
COMPILANDO CONOCIMIENTO

Redes Computacionales

CIENCIAS DE LA COMPUTACIÓN

Oscar Andrés Rosas Hernandez
Laura Andrea Morales López

Febrero 2018

Índice general

I Las Redes en General	5
1. Introducción	6
1.1. Definición	7
1.1.1. Redes Computacionales	7
1.2. Clasificación de Redes	8
1.2.1. Por su Alcance	8
1.2.2. Por su Relación Funcional	8
1.2.3. Por su Medio de Transmisión	8
1.2.4. Por el Tipo de Transferencia	8
1.3. Topología de Redes	9
2. Ideas Practicas	11
2.1. Dirección MAC	12
2.2. Proceso de Encapsulación de Datos	13
3. Manejo de Errores	14
3.1. Chequeo de Redundancia Cíclica: CRC	15
3.1.1. Definición	15
3.1.2. Componentes	15
3.1.3. Ejemplos	16
3.2. Hamming	17
3.3. Suma de Comprobación: CheckSum	18
3.4. Bit de paridad	18

II	Capas	19
4.	Capa de Red	20
4.1.	Definición	21
4.2.	Funciones de la capa de red	21
5.	Capa de Transporte	22
5.1.	Definición	23
5.2.	Funciones de la capa de red	23
III	Protocolos	24
6.	En General	25
6.1.	Definiciones	26
6.1.1.	Características	26
7.	Protocolo OSI	27
7.1.	Definición	28
7.2.	Partes	28
7.3.	Diagrama OSI (Y su comparación vs IP)	29
8.	Protocolo Ethernet	30
8.1.	Frame	31
8.1.1.	Explicación	31
9.	Protocolo IP	35
9.1.	Definiciones	36
9.2.	Dirección IPv4	37
9.2.1.	Problemas con IPv4	37
9.3.	Dirección IPv6	38
9.3.1.	Haciendo un poco más fáciles las Direcciones IPv6	39
9.4.	Direcciones Especiales	40
9.5.	Clases IP	40

9.5.1. Clases A	41
9.5.2. Clases B	42
9.5.3. Clases C	43
9.5.4. Clases D	44
9.5.5. Clases E	44
9.6. Subredes	45
10.Protocolo TCP	46
10.1. Header - Encabezado	47
10.1.1. Explicación	48
11.Protocolo UDP	49
12.DHCP	50
12.1. Introducción	51
13.DNS	52
13.1. Introducción	53
14.Protocolo HDLC	54
15.Protocolo HDLC Nivel de Enlace de datos Capa:Acceso a la Red	55
15.1. High-level Data Link Control	55
15.2. Características comunes.	55
15.3. Tipos de Estaciones	56
15.3.1. Primarias	56
15.3.2. Secundarias	56
15.3.3. Combinadas	56
15.4. Transferencia de datos(Control)	56
15.4.1. NRM	56
15.4.2. ABM	56
15.4.3. ARM	56
15.5. Estructura de la trama	56

15.5.1. Flag	56
15.5.2. Dirección	56
15.5.3. Control	56
15.5.4. Secuencia para comprobar la trama	57
16.LLC	58
 IV Aparatos Físicos	 59
17.Hub	60
18.Switch	61
19.Routers	62
20.Access Points: Puntos de Acceso	63

Parte I

Las Redes en General

Capítulo 1

Introducción

1.1. Definición

1.1.1. Redes Computacionales

Decimos que una red de computadoras es un conjunto de nodos (sean computadoras personales, servidores, telefonos, etc...) interconectados por un medio físico y que se implementa una pila de protocolos para poder comunicarse entre si y compartir recursos.

Decimos que una red tiene que desempeñar las siguientes funciones:

- Encapsulamiento
Es decir
- Control de Errores
- Direccionamiento
- Multiplexión
- Segmentación y Ensamblado
- Servicios de Transmisión
- Control de Flujo
- Entrega en Orden
- Control de Conexión

1.2. Clasificación de Redes

1.2.1. Por su Alcance

- **PAN** (*Personal Area Network*) Redes de Área Personal
A una distancia de aprox 1m. Con una extensión de $1m^2$
- **LAN** (*Local Area Network*) Redes de Área Local
A una distancia de 10m-1km. Con una extensión de un cuarto.
- **CAN** Redes de Área Campus
- **MAN** (*Metropolitan Area Network*) Redes de Área Metropolitana
A una distancia de 10km. Con una extensión de una ciudad.
- **WAN** (*Wide Area Network*) Redes de Área Amplia
A una distancia de 1000km. Con una extensión de un País o Continente.
- **GAN** (*Global Area Network*) Redes de Área Global
Con una extensión de un mundo.

1.2.2. Por su Relación Funcional

- **Cliente Servidor**
- **Igual a Igual (P2P)**

1.2.3. Por su Medio de Transmisión

- **Alámbricas ó Guiado:**
Cosas como: Cable Coaxial, Fibra Óptica, Cable Trenzado o UTP
- **Inalámbricas ó No Guiado:**
Cosas como: Infrarrojo, LI-FI, WI-FI, Bluetooth

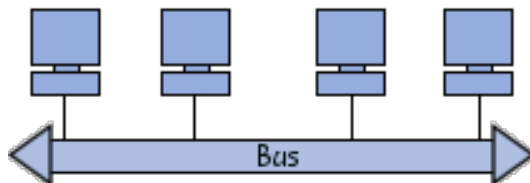
1.2.4. Por el Tipo de Transferencia

- **Simplex:** Si es que solo se puede enviar información en un sentido
- **Half-Duplex:** Si es que se puede enviar información en ambos sentidos, pero solo una a la vez.
- **Full-Duplex:** Si es que se puede enviar información en ambos sentidos, incluso ambos a la vez.

