Sistema de comunicaciones seguras con segmentación virtual de dominios

Avance de proyecto

Alberto Daniel Lange

Ingeniería en Telecomunicaciones
Instituto Balseiro

25 de febrero de 2025

1 Introducción
2 Revisión bibliográfica
3 Desarrollo
4 Conclusiones

1 Introducción
2 Revisión bibliográfica
3 Desarrollo
4 Conclusiones

Introducción

- Desarrollo de un encriptador para asegurar las comunicaciones entre sitios.

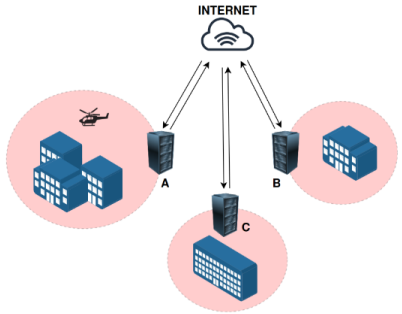
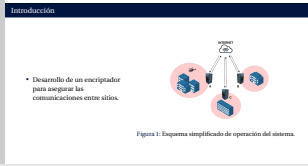


Figura 1: Esquema simplificado de operación del sistema.



Antes que nada quiero dar un pantallazo general de lo que se trata este proyecto. ¿A que nos referimos con sistema de comunicaciones seguras? Se trata del desarrollo de un dispositivo encriptador destinado a asegurar las comunicaciones entre sitios, es decir, que permitan formar una red segura entre estos sitios. En la figura se puede ver un esquema simplificado de la operación del sistema. Estos dispositivos hacen de interfaz entre dos dominios, el rojo, que no está encriptado, y el negro, que sí lo está, que es la parte externa.

Motivación

- Abordaje novedoso a las soluciones de encriptación de redes.
- Necesidad de una solución propia y auditable.

Este proyecto surge como una posibilidad de abordar una solución de encriptación de redes de datos con un enfoque novedoso como es la segmentación virtual de dominios. Además, es importante contar con una solución propia, cuyo desarrollo y mantenimiento no dependa de terceros y pueda ser auditada visto casos como el de Crypto AG, una empresa proveedora de equipos de cifrado con backdoors, que por mucho tiempo fue, en secreto, propiedad de entidades gubernamentales.

Objetivos

- Validar la viabilidad de realizar segmentación virtual de dominios.
- Realizar una prueba de concepto del enfoque propuesto.
- Implementar una propuesta de solución auditable y documentada.

Objetivos
- Validar la viabilidad de realizar segmentación virtual de dominios.
- Realizar una prueba de concepto del enfoque propuesto.
- Implementar una propuesta de solución auditable y documentada.

Respecto a los objetivos de este proyecto se plantean estos tres puntos. El primero, validar si es viable realizar segmentación virtual de dominios, responde al ¿para qué? del proyecto. Los otros objetivos se corresponden con la solución propuesta, el ¿qué? de este proyecto. Lo que se plantea es obtener un encriptador funcional, completamente auditable y documentado.

Quiero arrancar con una revisión de algunos conceptos que nos introducen a los sistemas de comunicaciones seguros y que permiten comprender que se busca como propuesta de solución.

Triada CID

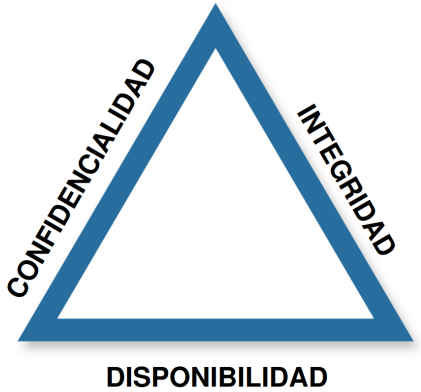


Figura 2: Modelo CID.

Triada CID
CONFIDENCIALIDAD
INTEGRIDAD
DISPONIBILIDAD
Figura 2: Modelo CID.

La triada CID es un modelo que constituye la base para el diseño de sistemas de seguridad. Se compone de estos tres conceptos fundamentales.
La confidencialidad implica garantizar que los datos no sean accesibles por usuarios no autorizados. Los esfuerzos en esta área se centran en los métodos de encriptación.
La integridad asegura la autenticidad de los datos, es decir, que estos estén libres de alteraciones. Aqui se pueden mencionar el uso de hashes o firmas digitales.
La disponibilidad garantiza que los datos sean accesibles cuando se los necesite. Un sistema débil en este aspecto puede ser víctima de ataques de denegación de servicio, por ejemplo. Para esto se refuerza el sistema incluyendo redundancia y algoritmos de detección de ataques.

