Durcissement du noyau Linux

l'utile et le futile dans le patch grsecurity

RÉSIST

28 janvier 2002

par Denis Ducamp - HSC Toulouse
Denis.Ducamp@hsc.fr - http://www.hsc.fr/
Denis.Ducamp@groar.org - http://www.groar.org/

HSC - Hervé Schauer Consultants

- cabinet de 16 consultants en sécurité
  - Unix
  - Réseaux TCP/IP
  - Windows NT/2k/...
- agence de 2 consultants sur Toulouse
  - depuis novembre 2001
- Intervient dans de nombreuses conférences
  - OSSIR Paris : SUR et SWNT
    - et maintenant RÉSIST à Toulouse ;-)
  - EFE / Linux Expo / Reed-OIP / Netsec / Infosec / etc.
- Toutes nos présentations sont accessibles sur notre serveur web
  - ainsi que de nombreux articles dans la rubrique tips,
  - des rapports de veille, des articles, des supports de cours, etc.
- Logiciels libres depuis toujours
  - 386BSD, Linux, FreeBSD, OpenBSD
  - sendmail, bind, Postfix, apache, OpenSSL, OpenSSH, etc.
  - auteur de quelques logiciels libres originaux en sécurité
    - filterrules, nstreams, idswu, babelweb, subweb, etc.
- Mais maîtrise aussi de nombreux logiciels et matériels commerciaux

HSC - prestations

- Audits de sécurité
- Tests d'intrusion
- Audits de sources

- Veille en vulnérabilités
- Veille technologique et stratégique de l'actualité en sécurité

- Cours
  - Risques TCP/IP
  - Linux / Unix / Windows
  - Postfix, bind, etc.
  - IPSec
  - Internet/Intranet
  - etc.

- Conseils en architectures
- etc.

Introduction

- Le patch grsecurity
  - et les autres : OpenWall, Pax, etc.
- Quelques notions
  - chroot / set[ug]id
  - débordement de tampons
  - /proc / noms de fichiers temporaires prévisibles / fork-bomb
  - Journalisation / audit
  - ptrace() / OS-fingerprint / tpe
- Les options de grsecurity
  - Buffer Overflow Protection
  - Access Control Lists
  - Filesystem Protections
  - Kernel Auditing
  - Executable Protections
  - Network Protections
  - Sysctl support
  - Miscellaneous Enhancements
- Quelques cas concrets
  - Le serveur d'un particulier / de e-commerce / de la NSA

Le patch grsecurity

- Le patch grsecurity regroupe un grand nombre d'options de durcissement du noyau linux.
- Le durcissement noyau est l'interdiction
  - d'appeler certains appels systèmes
    - dans un certain nombre de cas
  - d'accéder à certaines données sensibles
- Cette présentation fait le tour de chacune des options en expliquant
  - contre quelles attaques respectives elles sont écrites pour lutter
  - et si réellement elles sont utiles.
- Home Page : http://www.grsecurity.net/
- Version testée : Jan 21 01:34 grsecurity-1.9.3a-2.4.17.patch

Les autres patches

- Ce patch est basé sur de nombreux autres patches dont :
  - Openwall
    - http://www.openwall.com/linux/
    - par Solar Designer
    - disponible pour 2.2.19/20 et 2.0.39
    - fait partie du projet Openwall, voir aussi Owl
  - PaX
    - http://pageexec.virtualave.net/
    - par The PaX Team
    - disponible pour 2.2.19/20 et 2.4.16
  - Bien d'autres qui ont été portés sous Linux 2.4
    - ex : Trusted Path Execution, cf Phrack

Pourquoi ce patch ?

- Pourquoi existe-t'il ?
  - pour permettre à des administrateurs d'ajouter un grand nombre d'options de façon simple
    - toutes les options sont contenues dans un seul patch
    - ce patch est mis à jour à chaque nouvelle version de noyau
  - pour permettre d'avoir sous Linux 2.4 des patches écrits pour Linux 2.0/2.2
- Pourquoi est-il étudié ici ?
  - parce que c'est le patch qui regroupe le plus grand nombre d'options
    - c'est l'occasion de voir ce qui est utile...
- Attention : cet ensemble de patches est complexe
  - certains patches peuvent être en cours de développement
    - donc bugs :
      - du dénis de services...
      - aux contournements de protections.
  - certains patches peuvent être supprimés d'une version à l'autre
    - ex : "stealth networking" est remplaçable par iptables

Quelques notions

- chroot / set[ug]id
- débordement de tampons
- /proc / noms de fichiers temporaires prévisibles / fork-bomb
- Journalisation / audit
- ptrace() / OS-fingerprint / tpe

Quelques notions (1)

- chroot
  - L'appel système chroot permet de limiter un processus à une sous-arborescence du disque
    - et de limiter les risques si l'application est vulnérable
  - +
    - L'accès disque du pirate est limité à cette arborescence
  - -
    - La mise en cage d'un processus tournant sous root est inutile
      - création de périphériques : accès direct au disque
      - double chroot : évasion de la cage
      - etc.
- set[ug]id
  - L'appel système setuid (setgid) change l'utilisateur (le groupe principal) exécutant le processus
  - +
    - Seul un processus (setuid) root peut changer d'identité
  - -
    - Ce processus peut dans certains cas récupérer les droits root
  - Un programme setuid (setgid) s'exécute avec les privilèges de l'utilisateur (du groupe) propriétaire
    - si un tel programme est vulnérable
      - alors tout utilisateur peut obtenir les privilèges du propriétaire

