**Sarcina**: De monitorizat backupurile configurate pe NAS, in caz de eroare de deschis alerta

**Tinta**: toate NAS cu backup

**Realizare**:

La moment sunt NAS in Matelica pentru diferite servere interne, si in cloud: pentru serverele din cloud.

**Logica realizarii**:

Pentru monitorizare vom avea nevoie de un server central unde vom trimite logurile. De acolo logurile vor fi analizate de Zabbix

La moment in laborator avem un NAS si un server centos 8 in calitate de Syslog server

NAS:

Hostname – Hshare

IP – 172.16.5.3

Syslog server

IP – 172.16.6.43

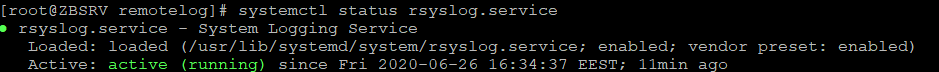
Vom configura pe serverul Syslog primirea mesajelor remote si modul de formatare a astfel de mesaje. In NAS se va configura ca logul backup sa fie exportat utilizand syslog pe serverul syslog (serverul Zabbix la noi) in formatul IP.log.

**Rezolvare**:

1. Ne logam prin ssh in serverul Syslog. Verificam daca serviciul rsyslog (serviciu ce raspunde de loguri) ruleaza

*systemctl status rsyslog.service*

Tb sa primim raspuns ca serviciul este activ



Daca serviciul nu ruleaza, il pornim noi

*systemctl enable --now rsyslog*

Daca serviciul nu este instalat, il instalam apoi il executam

*yum install rsyslog*

*systemctl enable –now rsyslog.service*

Odata ce serviciul ruleaza, intram in fisierul de configurare

*vi /etc/rsyslog.conf*

Aici gasim si decomentam liniile ce permit primirea mesajelor prin UDP si TCP prin portul 514

Ar tb sa avem in felul urmator dupa ce decomentam

*# Provides UDP syslog reception*

*# for parameters see http://www.rsyslog.com/doc/imudp.html*

*module(load="imudp") # needs to be done just once*

*input(type="imudp" port="514")*

*# Provides TCP syslog reception*

*# for parameters see http://www.rsyslog.com/doc/imtcp.html*

*module(load="imtcp") # needs to be done just once*

*input(type="imtcp" port="514")*

Apoi pina la directiva cu *### GLOBAL DIRECTIVES ###,* adica imediat dupa sectiunea cu TCP syslog reception adaugam template ce va formata mesajele primite

*# Template to receive logs*

*$template tplremote,"%timegenerated% %HOSTNAME% %fromhost-ip% %syslogtag%%msg:::drop-last-lf%\n"*

*$template RemoteLogs,"/var/log/remotelog/%fromhost-ip%.log"*

*\*.\* ?RemoteLogs;tplremote*

Apoi cream folderul unde vor fi stocate logurile remote: *mkdir –p /var/log/remotelog*

Restartam serviciul rsyslog pentru a activa setarile

*systemctl restart rsyslog*

Un pic de explicatii la template

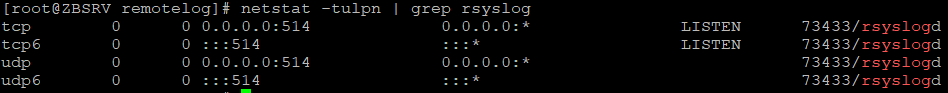
a) *$template tplremote +* ceea ce urmeaza pima la urmatorul template va formata mesajul in interiorul fisierului log. Astfel, in log-ul prvind clientul remote vom avea timpul mesajului, hostname, IP remote, syslogtag si mesajul

Exemplu: *Jun 26 16:35:15 APP\_NAS 172.16.5.3 Backup SYSTEM:#011[Local][backup] Backup task started.*

b) *$template RemoteLogs,"/var/log/remotelog/%fromhost-ip%.log"*  va inscrie mesajele remote in fisierul cu IP la clientul remote in calitate de nume in directorul: /var/log/remotelog

c) *\*.\* ?RemoteLogs;tplremote -*  activeaza toate mesajele remote sa fie formatate dupa cele 2 templat-uri

