Internet Protocol (IP)

2.2 IP

+ **How does this support my pentesting career?**

  - Understanding network attacks
  - Using network attack tools at their maximum
  - Studying other networking protocols

2.2 IP

- The **Internet Protocol** (IP) is the protocol that runs on the **Internet** layer of the Internet Protocol suite, also known as TCP/IP.

- IP is in charge of delivering the **datagrams** (IP packets are called datagrams) to the hosts involved in a communication, and it uses **IP addresses** to identify a host.

2.2.1 IPv4 Addresses

+ When you write a letter, you have to specify the recipient's **address** on the envelope before sending it. Similarly, the Internet uses its addressing scheme to deliver packets to the right destination.

+ Any host on a computer network, be it a private network or the Internet, is identified by a **unique IP address**.

2.2.1 IPv4 Addresses

EXAMPLE

+ The vast majority of networks run IP **version 4** (IPv4).
+ An IPv4 address consists of four bytes, or octets; a byte consists of 8 bits.

```
73.5.12.132
```

2.2.1 IPv4 Addresses

+ A dot delimits every octet in the address.

$$\underbrace{73}_{\text{First}}.\underbrace{5}_{\text{Second}}.\underbrace{12}_{\text{Third}}.\underbrace{132}_{\text{Fourth}}$$

2.2.1 IPv4 Addresses

+ As you may recall from the introduction module, with 8 bits, you can represent **up to 2^8 different values** from 0 to 255.

+ This does not mean that you can **assign** any address starting from 0.0.0.0 to 255.255.255.255 to a host. Some addresses are **reserved** for special purposes.

2.2.2 Reserved IPv4 Addresses

- For example, some reserved intervals are:
  - **0.0.0.0 – 0.255.255.255** representing "this" network.
  - **127.0.0.0 – 127.255.255.255** representing the local host (e.g., your computer).
  - **192.168.0.0 – 192.168.255.255** is reserved for private networks.

- You can find the details about the special use of IPv4 addresses in RFC5735.

http://tools.ietf.org/html/rfc5735

2.2.3 IP/Mask

EXAMPLE

+ To fully identify a host, you also need to know its **network**. To do that, you will need an IP address and a **netmask**, or subnet mask.

+ With an IP/netmask pair, you can identify the network part and the host part of an IP address.

IP address:	192.168.5.100
Subnet mask:	255.255.255.0

2.2.3 IP/Mask

+ To find the network part you have to perform a **bitwise *AND* operation** between the netmask and the IP address.

+ In the following example, we are going to see how to find the network part of this IP address/mask pair:

```
192.168.33.12/255.255.224.0
```

2.2.3.1 IP/Mask CIDR Example

1

Convert the octets in binary form:

192.168.33.12

11000000.10101000.00100001.00001100

255.255.224.0

11111111.11111111.11100000.00000000

2.2.3.1 IP/Mask CIDR Example

2

Perform the *bitwise AND*:

IP: `11000000.10101000.00100001.00001100`

&

Mask: **`11111111.11111111.111`**`00000.00000000`

=

Network: `11000000.10101000.00100000.00000000`

Network prefix in decimal notation: `192.168.32.0`

2.2.3.1 IP/Mask CIDR Example

- 192.168.32.0 is the **network prefix**. You can identify the network by using the following notation:

  `192.168.32.0/255.255.224.0`

- Or, as the netmask is made by 19 consecutive "1" bits:

  `192.168.32.0/19`

- The latter is the **Classless Inter-Domain Routing (CIDR)** notation.

2.2.3.2 IP/Mask Host Example

- The address part not covered by the netmask is the **host part** of the IP address. You can find it by performing a bitwise *AND* with the **inverse of the netmask**.

- Let's look at an example with the same IP/mask.

2.2.3.2 IP/Mask Host Example

1

Convert the octets in binary form:

192.168.33.12

11000000.10101000.00100001.00001100

255.255.224.0

11111111.11111111.11100000.00000000

2.2.3.2 IP/Mask Host Example

2

Invert the netmask by performing a *bitwise NOT*:

$$\neg(11111111.11111111.11100000.00000000)$$

$$=$$

$$00000000.00000000.00011111.11111111$$

2.2.3.2 IP/Mask Host Example

3

Perform the final *bitwise AND*:

IP:	11000000.10101000.00100001.00001100
	&
¬Mask:	00000000.00000000.000**11111**.**11111111**
	=
Host:	00000000.00000000.00000001.00001100
Host part in decimal notation:	0.0.1.12

2.2.3.2 IP/Mask Host Example

- Moreover, the inverse of the netmask lets you know how many hosts a network can contain.
- In our example, we have 13 bits to represent the hosts; this means that the network can contain 2^{13} = **8192 different addresses**.

