PUCE Virtual

CLASE

7

ESTRUCTURA DE DATOS (PROGRAMACIÓN III)

Grafos y selección de estructuras

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

En esta penúltima semana se abordarán conceptos fundamentales relacionados con el uso de estructuras de datos en el contexto de la ciberseguridad, con especial énfasis en los grafos y su aplicación en el análisis de sistemas complejos. A medida que las infraestructuras tecnológicas evolucionan, también lo hacen las amenazas, lo que hace indispensable contar con herramientas que permitan modelar, analizar y comprender las relaciones entre dispositivos, usuarios y servicios. En este sentido, los grafos se presentan como una solución poderosa para representar redes, identificar vulnerabilidades, detectar nodos críticos y analizar posibles rutas de ataque, facilitando la toma de decisiones en entornos de seguridad informática.

Además, se estudiará la importancia de la correcta selección de estructuras de datos, considerando factores como el rendimiento, el uso de memoria y la complejidad computacional. Se realizará una comparación entre estructuras como listas, árboles y grafos, analizando sus ventajas, limitaciones y aplicaciones en escenarios reales de ciberseguridad. Asimismo, se discutirán los trade-offs de diseño y la justificación técnica de las decisiones tomadas al desarrollar sistemas seguros.

Finalmente, estos conocimientos permitirán a los estudiantes diseñar soluciones más eficientes, escalables y robustas, alineadas con las necesidades actuales de protección de la información y gestión de riesgos tecnológicos.

Clase 7:
Seleccionar las estructuras de datos más adecuadas según los requerimientos y características de distintos escenarios de ciberseguridad.

7. Grafos y selección de estructuras

Los grafos en ciberseguridad son fundamentales porque permiten representar de forma clara las relaciones entre componentes de un sistema y analizar el flujo de información, facilitando la detección de amenazas, vulnerabilidades, rutas de ataque y puntos críticos. Por ello, son ampliamente utilizados para fortalecer la seguridad y proteger datos.

Por otro lado, la selección de estructuras de datos es clave para desarrollar soluciones eficientes, ya que depende del tipo y volumen de información, así como de la complejidad y el tiempo de respuesta. Una elección adecuada optimiza el rendimiento, asegura la escalabilidad y mejora la toma de decisiones en ciberseguridad.

7.1. Grafos en escenarios de ciberseguridad

En ciberseguridad, los grafos son especialmente útiles debido a que los sistemas informáticos, como usuarios, dispositivos, servidores y flujos de tráfico, se encuentran altamente interconectados, lo que hace necesario contar con herramientas que permitan analizar dichas relaciones de forma estructurada. La Figura 1 muestra diferentes escenarios de los grafos aplicados en ciberseguridad.

Figura 1. Escenarios de grafos aplicados a la Ciberseguridad. Creación de autor Patricio Coba

Mediante el uso de grafos, es posible representar de manera explícita estas interacciones, facilitando la comprensión de sistemas complejos y dinámicos. Los grafos pueden adoptar diferentes formas, como dirigidos, no dirigidos o ponderados, lo que permite modelar distintos tipos de relaciones según el contexto.

Asimismo, posibilitan el uso de métricas avanzadas como la centralidad, los caminos mínimos y el clustering, las cuales resultan clave para identificar nodos relevantes o comportamientos inusuales. Entre los aspectos más importantes a considerar se encuentran la escalabilidad para manejar grandes volúmenes de datos, la interpretabilidad de las relaciones representadas y su integración con algoritmos de inteligencia artificial. Los grafos permiten modelar redes informáticas, detectar ataques distribuidos e identificar patrones de comportamiento malicioso, ya que ayudan a descubrir anomalías mediante el análisis de desviaciones en la estructura normal de la red.

7.1.1. Modelado de infraestructuras de red

El modelado de infraestructuras de red mediante grafos consiste en representar dispositivos, servicios y conexiones como una estructura matemática en la que los nodos corresponden a activos tecnológicos, como routers, servidores, computadoras o dispositivos de seguridad, mientras que las aristas representan las conexiones o relaciones existentes entre ellos dentro de la red. Esto permite obtener una representación visual clara de la topología de la red, lo que facilita su análisis y comprensión, especialmente en entornos complejos donde existen múltiples interacciones entre sistemas.

Los grafos permiten modelar tanto redes físicas como redes lógicas, incorporando atributos o pesos en las conexiones, como la latencia, el ancho de banda o el nivel de confianza de una comunicación. Para que este modelo sea efectivo, es fundamental considerar aspectos como la precisión del modelo, la actualización constante de la información y la inclusión de atributos relevantes que reflejen el estado real de la infraestructura tecnológica (Sommerville, 2011).

Este tipo de representación permite analizar superficies de ataque, simular la propagación de malware y evaluar la resiliencia de la red ante incidentes o fallos, lo cual es clave para diseñar estrategias de defensa más efectivas. Diversos estudios señalan que los grafos permiten representar Internet y otras redes como estructuras complejas altamente interconectadas, lo que facilita la detección de vulnerabilidades y el análisis de riesgos en sistemas informáticos (Tanenbaum & Wetherall, 2012; Cormen et al., 2009).

La figura 2 muestra un ejemplo de modelado de red que puede representarse a través de grafos.

