
sudo, sudoedit

Exécute une commande sous un autre utilisateur

sudo autorise un utilisateur à exécuter une commande en tant que super-utilisateur ou un autre utilisateur. L'utilisateur réel est utilisé pour déterminer le nom de l'utilisateur avec lequel vérifier la stratégie de sécurité

sudo support une architecture à plugin pour les stratégies de sécurité et le logging. La stratégie de sécurité détermine quels privilèges un utilisateur a en lançant sudo. La stratégie peut exiger que l'utilisateur s'authentifie. La stratégie de sécurité par défaut est sudoers, qui est configuré via /etc/sudoers, ou via LDAP.

Les stratégie de sécurité peuvent supporter le cache d'accréditifs pour permettre à l'utilisateur de lancer sudo pendant un certain temps sans retaper son mot de passe.

OPTIONS

- A**, **-askpass** Normalement, si sudo exige un mot de passe, il le lit depuis le terminal de l'utilisateur. -A lance un helper
- b**, **-background** Dans la commande en tâche de fond
- C**, **-close-from=num** Ferme tous les descripteurs de fichier supérieur ou égal à <num> avant d'exécuter une commande. Les valeurs inférieure à 3 ne sont pas permises
- E**, **-preserve-env** Indique à la stratégie de sécurité que l'utilisateur souhaite préserver ses variables d'environnement.
- e**, **-edit** Édite un ou plusieurs fichiers
- g group**, **-group=group** Lance la commande avec le groupe primaire spécifié au lieu du groupe primaire de l'utilisateur cible
- H**, **-set-home** Demande que la stratégie définisse la variable HOME au répertoire spécifié par l'utilisateur cible
- h**, **-host=host** Lance la commande sur l'hôte spécifié si le plugin de stratégie de sécurité supporte les commandes distantes.
- i**, **-login** Lance le shell de l'utilisateur cible. Cela signifie que les fichiers comme .profile ou .login sont lus. Si aucune commande n'est spécifiée, un shell interactif est lancé
- k**, **-reset-timestamp** Utilisé sans commande, invalide les accréditifs cachés de l'utilisateur. En d'autres termes, la prochaine commande sudo nécessitera un mot de passe. Utilisé avec une autre option ou une commande, ignore les accréditifs en cache.
- K**, **-remove-timestamp** Similaire à -k, excepté qu'il supprime les accréditifs cachés de l'utilisateur et ne peut pas être utilisé en conjonction avec une commande ou autre option
- l**, **-list** Sans commande, liste les commandes permises (ou interdites) pour l'utilisateur invoquant, ou -U. Si une commande est spécifié et permise par la stratégie, le chemin complet de la commande est affichées avec les arguments.
- n**, **-non-interactive** Empêche l'utilisateur d'entrer quoi que ce soit.
- P**, **-preserve-groups** Préserve le groupe de l'utilisateur invoquant
- p**, **-prompt=prompt** Utilise un prompt de mot de passe personnalisé :
 - %H** Étend le nom d'hôte incluant le nom de domaine
 - %h** Étend un nom d'hôte local sans le domaine
 - %p** Étend le nom de l'utilisateur dont le mot de passe est demandé
 - %U** Étend le nom de login de l'utilisateur cible
 - %u** Étend le nom de login de l'utilisateur invoquant
 - %%** le caractère '%'
- r**, **-role=role** Lance la commande avec le contexte SELinux qui inclus le rôle spécifié
- S**, **-stdin** Écrit le prompt sur stderr et lit le mot de passe depuis stdin au lieu du terminal

-
- s, **-shell** Lance le shell spécifié par la variable d'environnement SHELL si définie, ou le shell de l'utilisateur invoquant
 - t, **-type=type** Lance la commande avec le contexte de sécurité SELinux qui inclut le type spécifié. Non spécifié, le type par défaut est dérivé du rôle
 - U, **-other-user=user** avec -l, liste les privilèges de l'utilisateur
 - T, **-command-timeout=timeout** Définis un timeout pour la commande
 - u, **-user=user** Lance la commande sous cet utilisateur
 - v, **-validate** Met à jours les accreditifs en cache de l'utilisateur
 - Indique la fin des arguments

Les variables d'environnement à définir pour la commande peuvent également être passés sur la ligne de commande sous la forme VAR=value, sujettes à restriction.

Exécution de commande

Quand sudo exécute une commande, la stratégie de sécurité spécifie l'environnement d'exécution pour la commande. Typiquement, l'uid et gid réel et effectif sont définis pour correspondre à l'utilisateur cible, tel que spécifié dans la base de comptes, et le vecteur groupe est initialisé avec la base de groupes.

Les paramètres suivants peuvent être spécifiés par la stratégie de sécurité :

UID effectif et réel

GID effectif et réel

GID supplémentaires

La liste d'environnement

Répertoire de travail courant

Masque de création de fichier

Role et type SELinux

Priorité (nice)

Quand sudo lance une commande, il se fork, définit l'environnement d'exécution, et appelle execve dans le processus enfant. Le processus sudo principal attend que la commande soit complétée, puis passe le status de sortie de la commande à la fonction de fermeture de la stratégie de sécurité et quitte. Si un plugin de login est configuré ou si la stratégie de sécurité le demande explicitement, un nouveau pseudo-terminal est créé et un second process sudo est utilisé pour relayer les signaux de contrôle de job entre le pty de l'utilisateur et le nouveau pty. Ce processus rend possible, par exemple, se suspendre et résumer la commande. Sans cela, la commande dans un groupe de processus orphelin et ne recevra aucun signal de contrôle de job. Un cas spécial, si le plugin de stratégie ne définit pas de fonction close et aucun pty n'est requis, sudo exécute la commande directement au lieu de fork. le plugin sudoers ne définit une fonction close quand quand le login I/O est activé, un pty requis, ou les options pam_session ou pm_setcred sont activés.

