PUCE Virtual

CLASE
8

Arquitectura de computadoras

Introducción a vulnerabilidades comunes

QS WORLD UNIVERSITY RANKINGS

Educación de calidad

INTRODUCCIÓN DE LA CLASE

GRADO / POSGRADO / TECNOLOGÍAS

Estudia a tu tiempo
son interrupciones

EDUCACIÓN EN LÍNEA DE CALIDAD

¿Te has preguntado alguna vez cómo un error aparentemente pequeño en un programa puede comprometer todo un sistema informático? En esta última clase vamos a explorar precisamente eso. Nos adentraremos en el estudio de las vulnerabilidades comunes, comenzando por los desbordamientos de memoria, un tipo de fallo que ocurre cuando un programa no controla adecuadamente los límites de la memoria que utiliza. A través de este análisis, entenderemos cómo la interacción entre el software y la arquitectura del computador puede dar lugar a riesgos críticos, permitiendo desde la corrupción de datos hasta la ejecución de código malicioso.

Ahora bien, la seguridad no solo se ve afectada por errores internos en el manejo de memoria. También existen ataques que explotan los recursos del sistema desde el exterior, como los ataques de denegación de servicio (DoS y DDoS), cuyo objetivo es saturar el sistema y hacerlo inaccesible para los usuarios legítimos. En este contexto, desarrollarás el RDA 3, orientado a diseñar soluciones eficientes en términos de consumo de recursos, comprendiendo que un sistema seguro no solo protege la información, sino que también garantiza su disponibilidad y estabilidad. Así, esta clase te permitirá analizar problemas reales y proponer soluciones desde una perspectiva integral de ciberseguridad.

RDA 3: Diseñar soluciones eficientes en términos de consumo de recursos para diferentes tipos de aplicaciones.

Reto # 4.

12. INTRODUCCIÓN A VULNERABILIDADES COMUNES

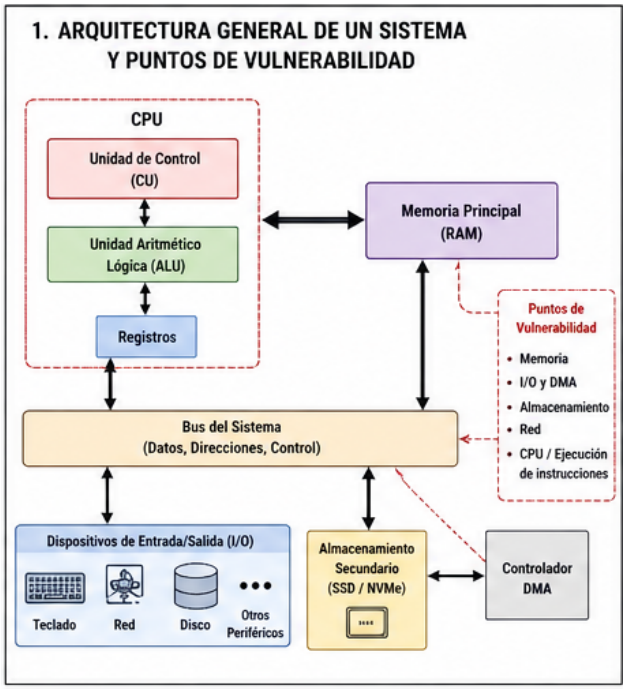
En Arquitectura de Computadoras aplicada a Ciberseguridad, es clave comprender vulnerabilidades originadas en el diseño interno de los sistemas. Los desbordamientos de memoria destacan por su impacto, ya que conectan errores de programación con la gestión de memoria a nivel de hardware, convirtiéndose en una amenaza crítica para la seguridad.

Un desbordamiento de memoria ocurre cuando un programa escribe más datos de los que un buffer puede almacenar, sobrescribiendo zonas contiguas. Esto puede corromper datos, alterar estructuras internas o modificar el flujo de ejecución, convirtiéndose en una vulnerabilidad ampliamente explotada en ciberseguridad.

La figura 1 muestra el diagrama del sistema computacional (CPU, memoria, I/O y bus) que identifica los principales puntos donde pueden surgir vulnerabilidades.

