



INSTITUTO POLITÉCNICO NACIONAL
ESCUELA SUPERIOR DE COMPUTO



PROYECTO CALCULADORA IP

NOMBRE DE LOS ALUMNOS:
GARCÍA QUIROZ GUSTAVO IVAN
ALEJANDRO JUÁREZ QUINTOS

GRUPO: 5CV3

NOMBRE DEL PROFESOR: ALCARAZ TORRES JUAN JESUS
REDES DE COMPUTADORAS

03/12/2023

Índice

Introducción	3
Objetivos	4
Material.....	4
Marco teórico.....	5
Subneteo	5
Principales conceptos del Subneteo	5
Broadcast	5
Dirección IP	5
Tipos de IPv4	5
Dirección de Broadcast	5
Dirección de red	5
Dirección de Host.....	6
Subred.....	6
Dirección de subred	6
¿Por qué practicar Subneteo?	6
Ejemplo.....	7
Desarrollo	8
Instrucciones.....	8
Código fuente	11
Capturas de pantalla	15
Conclusiones	16
Bibliografía	16

Introducción

La Calculadora IP facilita la obtención de información esencial a partir de una dirección IP y su máscara de red. Entre los datos que puedes obtener se encuentran la dirección de broadcast, la red, la Cisco wildcard mask y el número de IPs en la red. Además, al introducir un segundo valor de máscara de red, tendrás la capacidad de diseñar subredes, permitiéndote experimentar con diferentes configuraciones y observar los resultados.

Para comprender el proceso, es crucial entender el concepto de subneteo.

Subnetear implica tomar un conjunto de direcciones IP donde todas sean locales entre sí y dividir las en distintos rangos o subredes. Cada rango posee direcciones IP que son independientes de las demás.

Si deseas determinar cuántos hosts hay en un rango IP específico, el primer paso es identificar cuántos bits están asignados para los hosts. Por ejemplo, consideremos la dirección IP 192.107.2.4 con una máscara de red de 255.255.255.0. Una vez establecido que el ID de red es 192.107.2 y el ID de host es 4, puedes utilizar la fórmula $2^N - 2$, donde N es el número de bits para los hosts. En este caso, obtendríamos $2^8 - 2 = 254$ hosts. Esto significa que en nuestro ejemplo, la red 192.107.2.0 tiene 254 direcciones IP posibles, todas locales entre sí.

Para subdividir una red en subredes, es necesario quitar bits de la sección de hosts. Para planificar la segmentación, dos consideraciones importantes son el número de usuarios por red y la cantidad de redes necesarias. Cuantos más usuarios por red, menos redes necesitarás, y viceversa. Cabe destacar que por cada red creada, se pierden 2 IPs, una asociada al broadcast y otra a la identificación de la misma red.

Objetivos

- Crear una herramienta de red que sea accesible y fácil de usar para calcular información clave a partir de direcciones IP y máscaras de red.
- Permitir a los usuarios usar la herramienta que deberá ser capaz de calcular y mostrar de manera clara los detalles de las subredes resultantes.
- Contribuir a mejorar la comprensión del subneteo y sus implicaciones.
- Proporcionar información completa y la calculadora IP deberá generar información detallada, incluyendo direcciones de broadcast, redes, Cisco wildcard mask y el número de IPs en cada red. Esto ayudará a los usuarios a obtener una visión completa de la topología de la red.

Material

1 computadora personal.

1 software para compilar código.

Marco teórico

Subneteo

El Subnetting o subneteo que es la técnica de subdividir una gran red IP física en varias redes lógicas más pequeñas, de forma que cada una de estas subnets funcionen como una red individual respecto a envíos y recepción de paquetes, aunque sigan perteneciendo a una misma red principal y a un mismo dominio. Este proceso debe ser realizado cuidadosamente, para así no desaprovechar direcciones IPv4.

Principales conceptos del Subneteo

Para el entendimiento de la práctica del subnetting, será necesario aprender o recordar ciertos conceptos claves como broadcast, dirección IP, tipos de IPv4, subredes, máscara de subred, entre otros ejemplos de subneteo

Broadcast

También conocido como transmisión o radiodifusión, el broadcast se define como el mensaje transmitido a los integrantes de una red, sin necesidad de retroalimentación. Esta conexión multipunto de una red informática, se encarga de transmitir un paquete de datos a todos los miembros de una red de comunicación, mediante el uso de una dirección de *Broadcast*.