Encriptación simétrica

- Clave única.
- Requiere un canal seguro para el intercambio de la clave.
- La confidencialidad y autenticación dependen tanto de A como de B.
- Método eficiente.

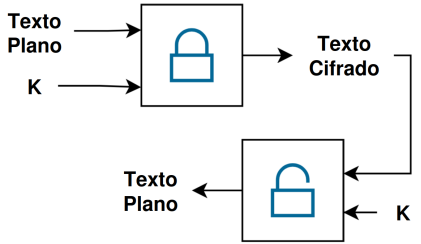
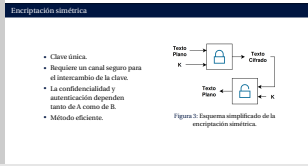


Figura 3: Esquema simplificado de la encriptación simétrica.



La encriptación simétrica es un método que utiliza una clave única para encriptar y desencriptar la información. En la figura esquematicé el proceso de encriptado y desencriptado de texto, vemos que se utiliza esta clave K en ambos procesos, lo que implica que tanto A como B deben conocer esta clave para poder comunicarse. Un problema surge en el acuerdo de esta clave, por que canales y cómo se transmite, además, si esta clave se ve comprometida por cualquiera de las partes, toda la comunicación queda expuesta. Más allá de esto, este es un método muy eficiente, lo que lo hace ideal para encriptar grandes volúmenes de información.

Encriptación asimétrica

- Par de claves relacionadas.
- Mantener la autenticidad y confidencialidad de lo que recibe A depende solo de A.
- Mayor costo computacional.

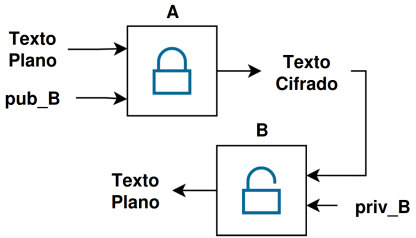


Figura 4: Esquema simplificado de la encriptación asimétrica.

La encriptación asimétrica es un método que utiliza un par de claves relacionadas, una clave pública y una privada. La clave pública se puede compartir con cualquier persona, y es utilizada para encriptar, mientras que la clave privada debe mantenerse en secreto y es utilizada para desencriptar. Aquí en la figura se tiene un mensaje de A hacia B, donde A utiliza la clave pública de B. El proceso inverso, un mensaje de B hacia A sería encriptado por B usando la clave pública de A.

Este método además permite que si yo soy A y mi clave privada se ve expuesta, esto solo afecta a la confidencialidad de lo que recibo, puedo seguir enviando mensajes encriptados a B. Como contra, este método es más costoso computacionalmente que la encriptación simétrica, no suele utilizarse para grandes volúmenes de información.

Acuerdo de claves Diffie-Hellmann

- Método para generar una clave compartida sin intercambio directo.
- Mitiga un problema de la encriptación simétrica.
- No resuelve el problema de autenticación.

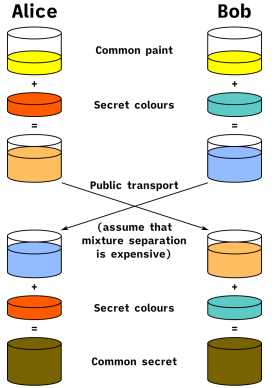
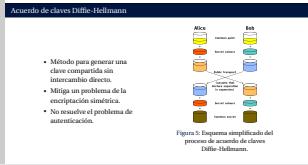


Figura 5: Esquema simplificado del proceso de acuerdo de claves Diffie-Hellmann.



El acuerdo de claves Diffie-Hellmann es un método para generar una clave compartida sobre un canal inseguro sin necesidad de intercambiarla directamente. De esta forma se mitiga el problema de la encriptación simétrica, que es la necesidad de un canal seguro para el intercambio de claves.

Este método se basa en la dificultad matemática del problema del logaritmo discreto en números grandes. Por lo general, los protocolos utilizados utilizan claves de 32 a 56 bytes.

Este esquema simplificado con pinturas representa el proceso de acuerdo de claves Diffie-Hellmann. A y B generan claves privadas y públicas, y luego intercambian las claves públicas. A partir de las claves públicas de A y B y sus claves privadas propias, cada uno puede obtener la misma clave como resultado.