1.3. Topología de Redes

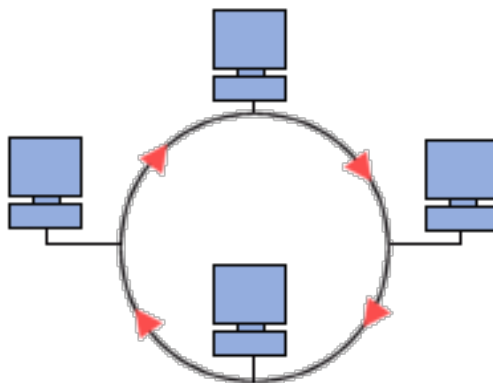
■ Bus

Todos el mismo medio de transmisión. Tampoco se pueden intentar comunicar más de 2 equipos al mismo tiempo, sino colecciones. No se necesita más que entrar al medio para poder permanecer a la Red. Son muy comunes. Si se daña el medio toda la red se viene abajo.



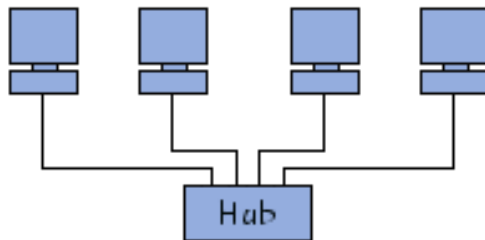
■ Anillo

Es muy seguro, no se puede añadir otro equipo fácilmente y no es bidireccional.



■ Estrella

Todos los equipos que están unidos a un mismo switch tendrán que enviar toda su información hacia el switch.



- **Árbol**

Es una extensión natural de la topología de árbol, es decir, es lo mismo de que topología de estrella de una topología de estrella, es decir, donde cada elemento de la topología de forma de estrella aquí esta dado por una topología de estrella en si misma.

- **Malla**

Es bueno... bueno una malla, donde no hay un solo camino ni forma de contactar a cada equipo. Es la mas robusta de todas.

Capítulo 2

Ideas Practicas

2.1. Dirección MAC

La dirección MAC es un identificador único de 48 bits para identificar la totalidad de dispositivos de red como por ejemplo tarjetas de red Ethernet, tarjetas de red wifi o inalámbricas, etc. Las direcciones MAC son identificadores únicos a nivel mundial para cada dispositivo y por lo tanto es imposible encontrar 2 dispositivos de red que tengan la misma dirección MAC. La entidad que se encarga de definir como se deben definir las direcciones MAC, es la IEEE. La dirección MAC consta de 48 bits y viene expresada en 12 dígitos hexadecimales.

Los primeros 24 bits, o 6 dígitos hexadecimales, de la dirección MAC contienen un código de identificación del fabricante o vendedor OUI.

Los últimos 24 bits, o 6 dígitos hexadecimales, están administrados por cada fabricante y presentan, por lo general el número de serie de la tarjeta

2.2. Proceso de Encapsulación de Datos

El proceso desde que los datos son incorporados al ordenador hasta que se transmiten al medio se llama encapsulación. Estos datos son formateados, segmentados, identificados con el direccionamiento lógico y físico para finalmente ser enviados al medio. A cada capa del modelo OSI le corresponde una **PDU** (Unidad de datos) siguiendo por lo tanto el siguiente orden de encapsulamiento:

1. Datos
2. Segmentos
3. Paquetes
4. Tramas
5. Bits

Debido a que posiblemente la cantidad de los datos sea deamasiada, la capa de transporte desde el origen se encarga de segmentarlos para así ser empaquetados debidamente, esta misma capa en el destino se encargará de reensamblar los datos y colocarlos en forma secuencial, ya que no siempre llegan a su destino en el orden en que han sido segmentados, así mismo acorde al protocolo que se esté utilizando habrá o no corrección de errores. Estos segmentos son empaquetados (paquetes o datagramas) e identificados en la capa de red con la dirección lógica o IP correspondiente al origen y destino. Ocurre lo mismo con la dirección MAC en la capa de enlace de datos formándose las tramas o frames para ser transmitidos a través de alguna interfaz. Finalmente las tramas son enviadas al medio desde la capa física.

El proceso inverso se realiza en el destino y se llama desencapsulación de datos.

Capítulo 3

Manejo de Errores

3.1. Chequeo de Redundancia Cíclica: CRC

3.1.1. Definición

Se usa en las tramas de acceso a la red. Un CRC es checksum que puede detectar la corrupción de datos que se almacenan y / o transmiten entre computadoras.

En términos generales, los CRC se calculan más eficientemente en hardware dedicado.

Dado un mensaje $\mathbf{M(X)}$ un mensaje de tamaño \mathbf{m} bits a ser transmitido, se le adicionan \mathbf{n} bits de redundancia para formar una trama de $\mathbf{T(X)}$ de tamaño $\mathbf{m + n}$ la cual esta formada de la siguiente forma:

$$T = 2^n M + FCS$$

3.1.2. Componentes

- **FCS:** Es la secuencia comprobación de la trama.

Se genera a partir del residuo de dividir $\frac{2^n M}{p}$ con n siendo el grado del polinomio.

Nota que no llevaremos acarreo.

- **P:** Es un polinomio generador de la redundancia, de orden \mathbf{n} y de tamaño $\mathbf{n+1}$ bits

Entonces enviamos T y si es que al llegar a la tarjeta de red $\frac{T}{P}$ nos deja un residuo de cero quiere decir que la trama es valida.

Nota que el polinomio tiene que cumplir con que su primer y ultimo bit sea 1.