Quelques notions (2)

- débordement de tampons
  - Le programmeur a prévu un tampon de longueur fixe pour une donnée
    - l'utilisateur envoie une donnée de longueur bien plus grande
    - l'exécution du programme est perturbée
      - du déni de service...

```
Program received signal SIGSEGV, Segmentation fault.
0x41414141 in ?? ()
```

      - à l'exécution contrôlée de code

```
Linux vaio 2.4.16 #3 Sat Dec 1 23:56:11 CET 2001 i686 unknown
uid=0(root) gid=0(root) groups=0(root)
```

- l'exploitation du débordement de tampon se fait par
  - envoie d'un shellcode (code exécutable lançant un shell)
    - et changement de l'adresse de retour d'une fonction en cours vers le shellcode envoyé
  - changement de l'adresse de retour vers une fonction de la biblioth èque libc
    - system(), strcpy(), etc.
- le tampon débordé peut aussi bien être placé
  - dans la pile, le tas, etc.
  - càd des espaces mémoire en exécution
    - qui ne devraient être qu'en lecture / écriture

Quelques notions (3)

- /proc
  - système de fichiers virtuels sous Linux
  - donne accès à des paramêtres du noyau
    - dont certains peuvent être modifiés au vol : /proc/sys
  - +
    - permet d'obtenir de nombreuses informations sur tous les processus
  - -
    - permet de savoir ce que font les autres utilisateurs
- Noms de fichiers temporaires prévisibles
  - l'attaquant crée le fichier avant l'utilisateur
    - et lui permet d'y écrire dedans
    - le fichier créé peut être un lien symbolique/dur ou un tube nommé
  - -
    - déni de service : écrasement de fichiers systèmes
    - escalade de privilèges : écrasement contrôlé de fichiers sensibles
    - vol d'informations sensibles
- fork-bomb
  - exécution d'un grand nombre de processus
  - exécution d'un grand nombre de processus par seconde
  - -
    - déni de service par consommation de ressources système : mémoire et/ou processeur

Quelques notions (4)

- Journalisation
  - une application (le noyau) génère un évènement
    - l'évènement est trié et enregistré par un démon spécialisé : syslogd
    - sous Linux c'est le démon klogd qui route les évènements noyau vers syslogd
  - la journalisation peut être déportée vers un autre serveur
    - utile car les pirates nettoient systématiquement leurs traces sur le système intrusé
- Audit
  - le noyau journalise tous les appels à certains appels systèmes
    - avec les paramêtres et/ou le résultat
- +
  - analyse de l'utilisation du système
    - en temps réel
    - en différé
- -
  - peut être très consommateur en ressources
    - déni de service par consommation de ressources système

Quelques notions (5)

- ptrace()
  - Cet appel système permet d'effectuer du debogage de processus sous Unix
- +
  - pour savoir comment fonctionne un processus
- -
  - pour changer l'exécution d'un processus en cours d'exécution

- OS-fingerprinting
  - Un ensemble de paquets IP non standards sont envoyés au systè me à analyser
  - L'ensemble des réponses est caractéristique de chaque type de système
    - deux versions différentes d'un même système peuvent souvent être différenciés

- Trusted path execution (tpe)
  - Permet de limiter l'exécution aux programmes se trouvant dans un répertoire de confiance
    - répertoire appartenant à root
    - et n'étant en écriture que pour root

Grsecurity

`CONFIG_GRKERNSEC`

- Permet tout simplement d'activer la sélection des groupes d'options principaux

  - Buffer Overflow Protection
  - Access Control Lists
  - Filesystem Protections
  - Kernel Auditing
  - Executable Protections
  - Network Protections
  - Sysctl support
  - Miscellaneous Enhancements

Groupe d’options 1/8

- Buffer Overflow Protection
- Access Control Lists
- Filesystem Protections
- Kernel Auditing
- Executable Protections
- Network Protections
- Sysctl support
- Miscellaneous Enhancements

OpenWall Non-executable Stack

`CONFIG_GRKERNSEC_STACK`

- Permet de rendre la pile d’exécution non exécutable
  - Il s’agit du patch de Solar Designer porté sous Linux 2.4
- Protège contre l’exécution de code dans la pile
  - Utilisée lors d’exploitations de vulnérabilités :
    - débordements de tampons dans la pile / chaînes de formats
- Ne protège pas dans le cas :
  - de débordements de tampons dans le tas
  - de retours dans la libc
- L’exécution de code dans la pile est utilisée dans la grande majorité des exploitations de vulnérabilités car c’est le cas d’école
  - donc cette option est très utile, mais :
    - elle ne dispense pas de mettre à jour ses services vulnérables
    - elle permet généralement de se prémunir contre le tout premier programme d’exploitation de chaque vulnérabilité
- Ne peut être sélectionné si "PaX protection" l’est