Arquitectura red/black

- Lineamientos para identificar y separar correctamente dominios de información.

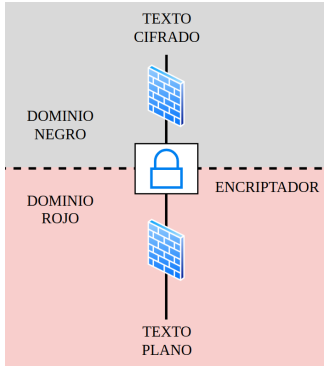
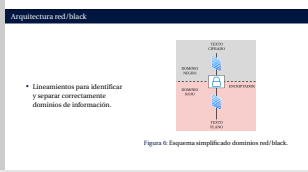


Figura 6: Esquema simplificado dominios red/black.



El concepto de arquitectura red/black representa la separación de los dominios que contienen información clasificada, o texto plano de aquellos que contienen información encriptada, o texto cifrado. Adoptar esta metodología permite identificar las interfaces entre dominios y diseñar medidas de seguridad adecuadas para aislarlas correctamente.

Esta idea de segregación de interfaces surge de las especificaciones militares TEMPEST, que provee lineamientos en el diseño físico de los equipos para evitar fugas de información a través de emisiones electromagnéticas. Aquí se encuentra el mayor desafío de este proyecto, que es lograr una implementación segura de esta arquitectura en un entorno virtualizado.

Hipervisores

Software que permite la ejecución de entidades virtuales independientes sobre hardware compartido.

- Tipo 1: ejecución sobre hardware.
- Tipo 2: ejecución sobre un sistema operativo.

Figura 7: Ejemplo de hipervisores.

Un hipervisor es un software que permite la ejecución y administración de máquinas virtuales sobre un hardware compartido. Gestiona los recursos de hardware y asegura el aislamiento entre las máquinas virtuales fuera de sus interfaces de comunicación.
Un hipervisor tipo 1 se ejecuta directamente sobre el hardware, sin necesidad de un sistema operativo intermediario. Ofrece mayor rendimiento y seguridad a costa de mayor complejidad de implementación. Un ejemplo de este tipo de hipervisor es seL4, el cual describiré más adelante.
Un hipervisor tipo 2 se ejecuta sobre un sistema operativo, es más versátil respecto a su implementación. Un ejemplo de este tipo de hipervisor es VirtualBox.

1 Introducción
2 Revisión bibliográfica
3 Desarrollo
Propuesta de solución
Laboratorios virtuales
4 Conclusiones

Plan de trabajo

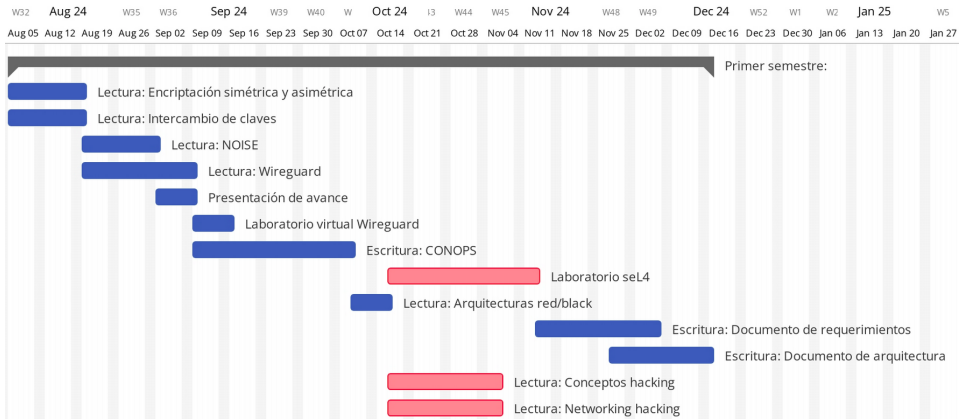


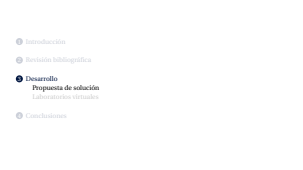
Figura 8: Plan de trabajo del primer semestre de proyecto.

Figura 8: Plan de trabajo del primer semestre de proyecto.

Quiero presentar antes que nada lo que fue el primer paso en este proyecto, que fue realizar el plan de trabajo del semestre. Esto me permitió mantener una metodología de trabajo ordenada en lo que es el proyecto.

