

1. Propuest de las tramas

Estática

Estructura de la trama

| Name | Size | | IP_SRC | (6 bytes) | SRC_MASK | (6 bits) | IP_DST | (6 bytes) | DST_MASK | (6 bits) | ENCRYPT | (1 bit) | SEQ_ORD | (1 byte) | PAYLOAD_SZ | (1 byte) | PAYLOAD | (160 bytes) | CRC | (4 bytes)

Total de bits: 1437 bits

Explicación de la estructura

La trama necesita saber su destino y fuente que tiene 6 octetos cada una IP_SRC y IP_DST. Cada una con su respectiva máscara necesaria para el enrutamiento de 6 bits cada una SRC_MASK y DST_MASK, donde pueden llegar a ser representadas de manera /48, /32, /16, etc. Luego tenemos una bandera ENCRYPT que nos indica si el payload está encriptado o no. Luego de esto tenemos el número que indica el orden y número de la trama SEQ_ORD, como el contenido del payload es irrelevante para las capas inferiores, podemos decir que los grupos de tramas son independientes, es decir, que este protocolo asegurará que lleguen en orden un grupo de 255 tramas y luego se reiniciará el contador nuevamente. El PAYLOAD_SZ nos dice hasta dónde se encuentra la información del payload y dónde empiezan los bits de relleno dentro del payload. EL PAYLOAD que contiene la información, si esta información es menor a 160 bytes, se utilizan bits de relleno en 0. Y por último el CRC que nos ayuda a realizar la suma de verificación para intentar detectar si hubieron errores en la trama.

Dinámica

Estructura de la trama

| Name | Size | | IP_SRC | (6 bytes) | SRC_MASK | (6 bits) | IP_DST | (6 bytes) | DST_MASK | (6 bits) | ENCRYPT | (1 bit) | SEQ_ORD | (1 byte) | PAYLOAD_SZ | (1 byte) | PAYLOAD | (n bytes) | CRC | (4 bytes)

Min total de bits: 157 bits

Explicación de la estructura

La estructura para la propuesta dinámica es bastante similar a diferencia que ahora PAYLOAD_SZ no cuenta la cantidad de bytes de la información del PAYLOAD, ahora representa la cantidad bits que indican el tamaño del PAYLOAD. Es decir, que después del header toma los primero n bits (determinados por PAYLOAD_SZ) y estos bits representan la cantidad de bytes de la información del payload y al final se coloca el CRC.

2. Pregunta 2

Una de las razones es el algoritmo de enrutamiento ya que permite que cada paquete se enrute independientemente lo que permite que se evadan congestiones y si se llegase a caer un enrutador simplemente se define una nueva ruta para continuar enviando los paquetes sin perder la conexión y los paquetes que iban a ese enrutador dependiendo de si ya se había confirmado o no su recepción, tal vez ni

quiera todos ellos sufran por la caída del enrutador . En el caso de servicio orientado a conexión si se cae un enrutador se pierde la conexión ya que se tiene una ruta definida para el envío de paquetes, por lo que todos sus paquetes se deben abortar en caso de una caída y pérdida de memoria del enrutador. También se requiere mucho tiempo y recursos en la fase de configuración del servicio orientado a conexión.