Detescan : un outil de détection de "scans" réseau

Denis Pugnère
Institut de Génétique Humaine / CNRS UPR 1142
141, rue de la Cardonille, 34396 Montpeller Cedex 5
Denis.pugnere@igh.cnrs.fr
14 octobre 2003

Résumé

Detescan a été développé pour fournir un résumé des événements et incidents réseaux survenus à la frontière d'un réseau de campus ou de laboratoire grâce à l'analyse des journaux de bord (logs) générés par les routeurs. Plusieurs types de routeurs ou de pare-feu sont supportés. Detescan fournit un rapport en deux parties comportant un résumé des événements, suivi par un échantillon des logs concernés par ces évènements.

Mots clefs

Filtrage, sécurité, scan réseau, détection scan

1 Introduction

A quoi servent les journaux de bord (appelés couramment *logs*, traces) ? On ne peut pas répondre à cette question sans avoir eu à résoudre un problème qui a nécessité la recherche d'informations. Or pour analyser un problème à postériori, il faut savoir ce qui s'est passé, pour cela on a besoin de se plonger dans les journaux de bord et remonter dans le temps pour trouver des traces des événements qui ont engendré le problème. Sans traces, la seule possibilité pour un ingénieur réseau de résoudre un problème est de tester une à une toutes les hypothèses de travail.

Sur Internet, des personnes recherchent sans arrêt des failles de sécurité dans les services réseaux accessibles ou sur les systèmes connectés au réseau. Les outils utilisés par ces personnes génèrent des paquets destinés à découvrir si des services réseaux sont disponibles. Ces paquets sont envoyés à une ou plusieurs machines, sur un ou plusieurs services réseaux. Cette technique est appelée *scan* (balayage) réseau. Quand un pare-feu ou un routeur est configuré pour stopper ces paquets, il est également possible de le configurer pour garder une trace des tentatives de *scans*. Cette trace peut être envoyée grâce au client *SYSLOG*[1] intégré au routeur, vers un serveur *SYSLOG* (généralement un serveur Unix ayant un *daemon SYSLOG* activé). Ce serveur *SYSLOG* enregistre les traces envoyées par le routeur dans un fichier.
Généralement, les évènements les plus fréquemment constatés dans les traces générées par les pare-feu ou les routeurs filtrants installés à l'entrée des réseaux de campus ou de laboratoires sont les *scans* réseaux.

L'analyse de *logs* (journaux de bord) des matériels réseaux est une tâche lourde et fastidieuse, c'est toutefois une tâche nécessaire qui incombe à l'administrateur réseau car il est important qu'il sache ce qui se passe sur le réseau qu'il administre. Comme l'administrateur réseau est garant du bon fonctionnement du réseau informatique, il doit être en mesure de détecter les anomalies (perte de connectivité, erreurs *CRC*, problème de routage, incompatibilités), les attaques (*scans, exploits,* déni de service…) sur le réseau informatique mis à la disposition des utilisateurs.
Cependant, il n'est matériellement pas possible de surveiller en continu ces journaux de bord toute la journée d'une part parce qu'il y a beaucoup d'autres choses à faire, mais aussi par ce que la quantité d'informations générée par ces matériels réseaux est considérable. Il est maintenant courant qu'un seul routeur soit capable de générer plusieurs dizaines de lignes de *logs* par seconde. L'analyse "au fil de l'eau" de ces *logs* est impossible sauf pour un outil automatique.

Certains outils comme Anapirate[1] ne supportent que les *logs* générés par des routeurs de marque Cisco et offrent un service de génération automatique de messages d'incident qui sont envoyés aux correspondants des réseaux d'où sont partis les scans. D'autres, comme fwlogwatch[2] sont aussi capables de générer des résumés des rapports sous format texte ou HTML, mais aussi de reconnaître les formats de logs d'IPFilter[3] et de SNORT[4]. On peut noter que Vigilog[5] offre des fonctionnalités équivalentes à Detescan, il offre une meilleure agrégation des *scans*, mais ne reconnaît que les logs des routeurs Cisco.