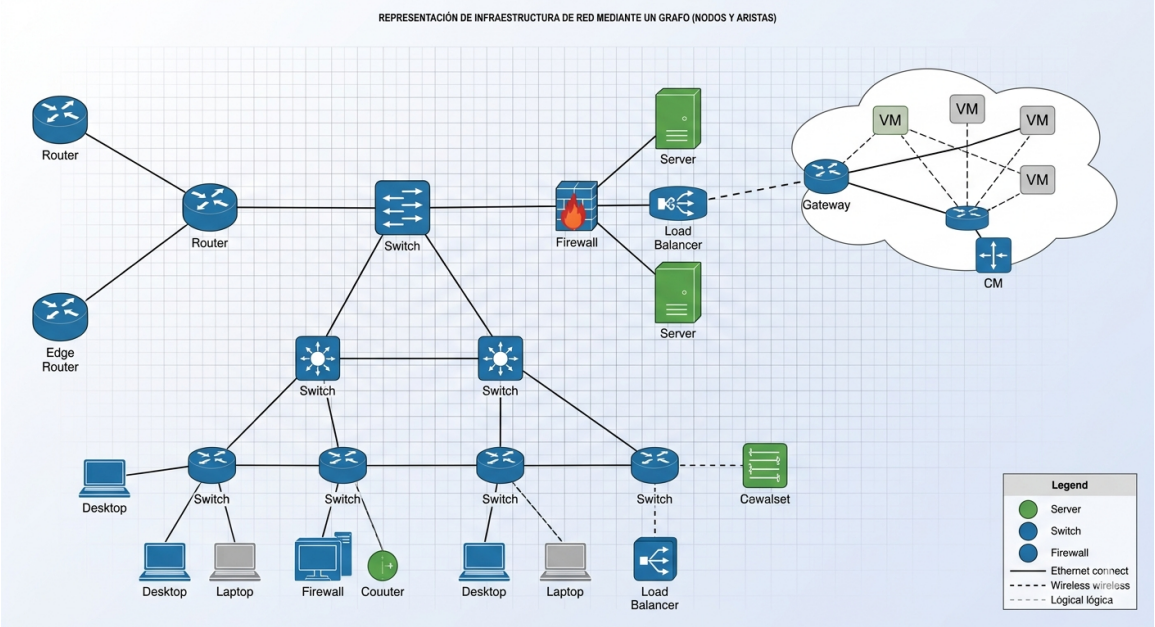


Figura 2. Ejemplo de modelado de Red. Creación de autor Patricio Coba

7.1.2. Detección de nodos críticos

Un nodo crítico en una red es aquel cuya falla o compromiso puede afectar de manera significativa el funcionamiento general del sistema, comprometiendo la disponibilidad, integridad o confidencialidad de la información. Estos nodos se identifican comúnmente mediante métricas de análisis de grafos, como la centralidad de grado, que mide la cantidad de conexiones directas; la centralidad de intermediación, que identifica nodos que actúan como puentes en la comunicación; y la centralidad de cercanía, que evalúa qué tan próximo está un nodo al resto de la red. Estas características permiten detectar cuellos de botella y puntos estratégicos dentro de la infraestructura (Sánchez, 2020).

La tabla 1 muestra los principales nodos críticos en una red, su riego, impacto y medidas de mitigación.

Tabla 1. Detalle de nodos críticos de una red. Creación de autor Patricio Coba

Nodo crítico	Descripción	Riesgo asociado	Impacto en la organización	Medidas de mitigación
Servidor principal	Aloja aplicaciones o servicios clave del negocio	Caída del sistema, pérdida de datos	Interrupción total de operaciones	Redundancia, backups, alta disponibilidad
Firewall	Controla el acceso entre redes internas y externas	Accesos no autorizados	Compromiso de la red interna	Configuración robusta, monitoreo continuo
Router	Gestiona el tráfico de red	Interrupción de conectividad	Pérdida de comunicación entre sistemas	Redundancia, mantenimiento periódico

Base de datos	Almacena información crítica	Robo o corrupción de datos	Pérdida de información sensible	Cifrado, backups, control de accesos
Servidor de autenticación	Gestiona credenciales y accesos	Suplantación de identidad	Accesos indebidos a sistemas	MFA, políticas de seguridad
Switch central	Conecta múltiples dispositivos dentro de la red	Caída de red interna	Interrupción de servicios internos	Equipos redundantes, segmentación de red
Sistema DNS	Traduce nombres de dominio a direcciones IP	Ataques de redirección (DNS spoofing)	Desvío de tráfico, pérdida de acceso a servicios	DNS seguro (DNSSEC), redundancia
Servidor de correo	Gestiona comunicaciones internas y externas	Phishing, malware	Compromiso de información y reputación	Filtros antispam, antivirus
Sistema de backups	Almacena copias de seguridad	Pérdida de respaldo	Imposibilidad de recuperación ante fallos	Backups periódicos, almacenamiento externo
SIEM	Sistema de monitoreo de eventos de seguridad	Falta de detección de amenazas	Respuesta tardía ante incidentes	Configuración adecuada, análisis continuo

Es fundamental analizar estos nodos, ya que su impacto en la disponibilidad del sistema es alto y representan objetivos potenciales para ataques dirigidos, como denegación de servicio o accesos no autorizados (Pérez & Gómez, 2021). La identificación de nodos críticos permite a las organizaciones priorizar medidas de protección, implementar redundancia y fortalecer la resiliencia de la red. En este sentido, el análisis de centralidad se convierte en una herramienta clave para detectar nodos relevantes dentro de redes complejas, facilitando la toma de decisiones estratégicas en la gestión de riesgos (Martínez, 2019).

