PROYECTO INTEGRADOR DE LA CARRERA DE INGENIERÍA EN TELECOMUNICACIONES

MI SÚPER INTERESANTE PI

Juan Teleco
Estudiante

Ing. Pepe Antenil
Director

Miembros del Jurado
Señor Malo 1 (INVAP)
Señora Mala 2 (Instituto Balseiro)

14 de Junio de 2025

Telecolandia

Instituto Balseiro
Universidad Nacional de Cuyo
Comisión Nacional de Energía Atómica
Argentina

A todos los telequitos

Índice de símbolos

Índice de contenidos

Índice de símbolos v

Índice de contenidos vii

Índice de figuras ix

Índice de tablas xi

Resumen xiii

Abstract xv

1. Introducción 1
- 1.1. Motivación y objetivos . . . 1
- 1.2. Concepto de operaciones . . . 2
  - 1.2.1. Contexto . . . 2
  - 1.2.2. Suposiciones y restricciones . . . 3
  - 1.2.3. Resumen del sistema propuesto . . . 4
  - 1.2.4. Objetivos, metas y justificación del sistema . . . 5
  - 1.2.5. Usuarios y modos de operación . . . 5
- 1.3. Requerimientos . . . 6
  - 1.3.1. Funcionales . . . 6
  - 1.3.2. De rendimiento . . . 7
  - 1.3.3. De interfaz . . . 7
- 1.4. Descripción de tecnologías . . . 8
  - 1.4.1. WireGuard . . . 8
  - 1.4.2. seL4 . . . 8
  - 1.4.3. CAmkES . . . 8

2. Arquitectura propuesta 9
- 2.1. Arquitectura lógica . . . 9
  - 2.1.1. Dominios . . . 9
- 2.2. Arquitectura física . . . 9
  - 2.2.1. Hardware . . . 9

3. Estrategia de modelos en entornos virtualizados 11
- 3.1. Sistema mínimo . . . 11
- 3.2. Sistema completo usando máquinas virtuales . . . 12
- 3.3. Sistema completo sobre seL4 . . . 12

3.4. Resumen 12

4. Implementación en entorno virtualizado **13**
4.1. Diseño del experimento 13
4.2. Procedimiento 13
4.2.1. Construcción de un kernel Linux con soporte para WireGuard 13
4.2.2. Generación de una imagen de sistema mediante Buildroot 13
4.2.3. Adaptación del ejemplo *zmq_samples* 13
4.2.4. Configuración del *passthrough* de interfaz Ethernet 14
4.3. Integración 14
4.4. Validación 14

5. Implementación en hardware **15**
5.1. SuperMicro SYS-E300-9D 15
5.2. Diseño del experimento 15
5.3. Procedimiento 15

A. Apéndice I **17**

Bibliografía **19**

Publicaciones asociadas **21**

Agradecimientos **23**

Índice de figuras

1.1. Comparación entre encriptación simétrica y asimétrica. 3
1.2. Método de Diffie-Hellmann para el acuerdo de una clave común. 3
1.3. Esquema simplificado del sistema de comunicaciones seguras. 4
1.4. Modos de operación del encriptador. 6
1.5. Esquema de comunicación zmq_samples. 8

3.1. Otro esquema mejorcito. 11
3.2. Otro esquema mejorcito. 12

Índice de tablas

1.1. Palabras clave utilizadas en la definición de requerimientos. 6

Resumen

Este es el resumen en castellano.
La tesis debe reflejar el trabajo desarrollado, mostrando la metodología utilizada, los resultados obtenidos y las conclusiones que pueden inferirse de dichos resultados.

Palabras clave: FORMATO DE TESIS, LINEAMIENTOS DE ESCRITURA, INSTITUTO BALSEIRO

Abstract

This is the title in English:
The thesis must reflect the work of the student, including the chosen methodology, the results and the conclusions that those results allow us to draw.

Keywords: THESIS FORMAT, TEMPLATES, INSTITUTO BALSEIRO

Capítulo 1

Introducción

1.1. Motivación y objetivos

En el ámbito de las comunicaciones, es fundamental contar con un sistema capaz de garantizar confiabilidad respecto a la autenticación, confidencialidad e integridad de la información transmitida. Cuando la comunicación se desarrolla sobre un medio considerado inseguro, asegurar el cumplimiento de estas propiedades implica abordar desafíos adicionales como mitigar la posibilidad de suplantación de identidad, encriptar la información para asegurar confidencialidad y detectar modificaciones del mensaje por agentes externos.

