

Croissance des volumes qui interrogent

07/03/2021 14:51 - François Poulain

Statut:	Fermé	Début:	07/03/2021
Priorité:	Normale	Echéance:	
Assigné à:	Christian P. Momon	% réalisé:	0%
Catégorie:		Temps estimé:	0.00 heure
Version cible:	Sprint 2021 mars		
Description			
Cf PJ (volumes et backups). C'est le backups qui m'interrogent. On a facilement 7Go/j non déduplicé en plus dans les backups, qui proviennent de /var/lib/postgresql. Ça m'interroge.			

Historique

#1 - 08/03/2021 13:39 - Christian P. Momon

- Fichier pad20210308a.png ajouté

- Statut changé de Nouveau à En cours de traitement

Je pense que c'est drôlement bien observé.

Une piste : le plugin photo.

Dans le graphique, on peut voir une augmentation significative (15 Go) le 23/02/2021 entre 14h et 17h :

pad20210308a.png

Pour rappel, les pads sont stockés sous forme de json contenant un champ timestamp. Trouvons une fourchette de timestamp compatible avec le créneau visé :

```
cpm@ocmstar (12:39:34) ~ 24 > date -d @1614080000; date -d @1614099999
mardi 23 février 2021, 12:33:20 (UTC+0100)
mardi 23 février 2021, 18:06:39 (UTC+0100)
```

Requêtons :

```
etherpad=# select count(*), sum(length(value)), sum(length(value))/1024/1024 as mo from store s where (s.value like '%"timestamp":161408%' or s.value like '%"timestamp":161409%') ;
count | sum | mo
-----+-----+-----
14222 | 1459047113 | 1391
(1 ligne)
```

```
etherpad=# select count(*), sum(length(value)) as taille ,sum(length(value))/1024/1024 as mo from store s where s.key like 'pad%' and (s.value like '%"timestamp":161408%' or s.value like '%"timestamp":161409%') ;
count | taille | mo
-----+-----+-----
14180 | 1459040414 | 1391
(1 ligne)
```

Donc à cette période, on constate la **création de 14180 lignes de pads pour un total de ~1,4 Go de données.**

Déterminons combien de lignes pèsent le plus :

```
etherpad=# select count(*), sum(length(value)), sum(length(value))/1024/1024 as mo from store s where s.key like 'pad%' and (s.value like '%"timestamp":161408%' or s.value like '%"timestamp":161409%') and length(s.value) > 100000 ;
count | sum | mo
-----+-----+-----
42 | 1454126556 | 1386
(1 ligne)
```

Donc l'augmentation de 16 Go constatée sur le graphique viendrait des 1,4 Go de ces 42 lignes. Ça fait un facteur 10 de différence. **Est-ce du à la gestion de Postgresql ?**

```
etherpad=# select count(*), sum(length(key)+length(value)), sum(length(key)+length(value))/1024/1024 as Mo fr
om store s ;
count | sum | mo
-----+-----+-----
10497954 | 5590464435 | 5331
(1 ligne)

=(^~^)=root@pad:/var/lib/postgresql# du -sch 11
25G total
```

Donc constat que ~5 Go de données prennent 25 Go d'espace disque, soit x5. On peut imaginer que pour de grosses données, ça passe à x10. Ok.

Déterminons le nom des pads concernés et leur taille :

```
etherpad=# select key,length(value) as taille from store s where s.key like 'pad%' and (s.value like '%"timest
amp":161408%' or s.value like '%"timestamp":161409%') order by taille desc limit 45 ;
key | taille
-----+-----
pad:gxxxxx1:revs:32800 | 34627097
[...] x 40
pad:gxxxxx1:revs:28700 | 34617731
pad:Terxxxxxxxxxxxxxxx | 71412
pad:9ltlxxxxxxxxxxxxx:revs:5400 | 63445
pad:9ltlxxxxxxxxxxxxx:revs:5300 | 62259
(50 lignes)
```