7.1.3. Dependencias y puntos únicos de falla

Un punto único de falla (Single Point of Failure, SPOF) es un componente dentro de un sistema cuya caída o mal funcionamiento puede provocar la interrupción total o parcial del servicio, afectando directamente la continuidad operativa de la infraestructura tecnológica. Este tipo de elementos suele estar asociado a dependencias jerárquicas, donde múltiples procesos o servicios dependen de un único nodo, lo que genera una vulnerabilidad estructural significativa dentro del sistema (Sánchez, 2020).

Es fundamental identificar estos puntos críticos para reducir riesgos y mejorar la resiliencia. Entre los aspectos más importantes a considerar se encuentran la implementación de mecanismos de redundancia, que permiten contar con componentes alternativos en caso de

fallo, y el balanceo de carga, que distribuye el tráfico o procesamiento entre varios nodos para evitar la sobrecarga de uno solo (Pérez & Gómez, 2021).

El análisis de SPOF resulta clave para identificar dependencias críticas y prevenir fallos catastróficos que puedan ser explotados por atacantes o que deriven en caídas del sistema. En este sentido, el uso de grafos permite analizar de manera eficiente las relaciones y dependencias complejas entre los distintos componentes, facilitando la detección de vulnerabilidades ocultas y mejorando la planificación de estrategias de mitigación (Martínez, 2019).

La tabla 2 muestra una lista de los puntos únicos de falla comunes, su riesgo asociado, impacto y medidas de mitigación.

Tabla 2. Lista de puntos de falla comunes. Creación de autor Patricio Coba

Punto único de falla (SPOF)	Descripción	Riesgo asociado	Impacto en la organización	Medidas de mitigación
Servidor único	Único servidor que soporta una aplicación crítica	Caída total del servicio	Interrupción de operaciones	Clúster, alta disponibilidad
Base de datos centralizada	Una sola instancia de base de datos	Pérdida o corrupción de datos	Pérdida de información crítica	Replicación, backups automáticos
Firewall único	Único punto de control de acceso a la red	Fallo en seguridad o bloqueo total	Exposición o aislamiento de la red	Firewalls redundantes
Router principal	Dispositivo principal de conectividad	Pérdida de conexión a internet/red	Inactividad operativa	Redundancia de routers
Servidor de autenticación	Único sistema de control de accesos	Imposibilidad de autenticación	Bloqueo de usuarios o accesos indebidos	Sistemas redundantes, MFA
Enlace de internet único	Un solo proveedor de conectividad	Caída de conexión externa	Interrupción de servicios online	Proveedores múltiples (failover)
Sistema DNS único	Único servidor DNS	Fallo en resolución de nombres	Acceso interrumpido a servicios	DNS redundante
Sistema de backups único	Un solo repositorio de respaldo	Pérdida de copias de seguridad	Imposibilidad de recuperación	Backups distribuidos y fuera de sitio
Switch central único	Único punto de conexión interna	Caída de red interna	Interrupción de comunicaciones internas	Switches redundantes
Plataforma en nube única	Dependencia de un solo proveedor cloud	Caída del servicio externo	Interrupción total de servicios	Estrategia multi-cloud o híbrida

Single point of failure. Página Web de IONOS Guía Digital que contiene conceptos, características, ejemplos varios y formas de aplicar. Enlace: https://www.ionos.com/es-us/digitalguide/servidores/seguridad/single-point-of-failure/

7.1.4. Análisis forense basado en grafos

El uso de grafos en el análisis forense digital se basa en la capacidad de representar y reconstruir eventos de seguridad mediante estructuras que reflejan las relaciones entre distintos elementos de un sistema. En este enfoque, los eventos, como accesos, ejecuciones de procesos o transferencias de datos, se modelan como nodos, mientras que las relaciones entre ellos ya sean temporales o causales, se representan mediante aristas. Esta forma de modelado permite visualizar de manera clara la secuencia de acciones dentro de un incidente, facilitando la comprensión de cómo se desarrolló un ataque (Sánchez, 2020).

Entre sus características más relevantes se encuentran la representación estructurada de eventos y la posibilidad de establecer relaciones que evidencian dependencias y cronologías. Asimismo, aspectos como la integridad de los datos y la trazabilidad son fundamentales, ya que garantizan que la información analizada sea confiable y que cada acción pueda ser rastreada de forma precisa (Pérez & Gómez, 2021).

Este enfoque permite reconstruir ataques, identificar el origen de amenazas y analizar patrones de comportamiento malicioso, lo que contribuye significativamente a la mejora de los mecanismos de respuesta ante incidentes. Los grafos se consolidan como una herramienta eficaz para el análisis profundo de eventos y la detección de actividades sospechosas en entornos complejos (Martínez, 2019).

7.2. Selección de estructuras de datos

La selección de estructuras de datos es fundamental en ciberseguridad, ya que es indispensable elegir la más adecuada según el problema que se desea resolver. Esta decisión influye directamente en la eficiencia del sistema, el tiempo de respuesta y el uso de recursos. Además, una correcta elección permite manejar grandes volúmenes de información, como logs o eventos, y facilita la detección oportuna de amenazas y comportamientos anómalos.