Algunas tareas involucraron la lectura como introducción a los conceptos necesarios para comprender el fin del proyecto y otras que permitieron aprender de forma práctica sobre las tecnologías a utilizar. También hay cierta documentación pensada como entregables que consideramos importante para enmarcar la propuesta de solución.

En azul están marcadas las tareas que ya fueron completadas, en rojo las que quedaron pendientes para el segundo semestre.



Método ARCADIA

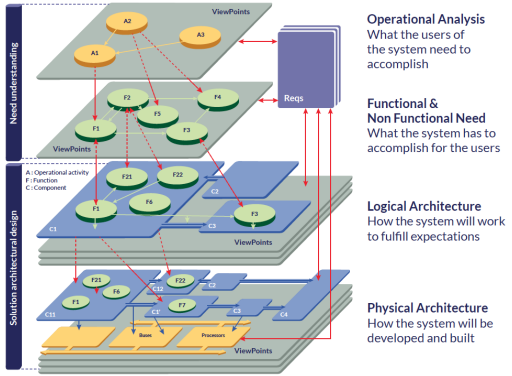


Figura 9: Desglose del método adoptado.

Figura 9: Desglose del método adoptado.

Quiero presentar brevemente la forma de trabajo que adoptamos para la elaboración de la propuesta de solución. El método ARCADIA se basa en la identificación de los requerimientos del usuario y la definición de los objetivos del sistema.
Se trabajó primero en entender las necesidades del usuario, esto es la capa superior de la figura, y comprende el análisis operacional. Luego definimos los requerimientos del sistema como las funciones que debe realizar el sistema para cumplir estas necesidades. Ya con esto documentado se puede abordar el diseño de la solución.

Si bien ya recorrimos todas etapas a lo largo del semestre, el diseño de la solución no está completo, y es probable que se identifiquen más necesidades, requerimientos a medida que se avance en el proyecto. Esto es más bien un proceso iterativo entre estas etapas, por lo que consideramos importante documentar cada necesidad, requerimiento y decisión de diseño.

Análisis operacional

- Definición del problema.
- Planteo de las necesidades del usuario.
- Alcance de la solución.

Figura 10: Concepto de operaciones.

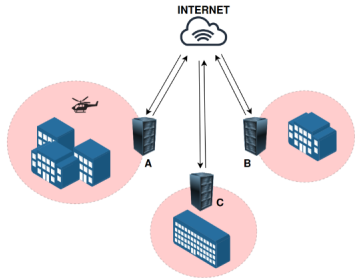
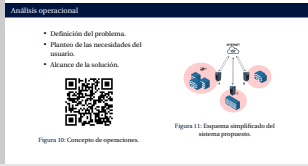


Figura 11: Esquema simplificado del sistema propuesto.



Partiendo de la primer capa del modelo, planteamos el concepto de operaciones del sistema, que se encuentra documentado y puede accederse a una copia desde este QR.
Esta etapa involucró definir el problema y el contexto en el que se desarrolla la solución. Se manifiestan las suposiciones y limitaciones del sistema propuesto, como puede ser el análisis de encriptación en capa física respecto a hacerlo en capa de red, por ejemplo.
Se documentó también el alcance de la solución, que es la definición de los límites del sistema y las funciones que debe cumplir, sin profundizar en cómo se va a lograr esto.

Este es un esquema simplificado del sistema propuesto. Cada nodo de la red segura se comunica con los demás a través de una conexión a Internet.

Modos de operación

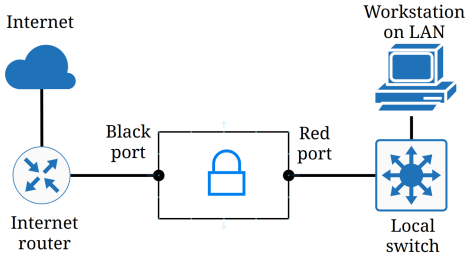


Figura 12: Nodo pequeño.

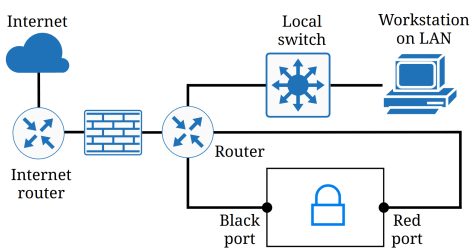
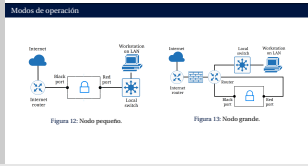


Figura 13: Nodo grande.