Gestion des signaux

Quand la commande est lancée comme enfant du processus sudo, sudo relai les signaux qu'il reçoit à la commande. Les signaux SIGINT et SIGQUIT sont seulement relayés quand la commande est lancées dans un nouveau pty ou quand le signal a été envoyé par un processus utilisateur, pas le kernel. Cela évite que la commande reçoive 2 fois SIGINT chaque fois que l'utilisateur utilise Contrôle-C. Certains signaux comme SIGSTOP et SIGKILL, ne peuvent être relayés à la commande. SIGTSTP doit être envoyé au lieu de SIGSTOP pour suspendre une commande.

Un cas spécial, sudo ne relai pas les signaux qui sont envoyés par la commande lancée. Cela empêche la commande de se tuer elle-même. Dans certains systèmes, reboot(8) envoie SIGTERM à tous les processus non-système autre que lui-même avant de redémarrer le système.

Cela empêche sudo de relayer SIGTERM, qui peut quitter avant que le système soit redémarré. En résultat, lancer un script qui appelle reboot ou shutdown via sudo peut terminer le système dans un état indéfini sauf si reboot ou shutdown sont lancés avec exec() au lieu de system().

Si aucun plugin de login n'est chargé et que la stratégie n'a pas défini de fonction close, définir un timeout de commande ou que la commande soit lancée dans un nouveau pty, sudo peut exécuter la commande directement.

Plugins

Les plugins peuvent être spécifiés via les directives Plugin dans sudo.conf. Ils peuvent être chargés comme objets partagés dynamiques ou compilés directement dans sudo. sudoers est le plugin par défaut

Valeur de sortie

Le code de sortie de sudo est le code de sortie de la commande exécutée. Si la commande s'est terminée à cause d'un signal, sudo envoie lui-même le signal qui termine la commande.

Sinon, sudo quitte avec une valeur de 1 s'il y a un problème de configuration/permission ou si sudo ne peut pas exécuter la commande. Dans le dernier cas, la chaîne d'erreur est affichée sur stderr. Si sudo ne peut pas stat(2) une ou plusieurs entrées dans le PATH de l'utilisateur, une erreur est affichée sur stderr.

Notes de sécurité

Sudo tente d'être sûr en exécutant des commandes externes. Pour éviter le spoofing de commande, sudo vérifie "." et "" en dernier en recherchant une commande dans le PATH de l'utilisateur. Noter cependant que la variable PATH n'est pas modifiée et est passée inchangée au programme que sudo exécute.

Les utilisateurs ne devraient jamais obtenir des privilèges pour exécuter des fichiers qui sont en écriture par l'utilisateur ou qui résident dans un répertoire en écriture par l'utilisateur. Si l'utilisateur peut modifier ou remplacer la commande il n'y a pas de limite à ce qui peut être lancé.

Noter que sudo normalement ne log que les commandes explicitement lancées. Si un utilisateur lance une commande telle que sudo su ou sudo sh, les commandes suivantes ne sont plus sujettes à la stratégie sudo. De même pour les commandes qui offrent des échappement shell. Si le login I/O est activé, les commandes suivantes sont loggées, mais il n'y a pas des logs traditionnels pour ces commandes. À cause de cela, une attention particulière doit être portée sur l'accès des utilisateurs aux commandes via sudo pour vérifier que la commande ne donne pas par inadvertance un shell root.

Pour éviter la fuite d'information potentiellement sensible, sudo désactive les coredumps par défaut durant l'exécution.

VARIABLES D'ENVIRONNEMENT

EDITOR Nom de l'éditeur par défaut pour -e (sudoedit), si SUDO_EDITOR ou VISUAL ne sont pas définis

MAIL Emplacement de la boîte mail de l'utilisateur

HOME Répertoire personnel de l'utilisateur

LOGNAME Nom de l'utilisateur

PATH Les chemins de recherche pour les commandes

SHELL Shell courant de l'utilisateur

SUDO_ASKPASS Spécifie le chemin du helper pour lire le mot de passe si aucun terminal n'est disponible ou avec -A

SUDO_EDITOR Éditeur par défaut pour -e (sudoedit)

SUDO_GID GID de l'utilisateur qui a invoqué sudo

SUDO_PROMPT prompt pour les mots de passe

SUDO_PS1 valeur PS1 pour le programme à exécuter

SUDO_UID UID de l'utilisateur qui a invoqué sudo

SUDO_USER nom de login de l'utilisateur qui a invoqué sudo

USER nom de l'utilisateur

USERNAME idem à USER

VISUAL Éditeur par défaut pour -e (sudoedit) si SUDO_EDITOR n'est pas définis

Exemples

obtenir une liste des répertoire non-lisible :

sudo ls /usr/local/protected

Liste le répertoire personnel de l'utilisateur yaz dans une machine où le système de fichier maintenant ~yaz n'est pas exporté en root

sudo -u yaz ls ~yaz

Éditer index.html en tant qu'utilisateur www

sudoedit -u www ~www/htdocs/index.html

Voir les logs système uniquement accessibles par root

sudo -g adm more /var/log/syslog

Lancer un éditeur en tant que jim avec un groupe primaire différent

sudoedit -u jim -g audio ~jim/sound.txt

Éteindre la machine

sudo shutdown -r +15 "quick reboot"