**200.1 Измерение производительности и решение проблем связанных с ней**

Студент должен уметь измерять производительность железа, пропускную способность сети, определять и исправлять возникающие проблемы.

**Изучаем**:

* метрики ЦПУ;
* метрики ОЗУ;
* метрики HDD;

Вне зависимости от используемых инструментов снятия метрик, для определения производительности CPU используются следующие характеристики:

* **user**: время, затраченное на исполнения кода на уровне пользователя (чаще всего код софта, не использующий железо);
* **system**: время, затраченное на исполнение кода на уровне ядра (чаще всего это ОС – операции ввода вывода, управление процессами и тд);
* **nice**: время, затраченное на выполнение операций с измененным приоритетом;
* **idle**: время простаивания CPU;
* **wait**: время ожидания ввода-вывода (чаще всего ожидание ответа от дисковой подсистемы);
* **hardware\_interrupts**: время затраченное на обработку аппаратных прерываний (события от железа);
* **software\_interrupts**: время затраченное на обработку программных прерываний (события от программного кода);
* **stolen**: время ожидания ответа от гипервизора (в случае виртуальной машины).

Для определения базовых параметров производительности можно использовать утилиты ***vmstat***, ***top*** и модный ***htop***. Для тестовой загрузки процессора мы использовали пакет stress.

------------------------------

Вне зависимости от используемых инструментов снятия метрик, для определения производительности оперативной памяти используются следующие характеристики:

* **total**: всего ОЗУ;
* **free**: реально свободно ОЗУ (т.е. количество никак не задействованной памяти, Linux стремится уменьшить ее до минимума);
* **used**: используется ОЗУ;
* **shared**: память разделяемая процессами (чаще всего используется для взаимодействия между процессами во избежание использования системных вызовов ядра);
* **cached**: кэшированная ОЗУ (например, набор данных, к которым часто обращается программа, может быть помещен в кэш ОЗУ с жесткого диска, для более быстрого доступа к ним);
* **buffered**: буферизированная ОЗУ (например, промежуточное хранилище данных перед обработкой, или перемещением их на диск). Часто можно увидеть buffered/cached как единое целое, логически показывающее ту область памяти, которую можно освободить при необходимости.
* **available**: количество памяти которую можно использовать без необходимости обращаться к swap (т.е. память, которая будет свободна если выкинуть все из кэшей и т.д.);
* **active**: память, активно используемая процессом;
* **inactive**: память, которая была выделена под процесс, но в данный момент им больше не используется;
* **swapped**: в свопе (на жестком диске – в разделе или в файле);

Для раздела (или файла) подкачки могут быть использованы метрики:

* **swapped\_in**: количество блоков, считанных за секунду с диска;
* **swapped\_out**: количество блоков, записанных за секунду на диск;

Здесь и далее: размер блока указывается при форматировании файловой системы, чаще всего его размер равен 4Кб.

<https://ru.wikipedia.org/wiki/%D0%A1%D0%B5%D0%BA%D1%82%D0%BE%D1%80_%D0%B4%D0%B8%D1%81%D0%BA%D0%B0>

<https://toster.ru/q/19776>

Удобным инструментом для мониторинга состояния ОЗУ является ***free***. Например:

*free -m*

------------------------------

Вне зависимости от используемых инструментов снятия метрик, для определения производительности жестких дисков (если конкретно – блочных устройств) используются следующие характеристики:

* **blocks\_in**: количество считанных блоков в единицу времени;
* **blocks\_out**: количество записанных блоков в единицу времени;
* **total\_read/write**: скорость чтения/записи между процессами и подсистемой ввода/вывода;
* **actual\_read/write**: скорость чтения/записи между подсистемой ввода/вывода и самими дисками;
* **tid**: thread id (идентификатор потока процесса ввода/вывода, аналогичный pid для вычислительных процессов);
* **prio**: приоритет (аналогично nice для процессов). Бывает idle (минимальный), be/0-7 (best effort, чем меньше число тем выше приоритет) и rt/0-7 (real time, максимальный, чем меньше число тем выше приоритет)
* **swapin**: используемая часть swap;
* **tps**: количество запросов к диску в секунду;
* **rrqm/s**: количество запросов на чтение в секунду;
* **wrqm/s**: количество запросов на запись в секунду;
* **r/s**: количество операций чтения в секунду;
* **w/s**: количество операций записи в секунду;
* **rkB/s**: скорость чтения в килобайт/сек;
* **wkB/s**: скорость записи в килобайт/сек;
* **avgrq-sz**: средний размер запроса к диску в секторах;
* **avgqu-sz**: средняя очередь запросов к диску в секторах;
* **await**: среднее время обслуживания запроса (включая нахождение в очереди на обработку) в миллисекундах;
* **r\_await**: среднее время обслуживания запроса на чтение (включая нахождение в очереди на обработку) в миллисекундах;
* **w\_await**: среднее время обслуживания запроса на запись (включая нахождение в очереди на обработку) в миллисекундах;
* **svctm**: среднее время обслуживания запроса в миллисекундах;
* **util**: условно говоря – насколько был занят диск (по аналогии с утилизацией процессора);

Удобным инструментом для мониторинга состояния жестких дисков является ***iotop*** (ставится отдельным пакетом), ***iostat*** и ***sar*** (входят в состав пакета ***sysstat***). Например:

*iostat -x*

*sar -b 1 1*

*sar -d 1 1*