3.1.3. Ejemplos

Ejemplo 1:

Dado $M(x) = '1010001101'$ y $P(x) = 110101$

Entonces $P(x)$ simplemente es la simplificación de $P(x) = 1x^5 + 1x^4 + 0x^3 + 1x^2 + 0x^1 + 1x^0$

Por lo tanto tenemos que:

$T = 2^n M + FCS$	Por definición
$= 2^5(1010001101) + FCS$	Por Ahora multiplicamos por 2^5
$= (101000110100000) + FCS$	Manejo de bits $M \ll 5$
$= (101000110100000) + res\left(\frac{2^n M}{p}\right)$	Por definición
$= (101000110100000) + rep\left(\frac{101000110100000}{110101}\right)$	Por definición
$= (101000110100000) + res((110101)(1101010110) + 01110)$	Nota que el residuo tiene que tener n bits
$= 101000110101110$	Magia

3.2. Hamming

Nada habla mas de un código :3

```

1 import json, math
2
3 def CreateHammingCode(RawData, Show = False):
4
5     Data = []
6     DataIterator = 0
7     ActualPow = 0
8     RawDataIterator = 0
9
10    while True:
11        if (DataIterator + 1) == (2**ActualPow):
12            Data += "X"
13            ActualPow += 1
14            DataIterator += 1
15        else:
16            Data += RawData[RawDataIterator]
17            RawDataIterator += 1
18            DataIterator += 1
19
20        if (RawDataIterator == len(RawData)):
21            break
22
23    if Show: print(f"Parity Bits: {''.join(Data)}")
24    if Show: print(f"Parity Bits: {ActualPow}")
25
26    for ParityBit in range(0, ActualPow):
27
28        CounterOf1 = 0
29        for Bit, BitValue in enumerate(Data):
30            if (Bit + 1) >> ParityBit & 1 == 1:
31                if (Bit != (2**ParityBit - 1)):
32                    if (Data[Bit] == "1"): CounterOf1 += 1
33
34            if (CounterOf1 % 2 == 0):
35                Data[(2**ParityBit - 1)] = "0"
36            else:
37                Data[(2**ParityBit - 1)] = "1"
38
39    return ''.join(Data)
40
41 def CheckErrorHammingCode(Data, Show = False):
42
43     ErrorAdress = 0
44     NumberOfParityBits = math.ceil(math.log2(len(Data)))
45
46     for ParityBit in range(0, NumberOfParityBits):
47         CounterOf1 = 0
48
49         for Bit, BitValue in enumerate(Data):
50             if (Bit + 1) >> ParityBit & 1 == 1:
51                 if (Data[Bit] == "1"): CounterOf1 += 1
52
53             if (CounterOf1 % 2 != 0):
54                 ErrorAdress += 2**(ParityBit)
55
56     if Show: print(f"Error Address: {ErrorAdress}")
57
58     if ErrorAdress == 0:
59         return Data
60
61     Data = list(Data)
62
63     if Data[ErrorAdress - 1] == "1":
64         Data[ErrorAdress - 1] = "0"
65     else:
66         Data[ErrorAdress - 1] = "1"
67
68     return ''.join(Data)
69
70 with open('HammingData.json', encoding='utf-8') as DataFile:
71     HammingData = json.loads(DataFile.read())
72
73     RawData = HammingData["ToEncodeByHamming"]
74     DataToCheck = HammingData["ToCheckByHamming"]
75
76     print(CreateHammingCode(RawData, True))
77     print(CheckErrorHammingCode(DataToCheck, True))

```

3.3. Suma de Comprobación: CheckSum

Este algoritmo permite verificar la integridad de la PDU y su calculo es de la siguiente manera:

- Ordena los datos en palabras de 16 bits
- Poner ceros en la posición del checksum y sumar con acarreo
- Suma cualquier acarreo fuera de los 16 bits
- Complementar a uno

3.4. Bit de paridad

Es un bit extra a agregar, el total de los bits debe ser par o impar, se rellena a necesidad para completar la paridad o desacompletarla. Se indica el tipo de paridad.

Parte II

Capas

Capítulo 4

Capa de Red

4.1. Definición

En esta capa se lleva a cabo el direccionamiento lógico que tiene carácter jerárquico, se selecciona la mejor ruta hacia el destino mediante el uso de tblas de enrutamiento a través del uso de protocolos de enrutamiento o por direccionamiento estático.

Protocolos de Capa de Red son por ejemplo: IP, IPX, RIP, IGRP, Apple Talk.

4.2. Funciones de la capa de red

La capa de red define cómo transportar el tráfico de datos entre dispositivos que no están conectados localmente en el mismo dominio de difusión, es decir, que pertenecen a diferentes redes.

Para conseguir esta comunicación se necesita conocer las direcciones lógicas asociadas a cada puesto de origen y de destino y una ruta bien definida a través de la red para alcanzar el destino deseado. La capa de red es independiente de la de enlace de datos y, por tanto, puede ser utilizada para conectividad de medios físicos diferentes.

Capítulo 5

Capa de Transporte

5.1. Definición

Es la encargada de la comunicación confiable entre host, control de flujo y de la corrección de errores entre otras cosas. Los datos son divididos en segmentos identificados con un encabezado con un número de puerto que identifica la aplicación de origen. En esta capa funcionan protocolos como UDP y TCP, siendo este último uno de los más utilizados debido a su estabilidad y confiabilidad.

5.2. Funciones de la capa de red

Para conectar dos dispositivos remotos es necesario establecer una conexión. La capa de transporte establece las reglas para esta interconexión. Permite que las estaciones finales ensamblen y reensamblen múltiples segmentos del mismo flujo de datos. Esto se hace por medio de identificadores que en TCP/IP reciben el nombre de número de puerto. La capa cuatro permite además que las aplicaciones soliciten transporte fiable entre los sistemas. Asegura que los segmentos distribuidos serán confirmados al remitente. Coloca de nuevo los segmentos en su orden correcto en el receptor. Proporciona control de flujo regulando el tráfico de datos.