Se han desarrollado soluciones privativas al problema descrito. Sin embargo, la inexistencia de *back-doors* y mecanismos de vigilancia es algo que no puede ser verificado por completo en dichas soluciones. Es por esto que en áreas como defensa y servicio diplomático es de interés contar con soluciones completamente auditables, que no dependan de licencias de exportación. A su vez, que la funcionalidad y validación formal del equipo no dependa de las funciones criptográficas utilizadas, y que estas puedan ser provistas o implementadas por el usuario final.

Borrador:

Este trabajo propone un abordaje novedoso a las soluciones de encriptación de redes, respondiendo a la necesidad de una solución propia y auditable...

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.

Método ARCADIA

1.2. Concepto de operaciones

El CONOPS es un documento utilizado para describir cómo se espera que opere un sistema desde el punto de vista del usuario final, sin brindar aún detalles técnicos de la solución. Esta sección es de utilidad para clarificar el propósito y uso del sistema, y de base para la definición de requerimientos.

1.2.1. Contexto

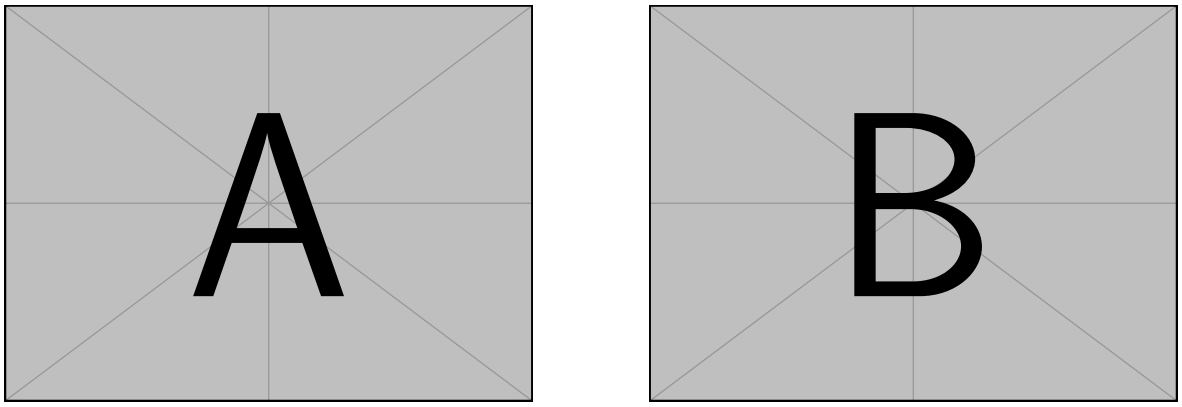
La tríada de la CIA *(Confidentiality, Integrity, Availability)* es un modelo que constituye la base para el desarrollo de sistemas de seguridad. Es utilizada para identificar vulnerabilidades de un sistema y proponer soluciones que cumplan con estos principios.

- **Confidencialidad**: consiste en proteger la información sensible de accesos no autorizados. Los métodos para reforzar este aspecto pueden involucrar la encriptación de la información y la implementación de controles de acceso.
- **Integridad**: refiere a asegurar la consistencia y confiabilidad de la información transmitida y mitigar el riesgo de que los mensajes sufran alteraciones por parte de agentes no autorizados. La implementación de firmas digitales es uno de los métodos empleados para reforzar este principio.
- **Disponibilidad**: cumplir con este aspecto requiere asegurar que la información sea accesible para usuarios autorizados cada vez que sea requerida. Un sistema robusto en este aspecto tiene que ser capaz de soportar ataques de denegación de servicio. Los métodos que refuerzan este aspecto pueden involucrar introducir redundancia a los componentes del sistema.