[1] SYSLOG : RFC 3164 : The BSD syslog Protocol : http://www.ietf.org/rfc/rfc3164.txt

2 Qu'est ce Detescan ?

L'outil Detescan a été développé pour aider l'administrateur réseau en analysant les traces des *logs* générées par différents types de routeurs, commutateurs ou pare-feu (ou potentiellement tout matériel réseau ayant une fonction de filtrage IP de niveau 3 ou 4) pour en produire une synthèse quotidienne (par sélection de la date), hebdomadaire ou mensuelle (en fonction de la périodicité de rotation des *logs*). Cette synthèse peut avoir plusieurs utilisations, comme :

- la détection de *scans* réseaux reçus (de provenance interne ou externe), en différenciant les protocoles (*IP*, *ICMP, TCP, UDP*), les ports *TCP* ou *UDP*, le type et le code *ICMP*,
- la détection de problèmes de configuration des postes clients (adressage *IP* incorrect, configuration *DNS* incorrecte...),
- la détection d'applications qui dans leur fonctionnement ont besoin de se connecter au client,
- la détection de erreurs de filtrage des filtres de paquets,
- la détection de l'utilisation de protocoles ou d'applications non prévus ni autorisés sur le réseau local et contraires à la charte informatique du laboratoire ou à celle de RENATER[2],
- la remontée d'informations vers le CERT RENATER.

A l'origine, Detescan à été développé au sein du LORIA[3] (Gabrielle Feltin, Bertrand Wallrich) puis modifié par Marwan Burelle et Jean-Claude Barbet du LRI[4]. Ce logiciel reconnaissait uniquement les traces des routeurs Cisco et générait une alerte par *scan* détecté. J'ai repris le développement de l'outil en lui ajoutant d'autres fonctionnalités.

Detescan est maintenant le fruit d'une œuvre collective de plusieurs administrateurs réseaux et développeurs, il est écrit en langage PERL, il reconnaît plusieurs formats de *logs* (stockés sur un serveur *SYSLOG*) provenant de différents types de routeurs, commutateurs de niveau 3 ou pare-feu, comme :

- routeurs/commutateurs Cisco IOS v10.x, 11.x et 12.x (contributions collectives),
- routeurs/commutateurs Foundry Networks (code de Joël Marchand),
- routeurs/commutateurs Cabletron (code de Philippe Weill),
- routeurs logiciels Tru64Unix screend (code de Bruno KRINER),
- routeurs/pare-feu logiciels Linux IPCHAINS -kernel 2.2.x- (code de Philippe Weill),
- routeurs/pare-feu logiciels Linux IPTABLES -kernel 2.4.x- (code de Denis Pugnère),
- plugin portscan de la sonde logicielle SNORT (Un*x) (code de Denis Pugnère),
- routeurs/commutateurs Allied Telesyn (code de Denis Pugnère),
- routeurs/commutateurs ExtremeNetworks ExtremeWare 6.x (code de Denis Pugnère),
- pare-feu Cisco PIX (code de Denis Pugnère).

D'autres personnes ont également contribué à la correction des bogues et à l'évolution de Detescan.

3 Ce que Detescan ne fait pas

Detescan n'est pas un outil de détection d'intrusion réseau (NIDS[5]) ou serveur (HIDS[6]) car il ne fait que comptabiliser les traces des paquets qui ont été stoppés par un routeur ou un pare-feu. La version de Detescan décrite ici ne génère pas d'alerte spécifique pour chaque *scan* détecté.

4 Conditions d'utilisation

Pour obtenir le journal de bord de votre routeur sur votre serveur de *logs*, il vous faut :

- un serveur *SYSLOG* (serveur Unix avec le *daemon SYSLOG* activé),
- configurer votre routeur pour qu'il envoie les traces du journal de bord sur le serveur *SYSLOG*,
- configurer des *access-list* du routeur avec une politique : "filtrage de tout sauf",
- programmer le routeur pour *logger* les tentatives de violation des *access-list*,
- avoir des tentatives de violation des *access-list*.

---

[2] RENATER : Réseau National de Télécommunications pour la Technologie, l'Enseignement et la Recherche
La charte de sécurité et de déontologie de RENATER : http://www.renater.fr/Securite/charte_securite.htm
[3] LORIA : Laboratoire lorrain de recherche en informatique et ses applications : http://www.loria.fr
[4] LRI : Laboratoire de Recherche en Informatique : http://www.lri.fr
[5] NIDS : Network Intrusion Detection System
[6] HIDS : Host Intrusion Detection System

De plus, pour pouvoir lancer Detescan, il vous faut :
- interpréteur *PERL* (v5.005 ou ultérieure),
- modules *PERL Getopt::Long* et *Date::Manip*,
- un Agent de Transfert de Messages (*MTA*) comme *Sendmail* ou autre (pour l'envoi des rapports),
- avoir accès au fichier(s) contenant les traces enregistrées par le serveur *SYSLOG*. Le script Detescan sera lancé avec en paramètre les fichiers contenant les traces des routeurs.