Ne asiguram ca portul 514 este deschis si pentru UDP si pentru TCP (serverul asculta aceste porturi)



Daca nu avem in system utilita netstat o instalam cu comanda: *dnf install net-tools*

Daca Selinux este activ permitem traficul cu urmatoarele reguli:

*# semanage port -a -t syslogd\_port\_t -p udp 514*

*# semanage port -a -t syslogd\_port\_t -p tcp 514*

Daca utilita semanage nu este prezenta o instalam cu comanda: *dnf install policycoreutils-python-utils*

Daca firewall este activ, permitem traficul cu regulile:

*# firewall-cmd --permanent --add-port=514/tcp*

*# firewall-cmd --permanent --add-port=514/udp*

*# firewall-cmd –reload*

Gata. Rsyslog este configurat sa primeasca mesaje

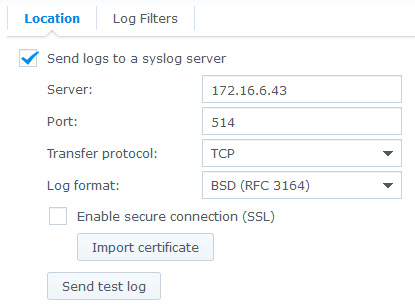
Acum e randul la clientul remote - NAS

2. Configuram ca NAS Hshare sa trimita logul despre backup pe serverul central de loguri – 172.16.6.43

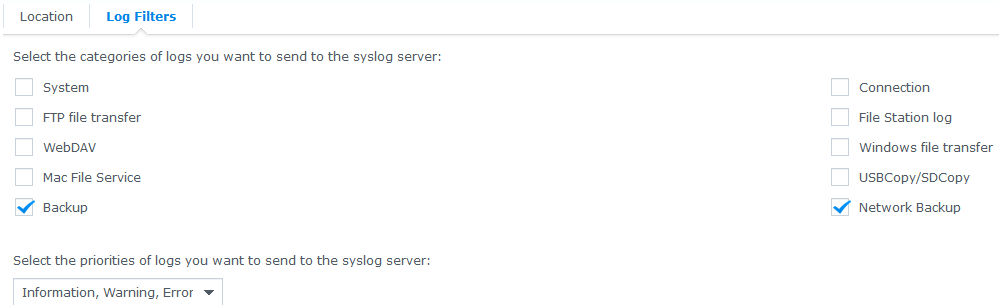
a) Pentru aceasta ne logam pe NAS prin interfata web, ne ducem la Log Center > Log Sending

b) Bifam Send logs to a syslog server

c) Introducem IP la Syslog server, portul, selectam protocolul de transmitere: UDP sau TCP. UDP este mai rapid, insa TCP este mai sigur si nu admite pierderea pachetelor de retea

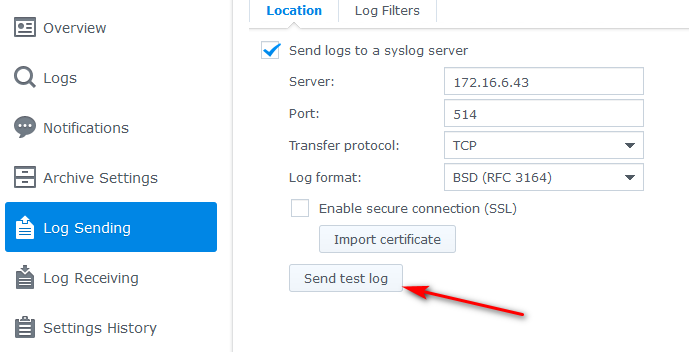


d) Ne ducem la tabul Log Filters si bifam Backup si Network Backup (la celelalte scoatem bifa). Astfel vom trimite mesaje remote doar pentru actiuni de Backup si Network Backup



e) Facem click pe Apply in partea dreapta, jos a paginii pentru a aplica setarile

Pentru a testa setarile, ne ducem inapoi in tabul Location si facem click pe Send test log



Ne logam pe serverul Syslog, in directorul */var/log/remotelog* unde verificam sa avem fisierul cu IP la NAS remote si mesajul de test



Mai departe vom configura monitorizarea in Zabbix

**Monitorizarea backupului prin script**

Deci, acum noi primim pe serverul syslog in folderul /var/log/remotelog log privind starea backupurilor efectuate de NAS.