Un canal inseguro es un medio de transmisión en el que la información se encuentra expuesta a ataques. La escucha pasiva, la suplantación de identidad y la denegación de servicio son algunos ejemplos de ataques típicos de un canal inseguro. Los primeros dos corresponden al área de confidencialidad en el modelo CIA, y pueden tratarse implementando métodos de encriptación. La encriptación de las comunicaciones garantiza la confidencialidad de las mismas y es la base de un sistema de comunicaciones seguras. Existen dos enfoques principales para realizar encriptación, el enfoque simétrico y el asimétrico:

- **Encriptación simétrica**: consiste en un método en el cual las partes utilizan una misma clave para la encriptación y desencriptación de la información. Esto trae como problema que cualquier entidad con acceso a dicha clave tiene la capacidad de leer y reescribir la información. Aún así, se trata de un método eficiente y muy utilizado para transmitir grandes volúmenes de datos. Cualquier implementación con este enfoque requiere como complemento de una forma segura de intercambiar la clave utilizada en un contexto de canales inseguros debido a que mantener la confidencialidad de la comunicación depende de que ambas partes resguarden la clave simétrica.
- **Encriptación asimétrica**: bajo este enfoque cada entidad posee un par de claves únicas, denominadas clave pública y clave privada, que guardan relación entre sí. La clave pública es utilizada para encriptar información y la clave privada correspondiente al mismo par es utilizada para desencriptarla. Este método no suele ser utilizado para la transmisión de grandes volúmenes de información debido a que, al ser de mayor complejidad computacional, se vuelve poco eficiente en estos casos. A diferencia del enfoque simétrico, aquí la confidencialidad de lo que transmite una de las partes depende de que la misma mantenga asegurada su clave privada.

El método de Diffie-Hellmann propone combinar ambos enfoques para lograr acordar, empleando el concepto de encriptación asimétrica, una clave simétrica de manera segura que puede ser usada



(a) Encriptación simétrica **(b)** Encriptación asimétrica

Figura 1.1: Comparación entre encriptación simétrica y asimétrica.

posteriormente para encriptar información. La principal ventaja del método radica en que ambas partes logran generar la misma clave simétrica sin transmitirla por el canal, mitigando un gran problema de seguridad del enfoque simétrico. La figura 1.2 describe las operaciones matemáticas sobre las cuales funciona el método de Diffie-Hellmann.

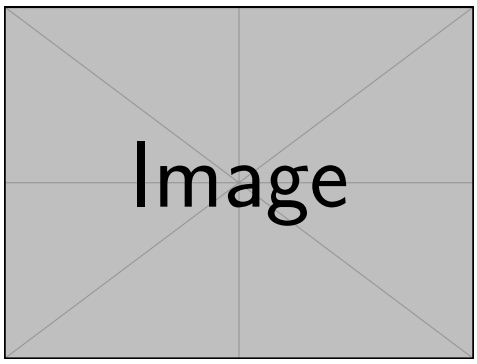


Figura 1.2: Método de Diffie-Hellmann para el acuerdo de una clave común.

Para los estándares de seguridad actuales, un esquema de establecimiento de comunicaciones requiere mayor complejidad para abordar desafíos como un ataque de tipo man-in-the-middle. Aún así, el método de Diffie-Hellmann es la base de numerosas implementaciones por su simplicidad y el elevado costo computacional que supone calcular la clave K conociendo únicamente las claves públicas A y B.

1.2.2. Suposiciones y restricciones

Distintas soluciones de encriptación pueden proponerse según la capa del modelo OSI que se analice. La encriptación a nivel de capa física minimiza las penalidades en rendimiento a costa de introducir complejidad al sistema, que se manifiesta en la necesidad de hardware dedicado de extremo a extremo de la red. Esta restricción puede volver inviable la encriptación a nivel de la capa física cuando el sistema de comunicaciones requiere de escalabilidad, principalmente por el costo de despliegue y mantenimiento. La encriptación en capas superiores provee mayor flexibilidad en la implementación del sistema. Si bien esto introduce una mayor latencia a la red, reduciendo el rendimiento de la misma, se pueden lograr valores aceptables de latencia con suficiente optimización. Las soluciones de encriptación en la capa 3 tienen la ventaja de ser independientes de la capa física, reduciendo la complejidad del sistema y permitiendo mayor flexibilidad en la implementación y compatibilidad con

infraestructura preexistente. Esto implica que, cualquier sitio que cuente con una conexión a Internet y un dispositivo de encriptación tiene la infraestructura suficiente para acceder a una red segura, denominada red privada virtual.

