DevOps Shack

System Monitoring in Linux

[Click Here To Enrol To Batch-6 | DevOps & Cloud DevOps]

System monitoring in Linux involves tracking and analyzing the performance and health of the system, including CPU usage, memory usage, disk I/O, network activity, and running processes. Several tools are available in Linux to monitor these aspects, ranging from command-line utilities to graphical applications.

Key System Monitoring Tools

1. top

top provides a real-time, dynamic view of the system's processes, including CPU and memory usage.

Usage:

top

Key Information:

- **PID:** Process ID
- **USER:** User that owns the process
- **PR:** Priority of the process
- **NI:** Nice value of the process
- **VIRT:** Virtual memory used by the process
- **RES:** Resident memory used by the process
- **SHR:** Shared memory used by the process
- **S:** Process status (R: running, S: sleeping, Z: zombie)
- **%CPU:** CPU usage
- **%MEM:** Memory usage

- **TIME+:** Total CPU time used by the process
- **COMMAND:** Command name or command line

2. htop

htop is an enhanced version of top with a more user-friendly interface and additional features.

Usage:

htop

Features:

- Color-coded display
- Easy process management (kill processes directly)
- Detailed system overview

Installation:

sudo apt install htop # For Debian/Ubuntu

sudo yum install htop # For CentOS/RHEL

3. vmstat

vmstat reports information about processes, memory, paging, block I/O, traps, and CPU activity.

Usage:

vmstat 1 5

Run vmstat with 1-second intervals for 5 iterations

Key Information:

- **procs:** Number of processes (r: runnable, b: blocked)
- **memory:** Memory usage (swpd: swapped, free: free memory, buff: buffers, cache: cache)
- **swap:** Swap activity (si: swap in, so: swap out)
- **io:** I/O activity (bi: blocks received, bo: blocks sent)
- **system:** System statistics (in: interrupts per second, cs: context switches per second)
- **cpu:** CPU usage (us: user, sy: system, id: idle, wa: wait)

4. iostat

iostat reports CPU and I/O statistics.

Usage:

iostat -x 1 5

Extended statistics with 1-second intervals for 5 iterations

Key Information:

- **tps:** Transactions per second
- **kB_read/s:** Kilobytes read per second
- **kB_wrtn/s:** Kilobytes written per second
- **await:** Average wait time for I/O requests

5. free

free displays the amount of free and used memory in the system.

Usage:

free -h

Display memory usage in human-readable format

Key Information:

- **total:** Total memory
- **used:** Used memory
- **free:** Free memory
- **shared:** Shared memory
- **buff/cache:** Buffers/cache
- **available:** Available memory

6. df

df reports the amount of disk space used and available on file systems.

Usage:

df -h # Display disk usage in human-readable format

Key Information:

- **Filesystem:** Name of the filesystem
- **Size:** Total size of the filesystem
- **Used:** Used space
- **Avail:** Available space
- **Use%:** Percentage of used space
- **Mounted on:** Mount point

7. du

du estimates file space usage.

Usage:

du -sh /path/to/directory # Display summary of disk usage for a directory

Key Information:

- **Size:** Size of the directory or file
- **File/Directory:** Name of the file or directory

8. netstat

netstat displays network connections, routing tables, interface statistics, masquerade connections, and multicast memberships.

Usage:

netstat -tuln # Display all listening TCP and UDP ports

Key Information:

- **Proto:** Protocol (TCP, UDP)
- **Recv-Q:** Receive queue size
- **Send-Q:** Send queue size
- **Local Address:** Local address and port number
- **Foreign Address:** Foreign address and port number
- **State:** Connection state (LISTEN, ESTABLISHED)

9. iftop

iftop displays bandwidth usage on an interface by host.

Usage:

sudo iftop -i eth0 # Monitor bandwidth usage on interface eth0

Features:

- Real-time bandwidth monitoring
- Source and destination IP addresses
- Bandwidth usage per connection

Installation:

sudo apt install iftop # For Debian/Ubuntu

sudo yum install iftop # For CentOS/RHEL

10. nmon

nmon (Nigel's Monitor) is a performance monitoring tool for CPU, memory, disks, network, NFS, processes, and resources.

Usage:

nmon

Features:

- Interactive monitoring
- Graphical display of various metrics
- Export data for further analysis

Installation:

sudo apt install nmon # For Debian/Ubuntu

sudo yum install nmon # For CentOS/RHEL

Example Scenarios

Scenario 1: Monitoring CPU and Memory Usage

1. **Using top to monitor CPU and memory usage:**

top

    - Press q to quit.

2. **Using htop for a more interactive view:**

htop

   - Press F6 to sort by different metrics.
   - Use arrow keys to navigate and F9 to kill a process.
   - Press q to quit.

3. **Using vmstat to get a quick overview:**

vmstat 1 5

   - This command provides a snapshot of system performance every second for 5 iterations.

Scenario 2: Monitoring Disk Usage

1. **Using df to check disk usage:**

df -h

   - This command displays disk usage in a human-readable format.

2. **Using du to check directory size:**

du -sh /var/log

   - This command displays the total size of the /var/log directory.

Scenario 3: Monitoring Network Activity

1. **Using netstat to check listening ports:**

netstat -tuln

   - This command displays all listening TCP and UDP ports.

2. **Using iftop to monitor bandwidth usage:**

sudo iftop -i eth0

   - This command monitors bandwidth usage on the eth0 interface.

Scenario 4: Comprehensive System Monitoring with nmon

1. **Starting nmon:**

nmon

   - Use keys c for CPU, m for memory, d for disks, n for network, and other keys as displayed to monitor different aspects of the system.
   - Press q to quit.

Summary

Linux provides a variety of tools for system monitoring, each suited for different aspects of system performance and health. Understanding and using these tools effectively helps in maintaining a well-performing and stable system. Regular monitoring can help detect issues early, optimize resource usage, and ensure smooth operation.