

Спасибо автору! Взято: <http://nikita.melnichenko.name/blog.php?id=11&topic=unix-way-temperature-monitor-in-5-minutes&lang=ru>

Вообще с помощью апплета stdin можно многое разместить на рабочем столе ;)

«UNIX прост. Но надо быть гением, чтобы понять его простоту.»
Деннис Ритчи.

Философия UNIX — не просто свод правил. Это идеология, которую применяешь каждый день, потому что так действительно удобно и просто. Бывает, что объяснить эту концепцию весьма трудно, но я попытаюсь сделать это на реальном примере — на одной из задач, которую мне недавно пришлось решать.

Мне нужно было понаблюдать за температурой процессора и видеокарты. Стандартный апплет в моей графической среде KDE почему-то не работал. Потратив немного времени, я так и не понял, как его оживить. Что ж, не проблема, «приготовим» свой апплет! Ведь я отлично знаю, как посмотреть текущую температуру устройств, значит нужно только вывести это на рабочий стол.

Одно из важнейших правил UNIX-way — правило композиции: «Разрабатывайте программы так, чтобы их можно было соединить с другими программами». Именно такие программы нам сейчас понадобятся. Вот, например, популярный [плазма-апплет stdin](#) позволяет показать вывод другой программы прямо на рабочем столе. Это очень кстати, поскольку наша задача сводится к более простой — написать скрипт, выводящий температуру всех устройств! Сразу возникают вопросы — сложен ли будет этот скрипт, сколько времени понадобится на его написание? Немного, ведь программы для считывания температуры у нас есть, и нужно только «причесать» их вывод.

Для просмотра температуры ядер процессора мы используем sensors.

```
# sensors
```

```
coretemp-isa-0000 Adapter: ISA adapter Core 0: +40°C (high = +100°C)
coretemp-isa-0001 Adapter: ISA adapter Core 1: +33°C (high = +100°C)
coretemp-isa-0002 Adapter: ISA adapter Core 2: +40°C (high = +100°C)
coretemp-isa-0003 Adapter: ISA adapter Core 3: +32°C (high = +100°C)
```

Температуру ядра видеокарты показывает специальная утилита от производителя.

```
# nvidia-settings -q [gpu:0]/GPUCoreTemp
```

```
Attribute 'GPUCoreTemp' (luna:0[gpu:0]): 45. 'GPUCoreTemp' is an integer attribute.
'GPUCoreTemp' is a read-only attribute. 'GPUCoreTemp' can use the following target types:
X Screen, GPU.
```

Для просмотра температуры жестких дисков у меня стоит сервер hddtemp, который по адресу <http://localhost:7634/> выводит в окне браузера следующее:

```
 |/dev/sda|MAXTOR
STM31000340AS|41|C|/dev/sdb|ST3750330AS|37|C|/dev/sdc|ST3200822A|43|C|
```

Здесь выводятся последовательно идентификатор устройства, название модели диска, его температура и единицы измерения.

Конечно, смотреть информацию в таком виде очень неудобно. Поэтому мы преобразуем вывод каждой программы. Например, текст от sensors подвергнем такой обработке:

```
# sensors | grep 'Core' | awk '{ print $1 " " $2 " " $3; }' | sed -e 's/+/ /' -e 's/^\s/CPU /'
```

Здесь написано, что нужно взять вывод от sensors, выбрать только строки, содержащие «Core» (утилита grep). Получится

```
Core 0: +40°C (high = +100°C) Core 1: +33°C (high = +100°C) Core 2: +40°C
(high = +100°C) Core 3: +32°C (high = +100°C)
```

Далее нужно выбрать 1-й, 2-й и 3-й столбцы (представим, что это текстовая таблица) и распечатать их через пробел (утилита awk). Получится

Core 0: +40°C Core 1: +33°C Core 2: +40°C Core 3: +32°C

Затем удалить «+» и добавить в начале строчки «CPU » (утилита sed). Получится

CPU Core 0: 40°C CPU Core 1: 33°C CPU Core 2: 40°C CPU Core 3: 32°C

В таком виде читать вывод гораздо удобнее.

Здесь мы снова использовали правило композиции — объединили работу четырех программ, чтобы достичь нужного результата. Каждая из небольших программ grep, awk, sed умеет качественно выполнять свою конкретную задачу, для которой эта программа создавалась. grep предназначена для выбора строк по критерию, awk работает с текстовыми таблицами, sed выполняет изменение строк. Вместе они — сила.

Переводить алгоритм преобразования текста на язык утилит — последнее дело. Главное — представлять, как можно в принципе добыть данные, т.е. что бы Вы делали с ними в редакторе. Например, для того, чтобы выделить «45» из вывода nvidia-settings, нужно выбрать строки с «Attribute», взять 4-й столбец, удалить точку. Просто, не правда ли? С температурой дисков ненамного сложнее. Здесь нам потребуется утилита netcat, которая выведет страницу по нужному хосту и порту (localhost, 7634). Как только это сделано, мы можем разделить результат на три строки, вставив перенос строк между «||», затем в полученной таблице выбрать 2-й, 4-й и 5-й столбцы, отформатировав их нужным образом, и заменить «/dev/» на «HDD ». Всё!

В результате получился следующий скрипт:

```
1. #!/bin/bash
2. #
3. # sensors.sh v0.1
4. # Copyright 2009, Nikita Melnichenko [http://nikita.melnichenko.name]
5. # License: GPL-2 (http://opensource.org/licenses/gpl-license.php)
6. #
7. # Output all sensors together.
8.
9. # CPU temperature
10. sensors | grep 'Core' | awk '{ print $1 " " $2 " " $3; }' | sed -e 's/+//' -e 's/^/CPU /'
11.
12. echo
13.
```

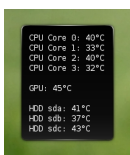
```
14. # GPU temperatures
15. gpu_temp=`nvidia-settings -q [gpu:0]/GPUCoreTemp | grep Attribute | awk '{ print $4 }' |
sed
-e
's/.$//'
\

16. echo "GPU: $gpu_temp°C"
17.
18. echo
19.
20. # HDD temperatures
21. netcat localhost 7634 | sed -e "s/|/|n|/g" | awk -F'|' '{ print $2 ": " $4 "°" $5 }' | sed -e "s,/
dev/,HDD ,"
```

Выводит он следующее:

```
CPU Core 0: 40°C CPU Core 1: 33°C CPU Core 2: 40°C CPU Core 3: 32°C GPU: 45°C
HDD sda: 41°C HDD sdb: 37°C HDD sdc: 43°C
```

Теперь осталось отдать скрипт апплету stdin, выставить нужный интервал обновления — и все готово — Вы можете смотреть температуру на рабочем столе в режиме реального времени.



Итак, огромное достоинство данного подхода — возможность собрать персональное решение из универсальных компонент как из кирпичиков. Возможно, для некоторых покажется сложным написание кода подобного рода. Конечно, чтобы интегрировать все компоненты за 5 минут нужен определенный опыт. Однако потраченное на его приобретение время окупается с лихвой. Попробуйте — и у Вас получится.

Никита Мельниченко.