En sistemas de criptografía, usualmente se utiliza el concepto de dominios rojo/negro para describir las partes del sistema que trabajan con información legible (dominio rojo) y aquellas que contienen información cifrada (dominio negro). La arquitectura de un sistema de comunicación seguro debe tener en cuenta este concepto para una correcta segregación de dominios, mitigando así la posibilidad de filtraciones indeseadas de información. La normativa actual que refiere a la segregación de dominios no contempla la segregación virtual, es decir, contener en un mismo dispositivo ambos dominios y aislarlos empleando herramientas de software. El implementar un sistema de este estilo capaz de cumplir las normas de seguridad existentes es un desafío importante.

1.2.3. Resumen del sistema propuesto

El sistema de comunicaciones seguras que se propone contempla el diseño de un único dispositivo, denominado encriptador, con la capacidad de formar un túnel VPN entre redes preexistentes, asegurando ciertos estándares de autenticación y confidencialidad sobre las comunicaciones entre dichas redes. El sistema opera a nivel de capa 3 del modelo OSI, y requiere de un mínimo de dos encriptadores para su funcionamiento, aunque también se encuentra prevista la escalabilidad del sistema. El propósito del sistema es brindar confiabilidad respecto a la autenticación, confidencialidad e integridad de la información transmitida entre redes interconectadas, afectando al mínimo el rendimiento de la red.

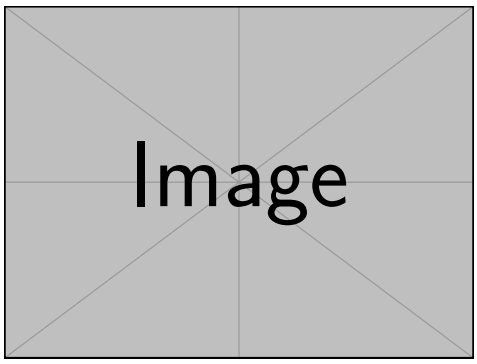


Figura 1.3: Esquema simplificado del sistema de comunicaciones seguras.

En el esquema de la figura 1.3 se describen los denominados dominios rojos, correspondientes a redes donde se trata con información sensible. El complemento de estos se denominan dominios negros, que usualmente se consideran canales inseguros donde la información proveniente de un dominio rojo requiere estar encriptada. El dispositivo encriptador actúa de interfaz entre dominios rojo y negro, motivo por el cual gran parte de la seguridad del sistema recae sobre este. Este dispositivo cuenta con acceso a claves utilizadas para establecer sesiones seguras con otros dispositivos, y al mismo tiempo cuenta con acceso a Internet. Debe poder garantizarse la seguridad en la gestión de estas claves y rechazar accesos no autorizados al dominio rojo.

1.2.4. Objetivos, metas y justificación del sistema

Un dispositivo en el cual los dominios rojo y negro no estén correctamente segmentados se encuentra con que la información confidencial respecto al túnel VPN como claves y permisos se hallan expuestos en el caso de una intrusión desde Internet o desde la propia red interna. Esto lleva a la posibilidad de que un agente no autorizado sea capaz de desencriptar y modificar información que viaje por el túnel, así como también acceder a dispositivos dentro de una organización.

En el diseño del encriptador, se propone implementar la segmentación virtual de dominios rojo/negro, esto es, representar los dominios como entidades virtuales independientes que ejecutan sobre hardware compartido. Este concepto puede implementarse a través de los denominados hipervisores, software que permite que varios sistemas operativos independientes trabajen juntos, compartiendo los mismos recursos físicos.

La segmentación permite que la entidad negra, la cual está conectada a Internet, realice el control del tráfico y oculte la existencia de la entidad roja para cualquier servicio fuera de la red virtual privada. Por otro lado, la entidad roja es la responsable de encriptar/desencriptar las comunicaciones entre organizaciones que se encuentren dentro de la red privada y de gestionar las claves de encriptación y los privilegios de usuarios. Esta segmentación permite un control estricto sobre el tráfico entre ambas entidades, mitigando la posibilidad de transferir información como claves contenidas en la entidad roja hacia la entidad negra.