5 Fonctionnement de Detescan

Il sélectionne uniquement les traces de paquets qui ont été filtrés par le routeur, les comptabilise et, quand un seuil (paramétrable) est dépassé, rapporte ces traces dans le rapport de synthèse.
Il est configurable par l'intermédiaire des paramètres fournis en ligne de commande. Il est par exemple possible de générer un rapport de *scans* pour une date particulière parmi les traces contenues dans un fichier de rotation hebdomadaire ou mensuelle. Il peut lire les *scans* dans des fichiers compressés (.gz ou .Z)
Le rapport de synthèse fourni par Detescan est constitué de deux parties :
- en première partie, un résumé contenant la liste de tous les *scans* de la période considérée : Cela permet d'avoir une vue générale des *scans* reçus pendant la période,
- en deuxième partie, chaque *scan* est également reporté, mais accompagné d'un extrait des *logs* correspondants : C'est utile quand on doit contacter un site particulier, on peut extraire le *scan* accompagné des quelques lignes de *logs* pour l'envoyer au site en question.

Le format des rapports générés par Detescan est reconnu par le CERT RENATER. L'agrégation des informations contenues dans les rapports envoyés au CERT RENATER leur permet de détecter les problèmes de sécurité potentiels sur les sites interconnectés. Il est conseillé aux sites qui le souhaitent, d'envoyer leur rapports quotidiennement au CERT RENATER.

5.1 Les fonctionnalités générales sont les suivantes :

- Il est possible de sélectionner d'une date d'événements parmi tous ceux contenus dans le(s) fichier(s) de logs. Si aucune date n'est sélectionnée, le rapport sera généré sur la base de tous les évènements contenus dans le(s) fichier(s) de *logs*.
- Les évènements pris en compte sont les suivants :
    - les paquets *loggués* et refusés d'une machine source sur un port destination,
    - les paquets *loggués* et refusés d'une machine source sur une machine destination.
- Il est possible d'appliquer d'un seuil (modifiable) de détection d'évènements :
    - ne sont rapportés que les *scans* sur plus de 5 ports par machine,
    - ne sont rapportés que les *scans* sur plus de 3 machines.
- Il est également possible d'exclure du rapport :
    - les adresses IP de certains réseaux (réseaux déportés, amis), ou de certaines machines (cas des tests réalisés depuis une machine externe),
    - les *scans* à destination de certains ports (*TCP* ou *UDP*), fonction utile dans quand on est victime de *scans* en grand nombre sur un service particulier.

5.2 Options et configuration

Quand on lance le script `detescan.pl` (sans paramètre ou avec le paramètre -h), on obtient le message suivant :

```
% ./detescan.pl
Detescan v20030207
Usage :
./detescan.pl -routeur=type [-date "AAAA/MM/JJ"] [-h] <fichier(s) de logs a lire>
 [-ignoreports=n°port,n°port] [-ignorehosts=@ip] [-ignorelocals=@ip] [-nbmach=x] [-nbport=x]
 [-nblig=x] [-noresolv] [-decalage=x] [-sujet="blabla"] [-dest=personne@email.fr]

 Lit dans le(s) fichier(s) de logs du routeur les traces de scan,
 en restreignant eventuellement la date de recherche (recherche
 sur un jour particulier) et reporte par mail les tentatives de scan

  -routeur=type : type de log généré par le routeur (parametre OBLIGATOIRE)
      type = ios : routeurs Cisco IOS v10.x, v11.x ou v12.x
      type = foundry : routeurs Foundry Networks
      type = ssr3 : Routeurs Cabletron et Xpedition V3.x
      type = screend : démon screend
```

```
      type = ipchains : Routeurs Linux avec IPCHAINS
      type = iptables : Routeurs Linux avec IPTABLES (Netfilter)
      type = allied : Switches Allied Telesyn
      type = snort : sonde snort (www.snort.org)
      type = extreme : Switches ExtremeNetworks (ExtremeWare 6.x)
      type = pix : Cisco PIX

 Les paramètres facultatifs sont :
  -date "AAAA/MM/JJ" : lire les logs pour un jour particulier, exemple "2001/04/13"
  -ignoreports=n°port[,n°port...] : detescan ne devra pas comptabiliser ce(s)
   port(s), exemple : -ignoreports=113,137
  -ignorehosts=@ip[,@ip...] : detescan ne devra pas comptabiliser ce(tte|s)
   machine(s) ou réseau externe (adresses source), exemple :
          -ignorehosts=192.168.0.1,10.11.12.13,172.16
  -ignorelocals=@ip[,@ip...] : detescan ne devra pas comptabiliser ce(tte|s)
   machine(s) ou réseau locale (adresse destination), exemple :
          -ignorelocals=192.168.0.1,10.11.12.13,172.16
  -nbmach=x : nombre de machines minimum concernées pour qu'un scan soit reporté
          defaut=3
  -nbport=x : nombre de ports minimum concernés pour qu'un scan soit reporté
          defaut=5
  -nblig=x : Nombre de lignes maximum que comportera le rapport détaillé (si 0, pas de rapport)
          defaut=10
  -noresolv : Pas de résolution de nom
  -decalage=x : adaptation du code de detescan.pl au format des fichiers syslog, x est le nombre
              de colonnes que le serveur syslog a ajouté.
  -sujet="blabla" : Sujet du mail
  -dest=email : Adresse e-mail de la personne qui recevra le rapport
  -h : afficher cette aide
```