Figura 1

Arquitectura de un sistema y computacional y sus puntos de vulnerabilidad



Autor: Milton Román

Desde la arquitectura, estos errores no son solo fallos de programación, sino resultado del modelo de gestión de memoria del sistema. La organización de la memoria, junto con el diseño del procesador y el sistema operativo, influye en el comportamiento ante errores (Stallings, 2019), por lo que la seguridad depende de cómo se gestionan y protegen los límites de memoria.

12.1 Desbordamientos de memoria

a) Naturaleza del desbordamiento de memoria

Un buffer es una región de memoria reservada para almacenar datos temporalmente. Cuando una aplicación excede el tamaño asignado a este buffer, los datos adicionales se escriben en direcciones adyacentes, lo que puede sobrescribir variables, direcciones de retorno o estructuras críticas del sistema.

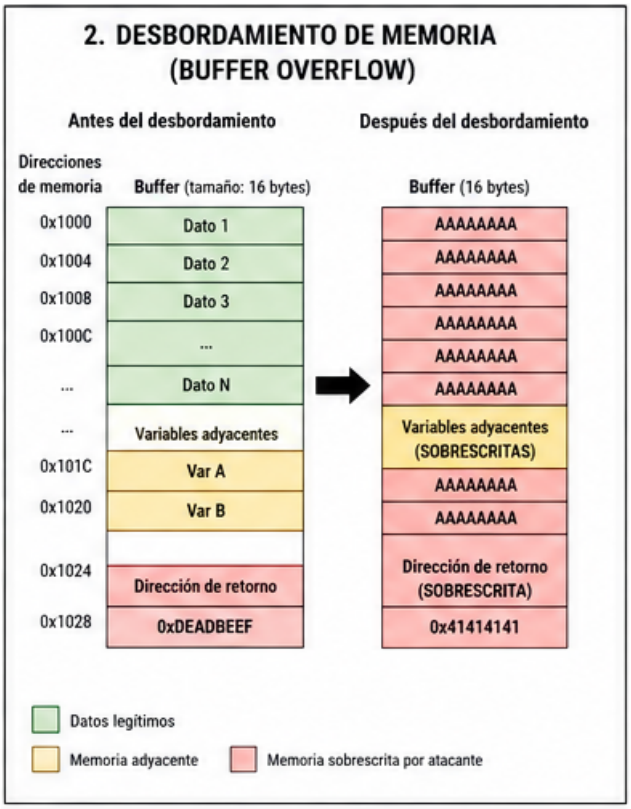
Este comportamiento es peligroso en arquitecturas con memoria organizada en pila y heap. La interacción entre hardware y software en la jerarquía de memoria es clave para el rendimiento,

pero también puede generar vulnerabilidades si no se controla adecuadamente el acceso (Hennessy et al., 2019).

La figura 2 muestra una comparación entre memoria antes y después del desbordamiento, mostrando cómo los datos sobrescriben zonas adyacentes.

Figura 2

Desbordamiento de memoria



Autor: Milton Román

b) Relación con la arquitectura del sistema

Los desbordamientos de memoria se relacionan con la ejecución de instrucciones y gestión de direcciones. En arquitecturas clásicas, la pila almacena direcciones de retorno y variables. Un desbordamiento puede sobrescribir estas direcciones y redirigir la ejecución del programa hacia código malicioso controlado por un atacante.

El entendimiento de la arquitectura interna del computador permite identificar cómo los errores de programación pueden escalar a vulnerabilidades de seguridad de alto impacto (Martínez Amador, 2012). Esto refuerza la idea de que la seguridad no es una capa externa, sino un elemento inherente al diseño del sistema.

c) Tipos principales de desbordamientos

Existen varios tipos de desbordamientos de memoria, entre los cuales destacan los siguientes:

