

een CentOS7 server monitoren met snmp

installatie

installeer de nodige software door:

```
yum install net-snmp snmp-utils
```

firewall

Zet de firewall open:

```
firewall-cmd --permanent --add-port=161/udp
firewall-cmd --permanent --add-port=162/udp
firewall-cmd --reload
```

configuratie

pas de configuratie aan:

```
vi /etc/snmp/snmpd.conf
```

Zet de betreffende regels als volgt:

```
rocommunity public

####
# First, map the community name "public" into a "security name"
#      sec.name      source                community
com2sec mynetwork 192.168.1.0/24 public
com2sec mynetwork localhost public

####
# Second, map the security name into a group name:
#      group.name sec.model sec.name
group MyROGroup v2c mynetwork

####
# Third, create a view for us to let the group have rights to:
#      name          incl/excl subtree      mask(optional)
view all included .1

####
# Finally, grant the group read-only access to the systemview view.
#      context sec.model sec.level prefix read write
```

```
notif
access MyROGroup "" any noauth exact all none
none
```

Om een proces en een directory te monitoren voeg je nog toe:

```
# dit is om het http proces te monitoren met nagios via snmp, critical is
>10 en <1
proc httpd 10 1

# dit is om de root partitie te bewaken critical is <100000 Mb vrij
disk / 100000

#dit is om de load te bewaken
load 12 14 14
```

Een custom OID/MIB maken

Voor sommige gewenste dat is geen snmp OID beschikbaar, bijvoorbeeld netwerk I/O per tijdseenheid.

Om deze waarden toch via snmp beschikbaar te maken kan je de uitkomst van een script als extra snmp waarde configureren.

We nemen het voorbeeld van het netwerk verkeer.

de standaard MIB's geven een cumulatief getal voor het aantal octets in en uit, dus geen bits per seconde.

De oplossing is een script dat continue deze waarde voor het netwerk verkeer uitrekent en in een file zet. Het uitlezen van de waarde in die file kunnen we dan als een simpel script aan de snmp configuratie toevoegen.

```
vi /usr/local/bin/calculate_network_traffic.sh
```

en zet hier in:

```
#!/bin/bash

while true
do

    #haal de cumulatieve waarde van de input en output octets op elke 5
    seconde en deel het verschil door 5

    inoct_1=$(snmpget -v 1 localhost -c public IF-MIB::ifInOctets.2 | cut -d'
' -f4)
    outoct_1=$(snmpget -v 1 localhost -c public IF-MIB::ifOutOctets.2 | cut -
d' ' -f4)
```

```
sleep 5

inoct_2=$(snmpget -v 1 localhost -c public IF-MIB::ifInOctets.2 | cut -d'
' -f4)
outoct_2=$(snmpget -v 1 localhost -c public IF-MIB::ifOutOctets.2 | cut -
d' ' -f4)

in_traffic=$((inoct_2 * 8) - (inoct_1 * 8)) / 5]
out_traffic=$((outoct_2 * 8) - (outoct_1 * 8)) / 5]
#echo $in_traffic
echo $in_traffic > /var/tmp/nettraffic
#echo $out_traffic
echo $out_traffic >> /var/tmp/nettraffic

done
```

op de Nagios server

test op de nagios server of de snmp waarden uit te lezen zijn:

```
snmpwalk -v 1 backup.auriel.nl -c public .1.3.6.1.4.1.2021.9.1.9
```

als het klopt krijg je output:

```
UCD-SNMP-MIB::dskPercent.1 = INTEGER: 65
```



From:
<https://wiki.auriel.nl/> -

Permanent link:
https://wiki.auriel.nl/doku.php?id=werkinstructies:snmp:monitoring_centos7



Last update: **2016/08/25 22:38**