7.2.1. Criterios de selección de estructuras

El proceso de selección de la estructura de datos adecuada consiste en elegir la forma más eficiente de organizar y manejar la información en función del problema a resolver, considerando tanto las necesidades operativas como las restricciones del sistema. Esta elección depende en gran medida del tipo de datos que se van a procesar, así como de la eficiencia requerida en términos de acceso, inserción, eliminación y búsqueda. Por ello, cada estructura de datos ofrece ventajas específicas que deben evaluarse cuidadosamente según el contexto (Sánchez, 2020).

Uno de los aspectos a considerar es la complejidad temporal de las operaciones, que influye directamente en el rendimiento del sistema; el uso de memoria, que determina el consumo de recursos; y la escalabilidad, que permite que la solución siga siendo eficiente a medida que crece el volumen de datos (Pérez & Gómez, 2021).

Una adecuada selección de estructuras de datos resulta clave para optimizar el análisis de grandes volúmenes de logs, así como para el procesamiento eficiente de eventos en tiempo real, lo que facilita la detección oportuna de amenazas y anomalías. De esta manera, una correcta elección contribuye a mejorar el desempeño de los sistemas de seguridad y la toma de decisiones estratégicas (Martínez, 2019).

La Tabla 3 muestra una comparativa de las principales estructuras de datos utilizadas en Ciberseguridad.

Tabla 3. Comparativa de estructuras de datos utilizadas en Ciberseguridad. Creación de autor Patricio Coba

Estructura de datos	Características principales	Ventajas	Desventajas	Aplicación en ciberseguridad
Lista	Estructura lineal y secuencial	Fácil implementación, bajo costo computacional	Búsqueda lenta en grandes volúmenes	Almacenamiento de logs y eventos
Pila (Stack)	LIFO (último en entrar, primero en salir)	Control de procesos, simplicidad	Acceso limitado	Análisis de llamadas, ejecución de procesos
Cola (Queue)	FIFO (primero en entrar, primero en salir)	Manejo ordenado de eventos	No permite acceso aleatorio	Procesamiento de eventos en tiempo real
Árbol	Estructura jerárquica	Búsqueda eficiente, organización estructurada	Complejidad de implementación	Control de accesos, clasificación de amenazas
Árbol binario	Cada nodo tiene máximo dos hijos	Búsqueda rápida (si está balanceado)	Puede degradarse si no está balanceado	Indexación de datos de seguridad
Tabla hash	Acceso mediante clave	Búsqueda muy rápida (O(1) promedio)	Colisiones, uso de memoria	Gestión de credenciales, detección de duplicados
Grafo	Relaciones complejas entre nodos	Alta flexibilidad, modelado realista	Alto costo computacional	Análisis de redes, detección de ataques

Árbol de decisión	Representación de decisiones y reglas	Interpretación sencilla	Puede crecer mucho	Sistemas de detección de intrusos (IDS)
Heap	Estructura basada en prioridad	Acceso rápido a elemento prioritario	No orden total	Priorización de alertas de seguridad

Cómo elegir la estructura de datos adecuada: una guía de decisión integral. Página web traducida al español donde se encuentran definiciones básicas de estructuras de datos y los criterios para elegir la más adecuada. Enlace: https://www-designgurus-io.translate.goog/blog/choosing-the-right-data-structure-a-comprehensive-decision-guide?_x_tr_sl=en&_x_tr_tl=es&_x_tr_hl=es&_x_tr_pto=tc&_x_tr_hist=true

7.2.2. Comparación entre listas, árboles y grafos

Estas estructuras cumplen funciones distintas en el manejo y organización de la información, y su elección depende del tipo de problema que se desea resolver. Las listas son simples y secuenciales, pero limitadas para relaciones complejas. Los árboles organizan datos jerárquicamente, aunque son rígidos. En cambio, los grafos ofrecen mayor flexibilidad para representar múltiples relaciones, siendo ideales para redes complejas.

En ciberseguridad, las listas sirven para registros, los árboles para estructuras de acceso y los grafos para modelar redes y detectar ataques. Por ello, los grafos son más potentes, aunque con mayor costo computacional.

La tabla 4 muestra una comparativa de ventajas y desventajas entre listas, árboles y grafos.

Tabla 4. Comparativa entre árboles, listas y grafos. Creación de autor Patricio Coba

Estructura	Tipo de relación	Ventajas	Desventajas	Uso en ciberseguridad
Lista	Lineal	Simple, rápida	Limitada en relaciones	Almacenamiento de logs
Árbol	Jerárquica	Organización clara	Poco flexible	Control de accesos
Grafo	Compleja	Alta flexibilidad	Mayor complejidad	Análisis de redes y ataques

7.2.3. Trade-offs de diseño

En el diseño de sistemas informáticos, los *trade-offs* o compromisos entre rendimiento, uso de memoria y complejidad son un aspecto fundamental al momento de seleccionar

estructuras de datos y algoritmos adecuados. Cada decisión implica sacrificar ciertas ventajas para obtener otras; por ejemplo, los grafos ofrecen una gran flexibilidad al permitir representar relaciones complejas entre múltiples entidades, lo que los hace ideales para modelar redes y dependencias, pero a costa de un mayor consumo de memoria y una mayor complejidad computacional en su procesamiento (Sánchez, 2020).

