

Admins - Anomalie #3450

Sur guarana, le reboot d'upgrade ne doit pas être impacté par un fsck massif des sauvegardes

12/11/2018 17:21 - Christian P. Momon

Statut:	Fermé	Début:	12/11/2018
Priorité:	Urgente	Echéance:	
Assigné à:	Quentin Gibeaux	% réalisé:	0%
Catégorie:		Temps estimé:	0.00 heure
Version cible:	Juin 2019	Temps passé:	0.00 heure
Difficulté:	2 Facile		

Description

Lors de la mise à jour de guarana le 12/11/2018, le reboot la machine a été super long (~40 minutes). Le dernier fsck datait de plus de 6 mois et les partitions de sauvegardes ont été vérifiées, ce qui a pris un certain temps. La conséquence a été une indisponibilité prolongée de la passerelle pour les 4 permanents du local : situation qu'il faut absolument éviter de reproduire.

Demande : dans la procédure de mise à jour de guarana, ajouter une étape pour :

- 1) détecter qu'un fsck massif va être déclenché ;
- 2) si c'est le cas alors appliquer le fsck massif avant de rebooter (couper backuppc, démonter la partition, et lancer manuellement le fsck...).

Historique

#1 - 12/11/2018 17:23 - Christian P. Momon

Extrait IRC #april-admin du 12/11/2018 :

```
16:08 < QGuLL> cpm_screen: https://serverfault.com/a/133699
16:10 < cpm_screen> voui
16:10 < QGuLL> voire : # mount -t ext2,ext3,ext4|while read i j; do tune2fs -l $i|sed -n '/[Mm]ount count/{s/.
*: *//;p}'|(read c; read m; [ $m -gt 0 -a $m -le $c ] && echofsck,count,$i); c="$(tune2fs -l $i|sed -n '/Next
check/{s/.*r: *//;p}')" ; [ -z "$c" ] || ([ `date +%s` -ge `date -d"$c" +%s` ] && echo
fsck,time,$i); done
16:11 < cpm_screen> cette version est copier-coller-able
```

#2 - 12/11/2018 18:35 - Christian P. Momon

Autre proposition de QGull

```
for part in `mount -t ext4 | awk '{print $1}'`; do echo "Partition : $part";tune2fs -l $part | grep '\(Mount c
ount\)\|\(Maximum mount count\)\|\(Last checked\)\|\(Check interval\)\|\(Next check after\)'; done
```

Adaptée à la seule partition de sauvegarde :

```
tune2fs -l /dev/mapper/guarana-backuppc--data | grep '\(Mount count\)\|\(Maximum mount count\)\|\(Last checked
\)\|\(Check interval\)\|\(Next check after\)'
```

Ce qui donne :

```
(April) root@guarana:~48 tune2fs -l /dev/mapper/guarana-backuppc--data | grep '\(Mount count\)\|\(Maximum moun
t count\)\|\(Last checked\)\|\(Check interval\)\|\(Next check after\) '
Mount count:                2
Maximum mount count:        27
Last checked:                Mon Nov 12 15:51:25 2018
Check interval:              15552000 (6 months)
Next check after:            Sat May 11 16:51:25 2019
```

#3 - 12/11/2018 18:52 - François Poulain

À votre place je ferai un script propre (python ?) et je l'intégrerais à nagios pour toutes les partitions des machines avec un signal warning si Mount count >= Maximum mount count ou si now >= Next check after.

En l'occurrence en python il y a pour plus de temps de le déployer et de le régler avec nagios que de l'écrire et le déboguer.

#4 - 12/11/2018 20:03 - François Poulain

pour me dégourdir les neurones :

```
import fileinput
import re
from datetime import datetime, timedelta

def parse(s):
    match = re.match(r'^(?P<int>\d+) \(.*\)$', s)
    if match:
        return parse(match.group('int'))
    try:
        return int(s)
    except:
        try:
            return datetime.strptime(s, '%c')
        except Exception as e:
            return s

tune2fs = {
    match.group('key'): parse(match.group('val'))
    for match in [
        re.match(r'^(?P<key>[^:]+):\s+(?P<val>.*)$', line)
        for line in fileinput.input()
    ]
    if match
}

try:
    # check mount count
    if tune2fs['Mount count'] >= tune2fs['Maximum mount count'] > 0:
        print("Mount count ({Mount count}) >= "
              "Maximum mount count ({Maximum mount count})".format(**tune2fs))
        exit(1)
    # check fsck delay
    if tune2fs['Check interval'] and datetime.now() >= tune2fs['Last checked'] + timedelta(0, tune2fs[
'Check interval']):
        print("Fsck delay has been reached")
        exit(1)

except Exception as e:
    print(tune2fs)
    print(e)
    print('Internal error: contact your preferred devops')
    exit(3)

print('Mount count and Fsck delay are OK')
exit(0)
```

Pour l'usage :

```
francois@renard:~$ sudo tune2fs -l /dev/sda1 | python check_tune2fs.py
Mount count and Fsck delay are OK
francois@renard:~$ ssh guarana tune2fs -l /dev/mapper/guarana-backuppc--data | python check_tune2fs.py
Mount count and Fsck delay are OK
```

Je doute fort qu'un oneliner fasse mieux. :)

#5 - 15/05/2019 14:18 - Quentin Gibeaux

- Assigné à mis à Quentin Gibeaux
- Priorité changé de Élevée à Urgente
- Version cible changé de Backlog à Mai 2019

#6 - 28/05/2019 14:19 - Quentin Gibeaux

Script ajouté à /srv/scripts/common sur les hosts

#7 - 28/05/2019 14:41 - Quentin Gibeaux

- Statut changé de Nouveau à Résolu

Doc guarana mise à jour.

#8 - 29/05/2019 21:51 - Quentin Gibeaux

- Statut changé de Résolu à En cours de traitement

Documenter la procédure de fsck manuel

#9 - 29/05/2019 22:23 - Quentin Gibeaux

- Version cible changé de Mai 2019 à Juin 2019

#10 - 26/06/2019 22:00 - Quentin Gibeaux

- Statut changé de En cours de traitement à Fermé