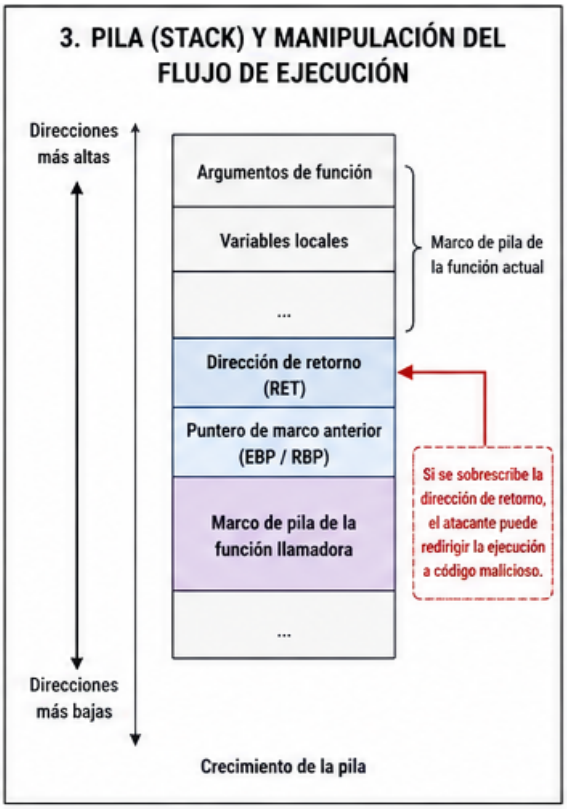
- Desbordamiento de pila (stack overflow): ocurre cuando se excede el espacio asignado en la pila, afectando direcciones de retorno y variables locales.
- Desbordamiento de heap: ocurre en memoria dinámica cuando estructuras asignadas en tiempo de ejecución son sobrescritas.

Estos tipos de fallos pueden derivar en corrupción de datos o ejecución arbitraria de código, lo que convierte a los desbordamientos en una de las principales amenazas a la integridad del sistema.

La figura 3 muestra la estructura de la pila que evidencia cómo un desbordamiento permite modificar la dirección de retorno y alterar el flujo del programa.

Figura 3

Pila y manipulación de flujo de ejecución



Autor: Milton Román

d) Impacto en la ciberseguridad

Desde la perspectiva de la ciberseguridad, los desbordamientos de memoria representan una de las vías más comunes para la explotación de sistemas. Un atacante puede aprovechar esta vulnerabilidad para ejecutar código no autorizado, escalar privilegios o comprometer completamente un sistema.

La evolución de las arquitecturas ha incrementado la complejidad de los sistemas y la superficie de ataque (Fabra Gallo et al., 2020). En entornos modernos, donde interactúan múltiples capas de software y hardware, los desbordamientos pueden propagarse fácilmente sin mecanismos de protección adecuados.

e) Mecanismos de protección a nivel de arquitectura

Para mitigar los desbordamientos de memoria, las arquitecturas modernas y sistemas operativos implementan diversas técnicas de protección. Entre las más importantes se encuentran:

- Canarios de pila: valores de verificación que detectan modificaciones no autorizadas en la pila antes de retornar de una función.
- ASLR (Address Space Layout Randomization): técnica que aleatoriza la ubicación de las direcciones de memoria para dificultar ataques.
- DEP (Data Execution Prevention): impide la ejecución de código en regiones de memoria destinadas a datos.

Estas medidas reflejan la evolución de la arquitectura hacia un enfoque donde la seguridad se integra directamente en el diseño del sistema.

f) Relación con la evolución de las arquitecturas

La evolución de las arquitecturas ha transformado la gestión de memoria y la seguridad. Se pasó de modelos simples a sistemas complejos con múltiples capas de abstracción, aumentando tanto la funcionalidad como los riesgos asociados en los sistemas computacionales (Ibarra Mayorga, 2009).

En arquitecturas modernas como RISC-V se integran mecanismos de protección más robustos a nivel de hardware, reduciendo la explotación de memoria. Jaramillo Villegas et al. (2022) destacan que estas arquitecturas incorporan seguridad desde el diseño, disminuyendo la dependencia exclusiva del software.

g) Relación con eficiencia y consumo de recursos

Aunque se asocian a seguridad, los desbordamientos también afectan la eficiencia. Un manejo incorrecto de memoria puede causar fallos, reinicios o consumo excesivo de recursos. Según el RDA 3, diseñar soluciones eficientes implica optimizar el rendimiento y reducir riesgos que comprometan la estabilidad del sistema.