Donc, le 23/02/2021, l'augmentation de 15 Go de l'espace disque vient d'un seul pad, gxxxxx1, dont 42 itérations pèsent chacune 34 Mo.

```
etherpad=# select substr(value, 0, 10000) from store where key='pad:gxxxxxxx1:revs:32800';
{"changeset":"Z:1sx7<
[...]
"144":["color","orange"],"145":["img","<img src=
bkSSAAAAAXNSR0IArs4c6QAAAI5lWEImTU0AKgAAAAGAAgESAAMAAAABAAEAAIdpAAQAAAABAAAAJgAAAAAABZADAAIAAAAUAAAAAJKGAACAAA
ASAAAAfKABAAAMAAAABAAEAAKACAAQAAAABAAAGVKADAAQAAAABAAAIcAAAAAAMDIxOjAyOjE5IDE5OjE5OjQ5AEFTQ01JAAAAU2NyZWVuc2hv
dOOZ0b4AAAHVAVRYdFhNTDpjb20uYWRvYmUueG1wAAAAAA8eDp4bXBtZXRhIHhtbG5zOng9ImFkb2JlOm5
```

Une image plain-text dans le json du pad ? En fait, dans chaque itération du pad ?!!! Argh !

Donc l'augmentation soudaine viendrait de l'utilisation du plugin https://www.npmjs.com/package/ep_image_upload

Recommandations :

- 1) désactiver dès que possible le plugin image ;
- 2) retirer de tout les json de pad les champs img ;
- 3) ne plus activer de plugin sans une vérification préalable de leur fonctionnement.

#2 - 08/03/2021 13:51 - Christian P. Momon

Le 08/03/2021, sur Freenode#april-admin :

```
13:39 < cpm_screen> PoluX: bravo pour ton sens de l'observation, j'ai procédé à l'analyse : https://agir.april
.org/issues/5270
13:42 < cpm_screen> QGuLL: PoluX: hurdman: toutlemonde: je veux bien que lisiez le #5270, pour voir si je n'ai
pas fait d'erreur. Vertige assuré :)
13:42 < PoluX> cpm_screen: lu
13:42 < PoluX> cpm_screen: j eplussoie les recomandations
13:43 < cpm_screen> (zut, pas le bon canal)
13:43 < PoluX> à part le 2) car je ne vois pas comment faire ça sans écrire un code adhoc
13:44 < cpm_screen> cool, merci
13:45 < cpm_screen> regex dans un substr sql ? voui, pas certain que ça existe
13:45 < PoluX> cpm_screen: désactive tt de suite le plugin stp et après on voit pour dropper/recrer le pad con
cerné
13:45 < cpm_screen> voui
```

Via l'interface admin web, désactivation des plugins :

- https://www.npmjs.com/package/ep_image_upload
- https://www.npmjs.com/package/ep_copy_paste_images (celui-là, on pouvait le garder)

#3 - 08/03/2021 13:54 - Christian P. Momon

Sur la page https://www.npmjs.com/package/ep_image_upload :

```
Images are converted to base64 and stored inside etherpad document Sample configuration in settings.json for u
```

sing with base64

Donc par défaut le stockage est bien fait dans le json du pad.

Mais il est possible d'activer un stockage hors du json :

Local (disk) storage needs config for accessing files from web

#4 - 08/03/2021 18:31 - Christian P. Momon

Statistiques sur les lignes de pad supérieures à 1Mo :

Count	Size	name
71	2400M	gen*****
62	334M	Com*****
6	47M	Cha*****
1	1M	9kb*****
====	====	
140	2782M	

