**EO 001.04.01 – Internet Protocol Reading Assignment**

**IPv4**

**What is IPv4? It routes most of today’s internet traffic**

The short answer to the question, “What is IPv4?”, is that it’s the fourth version of the internet protocol. IP, which stands for internet protocol, is the internet’s principal set of rules for communications.

In place for more than 35 years, the U.S. Department of Defense first deployed it on its ARPANET (Advanced Research Projects Agency Network) in 1983

Internet protocol version 4, IPv4, is also at a crossroads: its global IP address supply is exhausted. The internet is undergoing a gradual transition to the next version, IPv6, but not without challenges.

In this glossary entry, we’ll explore the basic components of the internet and how they work together, examine the fourth internet protocol version and its modern-day shortcomings, and touch on its IPv6 successor.

**Before IPv4, a little more on how the internet works**

**More details on IP**

IP is part of an internet protocol suite, which also includes the transmission control protocol. Together, these two are known as TCP/IP. The internet protocol suite governs rules for packetizing, addressing, transmitting, routing, and receiving data over networks.

IP addressing is a logical means of assigning addresses to devices on a network. Each device connected to the internet requires a unique IP address.

Most networks that handle internet traffic are packet-switched. Small units of data, called packets, are routed through a network. A source host, like your computer, delivers these IP packets to a destination host, such as a server, based on IP addresses in packet headers. Packet-switching allows many users on a network to share the same data path.

An IP address has two parts—-one part identifies the host, such as a computer or other device. And the other part identifies the network it belongs to. TCP/IP uses a subnet mask to separate them.

**Now, exactly what is IPv4?**

IP (version 4) addresses are 32-bit integers that can be expressed in hexadecimal notation. The more common format, known as dotted quad or dotted decimal, is x.x.x.x, where each x can be any value between 0 and 255. For example, 192.0.2.146 is a valid IPv4 address.

IPv4 addresses are 32-bit integers expressed in hexadecimal notation.

IPv4 still routes most of today’s internet traffic. A 32-bit address space limits the number of unique hosts to 232, which is nearly 4.3 billion IPv4 addresses for the world to use (4,294,967,296, to be exact).

**Today, we’ve run out**

Think about it: How many connected devices are in your household.

The median American household has five devices, including smartphones, computers and laptops, tablets, and streaming media devices. That doesn’t even include the range of devices that fall under the internet of things (IoT) category, such as connected thermostats, smart speakers, and doorbell cameras.

So, in today’s world of ultra-connected computer networks, where every stationary and mobile device now has an IP address, it turns out that 4.3 billion of them isn’t nearly enough.

In 2011, the Internet Assigned Numbers Authority (IANA), the global coordinator of IP addressing, ran out of IPv4 address space to allocate to regional registries. And regional registries have since depleted those allocations.

In 2015, the American Registry for Internet Numbers (ARIN), the regional registry for North America, began turning down requests for new blocks of numbers on IPv4. ARIN now has a waiting list for IPv4 space.

**Additional limitations**

Besides running out of address space, the IPv4 addressing system has some additional downsides:

About 18 million addresses were set aside for private addressing, drawn from a range known RFC 1918. Most organizations use private addresses on internal networks. However, devices on these local networks have no direct path to the public internet.

To access the public internet, devices with private addresses require a complex and resource-intensive workaround called network address translation (NAT)

**IPv6**

IPv6, the most recent version of the internet protocol, uses 128-bit address space. Unlike IPv4, both letters and numbers are used as identifiers (for example, 2002:db8::8a3f:362:7897). By implementing these changes, IANA created 2128 new IP addresses, which is about 340 undecillion or 340 billion billion billion billion. A whole lot.

With IPv6, a single network can have more addresses than the entire IPv4 address space. IPv6 exhaustion is basically impossible. (There is a hypothetical world IPv6 exhaustion counter out there. Nine million AD, anyone?).

Furthermore, routing tables are simpler. Admins can start from square one and be thoughtful and logical about deploying an addressing scheme. And there’s plenty of room to add more.

Security was also at the forefront when the IPv6 address space was built, while IPv4 has modern-day security measures tacked on after the fact. However, that’s not to say that you get a free pass to omit IPv6 space from your network security model. And the first IPv6 DDoS attack served as an important reminder.

**Eliminating private networks**

About 18 million IPv4 addresses were set aside for private addressing, drawn from a range known as RFC 1918. Most organizations use IPv4 private addresses on internal networks. However, devices with private addresses have no direct path to the public internet.