En la capa de transporte, los datos pueden ser transmitidos de forma fiable o no fiable. Para IP, el protocolo TCP (Protocolo de control de transporte) es fiable u orientado a conexión con un saludo previo de tres vías, mientras que UDP (Protocolo de datagrama de usuario) no es fiable, o no orientado a conexión, donde solo se establece un saludo de dos vías antes de enviar los datos.

TCP utiliza una técnica llamada ventanas, donde se establece la cantidad de envío de paquetes antes de transmitir; mientras que en el windowing o de ventana deslizante, el flujo de envío de datos es negociado dinámicamente entre el emisor y receptor. En las ventanas deslizantes o windowing cada acuse de recibo (ACK) confirma la recepción y el envío siguiente.

Parte III

Protocolos

Capítulo 6

En General

6.1. Definiciones

Decimos que un Protocolo es un conjunto de reglas que regulan el intercambio de información entre entes.

6.1.1. Características

- **Sintaxis:**

Es el formato de los datos, codificación y niveles de señal.

- **Semántica:**

Información de control para manejo de errores.

- **Temporización:**

Sincronización de velocidades y secuencias.

Capítulo 7

Protocolo OSI

7.1. Definición

Antes que nada, es un modelo de referencia. Pretende que los sistemas que son diseñados con base en el, se pueden comunicar sin problemas.

7.2. Partes

■ Capa Física:

Se encarga de la transmisión de datos, de cadenas de bits no estructurados sobre el medio físico y esta relacionado con:

- Voltaje necesario para representar cada bit
- Cuanto dura cada símbolo, es decir el tiempo de trama
- Si se realiza simultáneamente en ambos sentidos o no
- Como se establece una transmisión y como interrumpirla
- Especificar como serán los pines del conector de red, para que sirve cada pin pues

■ Capa de Enlace de Red:

Trabaja con direcciones físicas.

Proporciona el servicio de transferencia de datos (tramas) llevando a cabo la sincronización y corrección de datos, así como el control de flujo.

■ Capa de Red:

Trabaja con IP (es decir, direcciones lógicas) para poder conectar dos redes. Es responsable del establecimiento, mantenimiento y cierre de conexión.

También brinda las funciones de direccionamiento lógico y enrutamiento.

Se direccionan de manera lógica y no física para evitar problemas con el hardware.

■ Capa de Transporte:

Hablaremos de si será un archivo orientado a conexión (TCP), o si no esta orientado a conexión (UDP), es decir la importancia que le damos a si queremos los datos integros o si requerimos gran velocidad.

Proporciona seguridad, transferencia y transporte de datos entre los puntos finales también proporcionan mecanismos de control de flujo y de errores en el origen y destino.

Esta proporciona el control de la comunicación entre diferentes aplicaciones, establece, gestiona y cierra la comunicación entre aplicaciones

- **Capa de Sesión:**

Hablaremos los números de puerto, un identificador de programa que nos permite ejecutar varias aplicaciones al mismo tiempo, estas son permite ejecutar unos 65,536 aplicaciones en red TCP y otros 65,536 en UDP. Si, un montón.

- **Presentación:**

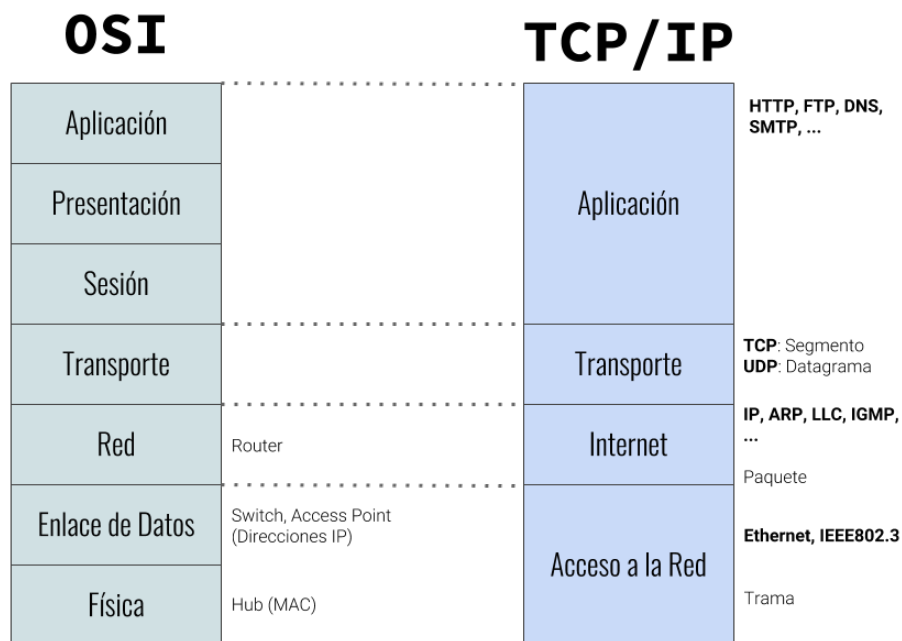
Poder transmitir los datos de manera transparente y sin importar la arquitectura de las computadoras de origen y destino.

- **Aplicación:**

Es donde trabajamos a nivel usuario y ... poquito más.

Proporciona un medio a los programas de aplicación para acceder a los servicios de red, contiene funciones de administración de aplicaciones distribuidas.

