

DOCUMENTO DE PROFUNDIZACIÓN

Programación III

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.