Présentation de Xinetd

Extended Internet Services Daemon

A.F.U.U Association Française des utilisateurs d'Unix
Bruno.Beaugrand@inria.fr

Juin 1994

Abstract

Ce topo a pour but de présenter de maniére générale les fonctionnalitées de ce service de substitution à INETD, largement utilisé sur les plates formes UNIX.

1 Présentation

C'est un remplaçant de inetd [1], ce programme n'est pas compatible avec inetd, son format de configuration étant complètement différent.

Le daemon xinetd:

- Lancé de la même manière que inetd (ex : dans le fichier de démarrage /etc/rc à l'emplacement de inetd), xinetd est le seul processus à l'écoute des connexions listées dans son fichier de configuration.

Xinetd permet d'avoir un controle d'access et de "tracer" les événements.

Xinetd peut être lancé avec de nombreux paramètres.

xinetd [parametres]

en voici une liste:

- -d (debug) - Permet d'obtenir une trace des événements (trés intérressant à la mise en route).
- -syslog (syslog facility) - Utilisation pour journaliser des événement au démarrage et à la maintenance de Xinetd (ajout, modification dans le fichier de configuration).
- -filelog (logfile) - idem que syslog mais rangé ici dans un fichier.
- -f config-file (fichier de configuration) - permet de spécifier un fichier de configuration différent du défault (/etc/xinetd.conf).
- -pid (process ID) - Le numéro du processus est affiché sur la sortie standard.
- -loop rate (boucle) - Limite le nombre de boucle avant de considérer le service en erreur et de le désactiver.
- -limit process limit - Limite le nombre de processus pouvant être démarré par Xinetd.
- -cc interval - spécifie un interval de temps pour une vérification du fichier de configuration de Xinetd.

  De plus, de nonbreuses opérations peuvent être réalisées sur le processus sur certains signaux.

[1] *The Internet Service daemon*

- USR1 - Relecture de configuration (soft).
- USR2 - Relecture de configuration (hard).
- QUIT - Fin du Programme.
- TERM - Termine la fin des services avant de terminer le programme.
- HUP - "Dump" la configuration dans un fichier (default : /tmp/xinetd.dump).
- IOT - Vérification de la structure des données.

  Voir le manuel xinetd(8), xinetd.log(5) et xinetd.conf(5).

2 Le Démarrage

Exemple du fichier de configuration dans /etc/rc

if [-f /usr/etc/xinetd -a -f /etc/xinetd.conf]; then

/usr/etc/xinetd -filelog /var/log/xinetd/filelog -cc 300 -pid; echo -n ' xinetd'

fi

3 Le fichier de configuration

Comparaison entre le fichier de configuration de inet et de xinetd.

3.1 Le fichier inetd.conf

Le fichier de configuration contient une entrée par ligne, décrivant le service que inetd doit gérer (l'entrée de ce fichier peut varier en fonction du systéme utilisé).

Le fichier inetd.conf se présente de cette manière:

```
# Ftp and telnet are standard Internet services.
#
ftp     stream  tcp     nowait  root    /usr/etc/in.ftpd        in.ftpd
telnet  stream  tcp     nowait  root    /usr/etc/in.telnetd     in.telnetd
#
```

- 1er champ : Le service.
- 2éme champ: Le type de socket.
- 3éme champ: Le protocole.

- 4éme champ: L'opération à faire sur le processus.
- 5éme champ: L'utilisateur sous lequel le serveur tourne.
- 6éme champ: L'endroit où se trouve le programme.
- 7éme champ: Le programme.

3.2 Le fichier xinetd.conf

Voici la configuration de xinetd correspondant au fichier inetd. Ici un champ par ligne.

```
service ftp
{
        socket_type    = stream
        protocol        = tcp
        wait            = no
        user            = root
        server          = /usr/etc/in.ftpd
}

service telnet
{
        socket_type    = stream
        protocol        = tcp
        wait            = no
        user            = root
        server          = /usr/etc/in.telnetd
}
```

A cela viennent s'ajouter des attributs suplémentaires:

- group : détermine le groupe pour le processus serveur.

- instances : détermine le nombre de serveur qui peuvent être actif simultanément.

- nice : détermine la priorité.

- server_args : détermine les argument à passer au serveur.

- only from : détermine si le service est disponible pour le host distant.

- no_access : détermine le host distant pour lequel ce service n'est pas accessible.

- access_times : détermine l'interval de temps durant lequel le service est accessible.