Dirección IP

El término dirección IP o dirección de protocolo de internet hace referencia a una representación numérica de la ubicación de un dispositivo en internet o una red local. Funciona como identificador de un dispositivo, y permite el envío de información entre equipos en una red.

Tipos de IPv4

En la cuarta versión del protocolo de internet, se puede identificar 3 tipos de direcciones IP que se encargan (cada una) del cumplimiento de una tarea específica, siendo estas:

Dirección de Broadcast

Es la última dirección IP dentro de una subred y permite la transmisión de datos a una multitud de nodos receptores que hacen parte de una misma subred.

Dirección de red

Es la dirección IP a la que se refiere la red o subred. La dirección de red incluye routers en sus tablas de enrutamiento para que puedan llegar a un destino establecido y para poder conocer el origen de un paquete.

Dirección de Host

Son las direcciones IP (subnetting ip) que se le asignan a los equipos finales de la red y es usada para identificar a un dispositivo particular que esté conectado a la red.

Subred

Es un rango de direcciones lógicas usadas para maximizar el espacio de direcciones IPv4. Permiten asignar parte del espacio de la dirección host a las direcciones de red, lo que hace posible que se tengan más redes. A ese espacio de dirección de host asignada a las nuevas direcciones se le denomina número de subred.

Dirección de subred

Una dirección de subred es utilizada para comparar la dirección del destino de un mensaje con la dirección del sistema principal de origen de este, con el objetivo de establecer si el destino está ubicado en la misma dirección que el origen, o si es posible llegar al destino mediante alguna de las interfaces locales.

¿Por qué practicar Subneteo?

Dividir una gran red en una serie de pequeñas subredes o practicar el subnetting puede traer muchos beneficios para el administrador y el sistema, como por ejemplo, el funcionamiento independiente de cada una de las subnets hace que el proceso de recolección de datos se realice más rápidamente.

Esto se debe a que el subnetting permite la ampliación o reducción local del rango de direcciones IP, lo que se traduce en una mayor claridad en las redes. Además de esto, la disminución o contención del tráfico de broadcast favorece a la velocidad y optimización de la red.

El subnetting también permite una mejora en la organización de la red, debido a que es posible usar cada subred creada para cumplir una función en específico, permitiendo un direccionamiento más manejable, que ayuda optimizar y ordenar la red en general.

Del mismo modo, esta práctica aumenta los niveles de seguridad y permite un mejor manejo del tráfico, que se traduce en una gestión más fácil del tráfico que entra y que sale, lo cual facilita por ejemplo, el permitir o denegar el tráfico entre los diferentes dispositivos.

Otra de las razones para dividir una red en redes o subredes más pequeñas, es que evita difusiones innecesarias que pueden ralentizar el funcionamiento de la red.

En resumen:

- **El proceso de recolección de datos se realice más rápidamente**
- **Mejora en la organización de la red**
- **Aumenta los niveles de seguridad y permite un mejor manejo del tráfico**
- **Evita difusiones innecesarias**

Ejemplo

Se usa la siguiente dirección de red 192.168.1.0 /24

Se piden 4 subredes y se debe calcular la cantidad de host y subredes.

11111111.11111111.11111111.00000000=255.255.255.0

$$\text{Cantidad de subredes} = 2^3 - 2 = 8 - 2 = 6$$

$$2^3 - 2 \geq 4$$

192.168.1.0 /27

11111111.11111111.11111111.11100000=255.255.255.224

$$\text{Cantidad de host} = 2^3 - 2 = 8 - 2 = 6$$

$$\text{salto de subred} = 256 - 224 = 32$$

No. de red	Dir. De subred	Dir. De Host inicio	Dir. De Host final	Dir. De broadcast
1	192.168.1.0	192.168.1.1	192.168.1.30	192.168.1.31
2	192.168.1.32	192.168.1.33	192.168.1.62	192.168.1.63
3	192.168.1.64	192.168.1.65	192.168.1.94	192.168.1.95
4	192.168.1.96	192.168.1.97	192.168.1.126	192.168.1.127

Desarrollo

Crearemos una calculadora de IP para hacer subredes usando el IDE Visual Studio Code para realizar y visualizar nuestro programa y lenguaje C para programarlo.

