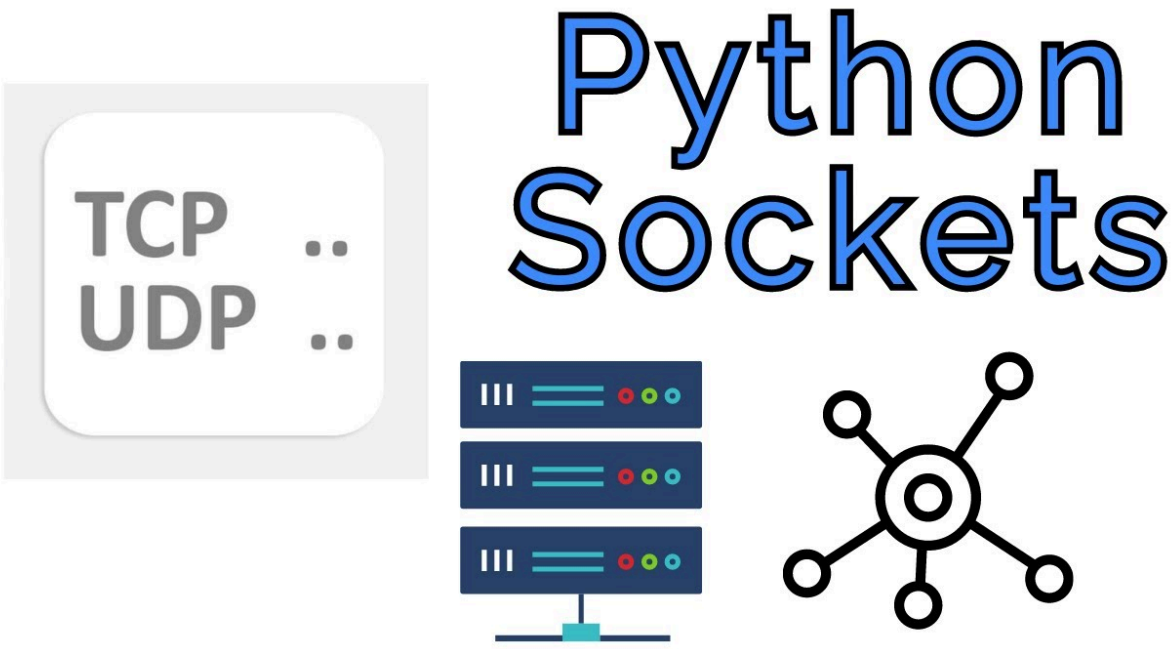
Tarea #991 Python scripts

Tarea #991



Abdiel Gabriel Hau Tun

20-02-24

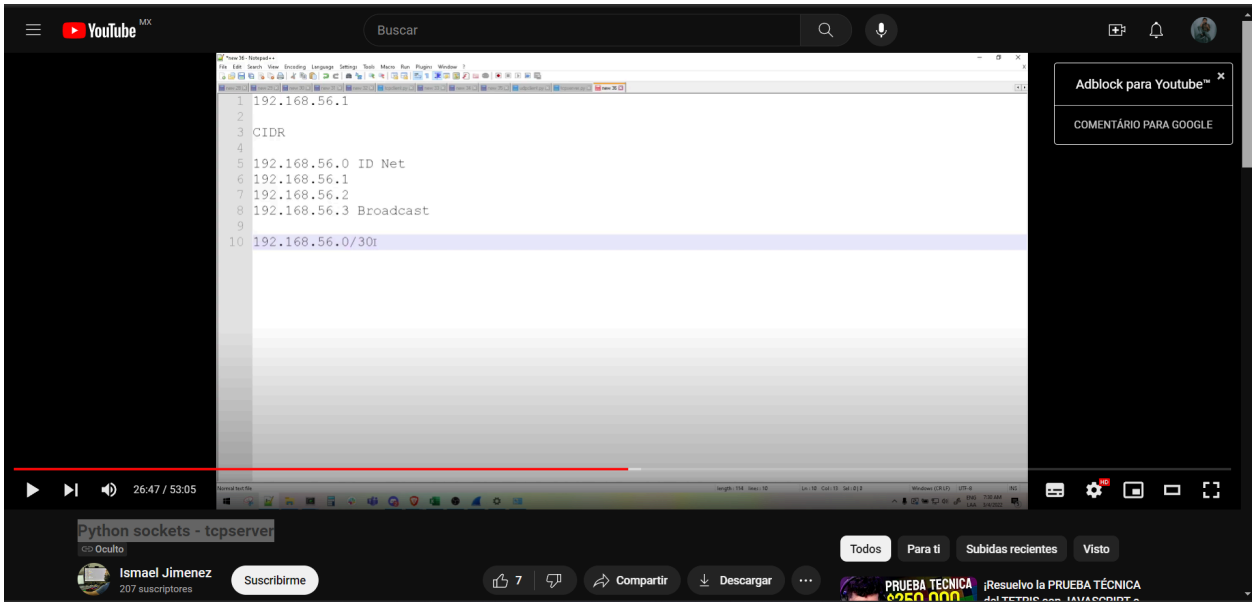
Seguridad de Datos

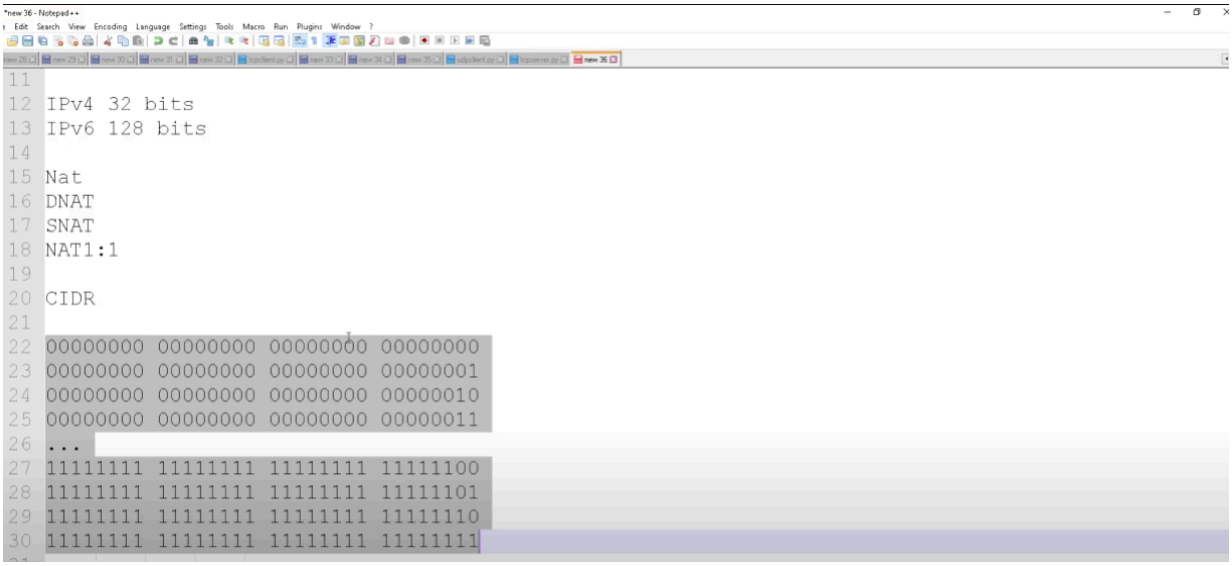
Docente: Ismael Jimenez

Python sockets - tcpserver

Un servidor TCP en Python utilizando sockets es una aplicación que escucha y acepta conexiones entrantes de clientes a través del protocolo TCP (Transmission Control Protocol). El servidor establece un punto final en una dirección IP y un número de puerto específicos, esperando conexiones de clientes que deseen comunicarse con él.

Cuando un cliente se conecta al servidor, se establece una conexión TCP bidireccional entre el cliente y el servidor, lo que permite la transferencia de datos en ambas direcciones de manera confiable y ordenada.