7.3. Diagrama OSI (Y su comparación vs IP)

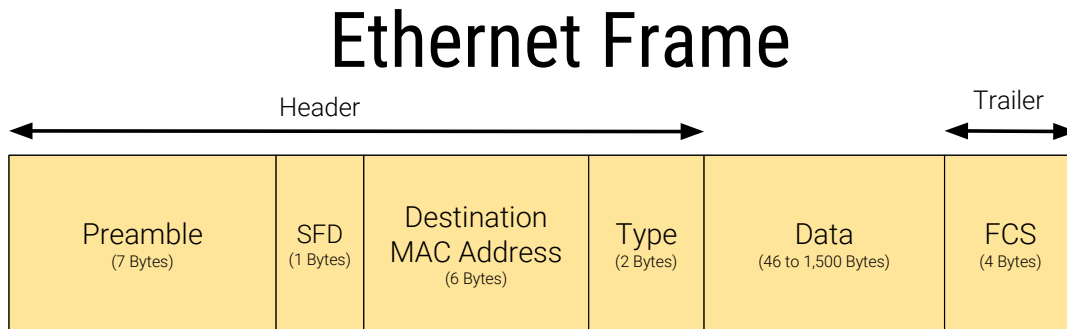


Capítulo 8

Protocolo Ethernet

8.1. Frame

Si el valor que leo en tamaño o tipo es menor a 1500 implica que sigue el protocolo LLC y el tamaño del mismo.



8.1.1. Explicación

Los campos que componen una trama ethernet son los siguientes:

- **Preámbulo (Preamble)**

Campo con una secuencia de bits utilizada para sincronizar y estabilizar el medio físico antes de iniciar la transmisión. Es una secuencia de unos y ceros, conocida que permite a los nodos saber que esta llegando un nuevo frame.

El patrón es el siguiente:

10101010 10101010 10101010 10101010 10101010 10101010 10101010

Tamaño: 7 Bytes

- **SFD (Start Frame Delimiter)**

Delimitador de inicio de trama. Campo que contiene la secuencia 10101011.

Indica el inicio de una trama de datos.

Tamaño: 1 Byte

- **Dirección de Destino (Destination Address)**

Campo que contiene la dirección MAC a la que se envía la trama.

El bit más a la izquierda del campo indica cuando la dirección es individual (indicado por un 0) o un grupo de direcciones (indicado por un 1).

El segundo bit desde la izquierda indica cuando la dirección destino es globalmente administrada (indicado por un 0).

La capa de enlace de datos del remitente añade la dirección de destino a la trama. La capa de enlace de datos del destinatario examina la dirección de destino para identificar los mensajes a recibir.

Tamaño: 6 Bytes

- **Dirección de Origen (Source Address)**

Campo que contiene la dirección MAC del dispositivo que envía la trama.

La dirección de origen es siempre una dirección individual y el bit más a la izquierda es siempre 0.

Con ella el receptor conoce a quien debe dirigir las respuestas del mensaje.

Tamaño: 6 Bytes

- **Tipo de Protocolo o Longitud**

Indica el tipo de protocolo de capa superior.

Este campo es el que distingue a las tramas IEEE 802.3 de las tramas Ethernet.

Valores para este campo iguales o menores de x05DC (1500 en decimal) indican que es una trama IEEE 802.3 y el valor representa la longitud del campo de datos.

Valores para este campo iguales o mayores de x0600 indican que es una trama Ethernet y el valor representa el tipo de protocolo.

Tamaño: 2 Bytes

- **Datos and Pad(Payload)**

Contiene los datos a transferir entre origen y destino. Si este campo fuera menor de 46 bytes se añade un campo de “relleno”, es decir pad para mantener el tamaño mínimo de paquete.

Este campo contiene los datos transferidos desde el origen hasta el destino. El tamaño máximo este campo es de 1500 bytes. Si el tamaño de este campo es menor a 46 bytes, entonces es necesario el uso del campo siguiente (Pad) para añadir bytes hasta que el tamaño de la trama alcance el valor mínimo.

Inicia en el byte 15 y se tomará hasta la longitud que se obtiene al multiplicar el último número a la derecha del byte 15 por 4. Generalmente son 20 posiciones, es decir, del byte 15 al byte 34.

Tamaño: 46 a 1,500 Bytes

- **FCS (Frame Check Sequence)**

Secuencia de verificación de trama.

Campo que contiene un valor de para control de errores, CRC (Cyclical Redundancy Check). La verificación de redundancia cíclica (CRC), consiste en un valor calculado por el emisor que resume todos los datos de la trama. El receptor calcula nuevamente el valor y, si coincide con el de la trama, entiende que la trama se ha

El campo FCS es generado ó calculado sobre los campos dirección de destino, la dirección de origen, el tipo/longitud y datos.

Tamaño: 4 Bytes

Capítulo 9

Protocolo IP

9.1. Definiciones

Debido a la cantidad de cables necesarios para conectar cada red con cada otra red del mundo no todas las redes tienen una conexión directa, es decir, no existe un cable entre tu red local y los servidores de Facebook por ejemplo.

Por eso existe el Protocolo IP que nos permite comunicarnos entre redes.

En resumen lo que permite es que tu red local solo este conectada a unas pocas redes y a varios routers, estos tienen algo llamado una tabla de direcciones, que les permite navegar entre redes hasta encontrar su destino.

El enrutamiento es parecido a la recursión, en el sentido en que no soluciona tu problema sino que solo te lleva un paso más cerca.