#5 - 12/03/2021 15:43 - Christian P. Momon

- Assigné à changé de Yves-Gaël Chény à Christian P. Momon

Le 12/03/2021 sur Freenode#april-chapril :

```
13:04 < QGuLL> DISK CRITICAL - free space: /srv/backups 9227 MB (1% inode=99%);
13:04 < QGuLL> 1% libre /o\
13:08 < PoluX> oula
13:09 < PoluX> faut arrêter de backupper le pad
13:09 < neox> PoluX: ah c'est le pad qui prend tout ça ?
13:10 < PoluX> QGuLL: le pad fait de la merde aux backups
13:10 < PoluX> et il faut virer /var/lib/pg des backups
13:10 < QGuLL> le /var/lib/postgresql peut être mis en ignore
14:27 < cpm_screen> j'ai supprimé le plugin qui pose problème mais le fait est que l'image est potentiellement
propager dans les nouvelles révision du pad
14:27 < cpm_screen> et donc la consommation est susceptible d'augmentée
14:27 < cpm_screen> j'ai passé du temps à regarder comment le nettoyer mais je suis inquiet de modifier direct
ement la structure du pad
14:28 < cpm_screen> je vais regarder un peu plus
14:34 < cpm_screen> puisque là, on a un gros problème technique et que ça ne vient que de 4 pads, je propose d
e les supprimer et de les recréer (copier/coller du contenu) manuellement avec un message au début pour préven
ir qu'on a du faire une action technique impérative
14:35 < QGuLL> je vote pour
14:35 < QGuLL> si on connaît les auteurs et qu'ils sont sur irc leur demander avant ?
14:36 < cpm_screen> c'est peu probable, je regarde
14:36 < QGuLL> juste si c'est isa ping la avant de parler courir
14:36 < QGuLL> ou fred, ou etienne
14:37 < cpm_screen> oui, bien vu
14:37 < cpm_screen> =(^^)=root@pad:~/Cpm# ll /root/Cpm/listgrosdpad-stats
14:38 < cpm_screen> dans ce fichier, tu as le nom des pads
14:38 < cpm_screen> pour info
14:40 < cpm_screen> rien que le nom des pads permet de déduire assez bien que ce ne sont pas des habitués d'irc
c
14:40 < QGuLL> 2.4G pour un pad c'est possible ? /o\
14:40 < cpm_screen> bah vala
14:41 < PoluX> cpm_screen: vaut supprimer le pad concerné, pas s'emmerder
14:41 < cpm_screen> le plugin photo est peut-être très bien mais il faut le configurer pour une sauvegarde ext
erne des fichiers
14:42 < cpm_screen> oui, mais sauver le contenu pour pouvoir le recréer
14:42 < cpm_screen> avant
14:42 < cpm_screen> comme ça on fait perdre l'historique à l'utilisateur mais pas le contenu
14:42 < cpm_screen> et pis comme c'ets facile on peut le faire
14:43 < cpm_screen> teslix: du coup on va pas faire via commandes psql, merci quand même à toi, et y aura d'au
tres besoins hein :D
14:44 < cpm_screen> là mon script de stats montre qu'un des pads problématique est passé de 71 révisions à 88,
32 Mo * 17 = 544 Mo de plus, boom
14:45 < cpm_screen> bon, je vais essayer de faire l'opération de suite
14:46 < PoluX> je vais exclure le var lib psql des backups et tout pruner
14:49 < teslix> cpm_screen: dac ;)
14:50 < PoluX> cpm_screen: sauf si tu me dis que tu met la db au régime
14:51 < cpm_screen> je suis en train de recréer les 4 pads problématiques donc la bd va perdre 3 Go dans les 2
```