La Figura 3 muestra el triángulo Trade-Offs aplicados a algoritmos.

Figura 3. Triángulo Trade-Offs aplicado a algoritmos. Creación de autor Patricio Coba

En contraste, las listas son estructuras simples y eficientes en términos de velocidad y uso de recursos, pero resultan limitadas cuando se requiere representar relaciones más elaboradas o multidimensionales. Estos compromisos deben evaluarse cuidadosamente según los objetivos del sistema.

Este equilibrio es especialmente importante, ya que se requiere procesar grandes volúmenes de datos en tiempo real sin sacrificar la precisión en la detección de amenazas. Por ejemplo, el uso de grafos puede mejorar significativamente la identificación de patrones de ataque, pero puede afectar el rendimiento si no se optimiza adecuadamente. Por ello, es necesario encontrar un balance entre precisión y eficiencia que permita garantizar la seguridad sin comprometer la operatividad del sistema (Pérez & Gómez, 2021; Martínez, 2019).

7.2.4. Justificación técnica de decisiones

La justificación técnica de decisiones en el diseño de sistemas implica argumentar de manera fundamentada la elección de determinadas estructuras de datos y enfoques computacionales en función de los requerimientos del problema. Este proceso no solo

considera aspectos teóricos, sino también criterios prácticos como el rendimiento, la escalabilidad, la mantenibilidad y la adecuación al contexto específico de aplicación.

Seleccionar una estructura de datos no es una decisión arbitraria, sino el resultado de un análisis que evalúa cómo se comportará el sistema ante distintos escenarios (Sánchez, 2020). Por ejemplo, el uso de grafos para la detección de ataques en ciberseguridad se justifica debido a su capacidad para representar relaciones complejas entre múltiples entidades, como usuarios, dispositivos y eventos, lo que permite identificar patrones de comportamiento malicioso que no serían evidentes con estructuras más simples. Esta capacidad resulta clave en entornos donde las amenazas son dinámicas y sofisticadas (Pérez & Gómez, 2021).

Asimismo, una adecuada justificación técnica permite respaldar decisiones relacionadas con la arquitectura de seguridad, facilitando la implementación de soluciones robustas y eficientes. Así se garantiza que la infraestructura tecnológica no solo sea funcional, sino también resiliente frente a posibles incidentes, optimizando recursos y mejorando la capacidad de respuesta ante amenazas (Martínez, 2019).

7.2.5. Diseño estructural del sistema final

El diseño estructural del sistema que los estudiantes deben desarrollar en su proyecto final debe centrarse en construir una solución funcional para un escenario realista de ciberseguridad, integrando de forma coherente estructuras de datos y algoritmos. El sistema debe organizarse en módulos que permitan gestionar, procesar y analizar la información, garantizando escalabilidad y mantenibilidad.

Es importante incorporar algoritmos de búsqueda para localizar datos relevantes y algoritmos de ordenamiento para optimizar el manejo de grandes volúmenes de información, como logs de seguridad. Además, los árboles pueden utilizarse para estructurar jerarquías, mientras que los grafos permiten representar redes y analizar posibles ataques.

El diseño debe considerar eficiencia computacional y buenas prácticas de ingeniería de software, justificando técnicamente cada decisión tomada.

Definición de los términos citados en la Semana 7

Nodo crítico. Un nodo crítico es un elemento dentro de una red o sistema cuya importancia radica en que su falla o compromiso puede afectar significativamente el funcionamiento general de la infraestructura, impactando la disponibilidad, integridad o confidencialidad de la información. Estos nodos suelen identificarse mediante métricas de análisis de grafos, como la centralidad, que permite determinar su relevancia dentro de la red (Sánchez, 2020).

Punto único de falla. Un punto único de falla es un componente de un sistema cuya interrupción provoca la caída total o parcial del servicio, debido a que no existe redundancia que permita mantener la operación. Este tipo de vulnerabilidad representa un alto riesgo en entornos tecnológicos, ya que concentra la dependencia en un solo elemento crítico (Pérez & Gómez, 2021).

Educación de calidad

REFERENCIAS

Cormen, T. H., Leiserson, C. E., Rivest, R. L., & Stein, C. (2016). Introducción a los algoritmos (3.ª ed.). MIT Press / McGraw-Hill Interamericana.