1.2.5. Usuarios y modos de operación

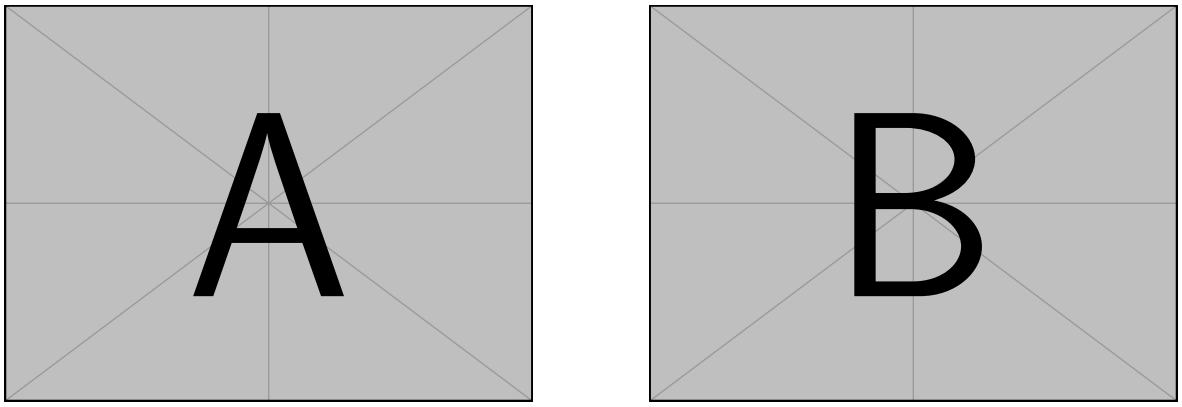
Usuarios

- **Administrador de red**: responsable de la configuración y mantenimiento del encriptador. Cuenta con acceso a la interfaz de administración del dispositivo y es el encargado de configurar el encriptador para funcionar dentro de la red segura. La configuración del encriptador incluye la definición de las redes que se conectarán a través del túnel VPN, la configuración de las claves de encriptación y la definición de los permisos de los usuarios.
- **Usuario final**: utiliza la red segura para compartir información sensible a otros nodos de la red. No tiene acceso a la configuración del encriptador y su interacción con el sistema se limita a utilizar la red segura como si de su red local se tratase.

Modos de operación

El diseño prevee que, una vez configurado el dispositivo, la operación del encriptador sea transparente para el usuario final, el cual no necesita interactuar con el dispositivo para utilizar la red segura.

- **Nodo pequeño:** este modo de operación es de utilidad en sitios que no cuentan con infraestructura de red. El encriptador opera también como router, procesando todo el tráfico del sitio y permitiendo que el tráfico no destinado a otro sitio de la red segura sea enrutado por fuera del túnel VPN. Esta funcionalidad es conocida como *split tunneling*. Es adecuado para instalaciones con tráfico moderado.
- **Nodo grande:** en instalaciones con tráfico elevado y que posiblemente cuenten con una infraestructura de red acorde, el equipo únicamente opera como encriptador de las comunicaciones seguras, y se conecta a un router preexistente para las comunicaciones no seguras. De esta manera solamente el tráfico dentro de la red segura es procesado por el encriptador.



(a) Sitio sin infraestructura de red. **(b)** Sitio con infraestructura de red.

Figura 1.4: Modos de operación del encriptador.

1.3. Requerimientos

El documento de requerimientos es utilizado para describir que necesita el sistema para cumplir las necesidades de los usuarios. Esta sección establece las bases para la solución propuesta. Se definen a continuación las palabras clave que se utilizarán para referirse a los requerimientos:

Palabra clave	Descripción
DEBE	Indica un requerimiento obligatorio que debe cumplirse para que el sistema funcione correctamente.
NO DEBE	Indica un requerimiento que no debe cumplirse, es decir, una restricción que debe evitarse.
DEBERÍA	Indica un requerimiento recomendado, pero no obligatorio. Se sugiere cumplirlo para mejorar la calidad del sistema.
NO DEBERÍA	Indica un requerimiento que se desaconseja cumplir, pero no es obligatorio evitarlo.
PUEDE	Indica una opción o característica que el sistema puede implementar, pero no es obligatoria.

Tabla 1.1: Palabras clave utilizadas en la definición de requerimientos.