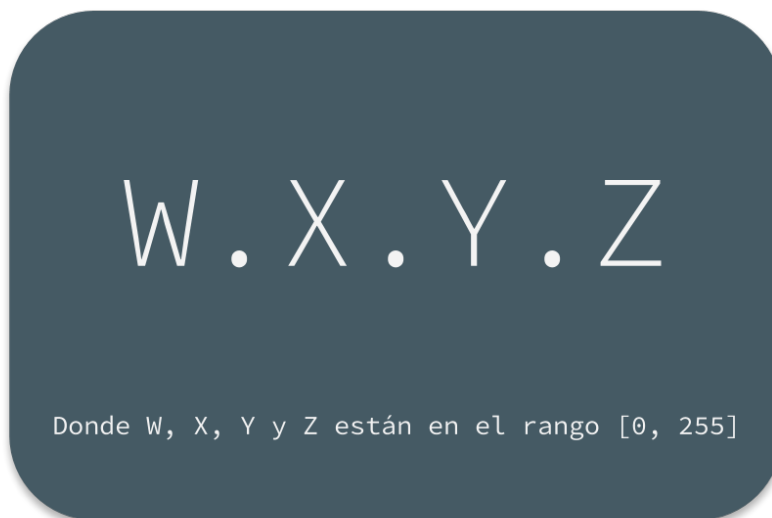
Direcciones IP

Es un identificador único (o casi, ya verás después porque). Necesitamos un identificador único porque es lo que nos permite enviar información y que la información que esperamos de regreso sepa a donde llegar.

9.2. Dirección IPv4

Como fue originalmente desarrollado este esquema podría alojar un identificador de **32 bits** a cada dispositivo que se quisiera conectar a internet. Esto nos daría algo así como 4 mil millones de posibles direcciones IP.

La convención es que estos serían representados como 4 conjuntos de 8 bits representados en decimal (una forma un poquito más amigable al público general), es decir:



Por ejemplo una IP v4 válida podría ser 140.247.220.12.

9.2.1. Problemas con IPv4

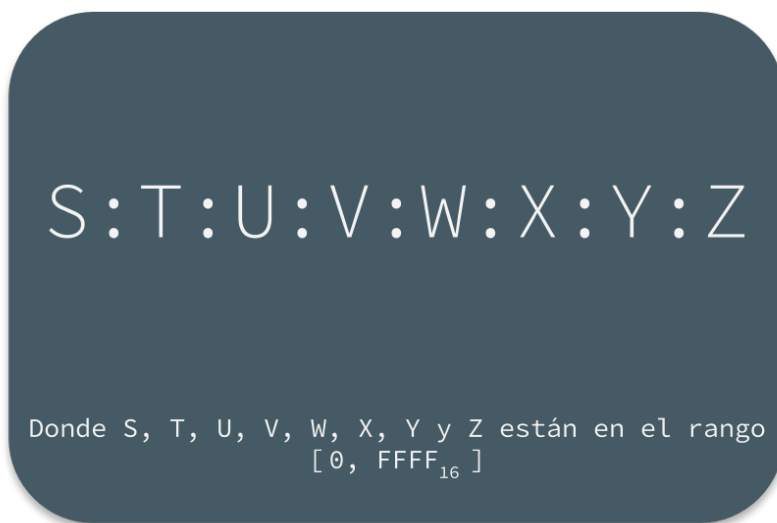
Ahora, recuerda que te dije que IP v4 acepta unos 4 mil millones de direcciones válidas, ahora el problema es que ahora mismo hay vivos mas de 7 mil millones de personas (A principios del siglo XXI) cada una con seguramente más de un dispositivo que quieran conectar a internet.

Por lo tanto tenemos que encontrar una forma de solucionar esto.

9.3. Dirección IPv6

Como vimos antes, ahora que parece que la cantidad de direcciones IPv4 se nos esta quedando corta, poco a poco estamos pasando de IPv4 a IPv6 que contará con nada menos y nada mas que **128 bits** para una dirección, es decir nos permitirá tener unas más o menos: 340, 282, 366, 920, 938, 463, 463, 374, 607, 431, 768, 211, 456 posibles direcciones IP. Un chingo.

La convención es que estos serían representados como 8 conjuntos de 65535 bits representados en hexadecimal (porque de otra manera sale un númeroote), es decir:



Por ejemplo una IPv6 podría ser `2001 : 0DB8 : 0000 : 0042 : 0000 : 8A2E : 0370 : 7334`

9.3.1. Haciendo un poco más fáciles las Direcciones IPv6

Ahora, todo esta mucho mejor que con IPv4, pero tenemos un pequeño problema, sus direcciones son moustrosamente enormes, por lo que tuvimos que hacer algunas simplificaciones para los humanos:

- Ignora los ceros dentro de cada grupo de 4 dígitos hexadecimales:

Ejemplo:

De 2001 : 0DB8 : 0000 : 0042 : 0000 : 8A2E : 0370 : 7334
a 2001 : 0DB8 : 0 : 42 : 0 : 8A2E : 370 : 7334

- Si tienes un montón de ceros pon :: y da por sentado que quien lee esta dirección tiene cerebro y puede entender que ahí van ceros:

Ejemplo:

De 2001 : 0DB8 : 0000 : 0042 : 0000 : 0000 : 0000 : 0000
a 2001 : 0DB8 : 0000 : 0042 ::

9.4. Direcciones Especiales

- **Ausencia de Dirección:** [0.0.0.0](#)

- **Broadcast:** [255.255.255.255](#)

- **Loopback (LocalHost):** [127.0.0.0](#)

Esta dirección nunca saldrá de nuestra máquina

- **Direcciones Privadas (RFC 1918):**

Aquí ya se complican las cosas, estas direcciones estan reservadas:

- Para la Clase A tenemos el rango [10.0.0.0](#) a [10.255.255.255](#)
- Para la Clase B tenemos el rango [172.16.0.0](#) a [172.31.255.255](#)
- Para la Clase B tenemos el rango [192.168.0.0](#) a [192.168.255.255](#)

9.5. Clases IP

En general, el sistema de direccionamiento IPv4 se divide en cinco clases de direcciones IP. Todas las cinco clases están identificadas por el primer octeto de dirección IP.