To access the public internet, these devices require a complex and resource-intensive workaround called network address translation (NAT).

IPv6 is NAT-free, enabling every device to communicate directly without intermediary steps.

**The challenges of implementing IPv6 DNS**

All of this change was born of necessity, but not everyone is on board. This is not just a configuration change. Think of it more like a challenging system migration.

**Examples of IPv6 challenges**

* Deployment of IPv6 is completely different from IPv4, requiring a steep learning curve to master.
* IPv4 and IPv6 are not directly interoperable. IPv6 is managed differently than IPv4, requiring a steep learning curve to master. IPv6 address formats are also longer, so they can’t be easily memorized or transcribed.
* It’s a lot of work to test all of your applications end-to-end in an IPv6 environment. And what may work well in a small test lab may fall apart when implemented at scale.
* Every part of your network chain (including every IPv6 DNS server) has to be compliant. Legacy network applications or devices hard-coded for IPv4 may lack IPv6 support.
* Specifically, most IoT devices do not support IPv6. If critical IoT devices on your network aren’t IPv6-ready, then you can’t transition your network at all. This a particularly tough conundrum for the healthcare industry.
* Tertiary content addressable memory (TCAM) quickly gets depleted when adding IPv6 addresses. TCAM stores access control lists on network routers. Routing vendors have allowed admins to tune how much TCAM to allocate to IPv4 and IPv6, with mixed results. Ultimately, enterprises end up having to buy more pricey TCAM.

**Complex enterprise implementation**

Enterprise implementation itself can be complex, with segmented steps and testing required at each point.

You might first start with your external-facing networks and services like web servers. Then go to your perimeter (or DMZ) networks and your data centers. And finally, your internal networks and devices. Just like your current network, you’ll need an IPv6 nameserver, DNS server, and all the rest.

It’s enough to say “thanks but no thanks” and stick with IPv4. Sure, the more workarounds that you add to your IPv4 network, the more you have to manage. But it works, you understand it, and you know the network won’t break.

There is no driving event such as a government mandate forcing the transition en masse. As a result, institutional inertia will be strong enough in most organizations to keep the status quo in place. The work involved in a transition simply isn’t worth it… yet.

**Subnet Masks**

**Subnet Mask Definition**

Every device has an IP address with two pieces: the client or host address and the server or network address. IP addresses are either configured by a DHCP server or manually configured (static IP addresses). The subnet mask splits the IP address into the host and network addresses, thereby defining which part of the IP address belongs to the device and which part belongs to the network.

The device called a gateway or default gateway connects local devices to other networks. This means that when a local device wants to send information to a device at an IP address on another network, it first sends its packets to the gateway, which then forwards the data on to its destination outside of the local network.

**What is Subnet Mask?**

A subnet mask is a 32-bit number created by setting host bits to all 0s and setting network bits to all 1s. In this way, the subnet mask separates the IP address into the network and host addresses.

The “255” address is always assigned to a broadcast address, and the “0” address is always assigned to a network address. Neither can be assigned to hosts, as they are reserved for these special purposes.

The IP address, subnet mask and gateway or router comprise an underlying structure—the Internet Protocol—that most networks use to facilitate inter-device communication.

When organizations need additional subnetworking, subnetting divides the host element of the IP address further into a subnet. The goal of subnet masks are simply to enable the subnetting process. The phrase “mask” is applied because the subnet mask essentially uses its own 32-bit number to mask the IP address.

**IP Address and Subnet Mask**

A 32-bit IP address uniquely identifies a single device on an IP network. The 32 binary bits are divided into the host and network sections by the subnet mask but they are also broken into four 8-bit octets.

Because binary is challenging, we convert each octet so they are expressed in dot decimal. This results in the characteristic dotted decimal format for IP addresses—for example, 172.16.254.1. The range of values in decimal is 0 to 255 because that represents 00000000 to 11111111 in binary.

**IP Address Classes and Subnet Masks**

Since the internet must accommodate networks of all sizes, an addressing scheme for a range of networks exists based on how the octets in an IP address are broken down. You can determine based on the three high-order or left-most bits in any given IP address which of the five different classes of networks, A to E, the address falls within.

(Class D networks are reserved for multicasting, and Class E networks not used on the internet because they are reserved for research by the Internet Engineering Task Force IETF.)

A Class A subnet mask reflects the network portion in the first octet and leaves octets 2, 3, and 4 for the network manager to divide into hosts and subnets as needed. Class A is for networks with more than 65,536 hosts.