Logica care o vom urmari in monitorizarea backupurilor va fi:

Zabbix cu ajutorul unui script va identifica taskurile de backup analizand logul de pe serverul syslog si va crea pentru fiecare in parte cate un item ce va intoarce informatie cand backupul a fost facut cu succes, si cate un trigger care va compara valorile intoarse de item. Daca nu va fi nici o diferenta intre ele, adica itemul a intors informatia precedenta, cee ce inseamna ca noul backup nu a fost executat – vom avea alerta

Pentru inceput este necesar sa configuram monitorizarea serverului Syslog de catre Zabbix server. Aceasta presupune instalarea agentului zabbix si conexiunea lui cu serverul zabbix. Dupa ce am instalat agentul, ne-am asigurat ca agentul comunica cu serverul, in configuratia la agent adaugam 2 randuri:

*UserParameter=backup[\*],/etc/zabbix/scripts/synobackup.sh "$1" "$2"*

*UserParameter=backupt[\*],/etc/zabbix/scripts/bptask.sh "$1" "$2"*

Apoi, pe serveryl syslog plasam in folderul /etc/zabbix/scripts scripturile *synobackup.sh* si *bptask.sh* si le facem executabile

Daca acest folder nu exista, il cream.

*mkdir /etc/zabbix/scripts*

*chown root:zabbix -R /etc/zabbix/scripts/*

*chmod 750 /etc/zabbix/scripts/*

Deoarece folosim comanda sed, va fi necesar dreptul de root. Permitem accesul zabbix ca root - in fisierul de configurare a agentului zabbix *(/etc/zabbix/zabbix\_agent.conf*) adaugam:

*AllowRoot=1*

*User=root*

Restartam agentul

*systemctl restart zabbix-agent*

Verificam ca utilizatorul zabbix (care a fost creat la instalarea agentului) sa poata executa aceste scripturi

*sudo -u zabbix /etc/zabbix/scripts/synobackup.sh NAS\_IP Backup\_name*

Ex. 

Scriptul ne va returna in format JSON backupurile configurate pe NAS.

Zabbix proxy

Daca NAS-ul nostru se afla dupa proxy, atunci comenzile se vor executa pe proxy. Pentru a monitoriza cu succes NAS aflate dupa proxy este necesar:

1. Sa configuram serverul Syslog pe zabbix proxy dupa modelul de mai sus

2. Sa configuram NAS-urile sa trimita logurile pe serverul zabbix\_proxy

3. Sa cream in cron un task care sa acorde drepturi agentului zabbix sa citeasca/modifice fisierele log

4. Sa copiem scripturile bptask.sh si synobackup.sh in folderul */etc/zabbix/scripts* si sa le acordam drepturi de executie: *chmod +x bptask.sh* si *chmod +x synobackup.sh*

5. Testam cu zabbix\_get daca se proceseaza corect datele pe zabbix\_proxy

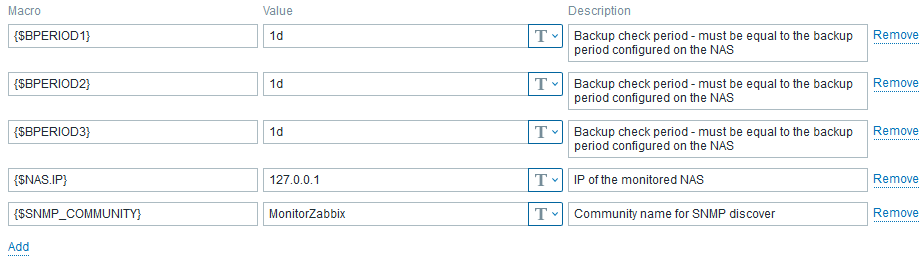
**Zabbix template**

Urmatorul pas este pregatirea unui template de monitorare, apoi il vom atasa la host