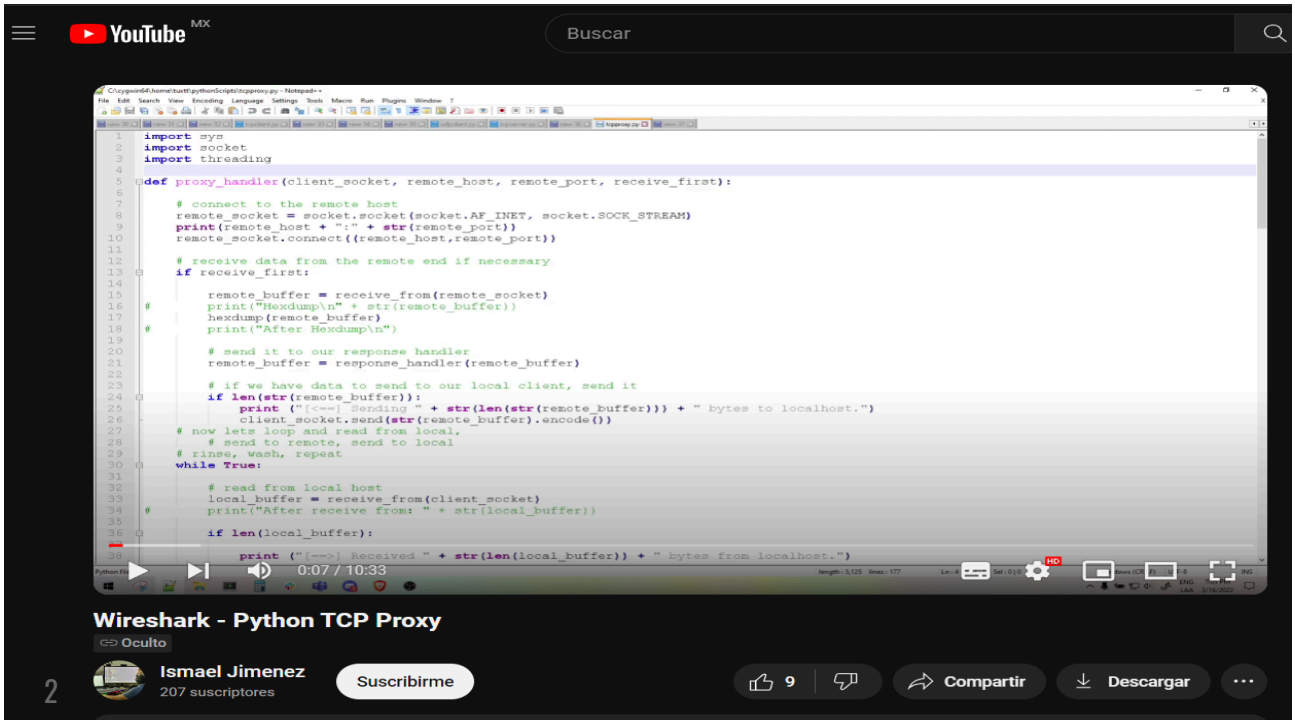


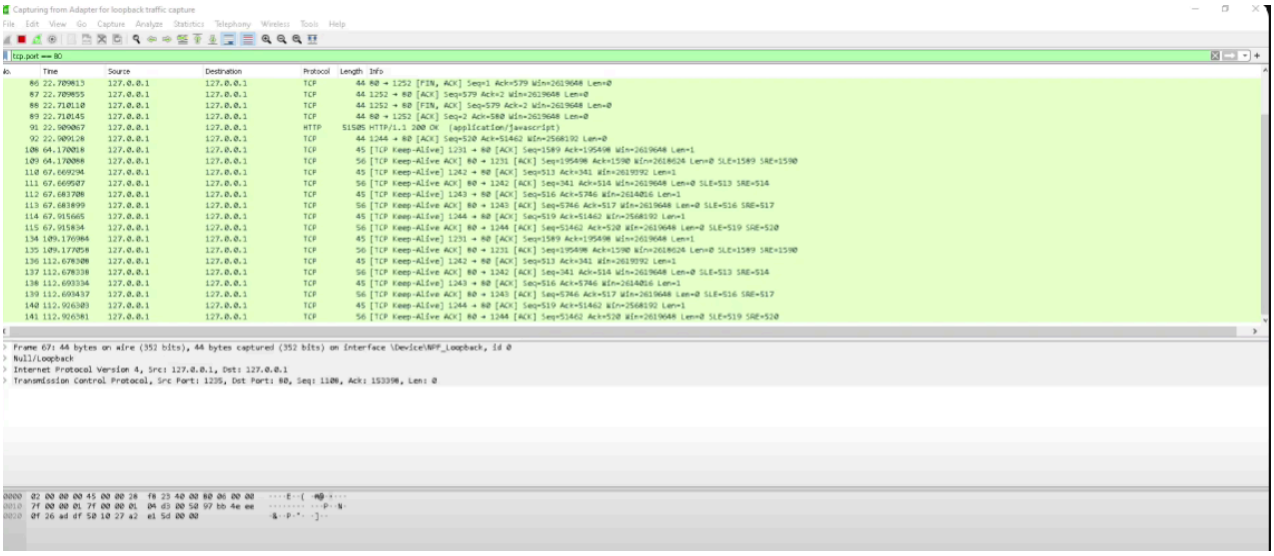
El script pcap_server.py es un programa Python que crea un servidor TCP y escucha en todas las interfaces de la computadora en el puerto 9999. Utiliza subprocesos para multiprocesamiento y múltiples conexiones. El servidor recibe mensajes, imprime una confirmación y envía una respuesta de "ack". El script pcap_client.py se utiliza para probar el servidor y se puede modificar para conectarse a la IP y al puerto del servidor. El vídeo demuestra la funcionalidad del servidor y del cliente, así como el uso de Wireshark para el análisis del tráfico. En conclusión, el video muestra una implementación básica de un servidor y un cliente TCP, con énfasis en comprender el comportamiento del servidor y capturar el tráfico de la red.

Wireshark - Python TCP Proxy

Un proxy TCP en Python es una aplicación que actúa como intermediario entre un cliente y un servidor TCP, interceptando y reenviando el tráfico entre ellos. Wireshark, por otro lado, es una herramienta de análisis de protocolos de red que permite capturar y analizar el tráfico en una red.

Cuando se combina Wireshark con un proxy TCP en Python, se puede realizar un análisis detallado del tráfico TCP que fluye a través del proxy. El proxy TCP en Python podría estar diseñado para realizar diversas funciones, como el registro de solicitudes y respuestas, la modificación de los datos en tránsito, la aplicación de políticas de seguridad, entre otras





En este video, se presenta un script de TCP proxy funcionando en Python 3, el cual es capaz de interceptar y modificar el tráfico que viaja a un puerto específico. Para ilustrar su funcionamiento, el narrador utiliza Wireshark para monitorear el tráfico en el loopback y filtra por el puerto 80.

Después de ejecutar el script, acceden a un sitio web a través del navegador en modo incógnito, utilizando la URL 127.0.0.1. El script intercepta la solicitud y la reenvía al sitio real, capturando toda la información en el tráfico entre el navegador y el sitio web.

El video explica que el script puede modificar el tráfico capturado, proporcionando dos funciones como ejemplo: perform_request_handler y response_handler. Estas funciones permiten alterar la solicitud del usuario y la respuesta del sitio web, respectivamente.

La actividad requerida para los espectadores es ejecutar el script TCP proxy, navegar a un sitio web específico a través de la URL 127.0.0.1, capturar pantallas de las ventanas relevantes, y subirlas a un repositorio como evidencia de la actividad completada.

El video menciona que aplicaciones como Squid proxy y Burp Suite tienen funcionalidades similares, pero en este caso, el script permite definir el comportamiento directamente en el código.