pkexec

Exécuter une commande sous un autre utilisateur

pkexec permet à un utilisateur autorisé d'exécuter le programmes spécifié sous un autre utilisateur. Si l'utilisateur n'est pas spécifié, utilise root.

Valeur de retour

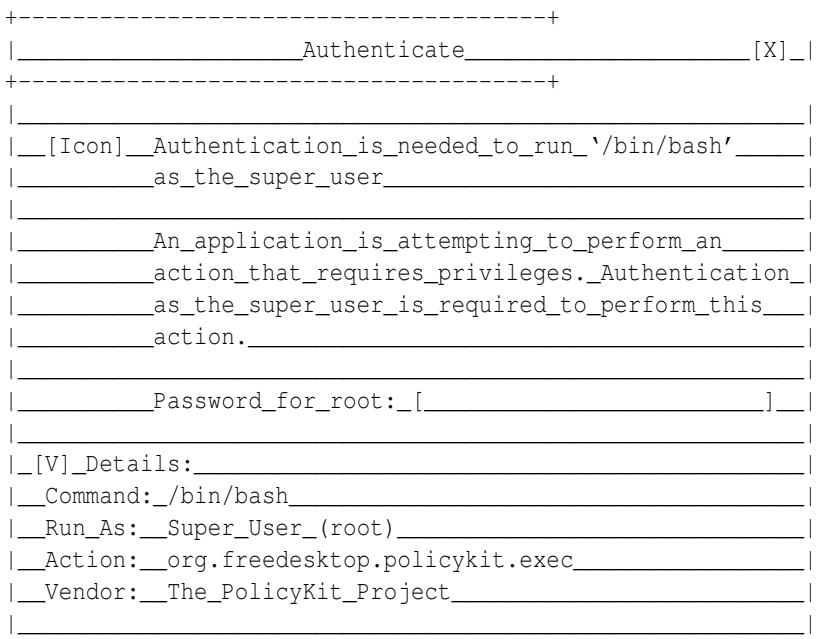
une fois terminé, la valeur de retour est la valeur de retour du programme. Si le processus appelant n'est pas autorisé ou qu'une autorisation ne peut pas être obtenue ou qu'une erreur s'est produite, pkexec quitte avec une valeur de retour de 127. Si l'autorisation ne peut être obtenue, pkexec quitte avec la valeur 126.

l'agent d'authentification

pkexec, comme tout autre application PolicyKit, utilise l'agent d'authentification enregistré pour le procéssus appelant. Cependant, si aucun agent d'authentification n'est disponible, pkexec enregistre son propre agent d'authentification. Ce comportement peut être désactivé avec –disable-internal-agent.

Notes de sécurité

Exécuter un programme sous un autre utilisateur est une opération privilégiée. Par défaut l'autorisation requise nécéssite une authentification administrative. De plus, le dialogue d'authentification présenté à l'utilisateur affiche le chemin complet du programme à exécuter dont l'utilisateur est au courant de se qu'il va se passer.



```
|_____________________________________[Cancel]_[Authenticate]_|
+-------------------------------------+
```

L'environnement dans lequel le programme tourne, est définis à un environnement minimal et sûr pour éviter d'injecter du code via LD_LIBRARY_PATH ou des mécanismes similaires. De plus, la variable d'environnement PKEXEC_UID est définie avec l'UID invoquant pkexec. En résultat, pkexec n'autorise pas de lancer des application X11 sous un autre utilisateur vu que $DISPLAY et $XAUTHORITY ne sont pas définis. Il y a 2 variables retenus si org.freedesktop.policykit.exec.allow_gui dans une action est définis à une valeur non-nul. C'est découragé, et utilisé uniquement pour compatibilité.

Autorisations requises

Par défaut, org.freedesktop.policykit.exec est requis sauf si un fichier de définition d'action est présent pour le programme en question. pour exiger une autre autorisation, cela peut être spécifié en utilisant l'annotation org.freedesktop.policykit.exec.path dans une action.

Exemple

Pour spécifier un type d'autorisation nécessaire pour exécuter le programme /usr/bin/pk-example-frobnicate sous un autre utilisateur, écrire simplement une définition d'action :

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE policyconfig PUBLIC
 "-//freedesktop//DTD PolicyKit Policy Configuration 1.0//EN"
 "http://www.freedesktop.org/standards/PolicyKit/1/policyconfig.dtd">
<policyconfig>

 <vendor>Examples for the PolicyKit Project</vendor>
 <vendor_url>http://hal.freedesktop.org/docs/PolicyKit/</vendor_url>
 <action id="org.freedesktop.policykit.example.pkexec.run-frobnicate">
  <description>Run the PolicyKit example program Frobnicate</description>
  <description xml:lang="da">Kør PolicyKit eksemplet Frobnicate</description>
  <message>Authentication is required to run the PolicyKit example program Frobnicate (user=$(user),
program=$(program), command_line=$(command_line))</message>
  <message xml:lang="da">Autorisering er påkrævet for at afvikle PolicyKit eksemplet Frobnicate
(user=$(user), program=$(program), command_line=$(command_line))</message>
  <icon_name>audio-x-generic</icon_name>
  <defaults>
   <allow_any>no</allow_any>
   <allow_inactive>no</allow_inactive>
   <allow_active>auth_self_keep</allow_active>
  </defaults>
  <annotate key="org.freedesktop.policykit.exec.path">/usr/bin/pk-example-frobnicate</annotate>
 </action>

</policyconfig>
```

placer ce fichier dans /usr/share/polkit-1/actions avec un nom qui ait du sens, par exemple l'espace de nom de l'action. Noter que pkexec ne valide pas les arguments passés au programme. Dans un cas normal (où l'authentification administration est requise à chaque fois que pkexec est utilisé), ce n'est pas un problème vu que si l'administrateur est un administrateur il peut simplement lancer pkexec bash pour devenir root.

Cependant, si une action est utilisée pour laquelle l'utilisateur peut retenir l'authorisation (ou si l'utilisateur est implicitement autorisé), tel qu'avec pk-example-frobnicate ci-dessus, cela peut être un problème de sécurité. Donc, les programmes pour lesquels l'autorisation

requise par défaut est changé, ne devraient jamais implicitement faire confiance à l'entrée utilisateur.