0 minutes qui viennent
14:51 < PoluX> ok ; j'essaie dans 20 mins :)
14:51 < cpm_screen> pardon, 3 Go de données effectives donc en données Postgresql, ça va faire dans les 25 Go en moins
14:52 < cpm_screen> vala, premier pad traité (c'est le plus léger, 1 révision seulement)
14:54 < cpm_screen> et de 2/4
14:56 < cpm_screen> et de 3/4
14:59 < PoluX> cpm_screen: je vais quand même exclure temporairement /var/lib/postgresql/ car il reste à 30G
15:04 < cpm_screen> bah nan, attends
15:04 < cpm_screen> voilà, j'ai recréé les 4 pads, donc normalement maintenant la base devrait avoir perdu dans les 30 Go.
15:05 < cpm_screen> et du coup le prochain backup devrait être « normal »
15:07 < PoluX> du -sh /var/lib/postgresql/
15:07 < PoluX> 30GI/var/lib/postgresql/
15:08 < cpm_screen> bon, d'abord vérifier que mon nettoyage est opérant
15:09 < cpm_screen> pour vérifier, faudrait générer un export pour comparer, car après tout c'est l'export qui est backupé
15:10 < PoluX> je suis en train de dumper la db pour le backu
15:13 < PoluX> le nouveau dump fait 331M
15:14 < cpm_screen> et du coup, c'est mieux 331 Mo ? Avant c'était combien ?
15:15 < cpm_screen> 14:59 < PoluX> cpm_screen: je vais quand même exclure temporairement /var/lib/postgresql/ car il reste à 30G
15:15 < cpm_screen> heu ... mais heu ... il est sauvegardé ce dossier ?
15:15 < cpm_screen> à quoi ça sert puisque'on fait le dump exprès ?
15:15 < PoluX> cpm_screen: on est descendu de 2Go sur le dump
15:15 < cpm_screen> haaaaaaaaaaaaaaaaaaaaaa \o/
15:16 < PoluX> cpm_screen: oui il était sauvegardé
15:16 < cpm_screen> oups, bah oui mais nan, faut pas hein on est d'accord que faut pas ?
15:16 < PoluX> ça sert à ne pas se soucier des détails pour la reconstruction quand tu as une petite db
15:16 < PoluX> c'est pas un mal
15:17 < PoluX> mais ça scale pas, c'est sûr
15:18 < cpm_screen> bah j'ignore comment dire à postgrsqli de faire le ménage dans ses tampons
15:18 < PoluX> cpm_screen: je ne connais pas assez postgres pour promettre que tout (user, grants, etc.) est contenu dans les dumps qu'on a
15:18 < PoluX> c'est l'intérêt de tout prendre :)
15:19 < cpm_screen> teslix: question pour toi, on vient de virer 3 Go de données de la base (30 Go en réel), et /var/lib/postgresql n'a pas maigri lui
15:19 < cpm_screen> teslix: comment le faire maigrir ?
15:19 < teslix> il faut faire un vacuum full de la table mais ça bloque les accès
15:20 < cpm_screen> pendant combien de temps ?
15:20 < teslix> après le moteur va réutiliser l'espace lors des prochaines insertions
15:20 < teslix> ça dépend des perf de ton stockage ;)
15:20 < cpm_screen> et c'est quoi la commande ?
15:22 < teslix> vacuum full matable;
15:22 < teslix> après ça bloque les accès hein
15:22 < teslix> au pire tu peux faire un ctrl-c pour interrompre
15:22 < teslix> et pendant la copie tu bouffer le double d'espace disque
15:22 < cpm_screen> oui, logique, ha oups
15:23 < cpm_screen> oki, si ça dure quelques minutes, ça le fera bien
15:23 < teslix> tu as des ssd ?
15:23 < cpm_screen> nan, donc oui, lent
15:23 < cpm_screen> teslix: merci bcp pour l'aide :)
15:23 < cpm_screen> PoluX: y a un backup en cours ?
15:24 < PoluX> cpm_screen: oui; le pruning est en cours
15:24 < cpm_screen> oki, parce que je n'arrive pas à faire borg list, merci
15:25 < PoluX> cpm_screen: borg il rame un peu là : <https://grafana.chapril.org/d/qfRrThRMk/icinga2-influxdb-host?from=now-1h&orgId=1&to=now&viewPanel=1&var-hostname=felicette.orbite.chapril.org&var-service=volume>
15:25 < PoluX> mais contrairement à pg il libère pour de vrai :)
15:26 < cpm_screen> huhu
15:28 < cpm_screen> PoluX: tu dit prune et backup qui prune, ça veut dire que tu as changé la conf et compte dessus ? Tu ne pensais pas supprimer manuellement des sauvegardes ?
15:28 < PoluX> cpm_screen: oui j'ai changé la conf et je compte dessus
15:29 < PoluX> pruning fini
15:31 < PoluX> et je crois que je vais laisser l'exclusion :)
15:31 < cpm_screen> oki, je pense que c'est sage pour ce service, et que les inconvénients sont mineurs (ou gérables)
15:42 < PoluX> cpm_screen: backup fini en 7 mins, il fait un consistency check
15:42 < PoluX> dedup size : 217.24 MB
15:42 < PoluX> c'est mieux
15:42 < cpm_screen> \o/