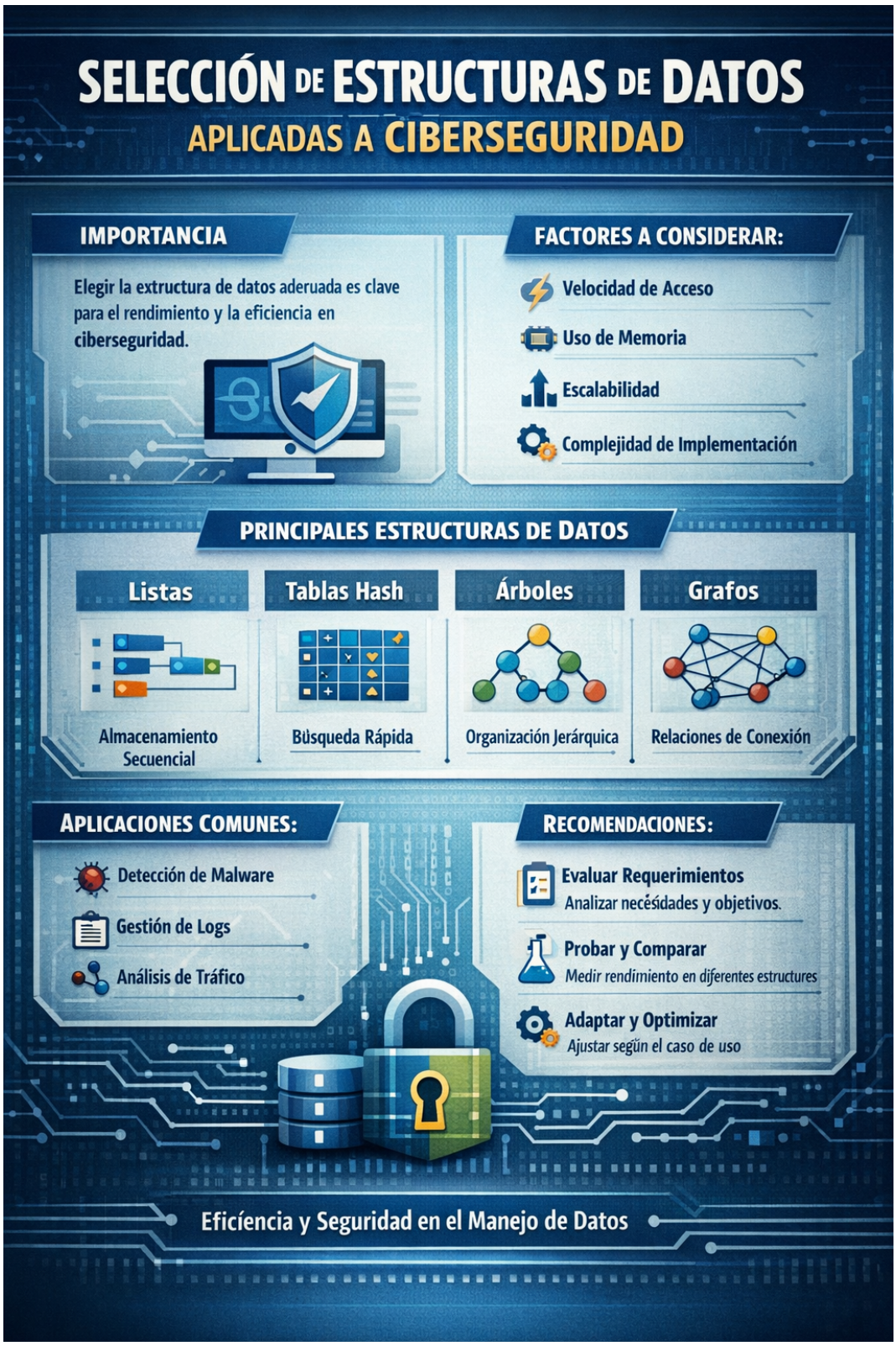
Martínez, L. (2019). Fundamentos de redes informáticas y seguridad. Editorial Alfaomega.

Pérez, J., & Gómez, R. (2021). Ciberseguridad: análisis y protección de infraestructuras digitales. Editorial Ra-Ma.

Sánchez, M. (2020). Estructuras de datos y algoritmos aplicados a redes. Editorial Paraninfo.

Puntos únicos de falla en Infraestructura Tecnológica. Infografía con el concepto, impactos, características, tipos y ejemplos en ciberseguridad.

Selección de estructuras de datos aplicadas a la ciberseguridad. Infografía con información sobre la importancia, factores a considerar, las principales estructuras, aplicaciones y recomendaciones.



Laboratorio de Contenido 1: Modelado de Infraestructuras de Red

Explicación básica

El modelado de infraestructuras de red consiste en representar los distintos componentes de una red informática (como routers, servidores, dispositivos y conexiones) mediante estructuras de datos, especialmente grafos. En este modelo, los nodos representan los dispositivos o activos, mientras que las aristas representan las conexiones o flujos de comunicación. Este enfoque permite analizar la topología de la red, identificar vulnerabilidades y comprender cómo se propaga la información o posibles amenazas dentro del sistema.

Objetivos

- Comprender cómo se modela una red utilizando grafos.
- Identificar los elementos clave de una infraestructura tecnológica.
- Analizar relaciones y dependencias dentro de una red.
- Aplicar el modelo en un escenario de ciberseguridad.

Caso práctico

Una empresa desea analizar su red interna para identificar posibles rutas de ataque y mejorar su seguridad.

Problema

Se cuenta con los siguientes elementos:

- Dispositivos: PC1, PC2, Servidor, Router, Firewall
- Conexiones entre ellos

Se requiere:

- Representar la red.
- Identificar conexiones críticas.
- Analizar posibles rutas de acceso no autorizado.

Aplicación paso a paso

Paso 1: Definir nodos y conexiones

Identificar los dispositivos y cómo están conectados.

