Использование профилировщиков производительности (perf, eBPF)

Профилирование производительности — это важный процесс, который помогает выявлять узкие места в системе и оптимизировать ее работу. В операционной системе НАЙС ОС существуют мощные инструменты для профилирования, такие как `perf` и eBPF (extended Berkeley Packet Filter). В этом руководстве мы рассмотрим, как использовать эти инструменты для анализа и оптимизации производительности.

Использование `perf` для профилирования производительности

Инструмент `perf` предоставляет широкий набор возможностей для мониторинга и анализа производительности системы. Он позволяет собирать данные о работе процессора, памяти, ввода/вывода и многого другого.

Установка `perf`

Для установки `perf` используйте команду:

```
sudo tdnf install perf
```

Основные команды `perf`

Рассмотрим основные команды `perf`, которые помогут в профилировании системы.

Запуск `perf top`

Команда `perf top` позволяет в реальном времени мониторить загруженность системы:

```
sudo perf top
```

Сбор данных с помощью `perf record`

Команда `perf record` собирает данные о производительности системы, которые можно проанализировать позднее:

```
sudo perf record -a -g
```

Параметры:

© 2026 NiceSOFT — z.niceos.ru

- `-a` — сбор данных со всех процессоров.
- `-g` — сбор данных о графе вызовов.

Анализ данных с помощью `perf report`

Команда `perf report` позволяет проанализировать собранные данные:

```
sudo perf report
```

Профилирование конкретного процесса

Для профилирования конкретного процесса используйте его идентификатор (PID):

```
sudo perf record -p PID
```

Анализ использования CPU

Команда `perf stat` отображает статистику использования CPU:

```
sudo perf stat -a
```

Использование eBPF для профилирования производительности

eBPF — это мощный инструмент для анализа производительности и безопасности, который позволяет загружать и выполнять программы в ядре Linux. eBPF используется для мониторинга сетевого трафика, трассировки системных вызовов и профилирования производительности.

Установка eBPF инструментов

Для использования eBPF инструментов в НАЙС ОС можно установить `bcc` (BPF Compiler Collection) и `bpftrace`.

Установка bcc

```
sudo tdnf install bcc-tools
```

Установка bpftrace

```
sudo tdnf install bpftrace
```

Основные команды bcc

Рассмотрим основные команды bcc, которые помогают в профилировании системы.

Использование `execsnoop` для мониторинга выполнения процессов

```
sudo /usr/share/bcc/tools/execsnoop
```

Эта команда отслеживает запуск процессов в системе.

Использование `opensnoop` для мониторинга открытия файлов

```
sudo /usr/share/bcc/tools/opensnoop
```

Эта команда отслеживает системные вызовы `open()`, что позволяет мониторить доступ к файлам.

Использование `profile` для сбора профиля CPU

```
sudo /usr/share/bcc/tools/profile
```

Эта команда собирает данные о использовании CPU.

Основные команды bpftrace

bpftrace предоставляет более гибкие возможности для профилирования и анализа системы.

Мониторинг системных вызовов

Для мониторинга системных вызовов используйте следующую команду:

```
sudo bpftrace -e 'tracepoint:syscalls:sys_enter_* { @[probe] = count(); }'
```

Эта команда отслеживает все входящие системные вызовы и считает их количество.

Профилирование использования CPU

Для профилирования использования CPU используйте следующую команду:

```
sudo bpftrace -e 'profile:hz:99 { @[comm] = count(); }'
```

© 2026 NiceSOFT — z.niceos.ru

Эта команда собирает данные о использовании CPU каждые 10 миллисекунд.

Мониторинг задержек дисковой подсистемы

Для мониторинга задержек ввода/вывода используйте следующую команду:

```
sudo bpftrace -e 'tracepoint:block:block_rq_issue { @start[args->rq] = nsecs; }
tracepoint:block:block_rq_complete { @latency = hist(nsecs - @start[args->rq]);
}'
```

Эта команда отслеживает задержки выполнения операций ввода/вывода.

Анализ и оптимизация производительности

После сбора данных с помощью `perf` и eBPF, следующим шагом является анализ этих данных и оптимизация системы на основе полученных результатов.

Идентификация узких мест

Анализируйте собранные данные для выявления узких мест в системе, таких как высокое использование CPU, частые системные вызовы, задержки ввода/вывода и т.д.

Пример анализа с `perf report`

Используйте `perf report` для анализа собранных данных и выявления процессов, потребляющих наибольшее количество ресурсов:

```
sudo perf report
```

Оптимизация использования ресурсов

После идентификации узких мест применяйте различные методы оптимизации для улучшения производительности системы:

- Оптимизируйте код приложений для уменьшения нагрузки на процессор и память.
- Настройте параметры системы и ядра для улучшения распределения ресурсов.
- Используйте более производительное оборудование при необходимости.

Оптимизация использования CPU

Изменяйте приоритеты процессов, распределяйте нагрузку между процессорами и используйте многопоточность для улучшения производительности:

```
sudo renice -n -10 -p PID
```

© 2026 NiceSOFT — z.niceos.ru

Оптимизация использования памяти

Настройте параметры свопинга, используйте кэширование и оптимизируйте использование памяти приложениями:

```
sudo sysctl -w vm.swappiness=10
```

Оптимизация дисковой подсистемы

Используйте более производительные диски, настройте планировщики дисковых операций и применяйте технологии кэширования:

```
echo mq-deadline | sudo tee /sys/block/sdX/queue/scheduler
```

Оптимизация сетевых ресурсов

Настройте QoS для приоритизации важного трафика, оптимизируйте параметры TCP/IP и используйте высокоскоростные сетевые интерфейсы:

```
sudo sysctl -w net.ipv4.tcp_window_scaling=1
```

Использование профилировщиков производительности `perf` и eBPF позволяет эффективно выявлять и устранять узкие места в системе НАЙС ОС. Эти инструменты предоставляют мощные возможности для сбора и анализа данных о производительности системы, что помогает принимать обоснованные решения по оптимизации и улучшению работы системы. Регулярное профилирование и мониторинг позволяют поддерживать высокую производительность и стабильность системы.

© 2026 NiceSOFT — z.niceos.ru