audit.rules

Jeu de règles chargés dans le système d'audit

Le fichier audit.rules est un fichier contenant des règles d'audit chargés par auditd au démarrage. auditctl est utilisé pour cette opération. Les régles d'audit sont de 3 type : control, file, et syscall

contrôle

Les commandes de contrôle configurent le système d'audit. Ces commande incluent typiquement la suppression de toutes les règles, définir la taille de file backlog, définir le mode d'erreur, etc.

Système de fichier

Ces régles sont parfois appelées des watchs. Ces règles sont utilisée pour auditer les accès à des fichiers particuliers. La syntaxe suit généralement ce format : **-w path-to-file -p permissions -k keyname** où les permissions sont une des suivantes : r,w,x,a

appels système

Ces règles sont chargées dans un moteur qui intercepte chaque syscall que tous les programmes font dans le système. Noter que cela impacte les performances.

Le kernel a 4 règles : task, exit, user et exclude.

task Cette liste est vérifie seulement durantles syscall fork ou clone. Rarement utilisé en pratique.

exit Est l'endroit où tous les syscall et requêtes d'audit système sont évalués

user Utilisé pour filtrer (supprimer) certains évènements qui viennent du userspace.

exclude Utilisé pour exclure certains évènements. Le champ msgtype est utilisé pour indiquer au kernel les types de message à ne pas enregistrer.

Les règles syscall prenne la forme générale :**-a action,list -S syscall -F field=value -k keyname**

-a indique d'ajouter une règle à la fin de la liste.

action et list sont séparés par une virgule. les listes valides sont task, exit, user et exclude, les actions valide sont always ou never (créer un évènement)

La suite de la règle est normalement -S, qui peut être soit un nom ou un numéro de syscall.

L'option -F affine le matche. Voir auditctl pour une liste complète de champ.

Notes

En faisant une investigation, on commence normalement avec aureport pour avoir une idée de ce qui se passe dans le système. Ce rapport indique principalement les évènements hardcodés par le système d'audits tels que les login/logout, authentifications, anomalies système, etc.

aureport –start this-week

Ensuit, pour obtenir une seconde vue des règles chargées :

aureport –start this-week –key –summary

Cela donne une liste ordonnée des clés associées avec les règles Si par exemple, une règle syscall sur un échec d'ouverture d'un fichier avec EPERM, avec une clé nommée access :

-a always,exit -F arch=b64 -S open -S openat -F exit=-EPERM -k access

On peut isoler ces erreurs avec auseurch et envoyer le résultat à aureport.

ausearch –start this-week -k access –raw | aureport –file –summary

Supposons que l'on souhaite voir quels utilisateurs se sont vu refuser l'accès :

ausearch –start this-week -k access –raw | aureport –user –summary

Pour afficher beaucoups d'accès échoués à un fichier particulier, on peut lancer un rapport pour voir qui le fait :

ausearch –start this-week -k access -f /path-to/file –raw | aureport –user -i

Ce rapport donne les tentatives d'accès par personne. Pour voir un évènement particulier, regarder la date et l'heure. Assumant que l'évènement est le 822 à 2 :30 le 09/01*2009.

ausearch –start 09/01/2009 02 :30 -a 822 -i –just-one

Sélectionne le premier évènement de ce jour.

Exemples

La règle suivante montre comment auditer les erreurs d'accès aux fichiers à cause de problèmes de permission. Noter que ça demande 2 règles pour chaque architecture :

-a always,exit -F arch=b32 -S open -S openat -F exit=-EACCES -k access
-a always,exit -F arch=b32 -S open -S openat -F exit=-EPERM -k access
-a always,exit -F arch=b64 -S open -S openat -F exit=-EACCES -k access
-a always,exit -F arch=b64 -S open -S openat -F exit=-EPERM -k access