Paso 2: Modelar con grafo en Python

```
import networkx as nx

G = nx.Graph()

# Nodos
dispositivos = ["PC1", "PC2", "Servidor", "Router", "Firewall"]
G.add_nodes_from(dispositivos)

# Conexiones
G.add_edges_from([
```

```
    ("PC1", "Router"),
    ("PC2", "Router"),
    ("Router", "Firewall"),
    ("Firewall", "Servidor")
])

print("Conexiones:", G.edges())
```

Paso 3: Analizar la red

- Identificar nodos con más conexiones.
- Detectar posibles puntos críticos.

```
print(nx.degree_centrality(G))
```

Conclusiones

- El modelado de infraestructuras de red mediante grafos permite representar de forma clara y estructurada las relaciones entre los distintos componentes de un sistema.
- En ciberseguridad, este enfoque facilita la identificación de vulnerabilidades, rutas de ataque y nodos críticos, mejorando la toma de decisiones para proteger la infraestructura. Además, permite simular escenarios y evaluar el impacto de posibles incidentes antes de que ocurran.

Laboratorio de Contenido 2: Comparación entre Listas, Árboles y Grafos

Explicación básica

Las estructuras de datos como listas, árboles y grafos permiten organizar y gestionar información de distintas formas según la complejidad del problema. Las listas son estructuras lineales que almacenan datos de forma secuencial, los árboles organizan la información de manera jerárquica y los grafos permiten representar relaciones complejas entre múltiples entidades. En ciberseguridad, esta diferencia es clave, ya que los datos pueden variar desde simples registros hasta redes altamente interconectadas.

Objetivos

- Comprender las diferencias entre listas, árboles y grafos.
- Identificar ventajas y limitaciones de cada estructura.
- Aplicar cada estructura en un contexto de ciberseguridad.
- Justificar la selección de una estructura según el problema

Caso práctico

Una organización desea implementar un sistema que gestione eventos de seguridad, controle accesos y analice relaciones entre dispositivos para detectar ataques.

Problema

Se requiere:

- Almacenar logs de eventos.
- Gestionar jerarquías de usuarios y permisos.
- Analizar conexiones entre dispositivos para detectar amenazas.

¿Qué estructura de datos utilizar en cada caso?

Aplicación paso a paso

Paso 1: Identificar necesidades

- Datos secuenciales → logs
- Jerarquía → usuarios
- Relaciones complejas → red

Paso 2: Seleccionar estructuras

- Lista → almacenamiento de logs
- Árbol → estructura de permisos
- Grafo → análisis de red

Paso 3: Implementación básica en Python

```
import networkx as nx

# Lista (logs)
logs = ["login", "logout", "error"]
```

```
# Árbol (jerarquía simple)
arbol = {"admin": ["user1", "user2"]}

# Grafo (red)
G = nx.Graph()
G.add_edge("PC1", "Servidor")
G.add_edge("Servidor", "Router")

print("Logs:", logs)
print("Árbol:", arbol)
print("Grafo:", G.edges())
```

Conclusiones

- Cada estructura de datos cumple un propósito específico y su correcta elección depende del problema a resolver.
- Las listas son útiles para datos simples y secuenciales, los árboles permiten organizar información jerárquica de manera eficiente y los grafos son ideales para modelar relaciones complejas.
- En ciberseguridad, la combinación de estas estructuras permite construir sistemas más completos, eficientes y capaces de analizar distintos tipos de información, mejorando la detección y prevención de amenazas.

Laboratorio de Contenido 3: Trade-offs de Diseño

Explicación básica

Los *trade-offs de diseño* se refieren a los compromisos que se deben asumir al seleccionar estructuras de datos y algoritmos, ya que no es posible optimizar simultáneamente todos los factores, como rendimiento, uso de memoria y complejidad. En ciberseguridad, estos compromisos son especialmente relevantes, ya que los sistemas deben procesar grandes volúmenes de datos en tiempo real sin sacrificar precisión en la detección de amenazas. Por ejemplo, una solución muy rápida puede consumir más memoria, mientras que una más simple puede ser menos eficiente ante grandes volúmenes de datos.

Objetivos

- Comprender el concepto de trade-offs en diseño de sistemas.
- Analizar el impacto del rendimiento, memoria y complejidad.
- Aplicar criterios de selección en un escenario de ciberseguridad.
- Evaluar diferentes soluciones y justificar decisiones técnicas

Caso práctico

Una empresa necesita analizar registros de acceso (logs) para detectar intentos de intrusión en tiempo real. Se debe elegir la mejor estructura de datos para almacenar y procesar esta información.

Problema

Se tienen miles de eventos por segundo y se requiere:

- Búsqueda rápida de patrones sospechosos.
- Bajo consumo de memoria.
- Respuesta en tiempo real.

¿Qué estructura de datos elegir y por qué?

Aplicación paso a paso

Paso 1: Identificar requerimientos

- Alta velocidad → prioridad en rendimiento
- Gran volumen → considerar memoria
- Análisis complejo → flexibilidad

Paso 2: Evaluar opciones

- Lista → rápida pero búsqueda lenta
- Tabla hash → rápida en búsquedas
- Grafo → potente pero costoso

Paso 3: Selección

Se elige tabla hash para búsquedas rápidas de eventos sospechosos.