Por ejemplo, sistemas embebidos o aplicaciones en tiempo real deben equilibrar el uso eficiente de memoria con mecanismos de protección, evitando sobrecargas que afecten el desempeño general.

h) Relación con almacenamiento, I/O, DMA y virtualización

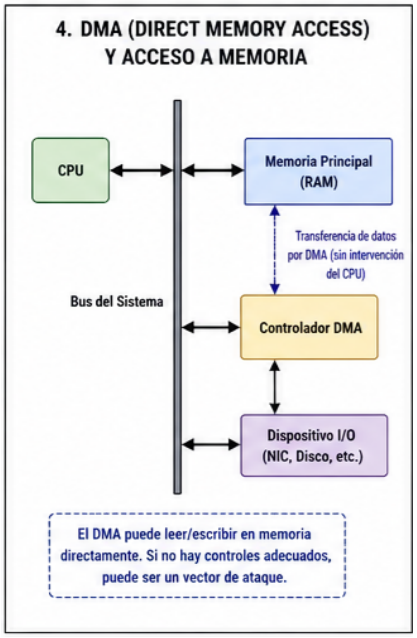
Aunque los desbordamientos de memoria se originan en la memoria principal, su impacto se extiende a almacenamiento (SSD, NVMe), dispositivos de entrada/salida y mecanismos como el DMA. Este último accede directamente a la memoria sin intervención del CPU, generando posibles riesgos. En entornos virtualizados, estas vulnerabilidades pueden afectar máquinas virtuales y contenedores, comprometiendo el aislamiento entre entornos y la seguridad del sistema.

La interacción entre los distintos componentes del sistema es compleja y requiere mecanismos de control estrictos para evitar fallos de seguridad derivados del acceso no autorizado a memoria (Stallings, 2019).

En la figura 4 se muestra el diagrama del acceso directo a memoria sin intervención del CPU, destacando riesgos de seguridad asociados.

Figura 4

DMA y acceso a memoria



Autor: Milton Román

i) Consideraciones en virtualización

En entornos de virtualización, los desbordamientos de memoria pueden tener consecuencias adicionales, ya que un fallo en una máquina virtual podría potencialmente afectar al hipervisor o a otras máquinas virtuales si existe una vulnerabilidad en el aislamiento. Esto se conoce como “escape de VM” en contextos más avanzados de seguridad.

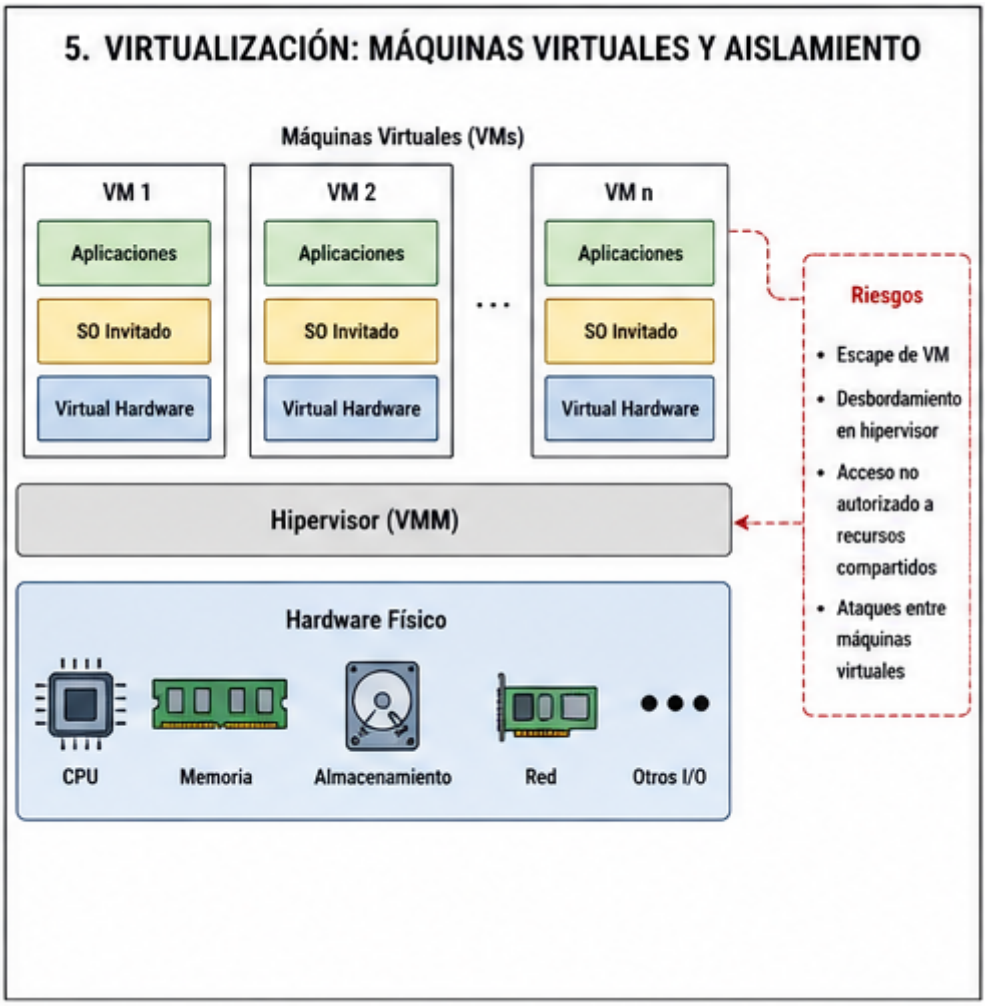
Por ello, el diseño de arquitecturas seguras debe considerar no solo la memoria física, sino también cómo se abstrae y protege en entornos virtualizados.