  Concernant les deux facilitées précédentes les formats peuvent être comme ceci:

- une adresse IP
- un ensemble d'adresses
- un réseau (cf /etc/networks).
- un "hostname".

exemple:

```
defaults
{
        log_type        = FILE /var/log/xinetd/defaults
        log_on_success  = PID
        log_on_failure  = HOST RECORD
        only_from       = 194.25.2.0 128.22.0.0
}

defaults
{
        log_type        = FILE /var/log/xinetd/defaults
        log_on_success  = PID
        log_on_failure  = HOST RECORD
        only_from       = 194.25.2.{1,3,4,5}
}

defaults
{
        log_type        = FILE /var/log/xinetd/defaults
        log_on_success  = PID
        log_on_failure  = HOST RECORD
        only_from       = reseau-autorise
}
```

Avec dans le fichier /etc/network :

```
reseau-autorise 1.2.3
```

- log_type : indique la manière de garder les traces avec SYSLOG ou dans un FICHIER.
- log_on_success : détermine le type d'information à garder en cas de réussite avec le type d'informations complémentaires à garder.
  - PID : log le numéro du processus serveur.
  - HOST : log l'adresse du distant.
  - USERID : log le numero de l'uid distant (cf RFC 1413).
  - EXIT : log la fin du service.

- DURATION : log la durée.

- log_on_failure : détermine le type d'information à garder en cas de refus à cela des informations complémentaires peuvent être logées.
  - HOST : log l'adresse du distant.
  - USERID : log le numéro de l'uid distant (cf RFC 1413).
  - ATTEMPT : acquittement.
  - RECORD : Informations sur le distant.

- env rpc_version, port, ...

  Il n'est pas nécessaire de définir chaque attribut pour chaque service.Un attribut default { .... } peut être défini dans la configuration.

```
service ftp
{
        socket_type     = stream
        protocol         = tcp
        wait             = no
        user             = root
        instances        = 1
        server           = /usr/etc/in.ftpd
        log_type        = FILE /var/log/xinetd/info-ftp
        log_on_success  += USERID PID HOST EXIT DURATION
        log_on_failure  += USERID HOST ATTEMPT RECORD
        access_times    = 8:00-19:00
        only_from       = 155.31.132.12
}
```

4 Trace des fichiers de LOG

4.1 Le fichier filelog

- Au démarrage de la machine :

  `94/6/6@21:35:13 xinetd[115]: NOTICE: Started working: 27 available services`

- Après un kill -USR1 du processus Xinetd: (liste l'ensemble de la configuration et notifie l'entrée. Indique si un probléme concernant la syntaxe apparaît.

```
94/6/4@15:25:38 xinetd[118]: NOTICE: Starting hard reconfiguration
94/6/4@15:25:39 xinetd[118]: NOTICE: readjusting service ftp
94/6/4@15:25:39 xinetd[118]: NOTICE: readjusting service telnet
94/6/4@15:25:39 xinetd[118]: NOTICE: readjusting service name
94/6/4@15:25:39 xinetd[118]: NOTICE: readjusting service shell
94/6/4@15:25:39 xinetd[118]: NOTICE: readjusting service login
94/6/4@15:25:39 xinetd[118]: NOTICE: readjusting service exec
94/6/4@15:25:39 xinetd[118]: NOTICE: readjusting service comsat
94/6/4@15:25:39 xinetd[118]: NOTICE: readjusting service talk
94/6/4@15:25:39 xinetd[118]: NOTICE: readjusting service uucp
94/6/4@15:25:39 xinetd[118]: NOTICE: readjusting service finger
94/6/4@15:25:39 xinetd[118]: NOTICE: readjusting service time-stream
94/6/4@15:25:39 xinetd[118]: NOTICE: readjusting service time-dgram
94/6/4@15:25:39 xinetd[118]: NOTICE: readjusting service echo-stream
94/6/4@15:25:39 xinetd[118]: NOTICE: readjusting service echo-dgram
94/6/4@15:25:39 xinetd[118]: NOTICE: readjusting service discard-stream
94/6/4@15:25:39 xinetd[118]: NOTICE: readjusting service discard-dgram
94/6/4@15:25:39 xinetd[118]: NOTICE: readjusting service daytime-stream
94/6/4@15:25:39 xinetd[118]: NOTICE: readjusting service daytime-dgram
94/6/4@15:25:39 xinetd[118]: NOTICE: readjusting service chargen-stream
94/6/4@15:25:39 xinetd[118]: NOTICE: readjusting service chargen-dgram
94/6/4@15:25:39 xinetd[118]: NOTICE: readjusting service rquotad
94/6/4@15:25:39 xinetd[118]: NOTICE: readjusting service rstatd
94/6/4@15:25:39 xinetd[118]: NOTICE: readjusting service rusersd
94/6/4@15:25:39 xinetd[118]: NOTICE: readjusting service sprayd
94/6/4@15:25:39 xinetd[118]: NOTICE: readjusting service walld
94/6/4@15:25:39 xinetd[118]: NOTICE: readjusting service pop
94/6/4@15:25:39 xinetd[118]: NOTICE: Reconfigured: new=1 old=26 dropped=0 (servic
```

