

INSTITUTO TECNOLÓGICO DE COSTA RICA

REDES
PROYECTO #1 - WAVENET

Allan Rojas
Saúl Zamora

profesor
Kevin Moraga

1 Introducción

La comunicación es un proceso fundamental para compartir ideas. Sin embargo, la brecha tecnológica y de comunicación es más notoria en áreas donde los insumos que poseen las TELCOS para implementar las soluciones necesarias, son bajos. Debido a esto, es una buena opción producir nuevos métodos alternativos de comunicación. Un ejemplo son las redes mesh de proyectos como guifi.net y openmesh. Dado lo anterior, el objetivo del presente proyecto es conocer distintas soluciones que nos permitan acercarnos más a esa libertad de expresión y disminuir la brecha tecnológica.

2 Ambiente de desarrollo

- Raspberry Pi:
 - Python
- Receptor:
 - Java

3 Estructuras de datos usadas y funciones

- Se hace uso de Java *AudioFormat* para examinar y luego interpretar el formato de los datos de entrada.
- Se usa *DataLine.Info* para guardar el tamaño de buffer interno de almacenamiento.
- Se usa *TargetDataLine* para leer el audio capturado por el buffer del *DataLine*.
- Se usa una base de datos MySQL en el Raspberry para llevar cuenta de los clientes conectados a la red.

Con las clases e interfaces mencionadas, se hace un ciclo para leer el flujo de datos entrante, definir el formato y decifrar el contenido. Luego se imprime el mensaje recibido en la consola.

4 Instrucciones de ejecución

5 Bitácora de trabajo

5.1 Allan Rojas

- 29-09-2018:

- 3 horas – Buzzer Python
- 29-09-2018:
 - 4 horas – Database for Node Directory with MariaDB
- 30-09-2018:
 - 4 horas – Java Audio Listening Programming
- 30-09-2018:
 - 3 horas – Media Access to Raspberry
- 01-10-2018:
 - 5 horas – Mac Address Get / Add to Package
- 10-10-2018:
 - 6 horas – Package Generation and Onion Routing
- 19-10-2018:
 - 4 horas – Scapy Implementation add to Buzzer
- 29-10-2018:
 - 6 horas – Scapy Implementation add to Buzzer

Total de Horas Trabajadas : 35

5.2 Saúl Zamora

- 23-09-2018:
 - 2 horas - Investigar RFC.
- 24-09-2018:
 - 4 horas - Investigar Scapy.
- 25-09-2018:
 - 2 horas - Investigar scraping.
- 26-09-2018:
 - 2 horas - Investigar scraping.
- 27-09-2018:
 - 2 horas - Investigar onion routing.
- 28-09-2018:

- 2 horas - Investigar onion routing.
- 29-09-2018:
 - 5 horas - Investigar audio format en Java.
- 30-09-2018:
 - 2 horas - Investigar scraping.
- 01-10-2018:
 - 2 horas - Investigar como hacer el relay chat.
 - 2 horas - Investigar sobre el servidor IRC.
- 02-10-2018:
 - 2 horas - Documentación.
- 27-10-2018:
 - 4 horas - Documentación y RFC.
- 29-10-2018:
 - 2 horas - Documentación y RFC.

Total de horas trabajadas: 33 horas.

6 Comentarios finales

- El sistema maneja la conexión de nuevos nodos (clientes) en una base de datos.
- Se maneja el control de direcciones MAC.
- Los clientes no distinguen si los paquetes recibidos son para ellos, entonces simplemente reciben y leen todos los paquetes, no desechan nada.
- El algoritmo de cebolla todavía está incompleto.

7 Conclusiones

- Uno como usuario normalmente toma por sentado el funcionamiento de las redes de Internet. Después de intentar implementar una red con un comportamiento similar al de la red de Internet, es evidente la complejidad que lleva solucionar un problema como este.
- A diferencia del proceso de comunicación convencional (conversaciones, diálogos, etc), las implementaciones digitales de dicho proceso plasman todas las complejidades del proceso que como seres humanos simplemente ya sabemos o tomamos por sentado.

References

- [1] Gpiozero.readthedocs.io. (2018). *13. API - Output Devices — Gpiozero 1.4.1 Documentation*. [online] Available at: https://gpiozero.readthedocs.io/en/stable/api_output.html#buzzer
- [2] community., P. (2018). *Scapy*. [online] Scapy.net. Available at: <https://scapy.net/>
- [3] GitHub. (2018). *secdev/scapy*. [online] Available at: <https://github.com/secdev/scapy>
- [4] Docs.python.org. (2018). *15.1. hashlib — Secure hashes and message digests — Python 3.3.7 documentation*. [online] Available at: <https://docs.python.org/3.3/library/hashlib.html>
- [5] *Ieee802.org*. (2018). [online] Available at: <http://www.ieee802.org/IEEE-802-LMSC-OverviewGuide-02SEPT>
- [6] scraping, T. (2018). *Tor IP changing and web scraping*. [online] Dm295.blogspot.com. Available at: <https://dm295.blogspot.com/2016/02/tor-ip-changing-and-web-scraping.html>
- [7] *Eprint.iacr.org*. (2018). [online] Available at: <https://eprint.iacr.org/2011/308.pdf>
- [8] Onion-router.net. (2018). *Onion Routing: Path Selection Algorithms*. [online] Available at: <https://www.onion-router.net/Archives/Route/Alg1/ThreeConnected.html>
- [9] Onion-router.net. (2018). *Onion Routing: Investigation of Route Selection Algorithms*. [online] Available at: <https://www.onion-router.net/Archives/Route/index.html>
- [10] Es.wikipedia.org. (2018). *Internet Relay Chat*. [online] Available at: https://es.wikipedia.org/wiki/Internet_Relay_Chat
- [11] Es.tldp.org. (2018). *Instalación y configuración de un servidor de IRC: Introducción*. [online] Available at: <http://es.tldp.org/COMO-INSFLUG/COMOs/Servidor-IRC-Como/Servidor-IRC-Como-2.html>
- [12] *Rfc-editor.org*. (2018). [online] Available at: <http://www.rfc-editor.org/rfc/rfc1459.txt>