Podemos entonces diferenciar entre 2 grandes grupos:

- **Unicast:**

Son los clásicos, lo que se pueden dividir entre ID de la Red y ID del host Estas permiten la posibilidad de ejecutar enrutamiento.

Estan son las clases A, B, C

Multicast:

Puede que los host pertenezcan a diferentes segmentos de red, por lo tanto no es posible hacer la división clásica.

Estan son las clases D, E

9.5.1. Clases A

La Clase A son todas cuya representación en binario empieza por 0.

Características

- **Rango**

Sus direcciones estan en el rango de:

0.0.0.0 a 127.255.255.255

Pero solo utiles tenemos a:

1.0.0.1 a 126.255.255.254

- **Capacidad**

La Clase A puede tener:

- 126 Redes ($2^7 - 2$)
- 16777214 Hosts ($2^{24} - 2$)

- **Forma**

El primer octeto es de la forma $N.H.H.H$, donde $N = Red$ y $H = Host$

- **Máscara de Red**

La máscara de red entonces por defecto es 255.0.0.0

- **Cosas Especiales**

Los rangos 127.X.X.X se usan para loopback y para diagnóstico

IP Class A

0 (Class ID)	ID of Network (7 Bytes)	ID of Host (24 Bytes)
-----------------	----------------------------	--------------------------

9.5.2. Clases B

La Clase B son todas cuya representación en binario empieza por 10.

Características

- **Rango**

Sus direcciones estan en el rango de:

128.0.0.0 a 191.255.255.255

Pero solo utiles tenemos a:

218.0.0.1 a 191.255.255.254

- **Capacidad**

La Clase B puede tener:

- 16384 Redes ($2^{14} - 2$)
- 65534 Hosts ($2^{16} - 2$)

- **Forma**

Es de la forma $N.N.H.H$, donde $N = Red$ y $H = Host$

- **Máscara de Red**

La máscara de red entonces por defecto es 255.255.0.0

IP Class B



9.5.3. Clases C

La Clase C son todas cuya representación en binario empieza por 110.

Características

- **Rango**

Sus direcciones estan en el rango de:

192.0.0.0 a 223.255.255.255

Pero solo utiles tenemos a:

192.0.0.1 a 223.255.255.254

- **Capacidad**

La Clase C puede tener:

- 2097152 Redes ($2^{21} - 2$)
- 254 Hosts ($2^8 - 2$)

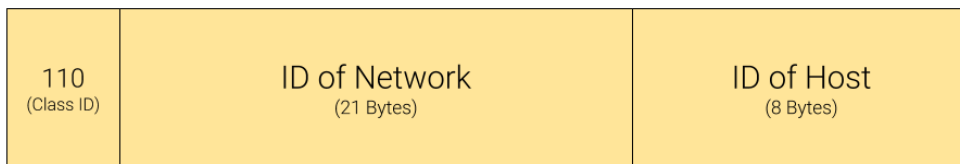
- **Forma**

Es de la forma $N.N.N.H$, donde $N = Red$ y $H = Host$

- **Máscara de Red**

La máscara de red entonces por defecto es 255.255.255.0

IP Class C



9.5.4. Clases D

La Clase C son todas cuya representación en binario empieza por 1110 Se pueden direccionar hasta 2^{28} grupos multicast.

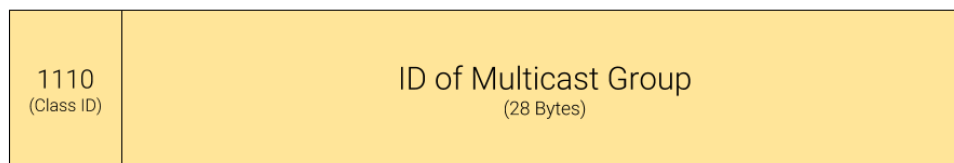
Es decir las direcciones en el rango:

224.0.0.0 a 239.255.255.255

Pero solo utiles tenemos a:

224.0.0.1 a 239.255.255.254

IP Class D



9.5.5. Clases E

La Clase C son todas cuya representación en binario empieza por 1111 No se usa :v

Tiene las direcciones en el rango:

240.0.0.0 a 255.255.255.255

9.6. Subredes

Las clasificación original de las dirección IPv4 provocan un gran desperdicio.

Ejemplo:

Sea la empresa ABC que se conforma de los sig departamentos.

- Ventas 25 Host.
- Mercadotécnia 4 Host.
- Producción 50 Host.
- Recursos Humanos 10 Host.