Gcc trampoline support

CONFIG_GRKERNSEC_STACK_GCC

- Permet d'activer le support des sauts par trampolines
  - Un saut par trampoline est l'exécution d'un morceau de code dans la pile pour accélérer un programme
    - Le patch doit différencier les sauts des exploitations de vulnérabilités
    - La bibliothèque glibc 2.0 contient plusieurs sauts et nécessite donc cette option
- En théorie ceci pourrait permettre de contourner le fait que la pile est non exécutable
  - En pratique aucun cas n'a été détecté.
- Nécessite "OpenWall Non-executable Stack" **CONFIG_GRKERNSEC_STACK**

PaX protection

CONFIG_GRKERNSEC_PAX

- Empêche l'exécution de code dans certaines pages mémoire
  - Normalement sous IA-32 toute page en lecture est en exécution
  - Il s'agit du patch PaX
- Protège contre l'exécution de code dans la pile, le tas...
  - Utilisée lors d'exploitations de vulnérabilités
- Incompatible avec certains programmes
  - ceux qui exécutent du code dans des espaces mémoires dynamiques obtenus par malloc() : XFree86, JRE, Wine...
- PaX est aujourd'hui contournable
  - voir l'article de Nergal : article 4 phrack 58
  - mais permet généralement de se prémunir contre le tout premier programme d'exploitation de chaque vulnérabilité
- Ne peut être sélectionné si "OpenWall Non-executable Stack" l'est

Emulate trampolines

CONFIG_GRKERNSEC_PAX_EMUTRAMP

- Le patch PaX est lui aussi incompatible avec les sauts par trampolines
  - cette option permet de les détecter pour en autoriser l'exécution
- Nécessite "PaX protection" **CONFIG_GRKERNSEC_PAX**

Restrict mprotect()

CONFIG_GRKERNSEC_PAX_MPROTECT

- Empêche que le processus réactive
  - le droit en exécution sur des pages non exécutables
  - le droit en écriture sur des pages en lecture seule
- Nécessite "PaX protection" **CONFIG_GRKERNSEC_PAX**

Randomize mmap() base

CONFIG_GRKERNSEC_PAX_RANDMMAP

- Rend aléatoire les adresses des programmes :
  - la pile d'exécution
  - les résultats de mmap
    - donc l'adresse de chargement des bibliothèques
  - l'adresse des programmes ELF dynamiques
- L'attaquant doit deviner les adresses des composants à utiliser
  - N'empèchée certainement pas :
    - l'exploitatation de chaînes de formats par analyse distante de la pile d'exécution
- Ne peut être sélectionné si "OpenWall Non-executable Stack" l'est

Read-only kernel memory

CONFIG_GRKERNSEC_KMEM

- Empêche root de modifier la mémoire du noyau
  - Empêche le chargement de root kits noyau
- Ne sert à rien si le support des modules n'est pas désactivé

Fixed mmap restrictions

CONFIG_GRKERNSEC_MMAPFIXED

- Empêche qu'un processus appelle la fonction mmap() avec
  - une adresse fixe
  - et un droit en exécution
- Permet d'empécher un programme de rendre inutile les résultats alé atoires de mmap()

Groupe d'options 2/8

- Buffer Overflow Protection
- Access Control Lists
- Filesystem Protections
- Kernel Auditing
- Executable Protections
- Network Protections
- Sysctl support
- Miscellaneous Enhancements

Oblivion ACL System

CONFIG_GRKERNSEC_ACL

- Permet d'activer le support du système des ACL Oblivion
- Ce système est basé sur le démon obvadm à récupérer séparément
  - http://www.grsecurity.net/obvadm-1.1a.tar.gz
- Permet de spécifier des ACL
  - de façon absolue sur tous les fichiers : file.acl
  - de façon relative à certains programmes : proc.acl
  - http://www.grsecurity.net/obvdoc.tar.gz
- Un fichier / une arborescence peuvent être :
  - en lecture
  - en exécution
  - en écriture
  - en ajout seulement
  - caché

Oblivion ACL System : options...

Ces options ne sont pas documentées...

- et nécessitent "Enable grsecurity ACL system" **CONFIG_GRKERNSEC_ACL**
- Maximum Number of Rulesets for Files, Processes
  - CONFIG_GR_MAX_RULESET 256
- Seconds in between log messages(minimum)
  - CONFIG_GR_FLOODTIME 3
- Default ruleset for programs without acls
  - Deny CONFIG_GR_DEF_DENY
  - Allow CONFIG_GR_DEF_ALLOW (défaut)
  - Deny_if_running_as_root CONFIG_GR_DEF_DENY_ROOT
- Enable ACL Debugging Messages
  - CONFIG_GR_DEBUG
- Path to gradm
  - CONFIG_GRADM_PATH "/sbin/gradm"
- Maximum tries before password lockout
  - CONFIG_GR_MAXTRIES 3
- Time to wait after max password tries, in seconds
  - CONFIG_GR_TIMEOUT 30

