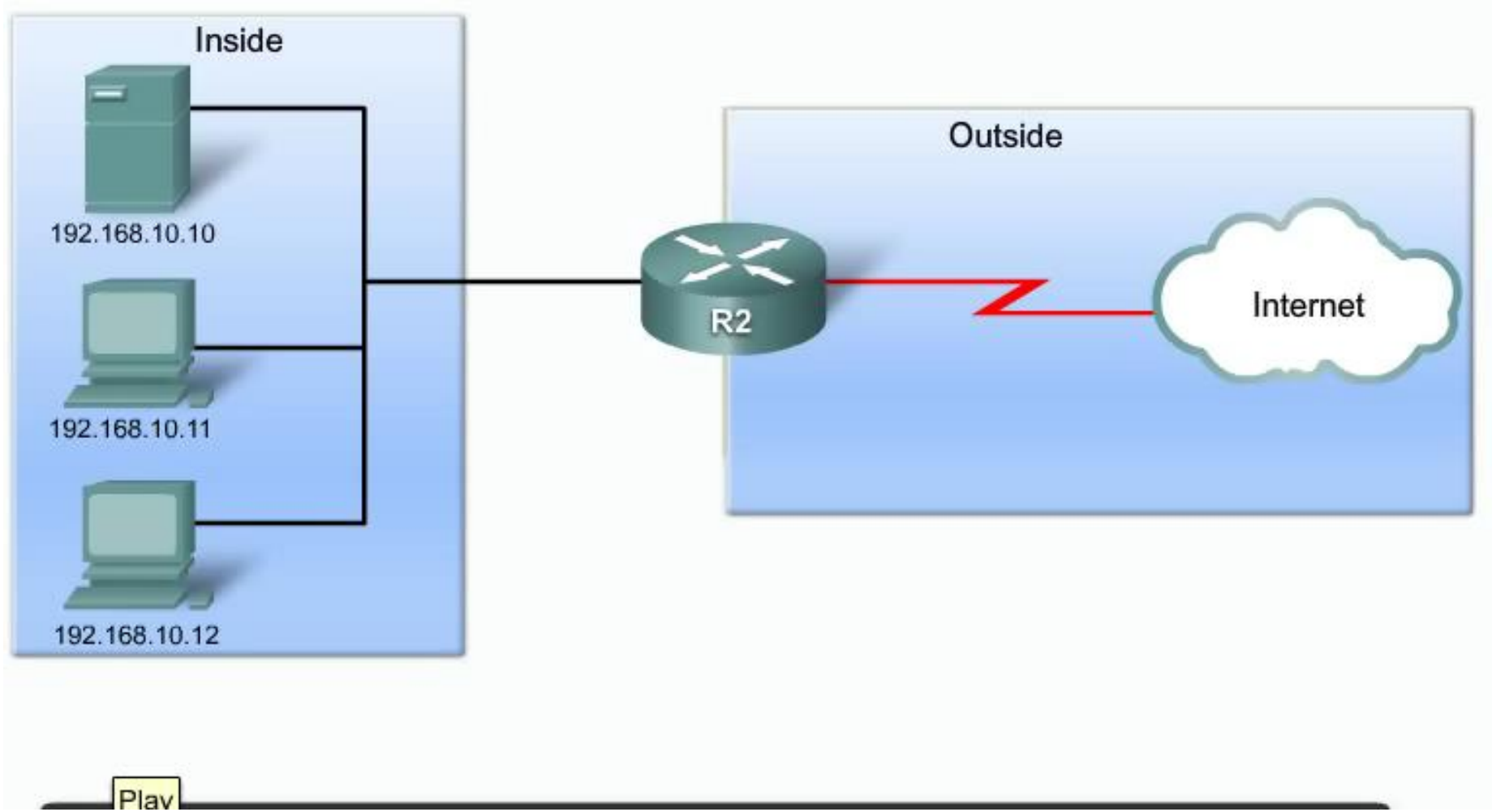
Port Address Translation NAT (PAT)

- PAT maps **multiple private** IPv4 addresses to a **single public** IPv4 address or a few addresses
- PAT uses the pair source port and source IP address to keep track of what traffic belongs to what internal client
- PAT is also known as **NAT overload**
- By also using the port number, PAT is able to forward the response packets to the correct internal device
- The PAT process also validates that the incoming packets were requested, thus adding a degree of security to the session

