

---

Subject: Monitoring OpenVZ resources using munin  
Posted by [Jan Tomasek](#) on Thu, 19 Jul 2007 12:37:00 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

Hello,

I wanted to monitor resource of my OpenVZ systems using Munin. I took inspiration from wiki page:

[http://wiki.openvz.org/Monitoring\\_openvz\\_resources\\_using\\_munin](http://wiki.openvz.org/Monitoring_openvz_resources_using_munin) but I needed mixture of features from old and new version. Nice thing on old version is possibility to draw multiple resources into one graph. Bad thing was speed.

I have rewritten old version to use `/proc/bc/<VEID>/resources`, and did some cleaning in code. Result is working nicely, you can see live output at <http://munin.cesnet.cz/munin/OpenVZ/index.html>.

I was thinking how to group some resources, not saying that my decision is best, but it works. I also prepared script `ln-vebc` which I run every hour from cron to create new links for new VE and delete links for old removed/stoped VE.

Both scripts are attached. If there is interest feel free put them on wiki, I just don't want to put there 3rd version of same.

Best regards

--

-----  
Jan Tomasek aka Semik  
<http://www.tomasek.cz/>

`#!/bin/bash`

`FILE=`mktemp /tmp/ln-vebc-XXXXXX``

`cd /etc/munin/plugins`

```
for resources in kmemsize \  
  lockedpages_privvmpages_shmpages_physpages_vmguarpages_oomgu arpages \  
  numproc \  
  numtcpsock_numflock_numpty_numsinfo_numothersock_numiptent \  
  tcpsndbuf_tpcrcvbuf_othersockbuf_dgramrcvbuf \  
  dcachesize \  
  numfile  
do  
  for VE in $(cat /usr/sbin/vzlist | sed "s/^ *//" | grep '^ [0-9]' | cut -f 1 -d " ")  
  do  
    ln -sf /usr/local/share/munin/plugins/vebc_ "vebc_"$resources_"$VE"
```

```

    echo "vebc_"$resources_"$VE >> $FILE
done
done

# remove no longer deserved links (ie. links pointing to machines
# which were destroyed or stoped)

find -type l -name vebc_* | sed "s/\./_/" | while read LN; do
    if grep ^$LN$ $FILE >/dev/null; then
        true
    else
        rm $LN
    fi
done

rm $FILE#!/bin/sh
#
# Munin's plugin to monitor OpenVZ bean counters.
#
# $Log$
# Revision 1.3 2007/07/19 12:57:00 Jan Tomasek <jan@tomasek.cz>
# * rewrited to work with /proc/bc/<VEID>/resources instead of
# /proc/user_beancounters, that simplified code and result
# is also bit faster.
# * added references to OpenVZ wiki
#
# Original revision taken from:
# http://wiki.openvz.org/Monitoring_openvz_resources_using_mun in
#
## family=auto
## capabilities=autoconf suggest

VEID=`basename $0 | sed -e 's/^vebc_.*_//';
STATS=`basename $0 | sed -e 's/^vebc_//' -e 's/_[0-9]*$//' -e 's/_/_/g`

if [ "$1" = "autoconf" ]; then
    if [ -r /proc/bc/0/resources ]; then
        echo yes
        exit 0
    else
        echo "no (/proc/bc/0/resources not found)"
        exit 1
    fi
fi

if [ "$1" = "suggest" ]; then
    if [ -r /proc/bc/0/resources ]; then
        cat /proc/bc/0/resources |

```

```

while read str; do
    vals=($str)
    echo ${vals[0]}
done

exit 0
else
exit 1
fi
fi

if [ ! -f /proc/bc/$VEID/resources ]; then
    exit 0;
fi

if [ "$1" = "config" ]; then
    #echo "graph_order down up"
    echo "graph_title VE$VEID: $STATS"
    echo "graph_vlabel bean counters"
    echo "graph_category VE$VEID"
    # Note on URLs. General graph info is by munin version 1.2.5
    # accepted even with HTML code. But for value.info it escapes URL,
    # I expect that authors of munin will note that in future and put
    # escaping even for graph.info.
    echo "graph_info VE bean counters info. Documentation of the OpenVZ resource management
is located at <a href=\"http://wiki.openvz.org/UBC\">http://wiki.openvz.org/UBC</a>."

    cat /proc/bc/$VEID/resources | while read name value top warn max stuff ; do
for statname in $STATS; do
    if [ "$name" == "$statname" ]; then
        URL="http://wiki.openvz.org/$name"
        if [ "$warn" == "0" ]; then
            warn=$max
        fi
        echo $name.label $name
        echo $name.warning $warn
        echo $name.critical $max
        echo $name.info Description of this resource is located at $URL
    fi
done
done
fi;

cat /proc/bc/$VEID/resources | while read name value stuff ; do
    for statname in $STATS; do
        if [ "$name" == "$statname" ]; then
            echo $name.value "$value";
        fi
    done
done

```

done  
done

---