Este programa en C se utiliza para obtener información sobre subredes a partir de una dirección IP y una máscara de subred en notación CIDR. Se explica una guía paso a paso de cómo usarlo:

Instrucciones

1. Ejecuta el programa. Se te pedirá que ingreses una dirección IPv4. Se debe ingresarla en formato de punto decimal (por ejemplo, 192.168.10.0).

```
Ingresa la direccion IPv4 ahora: 192.10.1.0
```

2. El programa verificará si la dirección IP que ingresaste es válida. Si no es válida, se te pedirá que ingreses una nueva dirección IP. Si es válida, el programa continuará.

```
Ingresa la direccion IPv4 ahora: 192.10.1.0  
Verificando: 192.10.1.0... Direccion verificada!
```

3. A continuación, se pedirá que ingreses una máscara de subred en notación CIDR. Debes ingresar un número entre 0 y 32.

```
Ingresa la direccion IPv4 ahora: 192.10.1.0  
Verificando: 192.10.1.0... Direccion verificada!  
Ingresa la mascara de subred en notacion CIDR ahora: 24
```

4. El programa verificará si el número CIDR que ingresaste es válido. Si no es válido, se te pedirá que ingreses un nuevo número. Si es válido, el programa continuará.

5. El programa imprimirá la dirección IP que ingresaste, junto con su clase correspondiente (A, B, C, D o E).


```
192.10.1.0/24 (Clase C)

Direccion de red: 192.10.1.0/24
Total hosts: 254
Primera direccion de host: 192.10.1.1
Ultima direccion de host: 192.10.1.254
Broadcast address: 192.10.1.255
Ingresa el numero de redes requeridas, o q para salir: █
```

7. A continuación, se te pedirá que ingreses el número de redes requeridas. Si ingresas 'q', el programa terminará. Si ingresas un número, el programa calculará la cantidad de bits necesarios para contener las redes requeridas.

```
Ingresa el numero de redes requeridas, o q para salir: 4
```

8. Si la cantidad de redes es demasiado grande para ser acomodada, se te informará y se te pedirá que ingreses un nuevo número de redes.

9. Finalmente, el programa imprimirá información sobre cada subred que se creará.

```
Total de subredes a ser creadas: 6
-----
Direccion de red: 192.10.1.0/27
Total hosts: 30
Primera direccion de host: 192.10.1.1
Ultima direccion de host: 192.10.1.30
Broadcast address: 192.10.1.31

Direccion de red: 192.10.1.32/27
Total hosts: 30
Primera direccion de host: 192.10.1.33
Ultima direccion de host: 192.10.1.62
Broadcast address: 192.10.1.63

Direccion de red: 192.10.1.64/27
Total hosts: 30
Primera direccion de host: 192.10.1.65
Ultima direccion de host: 192.10.1.94
Broadcast address: 192.10.1.95

Direccion de red: 192.10.1.96/27
Total hosts: 30
Primera direccion de host: 192.10.1.97
Ultima direccion de host: 192.10.1.126
Broadcast address: 192.10.1.127

Direccion de red: 192.10.1.128/27
Total hosts: 30
Primera direccion de host: 192.10.1.129
Ultima direccion de host: 192.10.1.158
Broadcast address: 192.10.1.159

Direccion de red: 192.10.1.160/27
Total hosts: 30
Primera direccion de host: 192.10.1.161
Ultima direccion de host: 192.10.1.190
Broadcast address: 192.10.1.191
```

Este programa debe ser compilado y ejecutado en un entorno que soporte el lenguaje de programación C, como GCC en Linux o MinGW en Windows.

Código fuente

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include <ctype.h>
#include <math.h>
#include <stdint.h>

int ipVerify(char* ipAddress, unsigned char* octetArray) {

    char* token;
    int i = 0;
    int j = 0;
    int periodCount = 0;
    token = strtok(ipAddress, ".");
    while (token != NULL) {
        for (j=0; j<strlen(token); j++) {
            if (isdigit(token[j]) == 0) {
                break;
            }
        }
        if (strlen(token) > 0 && strlen(token) < 4 && j == strlen(token)
            && atoi(token) < 256 && atoi(token) >= 0) {
            periodCount++;
            octetArray[i] = atoi(token);
        } else {
            break;
        }

        i++;

        token = strtok(NULL, ".");
    }

    if (periodCount != 4) {
        return 0;
    } else {
        return 1;
    }
}

void printSubnetInfo(uint32_t* addressOctets, int* CIDR, int* subnetBits) {
```

```

uint32_t netAddress;
uint32_t netMask;