NAT Overload



NAT Overload (Next Available Port using)



Comparing NAT and PAT

- NAT translates IPv4 addresses on a 1:1 basis between private IPv4 addresses and public IPv4 addresses
- PAT modifies both the address and the port number
- NAT forwards incoming packets to their inside destination by referring to the incoming source IPv4 address given by the host on the public network
- With PAT, there is generally only one or a very few publicly exposed IPv4 addresses
- PAT is also able to translate protocols that don't use port numbers such as ICMP. Each one of these protocols are supported differently by PAT

NAT

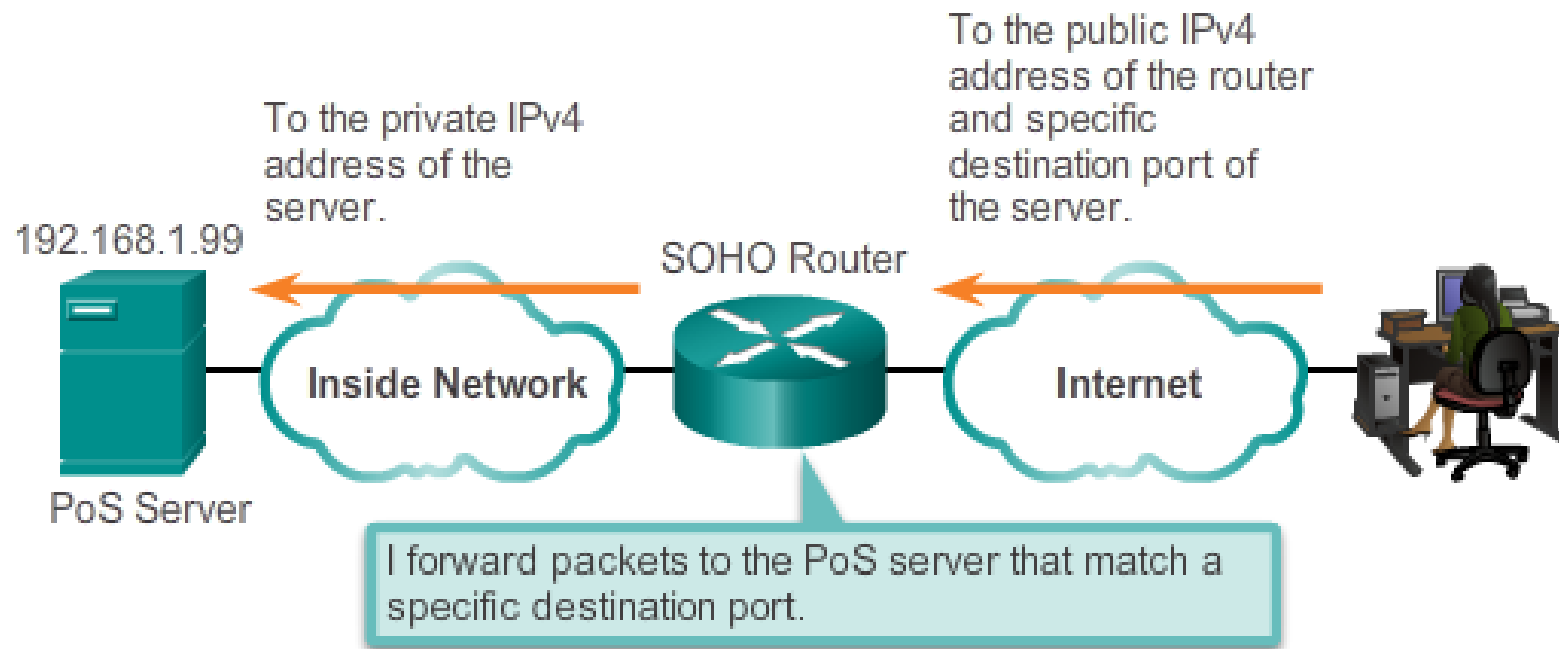
Inside Global Address Pool	Inside Local Address
209.165.200.226	192.168.10.10
209.165.200.227	192.168.10.11
209.165.200.228	192.168.10.12
209.165.200.229	192.168.10.13