A Class B subnet mask claims the first two octets for the network, leaving the remaining part of the address, the 16 bits of octets 3 and 4, for the subnet and host part. Class B is for networks with 256 to 65,534 hosts.

In a Class C subnet mask, the network portion is the first three octets with the hosts and subnets in just the remaining 8 bits of octet 4. Class C is for smaller networks with fewer than 254 hosts.

Class A, B, and C networks have natural masks, or default subnet masks:

* Class A: 255.0.0.0
* Class B: 255.255.0.0
* Class C: 255.255.255.0

You can determine the number and type of IP addresses any given local network requires based on its default subnet mask.

An example of Class A IP address and subnet mask would be the Class A default submask of 255.0.0.0 and an IP address of 10.20.12.2.

**How Does Subnetting Work?**

Subnetting is the technique for logically partitioning a single physical network into multiple smaller sub-networks or subnets.

Subnetting enables an organization to conceal network complexity and reduce network traffic by adding subnets without a new network number. When a single network number must be used across many segments of a local area network (LAN), subnetting is essential.

The benefits of subnetting include:

* Reducing broadcast volume and thus network traffic
* Enabling work from home
* Allowing organizations to surpass LAN constraints such as maximum number of hosts

**Network Addressing**

The standard modern network prefix, used for both IPv6 and IPv4, is Classless Inter-Domain Routing (CIDR) notation. IPv4 addresses represented in CIDR notation are called network masks, and they specify the number of bits in the prefix to the address after a forward slash (/) separator. This is the sole standards-based format in IPv6 to denote routing or network prefixes.

To assign an IP address to a network interface since the advent of CIDR, there are two parameters: a subnet mask and the address. Subnetting increases routing complexity, because there must be a separate entry in each connected router’s tables to represent each locally connected subnet.

**Port Numbers**

**What is a port?**

A port is a virtual point where network connections start and end. Ports are software-based and managed by a computer's operating system. Each port is associated with a specific process or service. Ports allow computers to easily differentiate between different kinds of traffic: emails go to a different port than webpages, for instance, even though both reach a computer over the same Internet connection.

**What is a port number?**

Ports are standardized across all network-connected devices, with each port assigned a number. Most ports are reserved for certain protocols — for example, all Hypertext Transfer Protocol (HTTP) messages go to port 80. While IP addresses enable messages to go to and from specific devices, port numbers allow targeting of specific services or applications within those devices.

**How do ports make network connections more efficient?**

Vastly different types of data flow to and from a computer over the same network connection. The use of ports helps computers understand what to do with the data they receive.

Suppose Bob transfers an MP3 audio recording to Alice using the File Transfer Protocol (FTP). If Alice's computer passed the MP3 file data to Alice's email application, the email application would not know how to interpret it. But because Bob's file transfer uses the port designated for FTP (port 21), Alice's computer is able to receive and store the file.

Meanwhile, Alice's computer can simultaneously load HTTP webpages using port 80, even though both the webpage files and the MP3 sound file flow to Alice's computer over the same WiFi connection.

**What are the different port numbers?**

There are 65,535 possible port numbers, although not all are in common use. Some of the most commonly used ports, along with their associated networking protocol, are:

* Ports 20 and 21: File Transfer Protocol (FTP). FTP is for transferring files between a client and a server.
* Port 22: Secure Shell (SSH). SSH is one of many tunneling protocols that create secure network connections.
* Port 25: Simple Mail Transfer Protocol (SMTP). SMTP is used for email.
* Port 53: Domain Name System (DNS). DNS is an essential process for the modern Internet; it matches human-readable domain names to machine-readable IP addresses, enabling users to load websites and applications without memorizing a long list of IP addresses.
* Port 80: Hypertext Transfer Protocol (HTTP). HTTP is the protocol that makes the World Wide Web possible.
* Port 123: Network Time Protocol (NTP). NTP allows computer clocks to sync with each other, a process that is essential for encryption.
* Port 179: Border Gateway Protocol (BGP). BGP is essential for establishing efficient routes between the large networks that make up the Internet (these large networks are called autonomous systems). Autonomous systems use BGP to broadcast which IP addresses they control.
* Port 443: HTTP Secure (HTTPS). HTTPS is the secure and encrypted version of HTTP. All HTTPS web traffic goes to port 443. Network services that use HTTPS for encryption, such as DNS over HTTPS, also connect at this port.
* Port 500: Internet Security Association and Key Management Protocol (ISAKMP), which is part of the process of setting up secure IPsec connections.
* Port 3389: Remote Desktop Protocol (RDP). RDP enables users to remotely connect to their desktop computers from another device.