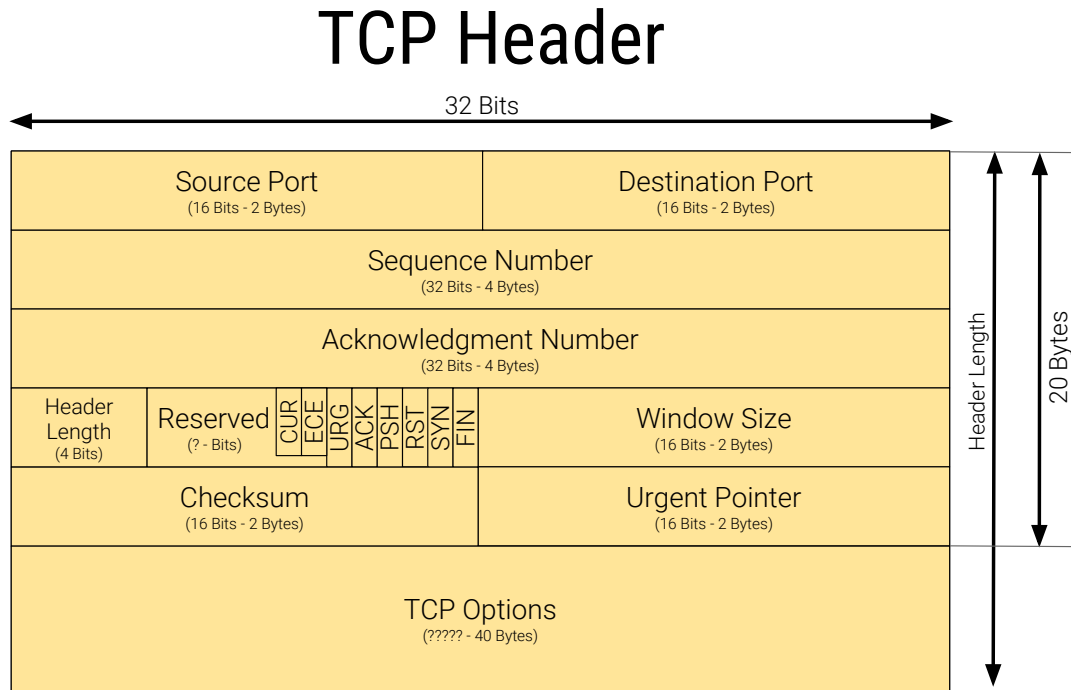
El esquema de direcciones podría ser :

- Opción A: [200.0.0.0/24](#)
Total Host: 89.
Esto implica que todos los host estan en una misma clase C y ademas de cualquier área pueden entrar a los demás host.
Sin mencionar el enorme desperdicio de host.
- Opción B: 1 Red de clase C para cada departamento.
Ventas [200.0.0.0/24](#)
Mercadotécnia [200.0.1.0/24](#)
Producción [200.0.2.0/24](#)
Recursos Humanos [200.0.4.0/24](#)
- Opción C: Dividir una red de la clase C en varias subredes.
 - [200.0.0.0/26](#)

Capítulo 10

Protocolo TCP

10.1. Header - Encabezado



10.1.1. Explicación

- **Source Port - Puerto de Origen**

Es solo el número de puerto de origen.

Tamaño: 16 bits - 2 bytes

Capítulo 11

Protocolo UDP

Capítulo 12

DHCP

Espera, espera ... ¿Cómo obtengo mi dirección IP?

12.1. Introducción

Este es un protocolo que nos permite dar una dirección IP a cada dispositivo. Está dado por *Dynamic Host Configuration Protocol*

Después de todo, si no existiera este protocolo, ¿Cómo obtengoo una dirección IP? No puedo tomar la que quiera, porque sino puede que haya más de una persona que piense como yo y elija mi misma dirección. Y si sabes lo que es que alguien en tu calle tenga el mismo número que tu, sabes que ahí tienes un problema.

Capítulo 13

DNS

Espera, espera ... ¿y cuál es la IP de Cats.com?

13.1. Introducción

Este es una aplicación que nos permite hacer una traducción de direcciones IP a nombres un poco más amigables para los seres humanos. Esta dada por *Domain Name System*. Es básicamente como las páginas amarillas del internet

Capítulo 14

Protocolo HDLC

Capítulo 15

Protocolo HDLC Nivel de Enlace de datos Capa: Acceso a la Red

15.1. High-level Data Link Control

Es un protocolo orientado a conexión de nivel de enlace.

15.2. Características comunes.

- Orientados a bit: provee una gran eficiencia con respecto a los protocolos orientados a byte, usando la estrategia de bit-stuffing (inserción de bit). También utilizar bits de control es otra ventaja, en vez de bytes
- Posee tres etapas en la comunicación.
 - Establecimiento Lógico de enlace(Logical Link Setup)
 - Transmisión de información.
 - Liberación de enlace.
- Control de Flujo
- Control de errores.
- Permite el sondeo de terminales.
- Protocolos de ventana deslizante.

15.3. Tipos de Estaciones

15.3.1. Primarias

15.3.2. Secundarias

15.3.3. Combinadas

15.4. Transferencia de datos(Control)

15.4.1. NRM

15.4.2. ABM

15.4.3. ARM

15.5. Estructura de la trama

En el control puede ser 1 o 2 bytes.

15.5.1. Flag

Siempre tiene un 01111110 y delimita el inicio y final de una trama.

Siempre que halla 5 "1" se agrega un cero y se transmite para evitar tener la flag y terminar la conexión antes de tiempo.

15.5.2. Dirección

Es de 1 byte y extendible, si bit menos significativo es 0 entonces utilizará un byte más, si bit menos significativo es 1 entonces es el último.

Por el primer byte son 127 estaciones. Por el segundo byte son 2^{14} estaciones.

15.5.3. Control

Es de 8 o 16 bits dependiendo de cuanto se ocupe en el tipo de trama. (Por ejemplo el extendido.) Las tramas no numeradas siempre valen 8 bits.

P/F Cuando es 0 significa null.

Cuando es 1 significa que hay que buscar el SAP(otro).

Los números de secuencia pueden ser de 3 o al 127 dependiendo del tipo de control de error y del tipo de transferencia de datos.

15.5.4. Secuencia para comprobar la trama

CRC

Checksum

Capítulo 16

LLC

Parte IV

Aparatos Físicos

Capítulo 17

Hub

Capítulo 18

Switch

Capítulo 19

Routers

Capítulo 20

Access Points: Puntos de Acceso

Bibliografía

- [1] Axel Ernesto Moreno Cervantes *Redes de Computación*. ESCOM, 2018.
- [2] Nidia Cortez. *Redes de Computadoras* ESCOM, 2018.