Dummynet - привязка процесса к CPU0

Опубликовано muff в Чт, 2014-11-27 02:43

Разбирался на днях с производительностью сервера, который работал в роли маршрутизатора.

Итак, что мы имеем в роли операционной системы:

```
# uname -rmo
FreeBSD 8.4-RELEASE-p7 amd64
```

Да, давно систему не обновляли... Ну и ладно.

В ходе диагностики обнаружил одну странность. Процесс **dummynet** потреблял слишком много ресурсов. Согласно выводу **top**:

```
  0 root      -68    0     0K   416K -       2 272:36 24.45% kernel{dummynet}
```

Небольшое лирическое отступление... **DUMMYNET** - это системное средство, которое дает возможность управления сетевым трафиком, проходящим через различные сетевые интерфейсы, путем применения имитации различной ширины полосы пропускания, ограничения размера очереди, имитации задержек и потерь пакетов.

На сетевых форумах уже стыкался с данной проблемой. Дело в том, что в 8-ой ветке **FreeBSD**, по умолчанию, системный тред **dummynet**, как и большинство других тредов, не привязан к конкретному **CPU**. Соответственно, планировщик перемещает его между ядрами процессора. А вот если "привязать" системный тред **dummynet** к **CPU0**, потребление ресурсов процессора тут же падает, при этом корректно выполняя все свои функции и задания.

Ну что ж, воспользуемся советом и привяжем процесс к **CPU0**. Но не тут то было. Поскольку процесс системный, его **PID** равен 0. Необходимо узнать **TID** процесса, и уже его "привязывать":

```
# procstat -at | grep dummynet     0 100098 kernel        dummynet        1   16 sleep   -
```

TID процесса в моем конкретном случае - 100098. Вот его и нужно привязать, воспользовавшись возможностями утилиты **cpuset**.

Чтобы в дальнейшем "облегчить себе жизнь", набросаем небольшой скрипт **cpuset_dummynet.sh** следующего содержания:

```
#!/bin/shAWK=/usr/bin/awkCPUSET=/usr/bin/cpusetGREP=/usr/bin/grepPROCSTAT=/usr/bin/p
rocstatPROCESS=dummynetTID=`$PROCSTAT -at | $GREP $PROCESS |$AWK '"/$PROCESS/" {prin
t $2}'`$CPUSET -l 0 -t $TID
```

После запуска скрипта, проверяем, что процесс dummynet действительно "привязан" к **CPU0**:

```
# procstat -at | grep dummynet     0 100098 kernel           dummynet           0
  16 sleep   -
```

Проверяем нагрузку:

```
0 root      -68    0     0K   416K -       0 279:33  0.00% kernel{dummynet}
```

Последний штрих - прописать запуск скрипта в **rc.local** для автоматического запуска после перезагрузки системы.

Источник (получено *2026-11-10 09:59*):
http://muff.kiev.ua/content/dummynet-privyazka-protsessa-k-cpu0