Groupe d’options 3/8

- Buffer Overflow Protection
- Access Control Lists
- Filesystem Protections
- Kernel Auditing
- Executable Protections
- Network Protections
- Sysctl support
- Miscellaneous Enhancements

Proc Restrictions

`CONFIG_GRKERNSEC_PROC`

- Restreint l’accès à /proc aux utilisateurs
  - qui ne peuvent consulter que leurs propres processus
  - qui ne peuvent accéder
    - ni à dmesg(8)
    - ni aux symboles noyaux et modules
- Les restrictions effectives dépendent des options sélectionnées
- Permet de maintenir une bonne confidentialité
  - sur des systèmes partagés par des personnes étrangères (FAI...)
  - sur des systèmes très sensibles

Restrict to user only

CONFIG_GRKERNSEC_PROC_USER

- Restreint l’accès à /proc à tous les utilisateurs non-root
- Nécessite "Proc restrictions" **CONFIG_GRKERNSEC_PROC**

Allow special group

CONFIG_GRKERNSEC_PROC_USERGROUP

- Permet à un groupe d’accéder de façon normale à /proc
- Le groupe principal actuel du processus n’est plus consultable
- Nécessite "Proc restrictions" **CONFIG_GRKERNSEC_PROC**
- Ne peut être sélectionné si "Restrict to user only" l’est

GID for special group

CONFIG_GRKERNSEC_PROC_GID 1001

- Le groupe pour lequel l'accès à /proc n'est pas restreint
- Nécessite "Allow special group" **CONFIG_GRKERNSEC_PROC_USERGROUP**

Linking restrictions

CONFIG_GRKERNSEC_LINK

- Empêche un processus de suivre un lien symbolique
  - dans un répertoire en +t (comme /tmp)
  - qui appartient à un autre utilisateur non-root
- Permet de lutter de façon efficace contre les attaques
  - utilisant des noms de fichiers temporaires prévisibles
- Ces attaques peuvent être de différents types
  - des dénis de services
    - destruction de fichiers systèmes sensibles
      - par ex. /etc/shadow
  - des escalades de privilèges
    - modification contrôlée de fichiers systèmes sensibles
      - par ex. /etc/passwd - /.rhosts

FIFO restrictions

CONFIG_GRKERNSEC_FIFO

- Empêche un processus de suivre un lien symbolique
  - dans un répertoire en +t (comme /tmp)
  - qui appartient à un autre utilisateur non-root
- Permet de lutter de façon efficace contre
  - les attaques utilisant des noms de fichiers temporaires prévisibles
- Ces attaques peuvent être de différents types
  - des vols d'informations sensibles
    - par ex. obtention d'informations après déchiffrement
  - de modifications de données au vol
    - par ex. insertion de backdoor dans main()

Secure file descriptors

CONFIG_GRKERNSEC_FD

- Assure que tout programme possède stdin, stdout et stderr ouverts
  - respectivement les fichiers 0, 1 et 2 en C
  - entrée et sortie standards, sortie d'erreurs
- Permet de lutter de façon efficace contre
  - des dénis de service
    - destruction de fichiers systèmes sensibles
      - par ex. /etc/shadow
  - des escalades de privilèges
    - modification contrôlée de fichiers systèmes sensibles
      - par ex. /etc/passwd - /.rhosts
  - des vols d'informations sensibles
    - par ex. /etc/shadow

Chroot jail restrictions

CONFIG_GRKERNSEC_CHROOT

- Permet d'accéder à certaines options de restriction des programmes mis en cage
  - Restricted signals
  - Deny mounts
  - Deny double-chroots
  - Enforce chdir("/") on all chroots
  - Deny (f)chmod +s
  - Deny mknod
  - Deny ptraces
  - Restrict priority changes
- Empêche les processus restreints d'accéder à certaines fonctions
  - normalement sans utilité pour eux
    - sauf pour s'évader de la cage
    - ou attaquer d'autres processus hors de la cage
- Normalement il est futile de restreindre des processus root
  - Utile dans le cas d'applications mal conçues ou d'origine douteuse

Restricted signals

CONFIG_GRKERNSEC_CHROOT_SIG

- Les processus en cage ne peuvent envoyer des signaux qu'aux processus dans la même cage
- Permet d'empécher les attaques
  - contre les programmes sensibles aux attaques par signaux
  - en dénis de service en tuant (aléatoirement) les autres processus
- Nécessite "Chroot jail restrictions" **CONFIG_GRKERNSEC_CHROOT**

Deny mounts

CONFIG_GRKERNSEC_CHROOT_MOUNT

- Les processus en cage ne peuvent monter ou démonter des partitions
- Empêche l'évasion de la cage en montant des partitions inaccessibles depuis la cage
- Nécessite "Chroot jail restrictions" **CONFIG_GRKERNSEC_CHROOT**

Deny double-chroots

CONFIG_GRKERNSEC_CHROOT_DOUBLE

- Les processus en cage ne peuvent pas se mettre en cage eux même
- Empêche l'évasion de la cage par double chroot
  - http://www.bpfh.net/simes/computing/chroot-break.html
- Nécessite "Chroot jail restrictions" **CONFIG_GRKERNSEC_CHROOT**