Los desbordamientos de memoria son una vulnerabilidad clave en Arquitectura de Computadoras por su relación con la gestión de memoria, ejecución de instrucciones y la interacción hardware-software. Su comprensión es esencial en ciberseguridad, ya que errores simples pueden escalar a fallos críticos. Además, afectan la seguridad, eficiencia y estabilidad. Su estudio debe integrarse con almacenamiento, periféricos, DMA y virtualización (Reto 4). La evolución de las arquitecturas destaca la importancia de incorporar mecanismos de protección desde el diseño para lograr sistemas más seguros y eficientes.

La figura 5 muestra una arquitectura con hipervisor y múltiples máquinas virtuales que muestra el aislamiento y posibles riesgos entre entornos.

Figura 5

Virtualización: máquinas virtuales y aislamiento



Autor: Milton Román

Para reforzar los conocimientos sobre desbordamiento de memoria, vista el siguiente enlace:

- **Título:** Que es y cómo opera un ataque buffer overflow
- **Descripción:** El video explica qué es un desbordamiento de memoria, cómo ocurre al exceder la capacidad de un buffer y cómo puede sobrescribir datos críticos. También muestra sus riesgos en ciberseguridad, como la ejecución de código malicioso y el control del sistema.
- **Enlace:** https://www.youtube.com/watch?v=8BxU0zst6sw

12.2 Ataques DoS y DDoS

En Arquitectura de Computadoras aplicada a Ciberseguridad, los ataques DoS y DDoS son amenazas clave para la disponibilidad. No buscan robar información, sino saturar recursos del sistema, provocando que deje de responder a solicitudes legítimas y afectando el funcionamiento normal de los servicios.

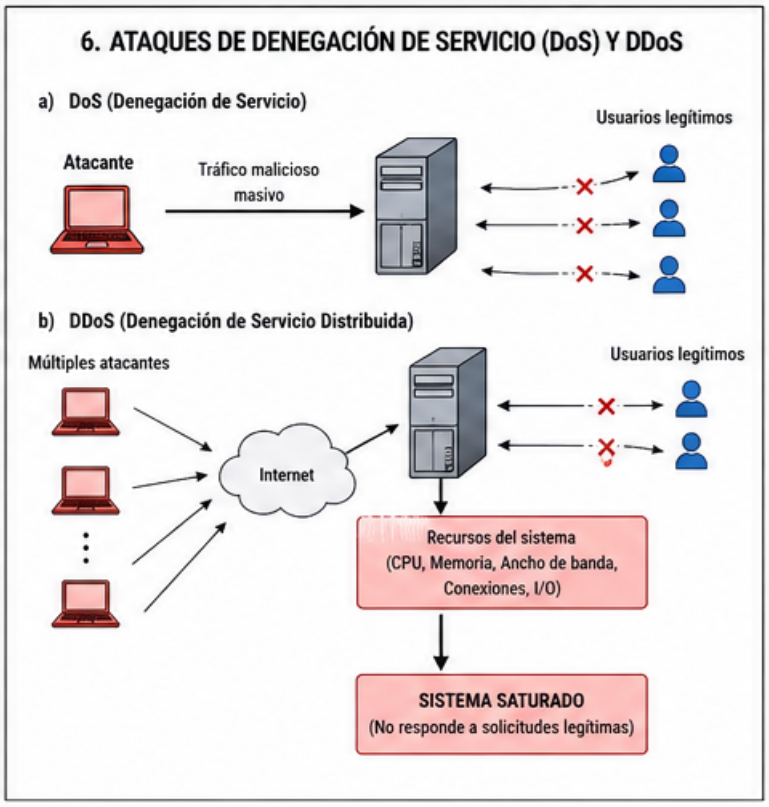
Desde la arquitectura de sistemas, los ataques DoS y DDoS explotan límites de CPU, memoria, red y E/S. Stallings (2019) indica que el rendimiento depende del equilibrio entre carga y

capacidad, por lo que la sobrecarga intencional constituye una amenaza crítica para la disponibilidad del sistema.