netMask = (0xFFFFFFFF << (32 - (*CIDR + *subnetBits)) & 0xFFFFFFFF);
netAddress = *addressOctets & netMask;
printf("\nDireccion de red: %d.%d.%d.%d/%d\n", (netAddress >> 24) &
0xFF, (netAddress >> 16) & 0xFF,
        (netAddress >> 8) & 0xFF, (netAddress) & 0xFF,
*CIDR + *subnetBits);
printf("Total hosts: %d\n", ((netAddress | ~netMask) - netAddress) - 1);
printf("Primera direccion de host: %d.%d.%d.%d\n", ((netAddress + 1) >>
24) & 0xFF, ((netAddress + 1) >> 16) & 0xFF,
        ((netAddress + 1) >> 8) & 0xFF, (netAddress + 1)
& 0xFF);

netAddress = netAddress | ~netMask;

printf("Ultima direccion de host: %d.%d.%d.%d\n", ((netAddress - 1) >>
24) & 0xFF, ((netAddress - 1) >> 16) & 0xFF,
        ((netAddress - 1) >> 8) & 0xFF, (netAddress - 1)
& 0xFF);

printf("Broadcast address: %d.%d.%d.%d\n", (netAddress >> 24) & 0xFF,
(netAddress >> 16) & 0xFF,
        (netAddress >> 8) & 0xFF, (netAddress) & 0xFF);
}

int main() {

    char ipAddress[18];
    char buffer[4];
    int CIDR;
    unsigned char* octetArray;
    octetArray = calloc(4, sizeof(char));
    uint32_t addressOctets;

    int subnetNumber;
    int subnetBits = 0;
    int totalSubnets = 0;
    uint32_t currentSubnet;
    int i;

    // conseguir la dirección IP
    while (1) {
        printf("Ingrese la direccion IPv4 ahora: ");

```

```

fgets(ipAddress, 17, stdin);
ipAddress[strlen(ipAddress)-1] = '\0';

printf("Verificando: %s... ", ipAddress);

// Verificador de IP
if (ipVerify(ipAddress, octetArray) == 0) {
    printf("IP no valida ingresada.\n");
} else {
    printf("Direccion verificada!\n");
    break;
}
}
while (1) {
    printf("Ingresa la mascara de subred en notacion CIDR ahora: ");
    fgets(buffer, 4, stdin);

    CIDR = atoi(buffer);

    if (CIDR > 0 && CIDR < 32) {
        break;
    } else {
        printf("Se ha introducido un CIDR invalido. Intenta otra
vez.\n");
    }
}

printf("\n%d.%d.%d.%d/%d ", octetArray[0], octetArray[1], octetArray[2],
octetArray[3], CIDR);

if (octetArray[0] > 239) {
    printf("(Clase E)\n");
} else if (octetArray[0] > 223) {
    printf("(Clase D)\n");
} else if (octetArray[0] > 191) {
    printf("(Clase C)\n");
} else if (octetArray[0] > 127) {
    printf("(Clase B)\n");
} else {
    printf("(Clase A)\n");
}

addressOctets = (octetArray[0] << 24) | (octetArray[1] << 16) |
(octetArray[2] << 8) | (octetArray[3]);
printSubnetInfo(&addressOctets, &CIDR, &subnetBits);

```

```

do {
    printf("Ingresa el numero de redes requeridas, o q para salir: ");
    fgets(buffer, 4, stdin);
    subnetNumber = atoi(buffer);

    if (subnetNumber == 0) {
        printf("Saliendo...\n");
        exit(0);
    }
    while (subnetNumber > totalSubnets) {
        subnetBits++;
        totalSubnets = pow(2, subnetBits) - 2;
    }
    if ((CIDR + subnetBits) > 31) {
        printf("La cantidad de redes es demasiado grande para ser
acomodada.\n");
    }
} while ((CIDR + subnetBits) > 31);

printf("\nTotal de subredes a ser creadas: %d\n-----
-----", totalSubnets);
for (i=0; i<totalSubnets; i++) {
    currentSubnet = (addressOctets & ((0xFFFFFFFF << (32 - CIDR)) &
0xFFFFFFFF))
        | i << (32 - (CIDR + subnetBits));
    printSubnetInfo(&currentSubnet, &CIDR, &subnetBits);
}

free(octetArray);

return 0;
}

```

Capturas de pantalla

Dirección de red: 192.168.0.1 /24

```
Ingrese la direccion IPv4 ahora: 192.168.0.1
Verificando: 192.168.0.1... Direccion verificada!
Ingresa la mascara de subred en notacion CIDR ahora: 24