Teniendo en cuenta una configuración de hardware genérica donde se implementará la solución se plantea la posibilidad de dos modos de operación. Se considera nodo pequeño a un sitio con poco tráfico, en el que el hardware del encriptador es capaz de direccionar todo el tráfico de la red según su destino sea Internet u otro nodo de la red segura. Si se trata de un nodo grande, puede que el hardware del encriptador no sea capaz de direccionar todo el tráfico. En este caso es necesario un router que cumpla esta función, de manera que solo el tráfico destinado a otro nodo red segura pasa por el encriptador.

Requerimientos

- **Funcionales**: renovación de claves, manejo de ataques DoS.
- **Rendimiento**: tasa de transferencia, número de nodos.
- **Interfaz**: administración, interfaces físicas.

Figura 14: Documento de requerimientos.

En un nivel menor de abstracción se definieron los requerimientos del sistema, estos son las cosas que el sistema debe hacer para cumplir con las necesidades planteadas en el concepto de operaciones. Como ejemplos están el tiempo asociado a la renovación de claves, la tasa máxima de transferencia entre nodos y las interfaces de administración del encriptador. Estos requerimientos se documentaron y pueden accederse a una copia desde este QR.

Arquitectura lógica

- Uso de un hipervisor con tres máquinas virtuales independientes.

Figura 15: Documento de arquitectura.

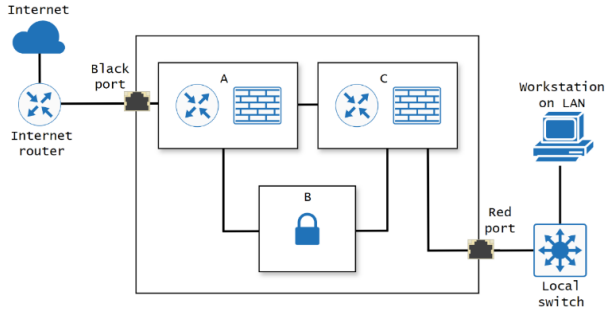
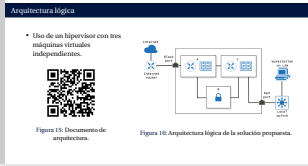


Figura 16: Arquitectura lógica de la solución propuesta.



Como última etapa queda proponer una arquitectura lógica para hacer cumplir los requerimientos del sistema, aquí ya estamos pensando sobre la solución.
Se planteó el uso de un hipervisor con tres máquinas virtuales independientes y aislados salvo por ciertas interfaces de comunicación definidas.
En esta etapa también se defieron las tecnologías a usar para implementar esta arquitectura.

Tecnologías a usar

NOISE

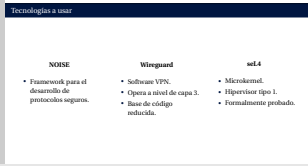
- Framework para el desarrollo de protocolos seguros.

Wireguard

- Software VPN.
- Opera a nivel de capa 3.
- Base de código reducida.

seL4

- Microkernel.
- Hipervisor tipo 1.
- Formalmente probado.



Para implementar la propuesta de solución se plantea el uso de tres tecnologías open-source.

- En primer lugar está NOISE, un framework para el desarrollo de protocolos seguros basado en el método Diffie-Hellmann que cuenta con una serie de protocolos formalmente probados, aplicaciones como WhatsApp son un ejemplo de uso de este framework en el cifrado de mensajería.
- Wireguard es un software que permite la creación de VPNs de forma sencilla y segura. Utiliza protocolos derivados del framework NOISE para el establecimiento de conexiones. Opera a nivel de capa 3 del modelo OSI. Un aspecto importante es que su código fuente es reducido, aproximadamente 4000 líneas de código, lo cual ayuda a reducir la superficie de ataque.
- seL4 es un microkernel que se ejecuta directamente sobre hardware con capacidad de funcionar como hipervisor. En este modo, la aislación de máquinas virtuales está formalmente probado, lo cuál es crítico en nuestro sistema. Está diseñado para ser seguro en sistemas críticos sin comprometer el rendimiento.

Laboratorio virtual - Wireguard

- Fundamentos de redes.
- Utilización de Wireguard.