1.3.1. Funcionales

- **Renovación de claves:** cada par de nodos realiza una renovación de claves efímeras cada 120 segundos para asegurar forward-secrecy. Esto significa que, si una clave es comprometida por alguna razón, no se comprometen las comunicaciones anteriores o futuras fuera del intervalo de tiempo especificado.
- **Seguridad ante intrusiones:** el equipo debe ser capaz de mitigar la posibilidad de intrusiones de agentes no autorizados vía software, tanto desde Internet o como desde la red local.
- **Detección de ataques DoS:** el sistema debe ser capaz de detectar y mitigar ataques de denegación de servicio.
- **Movilidad:** el sistema debe ser capaz de soportar la movilidad de los nodos y permitir a un nodo moverse entre redes sin interrupciones ni renegociaciones de claves.
- **Split-tunneling:** el sistema debe ser capaz de permitir y enrutar tráfico de ciertas aplicaciones o servicios por fuera del túnel VPN.

- **Segmentación de dominios:** el sistema debe aislar procesos contenidos en un dominio de información, como pueden ser archivos, contenida en otro dominio, independientemente de los privilegios que tenga este proceso.

1.3.2. De rendimiento

- **Tasa de transferencia:** el sistema debe ser capaz de lograr, de un nodo a otro, una tasa de transferencia de 950 Mbits/s de datos planos.
- **Número de nodos:** una red segura debe ser capaz de soportar hasta 250 dispositivos encriptadores, también denominados nodos.

1.3.3. De interfaz

- **Administración:** la configuración de funcionamiento del encriptador debe poder ser modificada únicamente por un administrador de red autorizado de manera local.
- **Interfaz de usuario:** el sistema debe ser transparente para el usuario final, es decir, no debe requerir de configuraciones adicionales para este.
- **Configuración de operación:** el modo de operación en red del sistema y otros parámetros de funcionamiento asociados deben ser configurables únicamente por el administrador de red.

1.4. Descripción de tecnologías

En este trabajo se combinarán distintas tecnologías que conformarán la solución final. A continuación se describen brevemente a modo de contextualizar al lector.

1.4.1. WireGuard

WireGuard es un protocolo de túneles VPN derivado de Noise Protocol Framework. Está diseñado para ser simple, rápido y eficiente, proporcionando una solución de código abierto con una base de código pequeña, lo que ayuda a minimizar la superficie de ataque. Su propósito es proporcionar una alternativa segura y eficiente a los protocolos VPN tradicionales, permitiendo la creación de túneles seguros entre dispositivos en una red.

1.4.2. seL4

Se trata de un microkernel con verificación formal matemática que garantiza que su implementación está libre de errores. Cuenta con una base de código muy reducida y proporciona un fuerte aislamiento entre procesos y una interfaz de comunicación segura. Es utilizado en sistemas críticos donde la seguridad y la confiabilidad son fundamentales.

1.4.3. CAmkES

Trabajar con un microkernel como seL4 implica que la mayoría de las funcionalidades del sistema deben implementarse en el espacio de usuario. Para simplificar este proceso, seL4 incluye CAmkES, un *framework* que facilita el desarrollo de sistemas de software modulares y seguros, diseñados específicamente para ejecutarse sobre seL4.

---

zmq_samples

Uno de los ejemplos de uso de VMs en CAmkES es el proyecto zmq_samples, que implementa un sistema de comunicación entre VMs utilizando la librería de mensajería ZeroMQ. Cada VM contiene una interfaz de red virtual eth0 que se conecta a las interfaces eth0 de las demás. En la figura 1.5 se esquematiza el funcionamiento de este sistema.

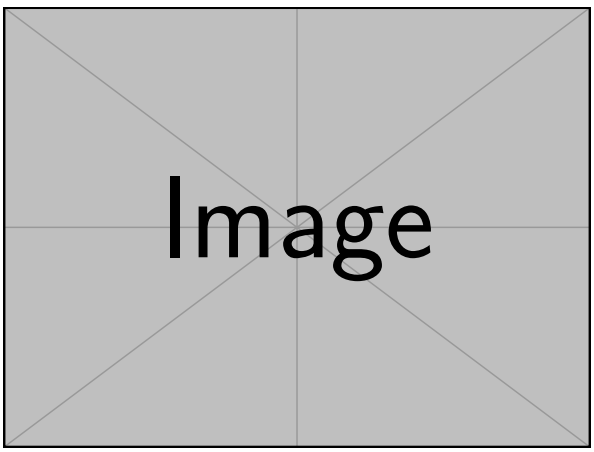


Figura 1.5: Esquema de comunicación zmq_samples.