PAT

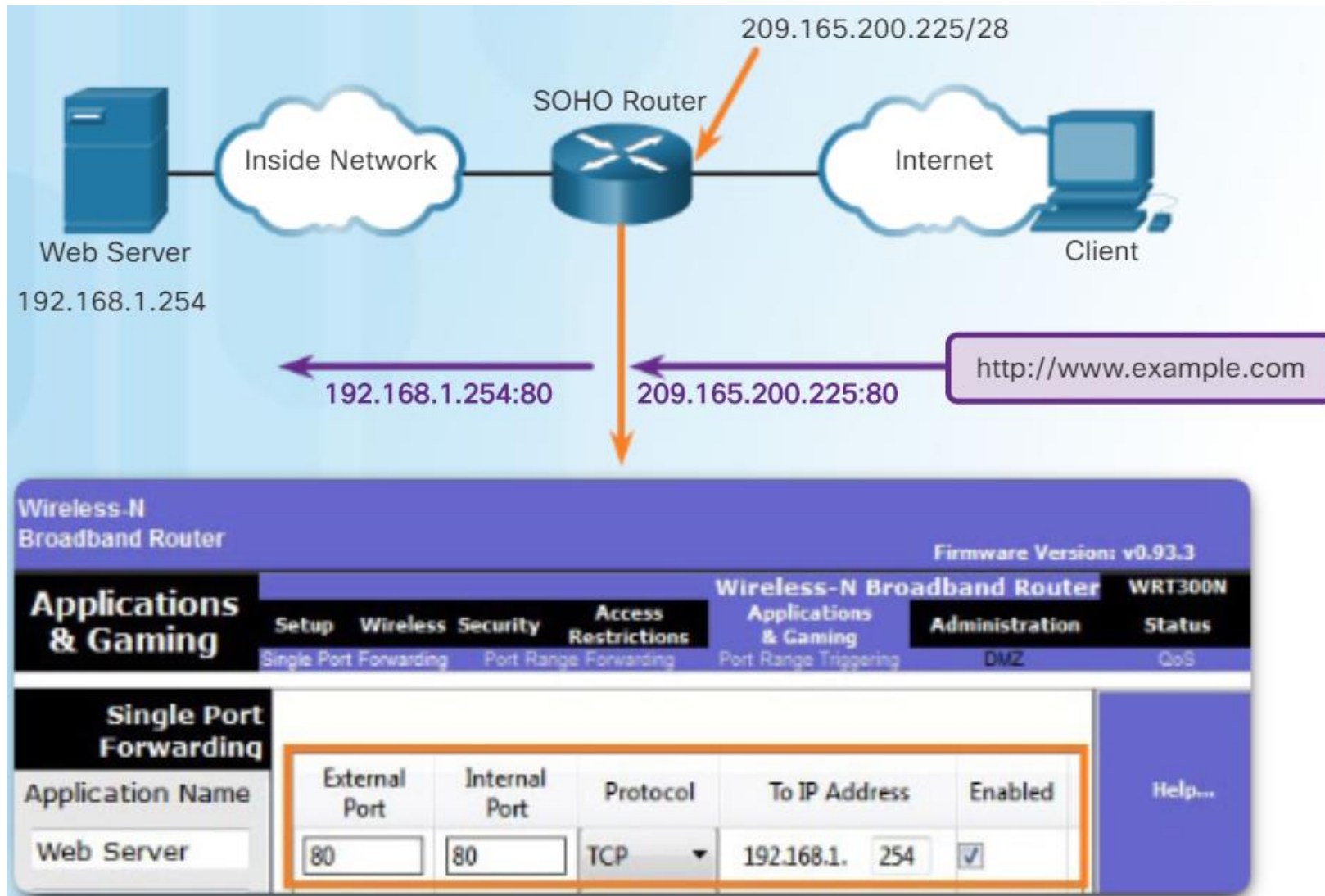
Inside Global Address	Inside Local Address
209.165.200.226:1444	192.168.10.10:1444
209.165.200.226:1445	192.168.10.11:1444
209.165.200.226:1555	192.168.10.12:1555
209.165.200.226:1556	192.168.10.13:1555

Port Forwarding

- Port forwarding is the act of forwarding a network port from one network node to another
- A packet sent to the public IP address and port of a router can be forwarded to a private IP address and port in inside network
- This is helpful in situations where servers have private addresses, not reachable from the outside networks



Port Forwarding Sample



Q&A

A light blue world map is centered in the background of the slide, showing the outlines of continents and countries.

Thank you!