

Управление и анализ производительности системы (`sysstat`, `sar`)

Добро пожаловать в подробное руководство по управлению и анализу производительности системы в НАЙС ОС. В данном документе мы рассмотрим использование пакета `sysstat` и команды `sar` для мониторинга и анализа производительности системы. Эти инструменты являются важными для системных администраторов, так как позволяют эффективно отслеживать и улучшать работу системы.

Введение в `sysstat`

`sysstat` — это набор утилит для мониторинга и анализа производительности системы в Unix-подобных операционных системах. В его состав входят такие инструменты, как `sar`, `iostat`, `mpstat` и другие. Эти утилиты позволяют собирать и анализировать данные о производительности процессора, дисков, памяти и сети.

Установка `sysstat`

Для установки пакета `sysstat` в НАЙС ОС используйте команду:

```
tdnf install sysstat
```

Основные утилиты `sysstat`

В составе пакета `sysstat` имеется несколько полезных утилит для мониторинга различных аспектов системы. Рассмотрим основные из них.

`sar`

`sar` (System Activity Reporter) — это утилита для сбора, отчета и сохранения данных о производительности системы. Она предоставляет информацию о процессоре, памяти, дисках, сети и других подсистемах.

Сбор данных с `sar`

Для запуска сбора данных с помощью `sar` используйте команду:

```
sar -o /var/log/sa/sar_data 1 3
```

Эта команда собирает данные каждые 1 секунду в течение 3 секунд и сохраняет их в файл `/var/log/sa/sar_data`.

Отчет по данным `sar`

Для просмотра отчета по собранным данным используйте команду:

```
sar -f /var/log/sa/sar_data
```

iostat

iostat — это утилита для мониторинга производительности дисковых устройств и системы ввода-вывода (I/O). Она предоставляет информацию о загрузке процессора и статистике ввода-вывода для дисков и других устройств.

Запуск **iostat**

Для запуска **iostat** используйте команду:

```
iostat
```

Команда **iostat** без параметров выводит общую информацию о процессорах и дисковых устройствах.

Примеры использования **iostat**

Для отображения информации о процессорах:

```
iostat -c
```

Для отображения информации о дисках:

```
iostat -d
```

Для отображения расширенной информации о дисках:

```
iostat -x
```

Для отображения информации о дисках с временными метками и выводом в килобайтах:

```
iostat -dk -t
```

mpstat

mpstat — это утилита для мониторинга загрузки процессора. Она предоставляет информацию о использовании процессора на уровне системы и отдельных процессоров.

Запуск **mpstat**

Для запуска **mpstat** используйте команду:

```
mpstat
```

Команда **mpstat** без параметров выводит информацию о загрузке процессора с момента

последней перезагрузки.

Примеры использования `mpstat`

Для отображения статистики загрузки процессора каждые 2 секунды:

```
mpstat 2
```

Для отображения статистики по каждому процессору отдельно:

```
mpstat -P ALL 2
```

Сбор и анализ данных с `sar`

`sar` является одной из наиболее мощных утилит пакета `sysstat`, предоставляя обширные возможности для сбора и анализа данных о производительности системы.

Сбор данных с `sar`

Для автоматического сбора данных с `sar` можно использовать `cron`. Рассмотрим пример настройки задания `cron` для регулярного сбора данных.

Настройка `cron` для `sar`

Откройте `crontab` для редактирования:

```
crontab -e
```

Добавьте следующую строку для сбора данных каждые 10 минут:

```
*/10 * * * * /usr/lib64/sa/sa1 1 1
```

Эта команда запускает `sa1`, скрипт для сбора данных `sar`, каждые 10 минут.

Отчет по данным `sar`

Для просмотра отчета по собранным данным используйте команду:

```
sar -A
```

Эта команда выводит полный отчет по всем доступным метрикам.

Анализ данных `sar`

`sar` предоставляет множество опций для анализа различных аспектов системы. Рассмотрим основные опции.

Процессор

Для анализа использования процессора используйте опцию `-u`:

```
sar -u
```

Эта команда выводит статистику по использованию процессора, включая процент времени, затраченного на выполнение пользовательских процессов, системных процессов, и простаивание.

Память

Для анализа использования памяти используйте опцию `-r`:

```
sar -r
```

Эта команда выводит информацию о используемой и свободной памяти, а также о swap.

Диски

Для анализа производительности дисков используйте опцию `-d`:

```
sar -d
```

Эта команда выводит статистику по чтению и записи на дисковые устройства.