2.2.4 Network and Broadcast Addresses

- There are two special addresses:
  - One with the host part made by all zeros.
  - Another with the host part made by all ones.

- These special addresses **were** used as the **network** and **broadcast** addresses, thus reducing by 2 the number of hosts on a given network. This technical limitation should be extinct (RFC1878) but is still used to keep compatibility with old equipment.

http://tools.ietf.org/html/rfc1878

2.2.5 IP Examples

+ Let’s recap by going over some IP examples.

2.2.5 IP Examples

10.54.12.0/24 (10.54.12.0/255.255.255.0)

- Contains 2^8 = 256 addresses
- 10.54.12.0 is the network address according to the pre-CIDR standard
- 10.54.12.255 is the broadcast address according to the pre-CIDR standard

2.2.5 IP Examples

192.168.114.32/27 (192.168.114.32/255.255.255.224)

+ Contains $2^5 = 32$ addresses
+ 192.168.114.32 is the pre-CIDR network address
+ 192.168.114.63 is the pre-CIDR broadcast address

2.2.5 IP Examples

- Given the network 172.16.2.0/23
  - 172.16.3.12 and 172.16.2.66 **are** in the same network
  - 172.16.3.240 and 172.16.4.2 **are not** in the same network

- The network 192.168.1.0/16
  - Does not make sense; a bitwise *AND* between 192.168.1.0 and 255.255.0.0 leads to 192.168.0.0 as network address
  - Could be a valid IP address in the 192.168.1.0/16 network

2.2.6 Subnet Calculators

- You can practice more on this topic by using a subnet calculator.

- Here are two subnet calculators you can check out:
  - A classful calculator
  - A CIDR calculator

http://www.subnet-calculator.com/
http://www.subnet-calculator.com/cidr.php

2.2.7 IPv6

- **IPv4** addresses are being consumed rapidly due to a large number of new devices connecting to the internet every day.

- One day IPv4 addresses might be exhausted.

2.2.7 IPv6

- As a 32-bit address, **IPv4** has 2^32 = **4.294.967.296** possible addresses.
- While a 128-bit **IPv6** address has **2^128 = 2^32 * 2^96** possible addresses.
- 2^96 is equal to **79 octillion addresses**

2.2.7 IPv6

- An **IPv6** address consists of **16-bit hexadecimal numbers** separated by a **colon (:)**. Hexadecimal numbers are case insensitive. In case zeros occur, they can be skipped.

- Let’s check out some IPv6 examples on the next slide.