6 Exemples d'utilisation

6.1 Quelques exemples de paramètres utilisés

Exemple de lancement de Detescan sur l'ensemble des fichiers de *log* (format généré par des routeurs de type Cisco) du répertoire `/var/log` et commençant par `cisco`, et sélection des évènements du jour précédent, puis envoi du rapport à l'adresse mèl configurée dans le script `detescan.pl` :

```
$ ./detescan.pl -routeur=ios -date "yesterday" /var/log/cisco*
```

Sélection des évènements du 19 novembre 2003 depuis le fichier compressé `/var/log/extreme.gz` (format généré par un routeur de marque Extreme Networks), génération d'un rapport contenant tous les *scans* de plus de 2 machines puis des *scans* sur plus de 2 services d'une même machine, cinq lignes maximum des *logs* correspondants seront incluses dans le rapport détaillé, seront exclus du rapport les *scans* à destination des ports *TCP* ou *UDP* n° 113 et 137, seront également exclus du rapport les *scans* provenant du réseau *IP* 192.168.0.0/16 et de la machine dont l'adresse *IP* est 10.2.3.4. Le rapport sera envoyé à l'adresse mèl `personne@email.fr` :

```
$ ./detescan.pl -routeur=extreme -date "2003/11/19" -nbmach=2 -nbport=2 -nblig=5 -ignoreports=113,137 \
       -ignorehosts=192.168,10.2.3.4 -dest=personne@email.fr /var/log/extreme.gz
```

Sélection des évènements d'il y a deux semaines (jour pour jour) depuis les fichiers compressés commençant par `routeur` contenus dans le répertoire `/home/archives` (format généré par un routeur de marque Foundry Networks), génération d'un rapport qui sera envoyé à l'adresse mèl configurée dans le script `detescan.pl` :

```
$ detescan.pl -routeur=foundry -date "2 weeks ago" /home/archives/routeur*.gz
```

Dans les exemples suivants, les identités des machines (adresses *IP* ou noms de machines ont été masqués).

6.2 Exemple de rapport de *scan* dû aux virus

L'exemple suivant montre que l'on recevait à l'époque des *scan* sur le port *TCP* 80. La quantité de *scans* était de plus en plus importante et inhabituelle. Ces *scans* ne semblaient pas être générés par des personnes car les outils utilisés scannent en

