

Fazendo backup com zfs send / zfs receive via ssh

Autoria de Fernando Buzon Macedo
10/11/2008
Última Atualização 21/11/2008

Fiz um pequeno script pra otimizar o processo de backup e manutenção de snapshots em sistema de arquivos zfs, espero que seja útil.

O script começa com alguns macros que informam:

```
dias="10" (Snapshots com mais de 10 dias no caso serão removidos)
tmp="/tmp" (Não quis chamar o 'zfs list' toda hora pq demora, preferi usar arquivo temporario.)
logdir="/var/log/snap" (arquivo onde serão salvos os logs)
```

```
nomepool="prefeitura" (esse é o nome do pool que eu vou trabalhar no servidor de produção)
```

Também tem essa opção de backup, caso esteja como 'enable' ele vai incrementar os snapshots gerados em um servidor de backup via ssh, só que vc deve usar o acesso ssh entre o servidor de produção e o que receberá o backup com chaves, percebam que eu não uso senha. Caso não esteja enable o script servirá apenas pra gerar novos snapshots e destruir os antigos.

```
backup="enable"
bkphost="192.168.0.2"
bkppool="backup" (Nome do pool no servidor backup, ja deve estar criado!)
sshport="2222"
sshuser="root"
```

Obs.: O script na verdade é meio burrão, vou explicar... rs

Caso ja tenha snapshots no seu pool e só agora vai passar a utilizar um servidor de backup, o script vai ver que vc ja tem snapshots criados no seu servidor de produção e ja vai pensar que esses snapshots também existam no servidor de backup, sendo assim vai tentar incrementa-los e não vai conseguir, vc, pra não ter trabalho, teria que destruir todos os snapshots do servidor de produção pra que ambos trabalhem em sincronia.

Sarrafo né? mas é assim... rs

Segue o script:

Grande abraço a todos!

Inicio-----

```
#!/bin/sh
```

```
dias="10"
tmp="/tmp"
logdir="/var/log/snap"
```

```
nomepool="prefeitura"
```

```
backup="enable"
bkphost="192.168.0.2"
bkppool="backup"
sshport="2222"
sshuser="root"
```

```
#####
hoje=`date | awk '{print $3}-${2}'`
epoch=`date +%s`

hora=`date | awk '{print $4}'`
echo "Inicio $hoje ( $hora )" > $logdir/$hoje.log
echo "" >> $logdir/$hoje.log

zfs list -t filesystem > $tmp/zfslist
zfs list -t snapshot > $tmp/snapshotlist

qtd_lista=`cat -n $tmp/zfslist | sed -n '/ /h; $ {x;p;}' | awk '{print $1}'`

b='2'
while [ $b -le $qtd_lista ]; do

pool=`sed "$b,$b!d" $tmp/zfslist | awk '{print $1}'`

#####
if test "$pool" != ""; then

teste=`grep $pool@ $tmp/snapshotlist`
if test "$teste" != ""; then

seg=`expr $dias '*' 60 '*' 60 '*' 24`
comp=`expr $epoch - $seg`

total=`grep $pool@ $tmp/snapshotlist | cat -n | sed -n '/ /h; $ {x;p;}' | awk '{print $1}'`

a='1'
while [ $a -le $total ]; do

ep_snap=`grep $pool@ $tmp/snapshotlist | sed "$a,$a!d" | awk '{print $1}' | awk -F '@' '{print $2}' | awk -F '_' '{print $2}'`

if test "$ep_snap" -lt "$comp"; then
snap=`grep $pool $tmp/snapshotlist | sed "$a,$a!d" | awk '{print $1}'`
zfs destroy $snap > /dev/null
echo "Destruido snapshot local $snap" >> $logdir/$hoje.log
if test "$backup" = "enable"; then
destsnap=`echo $snap | sed "s/$nomepoolV/$bkppoolV/g"`
echo "Destruindo snapshot remoto $destsnap em $bkphost (mais de $dias)" >> $logdir/$hoje.log
ssh -p $sshport $sshuser@$bkphost zfs destroy $destsnap
fi
fi

a=`expr $a + 1`
done

a=`expr $a - 1`

zfs snapshot $pool@"$hoje"_$epoch > /dev/null
echo "Criado snapshot $pool@"$hoje"_$epoch" >> $logdir/$hoje.log

if test "$backup" = "enable"; then
if test "$pool" != "$nomepool"; then
ultimosnap=`grep $pool $tmp/snapshotlist | sed "$a,$a!d" | awk '{print $1}' | awk -F '@' '{print $2}'`
npool=`grep $pool@ $tmp/snapshotlist | sed "$a,$a!d" | awk '{print $1}' | awk -F '@' '{print $1}' | sed "s/$nomepoolV/"`
echo "Incrementando backup $pool@$ultimosnap com $pool@"$hoje"_$epoch em $bkphost $bkppool/$npool" >>
$logdir/$hoje.log
zfs send -i $ultimosnap $pool@"$hoje"_$epoch | ssh -p $sshport $sshuser@$bkphost zfs recv -F $bkppool/$npool >>
$logdir/$hoje.log
fi
fi
```

```
else

zfs snapshot $pool@"$hoje"_$epoch > /dev/null
echo "Criado snapshot $pool@"$hoje"_$epoch" >> $logdir/$hoje.log

if test "$backup" = "enable"; then
  if test "$pool" != "$nomepool"; then
    npool=`echo $pool | sed "s/$nomepoolV//"`
    echo "Enviando 1o. backup de $pool@"$hoje"_$epoch para $bkphost $bkppool/$npool" >> $logdir/$hoje.log
    zfs send $pool@"$hoje"_$epoch | ssh -p $sshport $sshuser@$bkphost zfs recv $bkppool/$npool
  fi
fi

fi

else
  echo "Cade os pools?"
  echo "Cade os pools?" > $logdir/$hoje.log
fi

#####

b=`expr $b + 1`
done

hora=`date | awk '{print $4}'`
echo "" >> $logdir/$hoje.log
echo "Fim $hoje ( $hora )" >> $logdir/$hoje.log

fim-----

"O PUNK NUNCA MORRE!"
```