

Sauvegardes et garde-fous (Libres conseils 9/42)

Chaque jeudi à 21h, rendez-vous sur le framapad de traduction, le travail collaboratif sera ensuite publié ici même.

Traduction Framalang : Sky, LIAR, lerouge, yann, Goofy, peupleLa, KoS, Nys, Julius22, okram, 4nti7rust, zn01wr, lamessen

Des sauvegardes pour votre santé mentale

Austin Appel

Austin Appel, alias « scorche », est un professionnel de la sécurité informatique qui passe son temps à casser (il est dûment autorisé, évidemment) des choses précédemment réputées sécurisées. On le croise souvent enseignant le crochetage de serrure durant des conférences de sécurité et de hacking. Dans le monde de l'open source, il fait une foule de choses pour le projet Rockbox et a œuvré bénévolement pour le projet One Laptop Per Child (un ordinateur portable par enfant).

Les sauvegardes c'est bien. Les sauvegardes c'est super. Un administrateur compétent fait toujours des sauvegardes régulières. On apprend ça dans n'importe quel manuel traitant de l'administration des serveurs. Le problème c'est que les sauvegardes ne sont vraiment utiles qu'en cas d'absolue nécessité. Lorsque quelque chose de grave arrive au serveur ou à ses données et qu'on est forcé de se replier sur autre chose, les sauvegardes viendront à point nommé. Cependant, cela ne devrait jamais arriver, n'est-ce pas ? À n'importe quel autre moment, à quoi cela sert-il pour vous et votre environnement serveur d'avoir des sauvegardes ?

Avant d'aller plus loin, il est important de noter que ce conseil vaut pour les administrateurs serveurs des plus petits projets *open source* — la majorité silencieuse. Si vous maintenez des services qui vont engendrer une grande frustration, et même peut-être faire du tort s'ils sont indisponibles, vous devriez considérer ceci avec la plus grande circonspection.

Pour le reste d'entre nous qui travaillons sur d'innombrables petits projets ayant des ressources limitées, nous avons rarement deux serveurs séparés pour la production et les tests. En vérité, avec tous les services qu'un projet open source doit maintenir (système de gestion de version, services web, listes de diffusion, forums, ferme de compilation, bases de données, traceurs de bogues ou de fonctionnalités, etc.), des environnements de test séparés sont souvent de l'ordre du rêve. Malheureusement, l'approche courante de l'administration systèmes est d'avancer avec précaution et mettre les systèmes à jour uniquement en cas de nécessité absolue, afin d'éviter tout problème de dépendance, de code cassé, ou n'importe laquelle des millions de choses qui pourraient mal se dérouler. La raison pour laquelle vous êtes nerveux n'est pas que vous pourriez manquer d'expérience. Il est important de savoir que vous n'êtes pas seul dans ce cas. Que nous l'admettions ou non, beaucoup d'entre nous ont été (et sont probablement encore) dans cette situation. Il est triste que cette inaction — découlant de la peur de détruire un système fonctionnel — conduise souvent à des services en fonctionnement qui ont souvent plusieurs versions de retard, ce qui implique de nombreuses failles de sécurité potentiellement sérieuses. Cependant, soyez assuré que ce n'est pas la seule manière de jouer le jeu.

Les gens ont tendance à jouer un jeu différent selon qu'ils aient une infinité de vies ou qu'ils doivent recommencer depuis le début dès lors qu'une seule erreur a été commise. Pourquoi devrait-il en être autrement pour de l'administration systèmes ? Aborder le concept de sauvegardes avec un état d'esprit offensif peut complètement changer votre conception de l'administration systèmes. Au lieu de vivre dans la peur d'une *dist-upgrade* complète (ou de son équivalent pour yum, pacman, etc.), celui qui est armé de sauvegardes est libre de mettre à jour les paquets d'un serveur, confiant dans le fait que ces changements pourront être annulés si les choses tournent au vinaigre. La clé du succès réside tout entière dans l'état d'esprit. Il n'y a aucune raison d'avoir peur tant que vous avez vos données sauvegardées sous la main comme filet de sécurité lorsque vous sautez le pas. Après tout, l'administration système est une expérience d'apprentissage permanente.



Bien sûr, si vous ne validez pas vos sauvegardes, vous reposer sur elles devient un jeu très dangereux. Heureusement, les administrateurs systèmes expérimentés savent que le commandement « Garde des sauvegardes à jour » est toujours suivi par « Valide tes sauvegardes ». À nouveau, c'est un mantra que les gens aiment réciter. Ce qui, en revanche, ne tient pas de façon élégante dans un mantra entraînant est la manière de valider rapidement et simplement ses sauvegardes. La meilleure manière de dire qu'une sauvegarde est fonctionnelle est, bien sûr, de la restaurer (de préférence sur un système identique qui n'est pas en cours d'utilisation). Mais, une fois encore, en l'absence d'un tel luxe, on doit faire preuve d'un peu plus de créativité. C'est là (tout du moins pour les fichiers) que les sommes de contrôle peuvent vous aider à vérifier l'intégrité de vos fichiers sauvegardés. Dans *rsync*, par exemple, la méthode utilisée par défaut pour déterminer quels fichiers ont été modifiés consiste à regarder la date et l'heure de la dernière modification, ainsi que la taille du fichier. Cependant, en utilisant l'option '-c', *rsync* utilisera une somme de contrôle MD4 de 128 bits pour déterminer si les fichiers ont changé ou non. Bien que ce ne soit pas toujours la meilleure idée à mettre en œuvre à chaque fois en toute occasion — à cause d'un

temps d'exécution beaucoup plus long qu'un rsync normal et d'une utilisation accrue des accès disques — cette méthode permet de s'assurer que les fichiers sont intègres.

Le rôle d'un administrateur systèmes peut être éprouvant par moments. Il n'est cependant pas nécessaire de le rendre plus stressant que nécessaire. Avec le bon état d'esprit, certaines commandes de précaution apparemment à but unique et limité peuvent être utilisées comme des outils précieux qui vous permettent de progresser de façon agile, tout en gardant votre santé mentale intacte et la vitesse tant appréciée dans les projets open source.