#6 - 12/03/2021 15:48 - Christian P. Momon

Actions menées :

- pour les 4 pads problématiques :
 - copier/coller du contenu dans un pad temporaire,
 - suppression du pad via l'admin web,
 - création d'un pad avec le même nom,
 - copier/coller du pad temporaire dans le pad recréé,
 - ajout d'un message pour prévenir : « IMPORTANT Message de pad-support@chapril.org : le 12/03/2021, ce pad a été recréé avec la dernière version de son contenu, ceci afin de traiter un problème technique majeur sur la plateforme Chapril. Cette action exceptionnelle ne devrait pas se reproduire. Merci pour votre compréhension. »
- pour les sauvegardes :
 - exclusion de /var/lib/postgresql de la sauvegarde,
 - paramétrage temporaire du backup à 1 journée,
 - lancement de la sauvegarde avec le paramétrage temporaire,
 - vérification du pruning,
 - remise en place du paramétrage initial avec conservation de l'exclusion de /var/lib/postgresql.

Reste à faire :

- amaigrissement de /var/lib/postgresql via la commande `vacuum full mtable;`

20210312padpb.png

#7 - 12/03/2021 15:53 - Christian P. Momon

- Fichier 20210312padpb.png ajouté

#8 - 12/03/2021 22:37 - Christian P. Momon

Action d'amaigrissement de Postgresql :

```
=(^~)=root@pad:~# su - postgres
postgres@pad:~$ psql
psql (11.10 (Debian 11.10-0+deb10u1))
Saisissez « help » pour l'aide.
```

```
postgres=# \c etherpad
Vous êtes maintenant connecté à la base de données « etherpad » en tant qu'utilisateur « postgres ».
```

```
etherpad=# \dt
          Liste des relations
 Schéma | Nom | Type | Propriétaire
-----+-----+-----+-----
 public | store | table | etherpad
(1 ligne)
```

```
etherpad=# \d+ store
          Table « public.store »
 Colonne |          Type          | Collationnement | NULL-able | Par défaut | Stockage | Cible de statistique
s | Description
-----+-----+-----+-----+-----+-----+-----
 key     | character varying(100) |                  | not null  |            | extended |
 |
 value   | text                   |                  | not null  |            | extended |
 |
Index :
 "store_pkey" PRIMARY KEY, btree (key)
```

```
etherpad=# vacuum full store;
VACUUM
etherpad=#
```

Et du coup :

```
=(^~)=root@pad:/var/lib/postgresql# du -sch 11/
4,2G total
```

On passe de 26 Go à 4,2 Go. Ouf :-)

#9 - 12/03/2021 22:38 - Christian P. Momon

- Statut changé de En cours de traitement à Résolu

Tout semble revenu bien (sauvegarde et espace disque). Fermeture du ticket.

#10 - 13/03/2021 10:30 - François Poulain

- Fichier Capture d'écran de 2021-03-13 10-29-55.png ajouté

Confirmé.

#11 - 07/04/2021 22:59 - Quentin Gibeaux

- Statut changé de Résolu à Fermé

#12 - 08/04/2021 01:06 - Christian P. Momon

- Version cible changé de Backlog à Sprint 2021 mars

Fichiers

Capture d'écran de 2021-03-07 14-47-40.png	89 ko	07/03/2021	François Poulain
Capture d'écran de 2021-03-07 14-48-06.png	61,3 ko	07/03/2021	François Poulain
pad20210308a.png	47,4 ko	08/03/2021	Christian P. Momon
20210312padpb.png	60,6 ko	12/03/2021	Christian P. Momon
Capture d'écran de 2021-03-13 10-29-55.png	81,4 ko	13/03/2021	François Poulain