Capítulo 2

Arquitectura propuesta

2.1. Arquitectura lógica

Borrador:

- Definir los criterios de diseño.
- Presentar la arquitectura lógica del sistema.
- Describir los componentes principales y sus interacciones.
- Incluir diagramas de componentes y de despliegue.
- Argumentar las decisiones de diseño tomadas.

2.1.1. Dominios

2.2. Arquitectura física

Borrador:

- Describir las conexiones físicas entre los componentes del sistema.
- Passthrough.

2.2.1. Hardware

Capítulo 3

Estrategia de modelos en entornos virtualizados

Con el fin de abordar de manera ordenada la complejidad del sistema y validar progresivamente cada uno de sus componentes, se adoptó una estrategia basada en la construcción de modelos incrementales. Estos modelos permiten simular, probar y verificar distintas funcionalidades antes de integrarlas en la solución final.

Los modelos descritos a continuación tienen como finalidad:

- Ligar problemas concretos a cada modelo y resolverlos de forma independiente.
- Obtener una solución funcional en un entorno virtualizado como último paso previo a probarla sobre *hardware* real.

3.1. Sistema mínimo

Se propuso un primer modelo de baja complejidad complementario a la lectura de la documentación de WireGuard. La finalidad de este es familiarizarse con la configuración de un túnel VPN y su funcionamiento a través del análisis de paquetes de red. En el esquema de la figura 3.1 se representa la topología de este ensayo, implementada sobre GNS3.

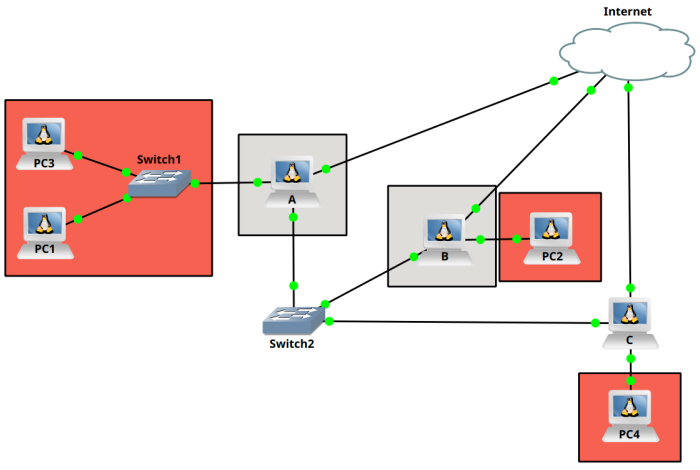


Figura 3.1: Otro esquema mejorcito.

En una siguiente iteración de este modelo se descartó la utilización de GNS3 y se optó por instanciar

cada VM en QEMU, y de esta manera adquirir mayor control sobre los dispositivos de red emulados. Un ejemplo de esto es la posibilidad de observar los parámetros PCI de las interfaces de red, lo cual es relevante para la implementación del *passthrough* de las mismas en la solución final.

Sobre este modelo además puede validarse el correcto funcionamiento de un kernel Linux modificado que luego se utilizará en cada VMM de seL4.

3.2. Sistema completo usando máquinas virtuales

Una vez planteada la arquitectura lógica del sistema, se procede a su simulación en GNS3. Este modelo permite limitar la complejidad de implementar la arquitectura a configurar la interconexión de las VMs que conforman el encriptador.

Este modelo tiene como objetivo validar la arquitectura lógica propuesta y verificar las funcionalidades de red pretendidas para el encriptador como puede ser el *split-tunneling*.

En la figura 3.2 se muestra la topología de red utilizada en GNS3 para simular el sistema completo. El encriptador se implementa como la interconexión mediante interfaces de red de tres VMs independientes.