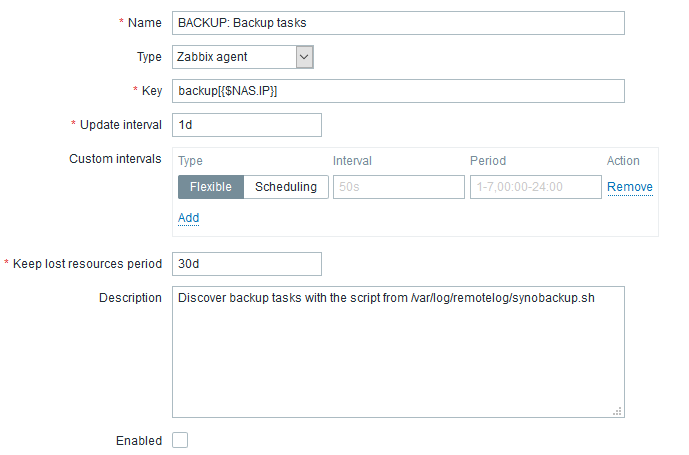
1. Pregatim template

a) Cream template, setam un nume: ex. NAS Backup

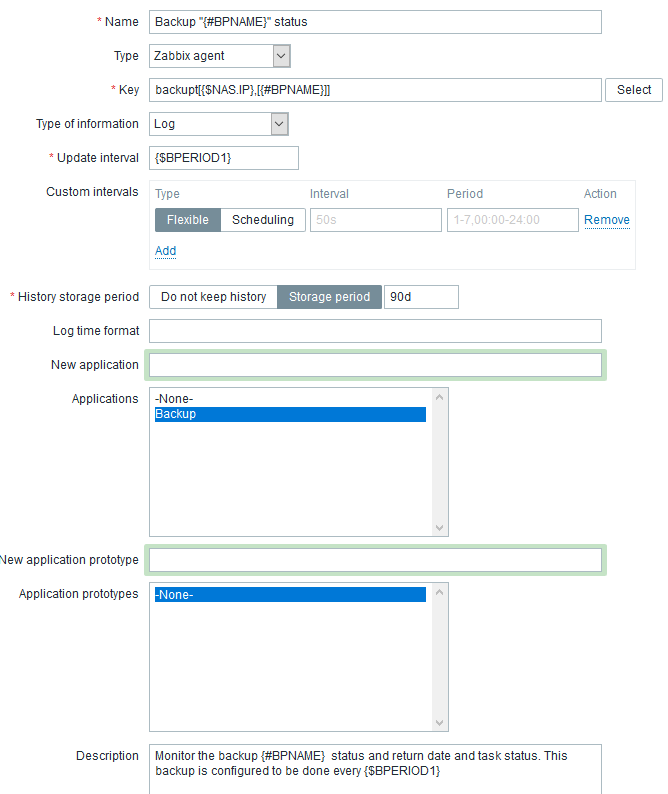
b) Adaugam Macros pentru perioade de monitorizare, IP la NAS si community name.



