

Resumen Tema 3

Monitorización de Servicios y Aplicaciones

Autor: @BlackTyson

Resumen teórico con ejercicios y explicaciones completas.

Índice

1. Introducción a la Monitorización	2
1.1. Definición	2
1.2. Importancia de la Monitorización	2
1.3. Objetivos Principales	2
2. Conceptos Fundamentales	2
2.1. Carga y Actividad	2
2.2. Monitores de Actividad	2
2.3. Clasificación de Monitores	2
3. Atributos Clave de los Monitores	3
4. Herramientas de Monitorización	3
4.1. <code>uptime</code>	3
4.2. <code>vmstat</code>	3
4.3. <code>sar</code>	3
5. Técnicas Avanzadas	3
5.1. Ley de Amdahl	3
5.2. Método USE (Utilización, Saturación, Errores)	4
6. Ejercicios Prácticos	4
7. Conclusión	4

1. Introducción a la Monitorización

1.1. Definición

La **monitorización** es el proceso de observar, analizar y optimizar el rendimiento de sistemas informáticos, evaluando variables clave como uso de CPU, memoria, red y almacenamiento. Su propósito es garantizar disponibilidad, eficiencia y escalabilidad.

1.2. Importancia de la Monitorización

- **Proactiva:** Detecta problemas antes de que afecten al sistema.
- **Predictiva:** Ayuda en la planificación de recursos y escalado.
- **Reactiva:** Permite soluciones rápidas ante incidentes.

1.3. Objetivos Principales

- Garantizar la **disponibilidad** de servicios.
- Optimizar el **rendimiento** de los recursos.
- Proporcionar datos para **auditorías** y cumplimiento normativo.

2. Conceptos Fundamentales

2.1. Carga y Actividad

- **Carga (workload):** Conjunto de tareas que demandan recursos.
- **Actividad:** Operaciones realizadas en respuesta a la carga.

2.2. Monitores de Actividad

Un **monitor de actividad** mide variables clave para el análisis del rendimiento. Sus acciones principales incluyen:

- Medir variables como uso de CPU o memoria.
- Procesar y almacenar datos.
- Visualizar resultados para el análisis.

2.3. Clasificación de Monitores

1. Por el momento de medición:

- **Eventos:** Miden cambios de estado (precisos, no continuos).
- **Muestreo:** Recogen datos periódicamente (continuos, estadísticos).

2. Por naturaleza:

- **Software:** Herramientas como `top` o `vmstat`.
- **Hardware:** Dispositivos físicos como sensores de red.

3. Por interacción:

- **Primer plano:** Tiempo real (e.g., gráficos en `htop`).
- **Segundo plano:** Recolección de datos para análisis posterior.

3. Atributos Clave de los Monitores

- **Sobrecarga:** Recursos consumidos por el monitor.
- **Exactitud:** Cercanía al valor real.
- **Precisión:** Consistencia en mediciones repetidas.
- **Resolución:** Cambio mínimo detectable en la medición.

4. Herramientas de Monitorización

4.1. uptime

Proporciona tiempo de actividad del sistema y carga promedio en los últimos 1, 5 y 15 minutos. Ejemplo:

```
1:21pm up 1 day, 4:09, 18 users, load average: 1.04, 0.30, 0.09
```

4.2. vmstat

Monitorea estadísticas clave de procesos, memoria, swap, I/O y CPU:

- **Memoria:** Libre, usada, caché.
- **CPU:** Uso de usuario (`%us`), sistema (`%sy`), inactividad (`%id`).

4.3. sar

Recopila estadísticas de actividad del sistema, mostrando parámetros como:

- Transferencias por segundo (`tps`).
- Bloques leídos/escritos por segundo (`bread/s`, `bwrtn/s`).

5. Técnicas Avanzadas

5.1. Ley de Amdahl

Evalúa el impacto de optimizaciones parciales en el rendimiento global:

$$S = \frac{1}{(1 - f) + \frac{f}{k}},$$

donde f es la fracción optimizable y k el factor de mejora.

5.2. Método USE (Utilización, Saturación, Errores)

- **Utilización:** Porcentaje de tiempo que el recurso está ocupado.
- **Saturación:** Tareas en cola esperando usar el recurso.
- **Errores:** Fallos relacionados con el recurso.

6. Ejercicios Prácticos

Ejercicio 1: Sobrecarga de un Monitor

Un monitor consume 10 ms de CPU por activación y se ejecuta cada 1 s. Calcule la sobrecarga:

$$\text{Sobrecarga (\%)} = \frac{\text{Recursos utilizados}}{\text{Capacidad total}} \times 100 = \frac{10 \text{ ms}}{1000 \text{ ms}} \times 100 = 1 \text{ \%}.$$

Ejercicio 2: Análisis de top

Dado el resultado:

$$\%Cpu(s) : 40,0 \%us, 10,0 \%sy, 5,0 \%wa, 45,0 \%id$$

Determine:

1. Tiempo dedicado a operaciones de I/O: 5 %.
2. Si el sistema está balanceado: Sí, con carga moderada.

Ejercicio 3: Optimización de Código

Un sistema tiene un tiempo de ejecución de 100 s, con 70 % optimizable y un factor de mejora $k = 3$. Calcule:

$$S = \frac{1}{(1 - 0,7) + \frac{0,7}{3}} = 1,82.$$

7. Conclusión

Este resumen integra teoría y práctica, proporcionando una visión integral del tema de monitorización de servicios y aplicaciones.