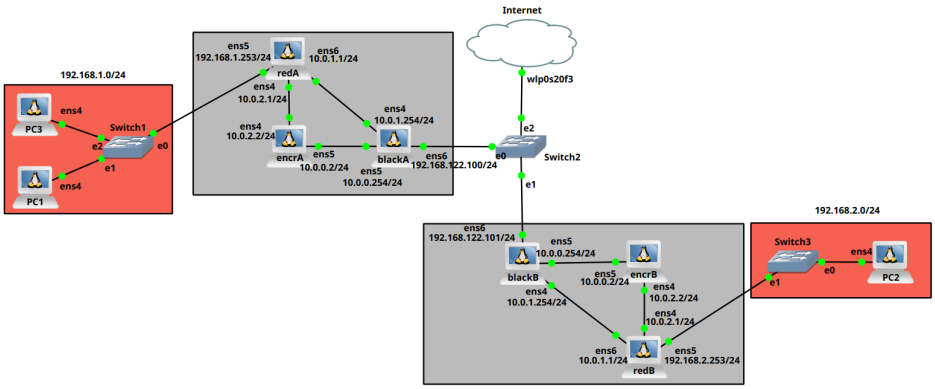


Figura 3.2: Otro esquema mejorcito.

Como *output* de este modelo se obtienen las configuraciones de red necesarias (tablas de enrutamiento, reglas de firewall, etc.) para darle funcionalidad al encriptador. Estas configuraciones se utilizarán posteriormente en la solución.

3.3. Sistema completo sobre seL4

- 4 instancias de QEMU: 2 PCs y 2 encriptadores.
- Descripción de zmq_samples. Virtual Switch. ZeroMQ. Esquemas.

3.4. Resumen

Tabla: modelo, nivel de complejidad, validaciones

Capítulo 4

Implementación en entorno virtualizado

4.1. Diseño del experimento

- Se realizará en completamente en host. ¿Por qué? Simplicidad de implementación.

4.2. Procedimiento

4.2.1. Construcción de un kernel Linux con soporte para WireGuard

- De www.wireguard.com/compilation/kernel-requirements se obtuvieron los requerimientos.
- wireguard-linux-compat patch
- seL4 soporta kernel 4.9[ref], se compiló este kernel con una adaptación de .config file original de los ejemplos camkes-vm
- Buscar referencias para justificar todo. [1]
- Drivers de red compatibles con el hardware utilizado.

4.2.2. Generación de una imagen de sistema mediante Buildroot

- Se utilizó la versión 2023.02.1 de Buildroot utilizando el kernel modificado.
- Se configuró el sistema de archivos para que contenga los binarios necesarios para el funcionamiento de WireGuard y las herramientas de red.

4.2.3. Adaptación del ejemplo *zmq_samples*

- Se adaptó el ejemplo de CAmkES para que funcione con el nuevo kernel y la imagen de sistema generada. Basicamente tocar el CMakeLists.txt
- ZeroMQ. Problema con iperf3 solucionado aumentando tamaño de buffers.
- Incrementar RAM de las VMs.

4.2.4. Configuración del *passthrough* de interfaz Ethernet

- Interfaz e1000 QEMU. PCI. BARS. IRQ.
- Modificar la configuración de la VMM camkes en seL4 para permitir el acceso a los recursos PCI correspondientes.
- Solución al problema de utilizar dos interfaces de red. Diferentes IRQ.
- Funcionamiento de passthrough (colas).

4.3. Integración

- Bridge en host. 4 instancias de QEMU. 2 PCs y 2 encriptadores.

4.4. Validación

Capítulo 5

Implementación en hardware

5.1. SuperMicro SYS-E300-9D

- Supermicro server sys-e300-9d with X11SDV-4C-TLN2F motherboard and intel xeon D-2123IT. Fotito del equipo.
- IPMI, COM1, SoL.

5.2. Diseño del experimento

- Que cosas esperamos validar. Rendimiento y que más?
- SetUP.

5.3. Procedimiento

- 1. Configurar redirección consola serie.
- 2. Bootear el Linux para obtener los parámetros para configurar el zmq_samples (lspci).
-

Apéndice A

Apéndice I

Bibliografía

[1] Laricchia, G., Armitage, S., Köver, A., Murtagh, D. J. Ionizing collisions by positrons and positronium impact on the inert atoms. En: Advances in Atomic, Molecular and Optical Physics, tomo 56, pág. 3. Academic Press, Inc., 2009. 13

Publicaciones asociadas

1. Mi primer aviso en la revista **ABC**, 1996
2. Mi segunda publicación en la revista **ABC**, 1997

Agradecimientos

A todos los que se lo merecen, por merecerlo