général linéairement ou pseudo-aléatoirement toutes les adresses d'un réseau *IP* durant une courte durée (inférieure à 10 minutes).
Or, dans le cas présent nous recevions des *scans* "lents" (quelques paquets par heure seulement) provenant d'un grand nombre d'adresses *IP* sources externes, durant plusieurs jours sur seulement un sous-ensemble des adresses *IP* de nos réseaux. Ce type de *scans* ressemble au fonctionnement des vers *Code-Red*[7] ou *Nimda*[8] :

```
From: <xxxx@igh.cnrs.fr> detescan
Subject: [detescan] Resume 23/09/2001

Parametres pour detection scans : nb machines minimum > 1, nb ports minimum > 1
Detescan.pl a détecté les scans suivants à partir des logs Cisco :

 Sep 23 00:56:22 : scan tcp de  vnd.xxxx.ru             sur le port 80 (www) ( 15 machines )
 Sep 23 18:28:10 : scan tcp de  c1859852-a.pinol1.sfba.xxxx.com sur le port 80 (www) ( 2 machines )
 Sep 23 04:27:54 : scan tcp de  dhcp-019-098.cns.xxxx.edu      sur le port 80 (www) ( 2 machines )

Logs de chacun des scans :

 Sep 23 00:56:22 : scan tcp de  vnd.xxxx.ru             sur le port 80 (www) ( 15 machines )
Sep 23 00:56:22 gate 5d10h: IPACL: list 101 denied tcp 195.9.xxxx.37(2705) -> x.y.z.177(80), 2 packets
Sep 23 02:47:35 gate 5d12h: IPACL: list 101 denied tcp 195.9.xxxx.37(4641) -> x.y.z.89(80), 2 packets
Sep 23 05:05:59 gate 5d14h: IPACL: list 101 denied tcp 195.9.xxxx.37(3151) -> x.y.z.48(80), 2 packets
Sep 23 06:10:56 gate 5d15h: IPACL: list 101 denied tcp 195.9.xxxx.37(2547) -> x.y.z.147(80), 2 packets
Sep 23 06:46:29 gate 5d16h: IPACL: list 101 denied tcp 195.9.xxxx.37(3591) -> x.y.z.108(80), 2 packets

 Sep 23 18:28:10 : scan tcp de  c1859852-a.pinol1.sfba.xxxx.com sur le port 80 (www) ( 2 machines )
Sep 23 18:28:10 gate 6d03h: IPACL: list 101 denied tcp 65.5.xxxx.242(2658) -> x.y.z.179(80), 1 packet
Sep 23 19:17:27 gate 6d04h: IPACL: list 101 denied tcp 65.5.xxxx.242(4526) -> x.y.z.52(80), 2 packets

 Sep 23 04:27:54 : scan tcp de  dhcp-019-098.cns.xxxx.edu      sur le port 80 (www) ( 2 machines )
Sep 23 04:27:54 gate 5d13h: IPACL: list 101 denied tcp 132.xxxx.19.98(4395) -> x.y.z.207(80), 1 packet
Sep 23 04:35:17 gate 5d14h: IPACL: list 101 denied tcp 132.xxxx.19.98(2128) -> x.y.z.69(80), 1 packet
```

6.3 Exemple de rapport de *scans* montrant différents types de problèmes

Cet exemple montre :
- qu'il y a des problèmes de configuration sur les postes clients :
  - les première et deuxième lignes du résumé montrent que les clients (internes) font des requêtes *NTP*[9] aux serveurs de temps : *time1.euro.apple.com* et *time2.euro.apple.com*
  - la cinquième ligne du résumé montre qu'un client fait des requêtes au serveur *DNS*[10] *ns1.club-internet.fr*
- que l'on a reçu des *scans* provenant de l'extérieur (troisième, quatrième et sixième lignes du résumé)

```
From: <xxxx@igh.cnrs.fr> detescan
Subject: [detescan] Resume 06/11/2001

parametres pour detection scans : nb machines minimum > 1, nb ports minimum > 1
detescan.pl a détecté les scans suivants à partir des logs Cisco :

 Nov 6 112959  scan udp de   time1.euro.apple.com    sur le port 123 (ntp) ( 2 machines )
 Nov 6 100352  scan udp de   time2.euro.apple.com    sur xxxx.igh.cnrs.fr ( 3 ports )
 Nov 6 120213  scan icmp de  165.21.xxxx.39          de type icmp (8/0) ( 3 machines )
 Nov 6 003654  scan tcp de   216.167.xxxx.211        sur le port 53 (domain) ( 3 machines )
 Nov 6 113252  scan udp de   ns1.club-internet.fr    sur xxxx.igh.cnrs.fr ( 6 ports )
 Nov 6 130229  scan tcp de   lns3-148.xxxx.w.club-internet.fr   sur le port 21 (ftp) ( 2 machines )

Logs de chacun des scans :

 Nov 6 112959  scan udp de   time1.euro.apple.com    sur le port 123 (ntp) ( 2 machines )
Nov  6 112959 gate 4w1d IPACL list 101 denied udp 194.151.19.93(123) -> x.y.z.34(123), 1 packet
Nov  6 145433 gate w1d IPACL list 101 denied udp 194.151.19.93(123) -> x.y.z.51(123), 1 packet

 Nov 6 100352  scan udp de   time2.euro.apple.com    sur xxxx.igh.cnrs.fr ( 3 ports )
```

