

IPV6 Completo

2000:0DB8:0000:0000:0000:0000:CCCC:0000/32

Cada um desses dígitos Hexadecimais é representado por um conjunto de 4 bits

HEXADECIMAL	DECIMAL	BINARIO
0	0	0000
1	1	0001
2	2	0010
3	3	0011
4	4	0100
5	5	0101
6	6	0110
7	7	0111
8	8	1000
9	9	1001
A	10	1010
B	11	1011
C	12	1100
D	13	1101
E	14	1110
F	15	1111

Como cada valor hexa decimal é formado por 4 bits, um IPV6 tem 32 números hexadecimais, logo um IPV6 possui $32 \times 4 = 128$ bits, logo o prefixo pode variar entre essa faixa de 0 a 128, alguns não podendo serem utilizados claro. Fazendo a conversão temos que:

HEXADECIMAL	BINARIO
2000	0010000000000000
0DB8	0000110110111000
0000	0000000000000000
0000	0000000000000000
0000	0000000000000000
0000	0000000000000000
0000	0000000000000000
CCCC	1100110011001100
0000	0000000000000000

A barra na segunda linha faz referência ao /32 do endereço IP, que está posicionado logo após o bit número 32, indicando que todos os valores a direita da barra podem mudar para cada IP de Host (Computador na rede) e que os valores a esquerda da barra devem permanecer inalterados.

Agora vamos para as divisões dessa rede, primeiramente devemos saber em quantas subredes devemos dividir a nossa rede. Suponha que o valor de subredes necessárias seja X, para calcular o novo barramento devemos seguir as seguintes etapas.

Primeiro devemos calcular o número de bits necessários que é dado pelo arredondamento sempre para cima do $N = \log_2(X)$,

após isso o seu novo barra será $\text{novoBarra} = \text{antigoBarra} + N$.

Exemplo quero dividir o IPV6 2000:0DB8:0000:0000:0000:0000:CCCC:0000/32 em 10 subredes. Assim temos que $X = 10$, logo $N = \log_2(X) = \log_2(10) = 3$. (alguma coisa). Arredondando esse resultado para cima temos que $N = 4$,

portanto $\text{novoBarra} = \text{antigoBarra} + N = 32 + 4 \rightarrow \text{novoBarra} = 36$, ou seja, o endereço IPV6 vai ter uma nova barra entre seus números

0010000000000000: 0000110110111000|: 0000|000000000000: 0000000000000000:
0000000000000000: 0000000000000000: 1100110011001100: 0000000000000000

Todos os Bits que estão à esquerda da primeira e os Bits que estão à direita da segunda barra não devem ser alterados. Caso ele peça que você coloque os endereços de rede e broadcast, para o endereço de rede você coloca 0 em todos os bits a direita e para broadcast você coloca 1 em todos os Bits a direita da segunda barra.

Agora finalmente vamos falar sobre como encontrar essas subredes, que nada mais é que a manipulação dos Bits que estão entre a primeira e a segunda Barra. Nos dois casos que vamos estudar, você deve implementar uma tabela que contém $2^N = 16$ colunas, que nada mais é que a quantidade de subredes que podemos montar a partir da quantidade de bits que estão entre as barras. No exemplo que estamos analisando, temos 4 Bits entre as barras, como cada Bit pode assumir 0 ou 1, nos temos 2 possibilidades de valores para cada Bit, logo temos $2 * 2 * 2 * 2 = 16$ possibilidades de valores, onde essa número representa a quantidade máxima de subredes que podemos montar.

Rightmost

O Rightmost representa uma divisão onde tendo 16 possibilidades de subredes, sempre que formos montar uma subrede dentro dessas 16 possíveis, montaremos uma após o outra. Onde isso é feito manipulando os bits que estão entre as barras.

Qual subrede vamos montar	Combinação dos Bits entre as barras	Valor dos bits em decimal
1° Subrede	0000	0
2° Subrede	0001	1
3° Subrede	0010	2
4° Subrede	0011	3
5° Subrede	0100	4
6° Subrede	0101	5
7° Subrede	0110	6
8° Subrede	0111	7
9° Subrede	1000	8
10° Subrede	1001	9
11° Subrede	1010	10
12° Subrede	1011	11
13° Subrede	1100	12
14° Subrede	1101	13
15° Subrede	1110	14
16° Subrede	1111	15

A tabela a seguir representa em cada coluna da linha 0 o valor em decimal dos Bits, e cada linha da coluna 0 representa qual subredes vamos criar, na junção dessas localizações temos onde cada subrede será criada.