2.2.7 IPv6

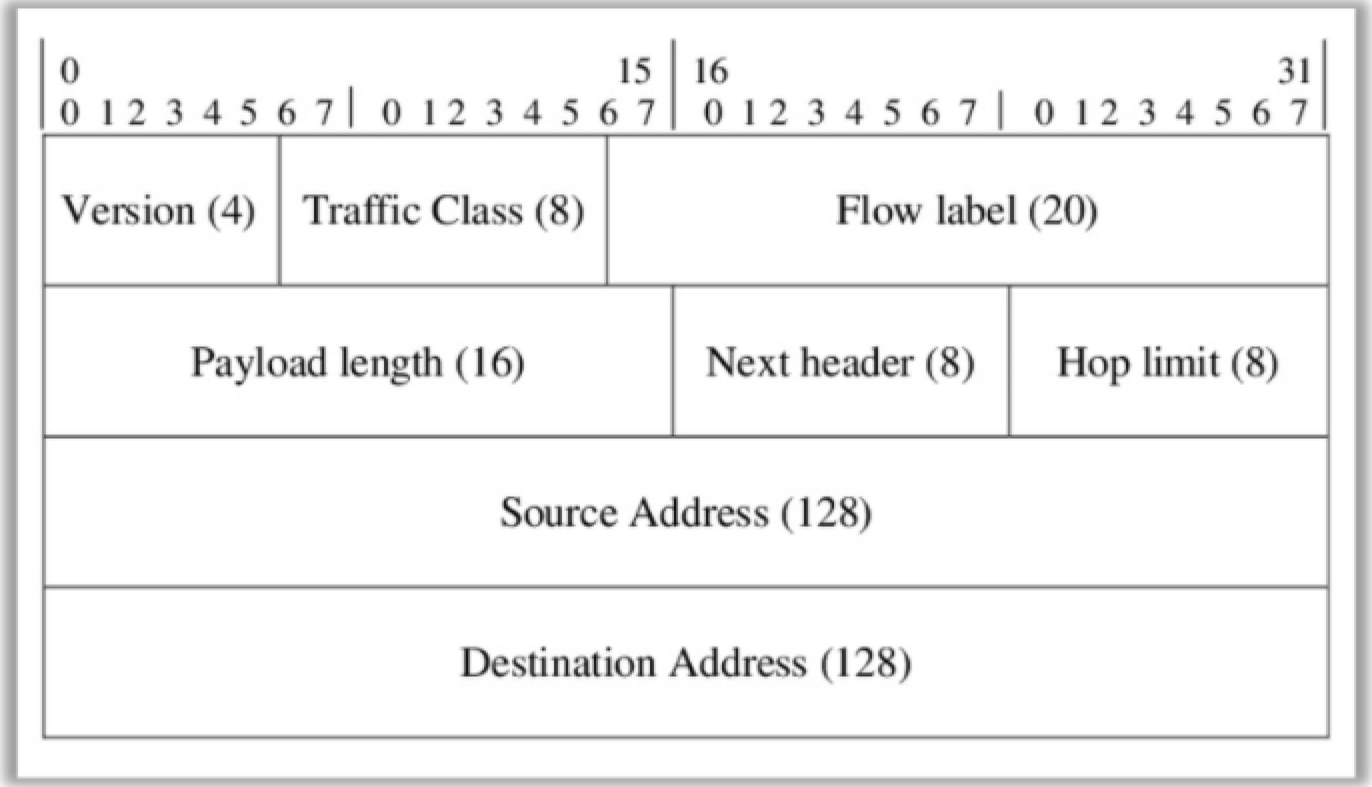
EXAMPLE

+ IPv6 addresses examples:

2001:0db8:0020:130F:0000:0000:087C:140B

2001:0db8:0:160F::850C:140B

2.2.7.1 IPv6 header



https://www.researchgate.net/profile/Dragos_Truscan/publication/31596630/figure/download/fig5/AS:340685715722244@1458237212506/IPv6-Header-Format.png

2.2.7.2 IPv6 forms

- IPv6 can be presented in following text representations:
  - Regular form: **1080:0:FF:0:8:800:200C:417A**
  - Compressed form: **FF01:0:0:0:0:0:0:43** becomes **FF01::43** as a result of skipping zeros
  - IPv4-compatible: **0:0:0:0:0:0:13.1.68.3** or **::13.1.68.3** after skipping zeros

2.2.7.3 IPv6 Reserved Addresses

+ IPv6 also has reserved addresses, which cannot be used like the reserved IPv4 ones.

+ For example:
  - ::1/128 is a loopback address
  - ::FFFF:0:0/96 are IPv4 mapped addresses

+ For more information, you can check RFC3513.