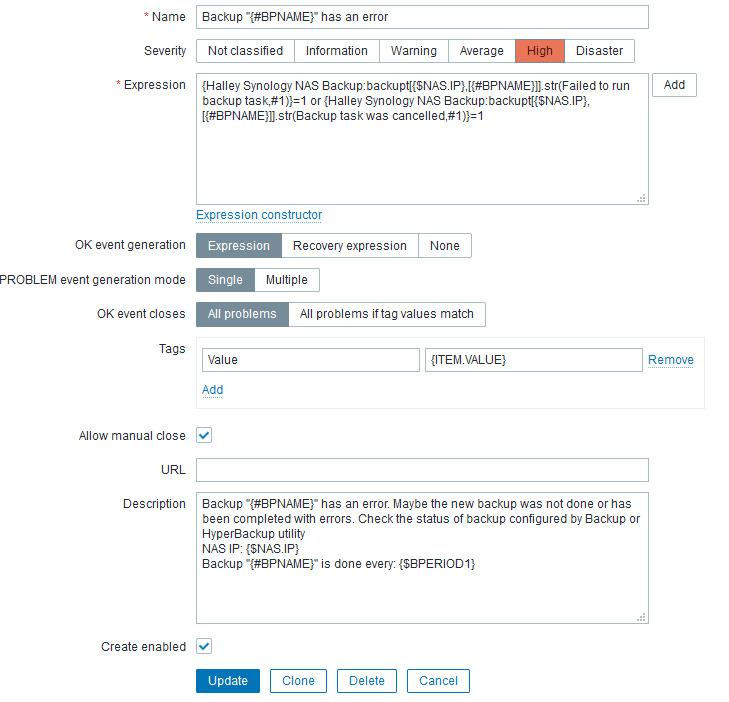
c) Mergem la Discovery rule, cream o noua regula de discovery cu cheia backup[{$NAS.IP}]



d) Adaugam item conform imaginii de mai jos. Cheia este *backupt[{$NAS.IP},[{#BPNAME}]],* unde NAS.IP este IP la NAS, BPName este numele la task-ul de backup. Setam ca itemul sa fie stocat in aplicatia Backup



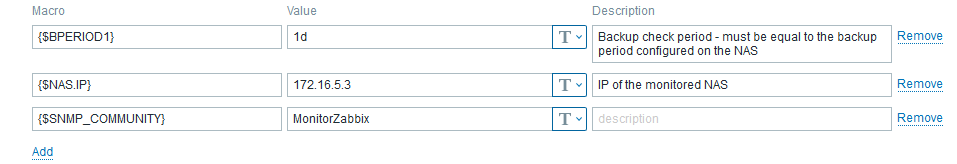
e) Adaugam trigger ce va verifica utima valoare intoarsa de item. Daca in valoarea intoarsa vor fi prezente expresiile: “Failed to run backup task” sau “Backup task was cancelled” – se va declansa problema



Problema declansata va afisa si mesajul cu eroare + ceva date despre hostul unde s-a depistat problema.

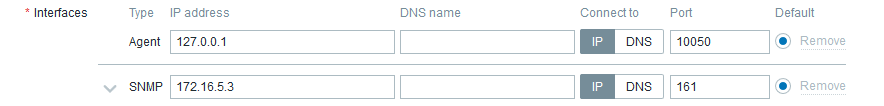
Dupa ce am creat template, este necesar de atasat template la hostul ce va fi monitorizat

1. Deci, mergem la setarile hostului, adaugam template, apoi configuram valorile macrosurilor, deja pentru hostul nostru

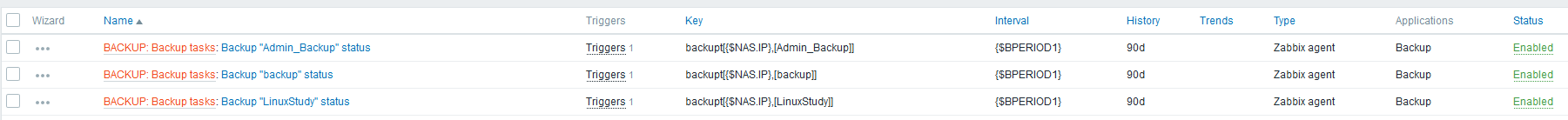


Deci, la mine perioada de executare a controlului backupului este de 1 zi (coincide cu perioada de executare a backupului), IP la NAS este 172.16.5.3, iar comunitatea SNMP – MonitorZabbix.