Сеть

Для анализа производительности сети используйте опцию `-n`:

```
sar -n DEV
```

Эта команда выводит информацию о трафике на сетевых интерфейсах.

Примеры использования *sar*

Рассмотрим несколько примеров использования *sar* для сбора и анализа данных о производительности системы.

Сбор данных каждые 10 секунд

Для сбора данных каждые 10 секунд в течение 1 минуты используйте команду:

```
sar -o /var/log/sa/sar_data 10 6
```

Эта команда собирает данные каждые 10 секунд и сохраняет их в файл `/var/log/sa/sar_data`.

Анализ использования процессора

Для анализа использования процессора используйте команду:

```
sar -u -f /var/log/sa/sar_data
```

Эта команда выводит статистику по использованию процессора из файла `/var/log/sa/sar_data`.

Анализ использования памяти

Для анализа использования памяти используйте команду:

```
sar -r -f /var/log/sa/sar_data
```

Эта команда выводит статистику по использованию памяти из файла `/var/log/sa/sar_data`.

Анализ использования дисков

Для анализа использования дисков используйте команду:

```
sar -d -f /var/log/sa/sar_data
```

Эта команда выводит статистику по использованию дисков из файла `/var/log/sa/sar_data`.

Анализ использования сети

Для анализа использования сети используйте команду:

```
sar -n DEV -f /var/log/sa/sar_data
```

Эта команда выводит статистику по использованию сети из файла `/var/log/sa/sar_data`.

Дополнительные утилиты пакета sysstat

Кроме `sar`, в пакет `sysstat` входят другие полезные утилиты для мониторинга производительности системы.

[pidstat](#)

`pidstat` — это утилита для мониторинга производительности процессов. Она предоставляет информацию о загрузке процессора, памяти и I/O на уровне отдельных процессов.

Запуск `pidstat`

Для запуска `pidstat` используйте команду:

pidstat

Команда `pidstat` без параметров выводит информацию о загрузке процессора процессами.

Примеры использования `pidstat`

Для отображения информации о загрузке процессора процессами каждые 2 секунды:

```
pidstat 2
```

Для отображения информации о загрузке памяти процессами:

```
pidstat -r
```

Для отображения информации о I/O процессами:

```
pidstat -d
```

nfsiostat

`nfsiostat` — это утилита для мониторинга производительности NFS (Network File System). Она предоставляет информацию о операциях чтения и записи на NFS серверах.

Запуск `nfsiostat`

Для запуска `nfsiostat` используйте команду:

```
nfsiostat
```

Команда `nfsiostat` без параметров выводит информацию о операциях чтения и записи на NFS серверах.

Примеры использования `nfsiostat`

Для отображения статистики NFS сервера каждые 2 секунды:

```
nfsiostat 2
```

cifsiostat

`cifsiostat` — это утилита для мониторинга производительности CIFS (Common Internet File System). Она предоставляет информацию о операциях чтения и записи на CIFS серверах.

Запуск `cifsiostat`

Для запуска `cifsiostat` используйте команду:

cifsiostat

Команда **cifsiostat** без параметров выводит информацию о операциях чтения и записи на CIFS серверах.

Примеры использования **cifsiostat**

Для отображения статистики CIFS сервера каждые 2 секунды:

```
cifsiostat 2
```

Использование графиков для анализа производительности

Для более наглядного анализа производительности системы можно использовать графики. Рассмотрим пример создания графиков на основе данных **sar** с помощью утилиты **ksar**.

Установка ksar

Для установки **ksar** загрузите последнюю версию с официального сайта и распакуйте архив:

```
wget https://sourceforge.net/projects/ksar/files/latest/download -O ksar.zip
```

```
unzip ksar.zip -d /opt/ksar
```

Создание графиков с ksar

Для создания графиков откройте **ksar** и загрузите файл данных **sar**:

```
java -jar /opt/ksar/ksar.jar
```

В интерфейсе **ksar** выберите **Data -> Load from file** и укажите путь к файлу данных **sar**.

Заключение

Мы рассмотрели основные аспекты управления и анализа производительности системы в НАЙС ОС с использованием пакета **sysstat** и команды **sar**. Эти инструменты позволяют эффективно собирать, анализировать и визуализировать данные о производительности системы, что является ключевым аспектом для поддержания ее стабильности и оптимальной работы. Продолжайте изучать и применять эти знания на практике для повышения надежности и производительности ваших систем.