Enforce chdir("/") on all chroots

CONFIG_GRKERNSEC_CHROOT_CHDIR

- La fonction chroot() force que le répertoire du processus soit la racine de la cage
- Empêche l'évasion de la cage par getcwd / fchdir
  - en forçant le répertoire de travail à être la racine de la cage
- Nécessite "Chroot jail restrictions" **CONFIG_GRKERNSEC_CHROOT**

Deny (f)chmod +s

CONFIG_GRKERNSEC_CHROOT_CHMOD

- Les processus en cage ne peuvent pas ajouter un bit suid/sgid via chmod() ou fchmod()
- Les bits suid/sgid n'ont pas de raison d'être dans un environnement restreint
  - Pour empécher les escalades de privilèges par des processus restreints
  - ou qu'un processus non privilégié et non restreint
    - obtienne des privilèges de la part d'un processus privilégié mais restreint
- Nécessite "Chroot jail restrictions" **CONFIG_GRKERNSEC_CHROOT**

Deny mknod

CONFIG_GRKERNSEC_CHROOT_MKNOD

- Les processus en cage ne peuvent pas appeler mknod()
- Empêche l'évasion de la cage par accès direct à des périphériques
  - via création de fichiers spéciaux en mode block ou caractère
- Nécessite "Chroot jail restrictions" **CONFIG_GRKERNSEC_CHROOT**

Deny ptraces

CONFIG_GRKERNSEC_CHROOT_PTRACE

- Les processus en cage ne peuvent pas appeler ptrace()
- Empêche l'évasion de la cage par débogage de processus hors de la cage
  - Les processus dans des environnements restreints n'ont pas de raison de déboguer d'autres processus
    - surtout si ces derniers ne sont pas restreints
- Nécessite "Chroot jail restrictions" **CONFIG_GRKERNSEC_CHROOT**

Restrict priority changes

CONFIG_GRKERNSEC_CHROOT_NICE

- Les processus en cage ne peuvent pas appeler nice() pour augmenter leur priorité
- Empêche les dénis de service par consommation de ressources processeur
- Nécessite "Chroot jail restrictions" **CONFIG_GRKERNSEC_CHROOT**

Capability restrictions within chroot

CONFIG_GRKERNSEC_CHROOT_CAPS

- Limite les privilèges des processus mis en cage pour leur interdire
  - d'insérer des modules : **CAP_SYS_MODULE**
  - d'ouvrir des périphériques raw : **CAP_SYS_RAWIO**
  - d'effectuer des tâches d'administration système et réseau : **CAP_SYS_ADMIN**
  - de transférer des privilèges : **CAP_SETPCAP**
  - et de configurer des tty : **CAP_SYS_TTY_CONFIG**
- Les processus dans des environnements restreints n'ont pas de raison d'effectuer ce type de tâches

Secure keymap loading

CONFIG_GRKERNSEC_KBMAP

- Limite à root le changement des touches et des touches de fonctions du clavier

Groupe d’options 4/8

- Buffer Overflow Protection
- Access Control Lists
- Filesystem Protections
- Kernel Auditing
- Executable Protections
- Network Protections
- Sysctl support
- Miscellaneous Enhancements

Single group for auditing

CONFIG_GRKERNSEC_AUDIT_GROUP

- Limite la journalisation des appels exec, chdir, (un)mount, et ipc à un groupe
- Empêche des possibilités de dénis de services via consommation de ressources par le système de journalisation
  - Nécessite que les processus et utilisateurs à surveiller soient dans un même groupe à spécifier

GID for auditing

CONFIG_GRKERNSEC_AUDIT_GID 1007

- Le groupe pour lequel les appels exec, chdir, (un)mount, et ipc sont journalisés
- Nécessite "Single group for auditing" **CONFIG_GRKERNSEC_AUDIT_GROUP**

Exec logging

CONFIG_GRKERNSEC_EXECLOG

- Active la journalisation des appels à execve()
- Limité à un groupe si "Single group for auditing" **CONFIG_GRKERNSEC_AUDIT_GROUP** est activé

Log execs within chroot

CONFIG_GRKERNSEC_CHROOT_EXECLOG

- Journalise tous les appels à execve() effectués par les processus mis en cage

Chdir logging

CONFIG_GRKERNSEC_AUDIT_CHDIR

- Active la journalisation des appels à chdir()
- Limité à un groupe si "Single group for auditing"
  CONFIG_GRKERNSEC_AUDIT_GROUP est activé

(Un)Mount logging

CONFIG_GRKERNSEC_AUDIT_MOUNT

- Active la journalisation des mountages et démontages de partitions
- Limité à un groupe si "Single group for auditing"
  CONFIG_GRKERNSEC_AUDIT_GROUP est activé

IPC logging

CONFIG_GRKERNSEC_AUDIT_IPC

- Active la journalisation des créations et destructions
  - des queues de messages
  - des sémaphores
  - des segments de mémoire partagés
- Limité à un groupe si "Single group for auditing"
  CONFIG_GRKERNSEC_AUDIT_GROUP est activé

Signal logging

CONFIG_GRKERNSEC_SIGNAL

- Journalise certains signaux importants comme SIGSEGV
  - ceci indique quand un programme fonctionne mal
  - ceci peut indiquer des tentatives infructueuses d’exploitation d’une vulnérabilité