La figira 6 ilustración de ataques DoS (una fuente) y DDoS (múltiples fuentes) saturando un servidor.

Figura 6

Ataques de denegación de servicio (DoS) y DDoS



Autor: Milton Román

a) Concepto de ataques DoS

Un ataque DoS ocurre cuando un único sistema intenta sobrecargar un servicio o recurso específico, enviando una gran cantidad de solicitudes o consumiendo recursos de forma excesiva. El objetivo es agotar la capacidad del sistema víctima, provocando lentitud extrema o caída del servicio.

Estos ataques pueden dirigirse a diferentes capas de la arquitectura, incluyendo:

- Capa de red (saturación de ancho de banda)
- Capa de aplicación (sobrecarga de peticiones HTTP o servicios)
- Capa de transporte (explotación de conexiones TCP/UDP)

Hennessy y Patterson explican que la eficiencia del sistema depende de la correcta gestión de la jerarquía de recursos, incluyendo CPU, memoria y entrada/salida, lo que implica que una saturación en cualquiera de estos niveles puede degradar significativamente el rendimiento general del sistema (Hennessy et al., 2019).

b) Ataques DDoS: evolución del problema

Los ataques DDoS representan una evolución de los DoS tradicionales, donde múltiples sistemas distribuidos, generalmente comprometidos, atacan simultáneamente un mismo objetivo. Estos sistemas forman redes conocidas como botnets, que amplifican el impacto del ataque al generar tráfico masivo desde múltiples fuentes.

La evolución de las arquitecturas de computadoras y la interconexión de sistemas ha incrementado la complejidad de los entornos tecnológicos, lo que a su vez amplía la superficie

de ataque y facilita la ejecución de amenazas distribuidas como los DDoS (Fabra Gallo et al., 2020).

c) Relación con la arquitectura de computadoras

Desde el punto de vista arquitectónico, los ataques DoS y DDoS afectan directamente la disponibilidad de los recursos del sistema. La CPU puede verse saturada por procesos excesivos, la memoria puede agotarse debido a conexiones abiertas o buffers mal gestionados, y los sistemas de entrada/salida pueden colapsar por exceso de solicitudes.

El conocimiento de la arquitectura interna del computador permite comprender cómo la sobrecarga de recursos puede degradar el funcionamiento del sistema hasta hacerlo inoperable (Martínez Amador, 2012). Esto demuestra que la seguridad no solo se relaciona con la protección de datos, sino también con la gestión eficiente de recursos.

d) Impacto en almacenamiento, I/O y DMA

En el contexto del Reto 4, los ataques DoS y DDoS también pueden afectar sistemas de almacenamiento como HDD, SSD y NVMe, ya que una alta tasa de solicitudes puede saturar los controladores de entrada/salida. Los dispositivos de I/O pueden convertirse en cuellos de botella si reciben más solicitudes de las que pueden procesar.

El DMA, aunque mejora la eficiencia del sistema al permitir transferencias directas a memoria sin intervención del CPU, también puede verse afectado indirectamente por la saturación de solicitudes que afectan el bus de memoria o los controladores.

La interacción entre los componentes del sistema debe estar cuidadosamente gestionada para evitar que la sobrecarga en un subsistema afecte el rendimiento global del sistema (Stallings, 2019).

e) Virtualización y contenedores

En entornos virtualizados, los ataques DoS y DDoS pueden tener un impacto aún mayor, ya que múltiples máquinas virtuales o contenedores comparten recursos físicos. Un ataque dirigido a una VM puede afectar al hipervisor o degradar el rendimiento de otras máquinas virtuales.