192.168.0.1/24 (Clase C)

Direccion de red: 192.168.0.0/24
Total hosts: 254
Primera direccion de host: 192.168.0.1
Ultima direccion de host: 192.168.0.254
Broadcast address: 192.168.0.255
Ingresa el numero de redes requeridas, o q para salir: 1

Total de subredes a ser creadas: 2
-----
Direccion de red: 192.168.0.0/25
Total hosts: 126
Primera direccion de host: 192.168.0.1
Ultima direccion de host: 192.168.0.126
Broadcast address: 192.168.0.127

Direccion de red: 192.168.0.128/25
Total hosts: 126
Primera direccion de host: 192.168.0.129
Ultima direccion de host: 192.168.0.254
Broadcast address: 192.168.0.255
```

Dirección de red: 148.204.1.0 /24

```
Ingrese la direccion IPv4 ahora: 148.204.1.0
Verificando: 148.204.1.0... Direccion verificada!
Ingresa la mascara de subred en notacion CIDR ahora: 24

148.204.1.0/24 (Clase B)

Direccion de red: 148.204.1.0/24
Total hosts: 254
Primera direccion de host: 148.204.1.1
Ultima direccion de host: 148.204.1.254
Broadcast address: 148.204.1.255
Ingresa el numero de redes requeridas, o q para salir: 1

Total de subredes a ser creadas: 2
-----
Direccion de red: 148.204.1.0/25
Total hosts: 126
Primera direccion de host: 148.204.1.1
Ultima direccion de host: 148.204.1.126
Broadcast address: 148.204.1.127

Direccion de red: 148.204.1.128/25
Total hosts: 126
Primera direccion de host: 148.204.1.129
Ultima direccion de host: 148.204.1.254
```

```
Ingrese la direccion IPv4 ahora: 148.204.1.0
Verificando: 148.204.1.0... Direccion verificada!
Ingresa la mascara de subred en notacion CIDR ahora: 10

148.204.1.0/10 (Clase B)

Direccion de red: 148.192.0.0/10
Total hosts: 4194302
Primera direccion de host: 148.192.0.1
Ultima direccion de host: 148.255.255.254
Broadcast address: 148.255.255.255
Ingresa el numero de redes requeridas, o q para salir: q
Saliendo...
```

Conclusiones

- GARCÍA QUIROZ GUSTAVO IVAN

El subneteo es esencial en esta práctica porque permite una gestión eficiente de direcciones IP al dividir una red en subredes más pequeñas. Esto facilita la optimización de recursos, la segmentación lógica de la red, la reducción del tráfico de broadcast, la mejora de la seguridad, la adaptabilidad a diferentes requisitos y la optimización del uso del ancho de banda. Además, el subneteo facilita el escalado de la red y contribuye a la reducción de colisiones, mejorando así el rendimiento general de la red. En resumen, el subneteo es una herramienta fundamental para diseñar y administrar redes de manera efectiva y eficiente.

- ALEJANDRO JUÁREZ QUINTOS

Las herramientas para comprender y organizar direcciones IP, como las calculadoras de IP, desempeñan un papel importante en la gestión de redes. Agilizan procesos importantes como descifrar la notación binaria y decimal, separar subredes, identificar etiquetas de direcciones IP e identificar direcciones de red y de transmisión.

Al comprimir las tareas clave de configuración de red, estas herramientas permiten optimizar la asignación de direcciones, mejorar la eficiencia y administrar su infraestructura de red de manera más efectiva.

Bibliografía

- Oviedo, B. O. B., Samaniego, E. S. M. & Murillo, J. M. O. (2018). Fundamentos de redes (1.a ed.). Grupo Compás.
- Yedlapalli, S. & Md, Y. R. (2022). Fundamentos de las redes y protocolos informáticos (Spanish Edition). Ediciones Nuestro ConocimieBibliografía
- Patrizio, A. (2019, diciembre 12). ¿Qué es una dirección IP? ¿Qué es una dirección IP?; Avast. <https://www.avast.com/es-es/c-what-is-an-ip-address>
- ¿Qué es Subnetting? (2022, marzo 14). KeepCoding Bootcamps. <https://keepcoding.io/blog/que-es-subnetting/nto>.