Fork failure logging

CONFIG_GRKERNSEC_FORKFAIL

- Journalise les appels à fork() qui ont échoué
- Ceci peut indiquer une tentative
  - de déni de service contre le système par fork-bombing
  - de dépassement de quota

Set*id logging

CONFIG_GRKERNSEC_SUID

- Active la journalisation des appels à set*id()
- Peut générer beaucoup d’évènements sur certains systèmes chargés
  - Cela dépend du comportement normal des applications
- Ne peut être sélectionné si "Log set*ids to root" l’est

Log set*ids to root

CONFIG_GRKERNSEC_SUID_ROOT

- Ne journalise que les appels à set*id() ou un utilisateur demande un changement vers l'utilisateur ou le groupe root
- Peut indiquer une exploitation de vulnérabilité
  - les shellcode commencent généralement par setreuid()
    - car /bin/bash2 se débarasse de ses privilèges
      - si le groupe et/ou l'utilisateur effectifs sont différents du groupe et/ou de l'utilisateur réel
      - voir l'option -p de bash(1) v2.x
    - car tcsh refuse de s'exécuter dans le même cas
- Ne peut être sélectionné si "Set*id logging" l'est

Time change logging

CONFIG_GRKERNSEC_TIME

- Journalise tous les changements de l'horloge système

Groupe d’options 5/8

- Buffer Overflow Protection
- Access Control Lists
- Filesystem Protections
- Kernel Auditing
- Executable Protections
- Network Protections
- Sysctl support
- Miscellaneous Enhancements

Exec process limiting

CONFIG_GRKERNSEC_EXECVE

- Les limites en ressources sont également vérifiées lors de execve()
  - Normalement ceci n’est fait que lors de fork()
- Permet d’empécher des contournements de limitations de ressources

Randomized PIDs

CONFIG_GRKERNSEC_RANDPID

- Permet de rendre aléatoire la génération des numéros de processus
- Permet de rendre plus complexes les attaques
  - par signaux
  - par débogage
- les numéros de processus des démons lancés au démarrage étant alé atoires

Altered default IPC permissions

CONFIG_GRKERNSEC_IPC

- Restreint les permissions par défaut à la création des segments de mé moire partagés
  - d'après le masque de l'utilisateur (umask)
  - quand l'utilisateur ne les spécifie pas
- Les droits d'accès par défaut sont ugo+rwx sous linux

Limit uid/gid changes to root

CONFIG_GRKERNSEC_TTYROOT

- Permet d'accéder à des options de limitation d'accès de root sur diffé rents types de consoles

Deny physical consoles (tty)

CONFIG_GRKERNSEC_TTYROOT_PHYS

- Interdit l'accès de root sur les consoles physiques
- Nécessite "Limit uid/gid changes to root" **CONFIG_GRKERNSEC_TTYROOT**

Deny serial consoles (ttyS)

CONFIG_GRKERNSEC_TTYROOT_SERIAL

- Interdit l’accès de root sur les consoles séries
- Nécessite "Limit uid/gid changes to root" **CONFIG_GRKERNSEC_TTYROOT**

Deny pseudo consoles (pty)

CONFIG_GRKERNSEC_TTYROOT_PSEUDO

- Interdit l’accès de root sur les pseudo consoles
- Nécessite "Limit uid/gid changes to root" **CONFIG_GRKERNSEC_TTYROOT**

Fork-bomb protection

CONFIG_GRKERNSEC_FORKBOMB

- Limite pour les utilisateurs du groupe spécifié
  - Le nombre total de processus autorisés
  - Le nombre de nouveaux processus par seconde autorisé
- Empêche les attaques en déni de service par fork-bombing

GID for restricted users

CONFIG_GRKERNSEC_FORKBOMB_GID 1006

- Le groupe auquel appartiennent les utilisateurs limités
- Nécessite "Fork-bomb protection" **CONFIG_GRKERNSEC_FORKBOMB**

Forks allowed per second

CONFIG_GRKERNSEC_FORKBOMB_SEC 40

- Nombre maximal de fork() par seconde pour un utilisateur limité
- Nécessite "Fork-bomb protection" **CONFIG_GRKERNSEC_FORKBOMB**

Maximum processes allowed

CONFIG_GRKERNSEC_FORKBOMB_MAX 20

- Nombre maximal de processus pour un utilisateur limité
- Nécessite "Fork-bomb protection" **CONFIG_GRKERNSEC_FORKBOMB**

Trusted path execution

CONFIG_GRKERNSEC_TPE

- Permet d'interdire l'exécution de programmes qui ne se trouvent pas dans un répertoire de confiance
- Permet d'interdire l'exécution de tout programme importé par un utilisateur
  - Empêche l'exécution d'exploitations de vulnérabilités locales
  - Inutile si un interpréteur comme perl est présent
    - car il est alors toujours possible de porter l'exploit en perl

