

## Smuxi - Bug # 1012: no stats provided by smuxi-server

<b>Status:</b>	New	<b>Priority:</b>	Normal
<b>Author:</b>	Andre n	<b>Category:</b>	Engine
<b>Created:</b>	01/05/2015	<b>Assigned to:</b>	
<b>Updated:</b>	01/05/2015	<b>Due date:</b>	
<b>Complexity:</b>			
<b>Found in Version:</b>			
<b>Subject:</b>	no stats provided by smuxi-server		
<b>Description:</b>	<p>when running smuxi-server there is no obvious way to discover how it is currently used. something like that but also with network usage (nethogs provide this but not for user and not a single process as i could see)</p> <pre>&lt;pre&gt;pidstat -ullhd -p `pgrep -fl mono   grep server   awk '{print \$1}'` 2 5&lt;/pre&gt;</pre> <p>A possible reason why such functionality could be wanted is, before doing the maintance work it would be usefull to see how heavy it is used by the end users.</p> <p>Another usecase might be to collect perfomance and usage data of the server, so the schedule of the maintance gets easier.</p> <p>Usefull information i would be interested in</p> <ul style="list-style-type: none"><li>* connected user/frontends</li><li>* uptime</li><li>* time since last connects (could be merged in --list-users)</li><li>* io or netio</li><li>* (size of buffers when they are limited)</li></ul>		

### History

**01/05/2015 12:39 PM - Mirco Bauer**

- Category set to Engine

**01/05/2015 12:40 PM - Mirco Bauer**

Not sure how this works. Can you provide a working/assumed example? It this using signals and prints to stdout or so?

**01/05/2015 01:39 PM - Andre n**

Mirco Bauer wrote:

> Not sure how this works. Can you provide a working/assumed example? It this using signals and prints to stdout or so?

me neither. (to be sure)

But i would split the functionality between an internal "API" to query and a dedicated binary for that to not affect the engine, when statistics are hanging or producing too much load.

For connected user/frontends i have to less insights how smuxi works, it might be good to react on a SIGnal or some method to query the state.

But for the rest the rest some notes:

you find network statistics for a process in:

```
<pre>cat /proc/`pgrep -fl mono | grep server | awk '{print $1}'`/net/dev</pre>
```

a tool to observe this for all processes might be nethogs, but it is written in c++ :/ So i'm not very motivated to analyze what they do

since i cannot say from monitoring the application from the outside if there is any user connected, i cannot

i didn't find any way to discover if users are connected to it.

If there is interest in i could to wrap the the provided oneliner up in a script.

If the performance impact is too high someone can still try to migrate from accessing proc or sysfs to using some lower API.