Paso 4: Ejemplo en Python

```
# Simulación de detección de intentos de acceso
logs = ["login_ok", "login_fail", "login_fail", "login_ok"]

contador = {}

for evento in logs:
    if evento in contador:
        contador[evento] += 1
    else:
        contador[evento] = 1

print(contador)
```

Conclusiones

- Los trade-offs de diseño son fundamentales en la construcción de sistemas de ciberseguridad, ya que obligan a equilibrar rendimiento, memoria y complejidad.
- No existe una solución perfecta, sino la más adecuada según el contexto.
- En este laboratorio se evidenció que estructuras como las tablas hash ofrecen un buen balance para análisis en tiempo real, aunque en escenarios más complejos podrían requerirse grafos u otras estructuras más avanzadas.

Estudio de caso 1: Selección de Estructuras de Datos

Contexto

En los sistemas de ciberseguridad actuales, se procesan grandes volúmenes de datos provenientes de logs, eventos de red y actividades de usuarios. La eficiencia en el manejo de esta información depende en gran medida de la correcta selección de estructuras de datos, ya que estas influyen directamente en el rendimiento, la velocidad de respuesta y el uso de recursos del sistema.

Problema

Una organización necesita desarrollar un sistema que permita analizar eventos de seguridad en tiempo real para detectar intentos de acceso no autorizado. El sistema debe procesar miles de registros por segundo, permitiendo búsquedas rápidas, bajo consumo de memoria y escalabilidad.

Solución Propuesta

Seleccionar estructuras de datos que optimicen el acceso y procesamiento de la información. En este caso, se propone el uso de tablas hash para búsquedas rápidas y listas para almacenamiento temporal de eventos.

Estrategia:

- Analizar los requerimientos del sistema (velocidad, volumen de datos).
- Evaluar distintas estructuras (listas, árboles, grafos, hash).
- Seleccionar la combinación que mejor equilibre rendimiento y consumo de recursos.

Implementación

```
# Simulación de análisis de eventos de seguridad
logs = ["login_ok", "login_fail", "login_fail", "intrusion", "login_ok"]

contador = {}

for evento in logs:
    if evento in contador:
        contador[evento] += 1
    else:
        contador[evento] = 1

# Identificar eventos sospechosos
for evento, cantidad in contador.items():
    if evento == "login_fail" and cantidad > 1:
        print("Alerta: múltiples intentos fallidos")
```

Resultados e Impacto en ciberseguridad

El uso de tablas hash permite una búsqueda eficiente en tiempo constante, mejorando la detección de patrones sospechosos en grandes volúmenes de datos. Esto reduce el tiempo de respuesta ante amenazas y mejora la capacidad de monitoreo en tiempo real.

Reflexión Técnica

La selección de estructuras de datos no debe basarse únicamente en la facilidad de implementación, sino en un análisis técnico que considere rendimiento, escalabilidad y contexto. En ciberseguridad, una elección adecuada puede marcar la diferencia entre detectar una amenaza a tiempo o no, evidenciando la importancia de tomar decisiones fundamentadas en el diseño del sistema.

ESTUDIO DE CASO 2: Comparación entre Listas, Árboles y Grafos

Contexto

En ciberseguridad, los sistemas deben manejar distintos tipos de datos, desde registros simples hasta relaciones complejas entre dispositivos y usuarios. Para ello, es fundamental seleccionar estructuras de datos adecuadas como listas, árboles y grafos, ya que cada una permite representar la información de manera diferente y optimizar ciertos procesos según el contexto.

Problema

Una empresa necesita desarrollar un sistema de monitoreo que permita: almacenar eventos de seguridad, gestionar jerarquías de usuarios y analizar conexiones entre dispositivos para detectar posibles ataques. El reto es determinar qué estructura de datos utilizar en cada caso.

Solución Propuesta

Utilizar una combinación de estructuras: listas para almacenar logs, árboles para representar jerarquías de acceso y grafos para modelar la red y detectar relaciones entre dispositivos.

Estrategia:

- Identificar el tipo de datos a manejar.
- Clasificar los requerimientos según su complejidad.
- Asignar la estructura más adecuada para cada necesidad

Implementación

```
import networkx as nx

# Lista (logs)
logs = ["login", "error", "logout"]

# Árbol (jerarquía)
usuarios = {"admin": ["user1", "user2"]}

# Grafo (red)
G = nx.Graph()
G.add_edge("PC1", "Servidor")
G.add_edge("Servidor", "Router")

print("Logs:", logs)
print("Usuarios:", usuarios)
```

```
print("Conexiones:", list(G.edges()))
# Grafo (red)
G = nx.Graph()
G.add_edge("PC1", "Servidor")
G.add_edge("Servidor", "Router")

print("Logs:", logs)
print("Usuarios:", usuarios)
print("Conexiones:", list(G.edges()))
```

Resultados e Impacto en Ciberseguridad:

El uso combinado de estas estructuras permite gestionar eficientemente distintos tipos de información. Las listas facilitan el almacenamiento, los árboles organizan accesos y los grafos permiten analizar relaciones complejas, mejorando la detección de amenazas.

Reflexión Técnica

No existe una única estructura ideal para todos los problemas. La combinación de listas, árboles y grafos permite construir sistemas más robustos y eficientes, evidenciando que una correcta selección impacta directamente en el rendimiento y la capacidad de respuesta en ciberseguridad.

PUCE Virtual

síguenos en:

www.pucevirtual.puce.edu.ec