Glibc protection

CONFIG_GRKERNSEC_TPE_GLIBC

- Empêche l'exécution de programme lorsque certaines variables d'environnement sont définies
  - LD_PRELOAD
  - /lib/ld-2.*
- Ces variables changent le comportement du programme /lib/ld-2.*
  - ce programme est utilisé par tous les programmes liés dynamiquement pour charger les bibliothèques dynamiques
  - ces variables permettent donc de contourner la protection par ré pertoire de confiance
- Nécessite "Trusted path execution" **CONFIG_GRKERNSEC_TPE**

Partially restrict non-root users

CONFIG_GRKERNSEC_TPE_ALL

- Permet d'autoriser aux autres utilisateurs d'exécuter des programmes si
  - ces programmes sont dans des répertoires qui appartiennent à l'utilisateur
  - ce répertoire n'est en écriture que pour l'utilisateur
- Autorise un utilisateur à exécuter ses propres programmes
- Empèche un attaquant de faire exécuter à un utilisateur victime un programme potentiellement malveillant
- Nécessite "Trusted path execution" **CONFIG_GRKERNSEC_TPE**

GID for untrusted users

CONFIG_GRKERNSEC_TPE_GID 1005

- Permet de limiter l'interdiction d'exécuter un programme qui ne se trouve pas dans un répertoire de confiance à un groupe donné
- Nécessite "Trusted path execution" **CONFIG_GRKERNSEC_TPE**

Restricted ptrace

CONFIG_GRKERNSEC_PTRACE

- Limite l’appel à ptrace() à root
  - le traçage des appels systèmes dans le noyau est aussi désactivé
  - les appels ptrace() autorisés sont journalisés
- L’appel ptrace() est inutilisé sur de très nombreux système
- Empêche
  - le vol d’informations dans d’autres processus
  - le changement de l’exécution d’autres processus

Allow ptrace for group

CONFIG_GRKERNSEC_PTRACE_GROUP

- Permet d’autoriser un groupe à appeler ptrace()
- Nécessite "Restricted ptrace" **CONFIG_GRKERNSEC_PTRACE**

GID for ptrace

CONFIG_GRKERNSEC_PTRACE_GID 1008

- Permet de spécifier le groupe autorisé à appeler ptrace()
- Nécessite "Restricted ptrace" **CONFIG_GRKERNSEC_PTRACE**

Groupe d’options 6/8

- Buffer Overflow Protection
- Access Control Lists
- Filesystem Protections
- Kernel Auditing
- Executable Protections
- Network Protections
- Sysctl support
- Miscellaneous Enhancements

Randomized IP IDs

CONFIG_GRKERNSEC_RANDID

- Permet de rendre totalement aléatoire la génération des numéros id des paquets ip
  - C'est le portage de la fonction de génération de OpenBSD
- Pas très utile
  - la fonction par défaut initialise à zéro le champ ID
    - pour les paquets avec le bit DF
    - donc pour les paquets syn/ack et rst tcp, udp courts

Randomized TCP source ports

CONFIG_GRKERNSEC_RANDSRC

- Permet de rendre totalement aléatoire le numéro de port source généré par le noyau
- Permet de rendre plus complexe des attaques de vols de données
  - contre des protocoles comme ftp passif

Altered Ping IDs

CONFIG_GRKERNSEC_RANDPING

- Le champ icmp.id des paquets icmp echo reply générés est identique au champ icmp.id des paquets icmp echo request reçus
- Permet de rendre plus complexe l'identification à distance de la pile IP

Randomized TTL

CONFIG_GRKERNSEC_RANDTTL

- Permet de rendre totalement aléatoire le champ TTL des paquets IP g énérés
- Permet de rendre plus complexe l'identification à distance de la pile IP
- Redessiner l'architecture d'accès n'est pas complexifié

TTL starting point

CONFIG_GRKERNSEC_RANDTTL_THRESH 64

- Permet de choisir la plus petite valeur de champ TTL à générer
- Nécessite "Randomized TTL" **CONFIG_GRKERNSEC_RANDTTL**

Socket restrictions

CONFIG_GRKERNSEC_SOCKET

- Permet d’accéder à des options de limitation des sockets Internet

Deny any sockets to group

CONFIG_GRKERNSEC_SOCKET_ALL

- Interdit à un groupe donné d’ouvrir des sockets clientes et serveurs
- Nécessite "Socket restrictions" **CONFIG_GRKERNSEC_SOCKET**

GID to deny all sockets for

CONFIG_GRKERNSEC_SOCKET_ALL_GID 1004

- Spécifie le groupe interdit d’ouvrir des sockets clientes et serveurs
- Nécessite "Deny any sockets to group" **CONFIG_GRKERNSEC_SOCKET_ALL**

Deny client sockets to group

CONFIG_GRKERNSEC_SOCKET_CLIENT

- Interdit à un groupe donné d’ouvrir des sockets clientes
- Nécessite "Socket restrictions" **CONFIG_GRKERNSEC_SOCKET**

GID to deny client sockets for

CONFIG_GRKERNSEC_SOCKET_CLIENT_GID 1003

- Spécifie le groupe interdit d’ouvrir des sockets clientes
- Nécessite "Deny client sockets to group"
  CONFIG_GRKERNSEC_SOCKET_CLIENT