	00	01	02	03	04	05	06	07	08	09	10	11	12	13	14	15
1°	1°															
2°		2°														
3°			3°													
4°				4°												
5°					5°											
6°						6°										
7°							7°									
8°								8°								
9°									9°							
10°										10°						
11°											11°					
12°												12°				
13°													13°			
14°														14°		
15°															15°	
16°																16°

A partir disso copiamos os Bits a esquerda da primeira barra e os a direita da segunda barra sem altera-los.

```
0010000000000000:0000110110111000|:XXXX|000000000000: 0000000000000000:
0000000000000000: 0000000000000000: 1100110011001100: 0000000000000000
```

Cada subrede no lugar de XXXX assumirá o valor da primeira tabela da coluna 2. Por exemplo a 12° Subrede, os bits entre as barras terão a cara 1011 (11 em decimal, e B em Hexadecimal), logo o IPV6 terá a seguinte cara:

```
0010000000000000:0000110110111000|:1011|000000000000: 0000000000000000:
0000000000000000: 0000000000000000: 1100110011001100: 0000000000000000
```

Agora devemos converter esses números para Hexadecimal Novamente

BINARIO	HEXADECIMAL
0010000000000000	2000
0000110110111000	0DB8
1011000000000000	B000
0000000000000000	0000
0000000000000000	0000
0000000000000000	0000
1100110011001100	CCCC
0000000000000000	0000

2000:0DB8:**B**000:0000:0000:0000:CCCC:0000/36

O Leftmost funciona da mesma forma que o Rightmost para o calcula da combinação dos Bits, porém você não usa eles do jeito que você calculou, para o Leftmost devemos inverter a Combinação dos Bits entre as barras, tendo assim a seguinte cara:

Qual subrede vamos montar	Combinação dos Bits entre as barras	Valor dos bits em decimal
1° Subrede	(Antes: 0000) – (Agora: 0000)	0
2° Subrede	(Antes: 0001) – (Agora: 1000)	8
3° Subrede	(Antes: 0010) – (Agora: 0100)	4
4° Subrede	(Antes: 0011) – (Agora: 1100)	12
5° Subrede	(Antes: 0100) – (Agora: 0010)	2
6° Subrede	(Antes: 0101) – (Agora: 1010)	10
7° Subrede	(Antes: 0110) – (Agora: 0110)	6
8° Subrede	(Antes: 0111) – (Agora: 1110)	14
9° Subrede	(Antes: 1000) – (Agora: 0001)	1
10° Subrede	(Antes: 1001) – (Agora: 1001)	9
11° Subrede	(Antes: 1010) – (Agora: 0101)	5
12° Subrede	(Antes: 1011) – (Agora: 1101)	13
13° Subrede	(Antes: 1100) – (Agora: 0011)	3
14° Subrede	(Antes: 1101) – (Agora: 1011)	11
15° Subrede	(Antes: 1110) – (Agora: 0111)	7
16° Subrede	(Antes: 1111) – (Agora: 1111)	15

A tabela a seguir representa em cada coluna da linha 0 o valor em decimal dos Bits, e cada linha da coluna 0 representa qual subredes vamos criar, na junção dessas localizações temos onde cada subrede será criada.

[illegible]

A partir disso copiamos os Bits a esquerda da primeira barra e os a direita da segunda barra sem altera-los.

0010000000000000:0000110110111000|:XXXX|000000000000: 0000000000000000:
0000000000000000: 0000000000000000: 1100110011001100: 0000000000000000

Cada subrede no lugar de XXXX assumirá o valor da primeira tabela da coluna 2. Por exemplo a 12° Subrede, os bits entre as barras terão a cara 1101 (13 em decimal, e D em Hexadecimal), logo o IPV6 terá a seguinte cara:

0010000000000000:0000110110111000|:1101|000000000000: 0000000000000000:
0000000000000000: 0000000000000000: 1100110011001100: 0000000000000000

Agora devemos converter esses números para Hexadecimal Novamente

BINARIO	HEXADECIMAL
0010000000000000	2000
0000110110111000	0DB8
1101000000000000	D000
0000000000000000	0000
0000000000000000	0000
0000000000000000	0000
1100110011001100	CCCC
0000000000000000	0000

Que é representada da seguinte forma

2000:0DB8:D000:0000:0000:0000:CCCC:0000/36

OBSERVAÇÃO: A CADA 4 BITS VOCÊ TEM UM NUMERO HEXADECIMAL, ENTÃO CASO VOCÊ TENHA QUALQUER QUANTIDADE MAIOR QUE 1 DE BITS ENTRE AS BARRAS, VOCÊ PODE ESTAR MEXENDO EM NUMEROS DIFERENTES. EXEMPLO

0010000000000000:0000110110111000:110|10|0000000000: 0000000000000000:
0000000000000000: 0000000000000000: 1100110011001100: 0000000000000000

EM VERMELHO SÃO OS BITS DE UM HEXADECIMAL, E EM AZUL OS BITS DE OUTRO HEXADECIMAL

Qual subrede vamos montar	RightMost	LeftMost
1° Subrede	00	00
2° Subrede	01	10
3° Subrede	10	01
4° Subrede	11	11

Para a 4° Subrede temos:

BINARIO (Rightmost)	HEXA (Rightmost)	BINARIO (Rightmost)	HEXA (Leftmost)
0010000000000000	2000	0010000000000000	2000
0000110110111000	0DB8	0000110110111000	0DB8
0001100000000000	1800	0001100000000000	1800
0000000000000000	0000	0000000000000000	0000
0000000000000000	0000	0000000000000000	0000
0000000000000000	0000	0000000000000000	0000
1100110011001100	CCCC	1100110011001100	CCCC
0000000000000000	0000	0000000000000000	0000