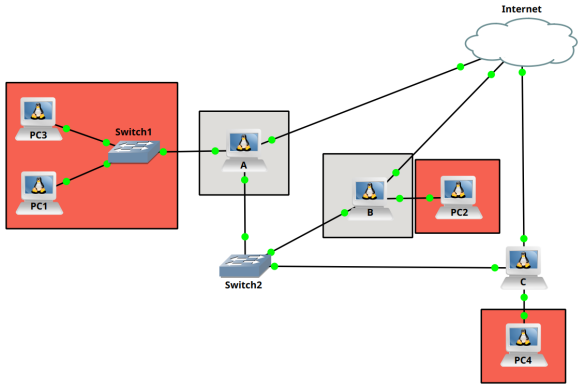
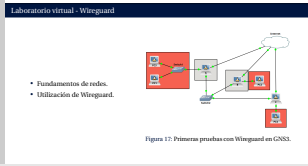


Figura 17: Primeras pruebas con Wireguard en GNS3.



Sobre GNS3, un hipervisor de tipo 2 se implementó una red como la de la imagen utilizando máquinas virtuales Linux. Se configuraron dos nodos, A y B, a modo de encriptadores usando Wireguard. Analizando el tráfico de la red se verificó que la comunicación entre los nodos esté encriptada y que el tráfico con destino a Internet se direccione por fuera del túnel VPN.

Laboratorio virtual - Wireguard

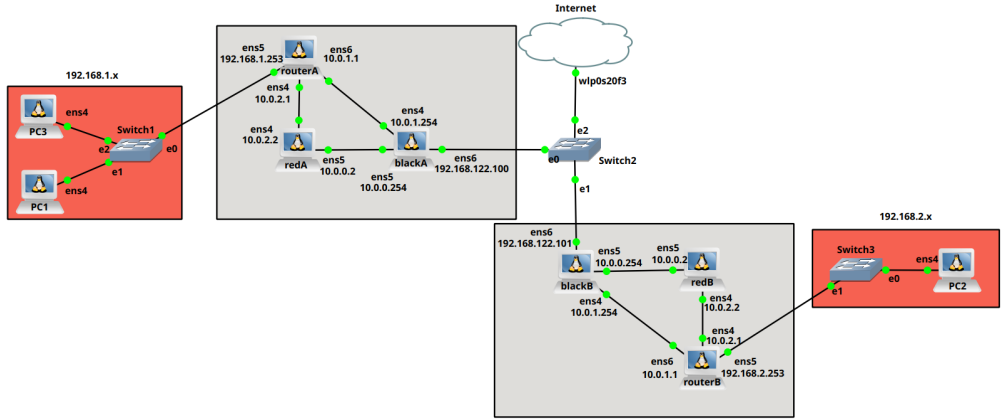
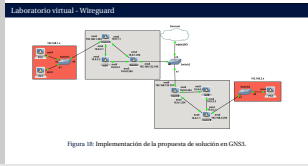


Figura 18: Implementación de la propuesta de solución en GNS3.

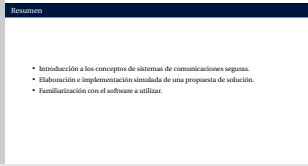


También se verificó esto sobre una red donde implementamos la arquitectura propuesta para los encriptadores utilizando tres máquinas virtuales independientes. Dos actuando como routers y una como encriptador. En este caso los paquetes con destino otro nodo de la red segura llegan a router", se encriptan en redz salen a Internet a través de la interfaz física de "black". El tráfico que no debe ser encriptado se dirige desde router"directamente a "black".

1 Introducción
2 Revisión bibliográfica
3 Desarrollo
4 Conclusiones

Resumen

- Introducción a los conceptos de sistemas de comunicaciones seguras.
- Elaboración e implementación simulada de una propuesta de solución.
- Familiarización con el software a utilizar.



En resumen, se estudiaron conceptos de sistemas de comunicaciones seguras y redes de datos. Se elaboró una arquitectura para la propuesta de solución y se la implementó de forma simulada en GNS3. Y por último, la realización del laboratorio virtual y los tutoriales permitieron familiarizarse con Wireguard y seL4, las tecnologías a utilizar en la propuesta de solución.

Trabajo a futuro

- Abordar la implementación simulada de la propuesta de solución en seL4 y posteriormente en hardware.
- Adquirir conceptos de hacking para evaluar la seguridad de la solución.
- Continuar con la documentación del proyecto.

Como trabajo a futuro se plantea continuar con la implementación virtual de la arquitectura propuesta en seL4 y posteriormente en hardware corriendo seL4. También falta adquirir conceptos que nos permitan evaluar la seguridad de la solución. Continuar con este proceso iterativo que es la documentación del proyecto.

¡Muchas gracias!
¿Preguntas?

¡Muchas gracias!
¿Preguntas?