Deny server sockets to group

CONFIG_GRKERNSEC_SOCKET_SERVER

- Interdit à un groupe donné d’ouvrir des sockets serveurs
- Nécessite "Socket restrictions" **CONFIG_GRKERNSEC_SOCKET**

GID to deny server sockets for

CONFIG_GRKERNSEC_SOCKET_SERVER_GID 1002

- Spécifie le groupe interdit d’ouvrir des sockets serveurs
- Nécessite "Deny server sockets to group" **CONFIG_GRKERNSEC_SOCKET_SERVER**

Groupe d’options 7/8

- Buffer Overflow Protection
- Access Control Lists
- Filesystem Protections
- Kernel Auditing
- Executable Protections
- Network Protections
- Sysctl support
- Miscellaneous Enhancements

Sysctl support

CONFIG_GRKERNSEC_SYSCTL

- Permet d’activer le support de la configuration dynamique des options grsecurity présentes
- Une arborescence /proc/sys/kernel/grsecurity est créée et un pseudo fichier par option est présent
- Si cette option est activée alors toutes les options sont désactivées par défaut
- Une entrée grsec_lock permet de rendre non modifiables toutes les entrées
  - À modifier après configuration des autres entrées

Groupe d’options 8/8

- Buffer Overflow Protection
- Access Control Lists
- Filesystem Protections
- Kernel Auditing
- Executable Protections
- Network Protections
- Sysctl support
- Miscellaneous Enhancements

BSD-style coredumps

CONFIG_GRKERNSEC_COREDUMP

- Change le nom des fichiers coredump
  - sous Linux tout fichier coredump s’appelle core
  - sous *BSD tout fichier coredump s’appelle core.nomDuProcessus
- Avec cette option les fichiers coredump s’appelle core.nomDuProcessus
- Les fichiers core sont générés dans le répertoire courant de l’application
  - Cette option peut permettre de récupérer les fichiers coredump de plusieurs applications différentes s’exécutant dans le même ré pertoire

Quelques cas concrets

- Le serveur d'un particulier
- Le serveur de e-commerce
- Le serveur de la NSA

Le serveur d'un particulier

- + Buffer Overflow Protection
- - Access Control Lists
- + Filesystem Protections
- - Kernel Auditing
- - Executable Protections
- - Network Protections
- - Sysctl support
- - Miscellaneous Enhancements

- Buffer Overflow Protection :
  - PaX est à choisir
    - car la protection mmap est absente du portage du patch openwall

- En fait ici le patch openwall est recommendable car il comporte
  - la protection contre la pile exécutable
    - avec émulation des sauts par trampoline
  - la protection mmap
  - la protection des fichiers temporaires
  - la protection des entrées/sorties

Le serveur de e-commerce

- + Buffer Overflow Protection
- - Access Control Lists
- + Filesystem Protections
- + Kernel Auditing
  - limité à un groupe bien précis
- - Executable Protections
  - + Trusted path execution
- - Network Protections
  - + Socket restrictions (ou ceci est applicable)
- - Sysctl support
- + Miscellaneous Enhancements

- Attention :
  - rien ne protège contre les scripts perl/asp
    - écrits rapidement au fond d'un garage...
  - ici les seules protections sont :
    - la conception sécurisée dès le début du projet
    - l'audit de code tout au long du projet

Le serveur de la NSA

- + Buffer Overflow Protection
- + Access Control Lists
- + Filesystem Protections
- + Kernel Auditing
  - non limité à un groupe
- + Executable Protections
- + Network Protections
  - Sauf Randomized IP IDs, TTL et Altered Ping IDs
- - Sysctl support
- + Miscellaneous Enhancements

- C'est une blague : la NSA préfère utiliser Windows
  - car c'est plus mieux bien certifié... ;-)
  - disons que c'est un serveur avec des informations ultra sensibles
- "Sauf Randomized IP IDs, TTL et Altered Ping IDs"
  - car il ne s'agit que de rendre plus difficile l'identification du systè me, ce qui est obtenable via bien d'autres moyens...
- "Sysctl support" :
  - une fois les bonnes choisies elles n'ont pas à être modifiées

Conclusion

- Il y a des choses utiles
  - pour se protéger
    - temporairement de failles non connues
    - le temps de la mise à jour
  - pour rendre plus complexe la tâche des pirates
  - pour permettre la détection de comportements anormaux
- Suivant le niveau requis, le nombre d'options à activer varie
  - certaines options ne peuvent être appliquées sur des systèmes trop complexes
  - un individu seul ne peut pas gérer correctement toute la journalisation générée par l'audit
- Cela ne dispense pas
  - de démons bien choisis et bien configurés
    - le durcissement du système est un pré-requis obligatoire
      - bastille linux peut être un bon début : http://www.bastille-linux.org/
    - sinon il sera possible de passer root localement
      - et de contourner bien des protections
  - de mises à jour aussi rapides que possible
    - dès la connaissance de toute vulnérabilité

Merci de votre attention

Vous pouvez poser vos questions...

et faire connaître vos remarques...

même par messagerie électronique.

puis réveiller discrètement ceux qui dorment ;-)

Bye, bye...