[7] Code-Red : http://securityresponse.symantec.com/avcenter/venc/data/codered.worm.html
[8] Nimda : http://securityresponse.symantec.com/avcenter/venc/data/w32.nimda.a@mm.html
[9] NTP : Network Time Protocol : RFC 1305, RFC 2030
[10] DNS : Domain Name System : RFC 1035

```
Nov  6 100352 gate w1d IPACL list 101 denied udp 194.151.19.94(123) -> x.y.z.31(49158), 1 packet
Nov  6 220407 gate w1d IPACL list 101 denied udp 194.151.19.94(123) -> x.y.z.31(49207), 1 packet

 Nov 6 120213  scan icmp de  165.21.xxxx.39          de type icmp (8/0) ( 3 machines )
Nov  6 120213 gate w1d IPACL list 101 denied icmp 165.21.xxxx.39 -> x.y.z.255 (8/0), 1 packet
Nov  6 164048 gate w1d IPACL list 101 denied icmp 165.21.xxxx.39 -> x.y.z.255 (8/0), 1 packet

 Nov 6 003654  scan tcp de   216.167.xxxx.211        sur le port 53 (domain) ( 3 machines )
Nov  6 003654 gate w0d IPACL list 101 denied tcp 216.167.xxxx.211(1307) -> x.y.z.237(53), 1 packet
Nov  6 003657 gate w0d IPACL list 101 denied tcp 216.167.xxxx.211(1308) -> x.y.z.238(53), 1 packet

 Nov 6 113252  scan udp de   ns1.club-internet.fr    sur xxxx.igh.cnrs.fr ( 6 ports )
Nov  6 113252 gate w1d IPACL list 101 denied udp 194.117.200.10(53) -> x.y.z.216(1768), 1 packet
Nov  6 113839 gate w1d IPACL list 101 denied udp 194.117.200.10(53) -> x.y.z.216(1770), 1 packet

 Nov 6 130229  scan tcp de   lns3-148.xxxx.w.club-internet.fr    sur le port 21 (ftp) ( 2 machines )
Nov  6 130229 gate w1d IPACL list 101 denied tcp 213.44.xxxx.148(4046) -> x.y.z.34(21), 1 packet
Nov  6 130238 gate w1d IPACL list 101 denied tcp 213.44.xxxx.148(4063) -> x.y.z.65(21), 1 packet
```

Il est possible d'exécuter Detescan toutes les nuits de manière automatique par l'intermédiaire de l'utilitaire *cron*, comme par exemple ceci :

```
#tous les jours a 7h00 du matin: detescan sur logs cisco
0 7 * * *      $HOME/cron/detescan.pl -routeur=ios11 -date yesterday /var/log/cisco*
```

7 Améliorations possibles

Dans les prochaines versions de Detescan, il est prévu de lui ajouter d'autres fonctionnalités comme :
- le support d'autres filtres *IP* (*ipfilter*),
- le support d'intervalles de dates,
- possibilité de sélectionner les adresses destinations dans le but d'offrir aux administrateurs de réseaux campus de générer des rapports pour les différents composantes de leur réseaux.

Toutes les suggestions sont les bienvenues.

8 Informations pratiques

Page web : *http://www.igh.cnrs.fr/perso/denis.pugnere/detescan*
Detescan disponible sur le serveur FTP à l'adresse : *ftp://ftp.igh.cnrs.fr/pub/unix/detescan/detescan.tar*
Liste de diffusion : *detescan [at] igh.cnrs.fr*

La version de Detescan décrite dans ce document est la v20030207.

Références

[1] Luc Veillon : Anapirate : *http://www.orleans.ird.fr/pub/anapirate/anapirate-site.html*
[2] Fwlogwatch : *http://cert.uni-stuttgart.de/projects/fwlogwatch*
[3] IPFilter : *http://coombs.anu.edu.au/~avalon/*
[4] SNORT : The Open Source Network Intrusion Detection System : *http://www.snort.org*
[5] Jose Marcio Martins da Cruz : Vigilog : *http://www.ensmp.fr/CC/vigilog*