Las arquitecturas modernas como RISC-V y los sistemas virtualizados requieren mecanismos de aislamiento robustos para garantizar que el consumo excesivo de recursos en un entorno no afecte a otros (Jaramillo Villegas et al., 2022). Sin una adecuada gestión de recursos, un ataque en una capa virtual puede escalar a nivel físico.

f) Tipos de ataques DoS y DDoS

Entre los principales tipos de ataques se pueden identificar:

- Ataques de volumen: saturan el ancho de banda con tráfico masivo.
- Ataques de protocolo: explotan debilidades en TCP/IP para consumir recursos del servidor.
- Ataques de aplicación: se enfocan en servicios específicos como HTTP, DNS o bases de datos.

Estos ataques pueden ser combinados para aumentar su efectividad y dificultar su mitigación.

g) Relación con eficiencia

Desde el punto de vista del RDA 3, los ataques DoS y DDoS evidencian la importancia del diseño eficiente de sistemas. Un sistema eficiente no solo debe optimizar el rendimiento bajo condiciones normales, sino también mantener la estabilidad ante cargas extremas.

El diseño de soluciones eficientes implica implementar mecanismos de control de tráfico, balanceo de carga y limitación de recursos por usuario o proceso, evitando que un solo punto de entrada consuma todos los recursos disponibles.

h) Mecanismos de protección

Para mitigar ataques DoS y DDoS, se implementan diversas estrategias a nivel de arquitectura y red:

- Firewalls con filtrado de tráfico

- Sistemas de detección de intrusos (IDS/IPS)
- Balanceadores de carga
- Limitación de tasa de solicitudes (rate limiting)
- Segmentación de redes y aislamiento de servicios

Estas medidas permiten distribuir la carga y evitar que un ataque afecte completamente al sistema.

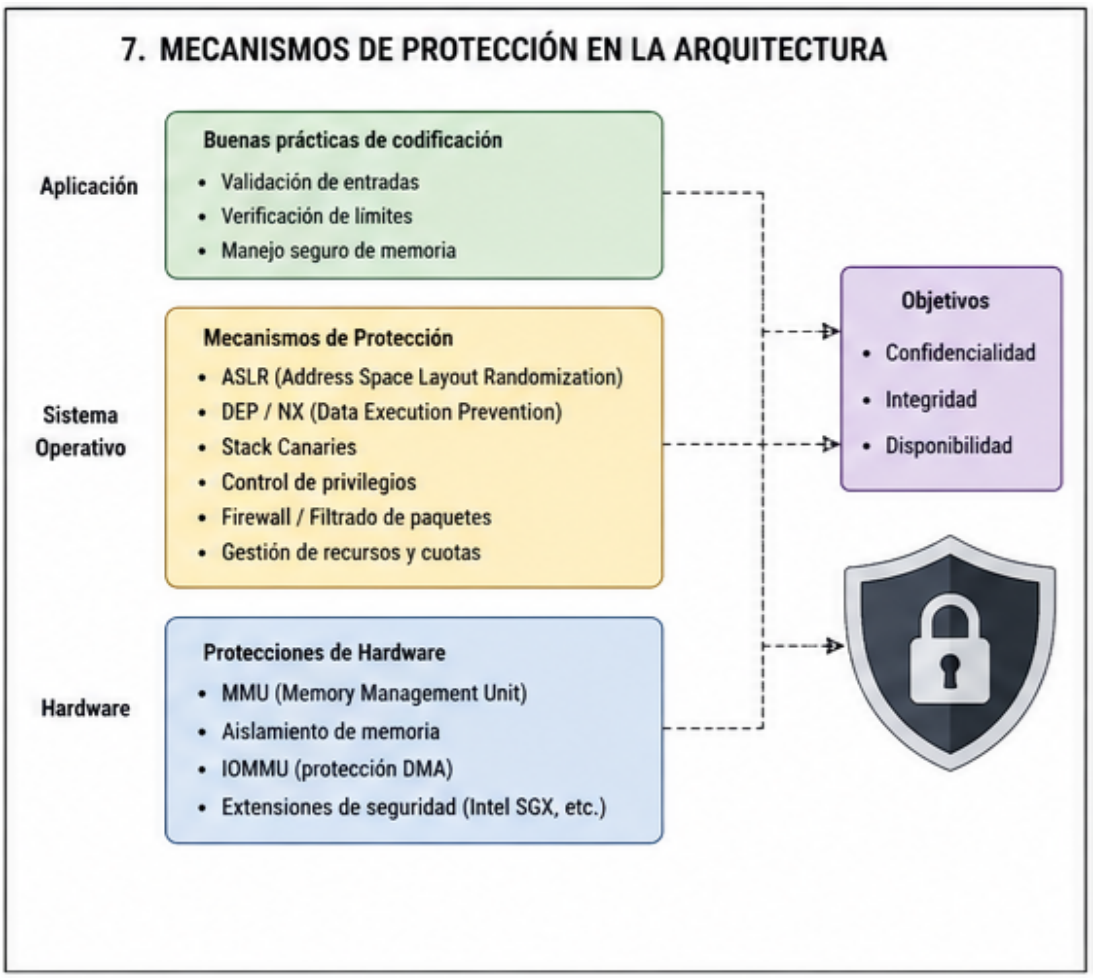
i) Evolución de arquitecturas y seguridad

