

5 – Tolerancia a fallos en comunicación con datos

5a – Incorporar Man-In-The-Middle

1. Crear una estructura que comunique 2 nodos finales a través de un nodo intermedio. Este nodo intermedio recogerá los datos del nodo de origen y lo reenvía al nodo de destino.
2. Construir un pequeño protocolo para intercomunicar ambos, con mensaje para aceptación de mensajes recibidos en el destino y con mensaje de error para solicitar reenvío.

Los MCU que proporciona Cisco Packet Tracer tienen sólo espacio para una tarjeta de red, por lo que es posible usarlos como nodos origen y destino, pero no es posible usarlo como nodo intermedio, ya que éste necesita 2 tarjetas. Para poder poner un nodo que tenga 2 tarjetas de red hay que construirlo usando un componente genérico llamado Things.

- 1) Crear 1 Thing con dos tarjetas Ethernet:
 - 1 Thing: **Components -> Boards -> Thing**
 - Clickamos y seleccionamos la pestaña **I/O Config**.
 - En Network Adapter, seleccionar PT-IOT-NM-1CE
 - En Network Adapter 2, seleccionar PT-IOT-NM-1CE
 - En la pestaña Config, se selecciona Ethernet0 y en el panel IP Configuration, IPv4 Address se pone una dirección IP de una red.
 - En la misma pestaña, seleccionamos Ethernet1 y en el panel IP Configuration, IPv4 Address ponemos una dirección IP de una red diferente a la anterior.
- 2) En Programming se crean 2 sockets.
 - Un socket se encargará de “escuchar” al origen. Otro socket se encargará de “escuchar” al destino.
 - Hay que recordar que cada socket debe estar asociado a un puerto diferente. Ese mismo puerto es el que debe configurarse para envío en el emisor.
- 3) Conectar los MCU con el Thing:
 - Al no estar conectados a través de componentes de red intermedios hay que interconectarlos con cables Ethernet cruzados: **Connections -> Cooper Cross-Over**

5b – Incorporar codificación

1. Rehacer el protocolo para incorporar en los paquetes con control de codificación (paridad, checksum, etc.)
2. El protocolo deberá, en el origen, codificar el código de control y añadirlo al paquete. El receptor deberá extraer el código de control, computar el código a partir de los datos recibidos y comparar dicho código con el código de control enviado. Si es igual, el paquete se considera sin error, mientras que, si es distinto, el paquete tiene error y se descarta el paquete y se solicita un reenvío al origen.

3. Hacer que el origen, de manera aleatoria, realice modificaciones en el paquete (incorporar, al menos, dos de las siguientes):
 - a. En un bit aleatorio:
 - i. Bit-flip: Intercambiar el valor de un bit (cambiar un 0 por un 1 o viceversa).
 - b. En varios bits:
 - i. Chunk flip: Intercambiar los valores individuales de varios bits consecutivos.
 - ii. Stuck-at-0: Poner a 0 varios bits consecutivos.
 - iii. Stuck-at-1: Poner a 1 varios bits consecutivos.
4. Enviar un conjunto de datos conocido. Incorporar diferentes códigos y comparar el funcionamiento. ¿Qué fallos detecta cada código?

5c – Ataque con un Man-In-The-Middle

1. Incorporar un nodo intermedio que se encargue de enrutar los paquetes desde origen a destino y las respuestas desde el destino e inyectar errores (siguiendo los mismos parámetros que en caso anterior) de manera aleatoria en los paquetes.
2. Comparar los diferentes códigos con respecto a los ataques.