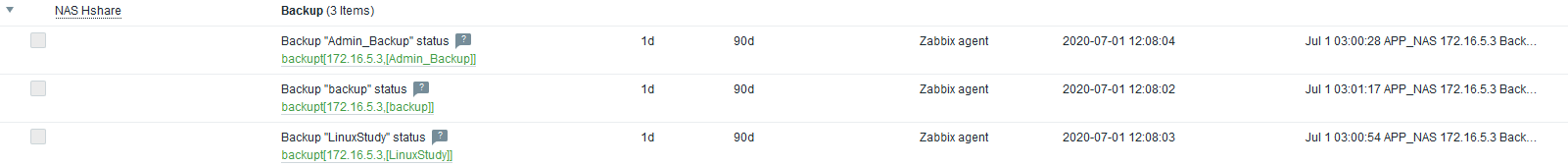
2. Este necesar ca in parametrii hostului adaugam adresa serverului syslog pentru agent



3. Verificam in latest data daca am primit noile date.



Primim datele



Anexa 1

**Scriptul synobackup.sh**

[root@zabbix scripts]# cat synobackup.sh

#!/bin/bash

export LC\_ALL=""

export LANG="en\_US.UTF-8"

### VARIABLES ###

BPATH="/var/log/remotelog"

IP="$1"

### Discover backup task and return in JSON format ###

if [[ -z "$1" || -z "$2" || -z "$3" ]]; then

##### DISCOVERY #####

BACKUPS=$(cat "$BPATH"/"$IP".log | grep "Backup task started" | cut -d '[' -f 3 | cut -d ']' -f1 | sort -u)

if [[ -n ${BACKUPS} ]]; then

JSON="{ \"data\":["

SEP=""

IFS=$'\n'

for BP in ${BACKUPS}; do

JSON=${JSON}"$SEP{\"{#BPNAME}\":\"${BP}\"}"

SEP=", "

done

JSON=${JSON}"]}"

echo ${JSON}

fi

exit 0

fi

Anexa 2

**Scriptul bptask.sh**

#!/bin/bash

# va extrage 1000 de randuri dupa cuvantul verificaT si le va analiza dupa cuvintele Failed to run backup task sau Backup task was cancelled

# va verifica daca exista cuvantul verificaT: daca da - verifica dupa el, daca nu, verifica intreg fisier

# apoi va sterge toate cuvintele verificaT si va adauga cuv. verificaT la urma. Astfel scriptul

# mereu va intoarce doar valorile noi

export LC\_ALL=""

export LANG="en\_US.UTF-8"

### VARIABLES ###

BPATH="/var/log/remotelog"

IP="$1"

BPNAME="$2"

BPTASK=$(cat "$BPATH"/"$IP".log | grep -A 1000 'verificaT' | grep -e "Failed to run backup task" -e "Backup task was cancelled" | grep "\[$BPNAME\]" | tail -n 1)

BPTASK2=$(cat "$BPATH"/"$IP".log | grep -e "Failed to run backup task" -e "Backup task was cancelled" | grep "\[$BPNAME\]" | tail -n 1)

CONDITION=$(cat "$BPATH"/"$IP".log | grep 'verificaT')

IFS=$'\n'

### Return backup task status ###

if [[ "$1" && "$2" ]]; then

if [[ "$CONDITION" ]]; then

if [[ "$BPTASK" ]]; then

echo "$BPTASK"

else

echo "Success. Backup is done"

fi

else

if [[ "$BPTASK2" ]]; then

echo "$BPTASK2"

else

echo "Success. Backup is done"

fi

fi

else

echo "Error. Parameters are provided wrong. Yo must introduce IP (x.x.x.x) and backup name in brackets "". Check script /etc/zabbix/scripts/bptask.sh"

fi

sudo sed -i 's/verificaT//g' "$BPATH"/"$IP".log 2> /dev/null

#if [[ "$CONDITION" ]]; then

# exit 0

#else

# echo 'verificaT' >> "$BPATH"/"$IP".log 2> /dev/null

#fi

echo 'verificaT' >> "$BPATH"/"$IP".log 2> /dev/null