La evolución de las arquitecturas de computadoras ha permitido desarrollar sistemas más resilientes frente a ataques de denegación de servicio. La transición hacia sistemas complejos y distribuidos ha incrementado tanto las capacidades como los riesgos asociados a la interconexión de sistemas (Ibarra Mayorga, 2009).

La figura 7 muestra un esquema de capas de seguridad (hardware, sistema operativo y aplicación) con mecanismos como ASLR, DEP y control de acceso.

Figura 7

Mecanismos de protección en la arquitectura



Autor: Milton Román

En arquitecturas modernas, la integración de mecanismos de seguridad desde el diseño es fundamental para garantizar la disponibilidad de los servicios.

Los ataques DoS y DDoS son críticos en Arquitectura de Computadoras por su impacto en la disponibilidad. Afectan CPU, memoria, almacenamiento, I/O y entornos virtualizados. Su análisis en el Reto 4 resalta integrar seguridad en todos los componentes, mientras el RDA 3 enfatiza diseñar sistemas eficientes y resistentes a sobrecargas. La evolución arquitectónica

demuestra que la resiliencia depende de una adecuada gestión de recursos y de incorporar mecanismos de protección desde el diseño del sistema.

Refuerza tus conocimientos sobre ataques DoS y DDoS ingresando al siguiente link:

- **Título:** Entiende los ATAQUES Dos y DDoS: ¿Cómo Funcionan?
- **Descripción:** El video explica de forma sencilla qué son los ataques DoS y DDoS, mostrando cómo se saturan los recursos de un servidor mediante múltiples solicitudes, afectando la disponibilidad del servicio y dejando sin acceso a usuarios legítimos.
- **Enlace**: https://www.youtube.com/watch?v=DAs_-3beIMU

Educación de calidad

REFERENCIAS

- **Fabra Gallo, J. M., Martínez Márquez, Y., & Valcárcel Izquierdo, N. (2020).** Estudio comparado de los contenidos de la asignatura Arquitectura de Computadoras: oportunidades de mejoras. Dialnet. Obtenido de https://dialnet.unirioja.es/servlet/articulo?codigo=8590371
- **Hennessy, Hennessy, J. L., & Patterson, D. A. (2019).** ARQUITECTURA DE COMPUTADORES: Un enfoque cuantitativo. McGrawHill. https://doi.org/1-55860-069-8
- **Ibarra Mayorga, M. F. (2009).** Evolución de las arquitecturas de las computadoras. eLibro. Obtenido de https://elibro.puce.elogim.com/es/lc/puce/titulos/28671
- **Jaramillo Villegas, J. A., Zuluaga Bucheli, H. M., & Sepúlveda Caviedes, C. (2022).** Arquitectura de computadoras con RISC-V. Universidad Tecnológica de Pereira. Digitalia. Obtenido de https://www.digitaliapublishing.com/a/136758/
- **Martínez Amador, H. (2012).** Arquitectura de computadoras: basado en competencias para nivel superior. Obtenido de https://elibro.puce.elogim.com/es/lc/puce/titulos/130397
- **Stallings, W. (2019).** Computer organization and architecture : designing for performance (11ava ed.). Pearson Education. Obtenido de https://os.ecci.ucr.ac.cr/ci0114/material/Stallings/Computer-Organization-Architecture-11th.pdf

PUCE Virtual

síguenos en:

www.pucevirtual.puce.edu.ec