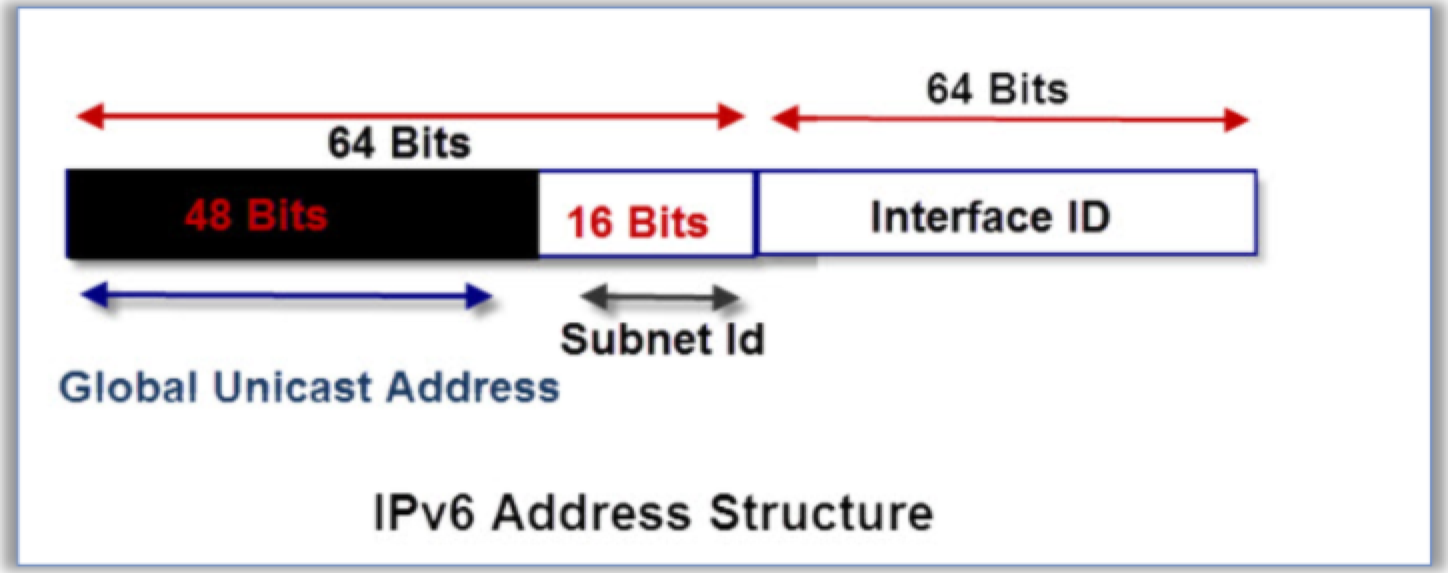
https://tools.ietf.org/html/rfc3513

2.2.7.4 IPv6 Structure

- An **IPv6 address** can be split in half (64 bits each) into a **network part** and a **device part.**

- Furthermore, the **first 64 bits** ends with a **dedicated 16-bits space** (one hex word) that can be used only for **specifying a subnet.**

2.2.7.4 IPv6 Structure



http://www.steves-internet-guide.com/ipv6-guide/

2.2.7.5 IPv6 Scope

Address Types and Scope

- IPv6 addresses have three types:
  - **Global Unicast Address** – These addresses are global ones and reside in global internet.
  - **Unique Local and Link Local** — reside only in Internal Networks.

2.2.7.5 IPv6 Scope

Address Types and Scope

Global Unicast Address --Scope Internet- Routed on Internet

Unique Local -- Scope Internal Network or VPN -Internally routable but Not routed on Internet

Link Local - Scope network link- Not Routed internally or externally.

http://www.steves-internet-guide.com/ipv6-guide/

2.2.7.6 IPv6 Translation

- **IPv6** addresses can also be translated to **binary**.
- One 4-digit hex word represents **16 binary digits**; we can see this demonstrated in the following way:
  - Bin **0000000000000000** = Hex 0000 (or just 0)
  - Bin **1111111111111111** = Hex FFFF
  - Bin **1101010011011011** = Hex D4DB

2.2.7.6 IPv6 Translation

Thus, 128-bit binary address looks like:

- 1111111111111111.1111111111111111.1111111111111111.11111111111111111.1111111111111111.1111111111111111.1111111111111111.1111111111111111

- And, the above can be represented by 8 hex words, separated by colons:

FFFF:FFFF:FFFF:FFFF:FFFF:FFFF:FFFF:FFFF

2.2.7.7 IPv6 Subnets

+ Like IPv4, an IPv6 address has a network portion and a device portion.

+ Unlike IPv4, an IPv6 address has a dedicated subnetting portion. On the next few slides, we'll show how the ranges are divided in IPv6.

2.2.7.7 IPv6 Subnets

- **Network Address Range**

  In IPv6, the first 48 bits are for Internet global addressing.

- 1111111111111111.1111111111111111.1111111111111111.00
  00000000000000.0000000000000000.0000000000000000.0000
  000000000000.0000000000000000

2.2.7.7 IPv6 Subnets

- **Subnetting Range**

  The 16 bits from the 49th to the 64th are for defining subnets.

- 0000000000000000.0000000000000000.00000000000000
  00.1111111111111111.0000000000000000.000000000000
  0000.0000000000000000.0000000000000000

2.2.7.7 IPv6 Subnets

Device (Interface) Range

The last 64 bits are for device (interface) ID's:

0000000000000000.0000000000000000.0000000000000000.0000
000000000000.1111111111111111.1111111111111111.11111111
11111111.1111111111111111

2.2.7.8 IPv6 Subnetting

- In **IPv6**, there are **prefixes** instead of subnets blocks. For example:

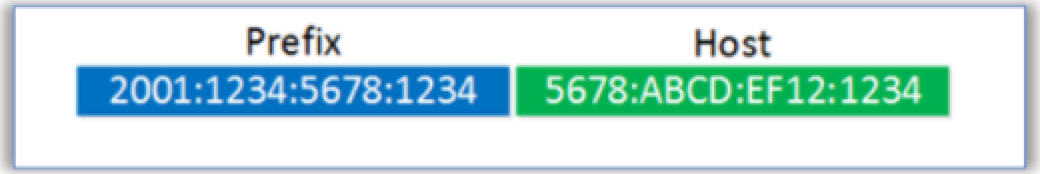
  2001:1111:1234:1234::/64

- In the above IPv6 address, the number after the slash (64) is the **number of bits that is used for a prefix**. Everything behind it can be used for **hosts** of the **subnet**.

https://networklessons.com/ipv6/how-to-find-ipv6-prefix/

2.2.7.8 IPv6 Subnetting

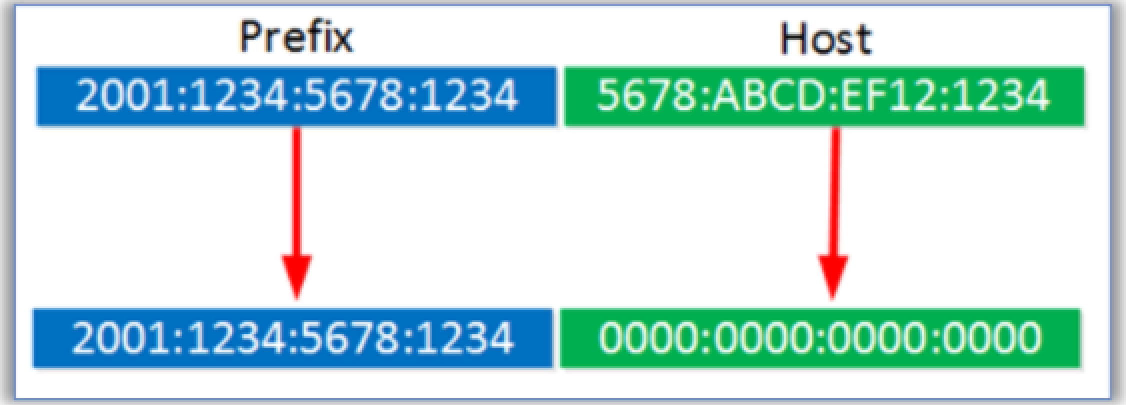
- As you may have noticed, /64 means that the first 64 bits are a prefix. And, as previously mentioned earlier, each 4-digit hex word is 16 bits, thus in following IPv6 address we can divide it as such:



https://networklessons.com/ipv6/how-to-find-ipv6-prefix/

2.2.7.8 IPv6 Subnetting

+ We confirmed that **2001:1234:5678:1234** is the prefix, but let's now focus on writing down a correctly formatted IPv6 address.



https://networklessons.com/ipv6/how-to-find-ipv6-prefix/

2.2.7.8 IPv6 Subnetting

- **2001:1234:5678:1234:0000:0000:0000:0000** is a valid prefix, but it can be shortened by omitting zeros, into following form:

2001:1234:5678:1234::/64

https://networklessons.com/ipv6/how-to-find-ipv6-prefix/

2.2.7.8 IPv6 Subnetting

+ You can practice more on this topic by using a subnet calculator.

+ Here is a calculator you can check out:
  - IPv6 Calculator

https://www.ultratools.com/tools/ipv6CIDRToRange

References

- RFC5735: http://tools.ietf.org/html/rfc5735
- RFC1878: http://tools.ietf.org/html/rfc1878
- A classful calculator: http://www.subnet-calculator.com/
- A CIDR calculator: http://www.subnet-calculator.com/cidr.php
- RFC3513: https://tools.ietf.org/html/rfc3513

References

- IPv6 Explained for Beginners: http://www.steves-internet-guide.com/ipv6-guide/
- How to find IPv6 Prefix: https://networklessons.com/ipv6/how-to-find-ipv6-prefix/
- IPv6 Calculator: https://www.ultratools.com/tools/ipv6CIDRToRange
- TACO IPv6 Router - a Case Study in Protocol Processor Design: https://www.researchgate.net/publication/31596630_TACO_IPv6_Router_-_a_Case_Study_in_Protocol_Processor_D

References

- The TCP/IP Guide: A Comprehensive, Illustrated Internet Protocols Reference: http://www.amazon.com/TCP-Guide-Comprehensive-Illustrated-Protocols/dp/159327047X/ref=sr_1_3?s=books&ie=UTF8&qid=1297880998&sr=1-3

- IPTables tutorial: https://www.frozentux.net/iptables-tutorial/iptables-tutorial.html