- Erreur dans la configuration.

```
94/4/13@23:29:45 xinetd[208]: NOTICE: readjusting service walld
94/4/13@23:29:45 xinetd[208]: NOTICE: Reconfigured: new=0 old=25 dropped=0 (services)
94/4/16@15:42:41 xinetd[104]: WARNING: Bad log-on-success flag: RECORD [line=11]
94/4/16@15:42:42 xinetd[104]: ERROR: Multi-threaded services cannot be intercepted: ft
```

5 Quelques fichiers de Log

5.1 finger

```
94/6/4@16:18:40: START: finger pid=1114 from=1.2.3.4
94/6/4@16:18:41: EXIT: finger status=0 pid=1114 duration=1(sec)
94/6/4@16:21:36: START: finger pid=1123 from=1.2.3.4
94/6/4@16:21:37: EXIT: finger
```

5.2 ftp

- Connexion Acceptée:

```
94/4/24@14:26:51: START: ftp pid=196 from=1.2.3.4
94/4/24@14:26:54: EXIT: ftp status=0 pid=196 duration=3(sec)
```

- Connexion Refusée:

```
94/4/24@14:27:25: EXIT: ftp status=0 pid=198 duration=28(sec)
94/6/11@16:04:36: FAIL: ftp address from=5.6.7.8
```

- Limitation du nombre de processus pour le service

```
94/6/11@16:06:45: START: ftp pid=244 from=1.2.3.4
94/6/11@16:06:48: FAIL: ftp service_limit from=1.2.3.4
94/6/11@16:07:10: EXIT: ftp status=0 pid=244 duration=25(sec)
```

- Rêquete effectuée dans une tranche horaire non allouée.

```
94/6/11@16:09:41: FAIL: ftp time from=192.31.252.17
```

- Commande spray:

```
94/2/25@11:21:29: FAIL: sprayd address from=1.2.3.2
94/2/25@11:22:20: START: sprayd pid=4711 from=4.5.6.7
```

- Commande talk:

```
94/6/14@10:16:49: FAIL: talk address from=1.2.3.2
```

6 Le Dump

Un kill -IOT “processus xinetd” donne :

```
94/6/11@16:22:29 xinetd[119]: INFO: Consistency check passed
```

Un kill -HUP “processus xinetd” donne :

```
94/6/11@16:25:26 xinetd[119]: INFO: generated state dump in file /usr/tmp/xinetd.dump
```

la trace du fichier dump:

```
more  /usr/tmp/xinetd.dump


INTERNAL STATE DUMP: xinetd Version 2.1.2
Current time: Sat Jun  4 15:08:27 1994
Services + defaults:
Service defaults
        No logging
Service = ftp
        State = Active
        Service configuration: ftp
                id = ftp
                socket_type = stream
                Protocol (name,number) = (tcp,6)
                Instances = 1
                Server = /usr/etc/in.ftpd
                Server argv = in.ftpd
                Access times = 08:00-19:00
                Only from:  1.2.3.4(NUMERIC)
                Logging to file: /var/log/xinetd/info-ftp (no limits)
                Log_on_success flags = HOST DURATION EXIT PID USERID
                Log_on_failure flags = HOST ATTEMPT RECORD USERID
        running servers = 0
        retry servers = 0
        attempts = 0
        service fd = 4
        shutdown function = (null)

Service = telnet
        State = Active
        Service configuration: telnet
                id = telnet
                socket_type = stream
                Protocol (name,number) = (tcp,6)

..........


active_services = 27
available_services = 27
descriptors_free = 218
running_servers = 0
Logging service = enabled
Shutdown service = enabled

max_descriptors = 256
process_limit = 0
config_file = /etc/xinetd.conf

END OF DUMP
```

7 Ou le récupérer

La version courante est la 2.1.4 sur le serveur FTP (ftp.irisa.fr dans pub/mirrors/xinetd).

8 Conclusion

L'ensemble des options de configuration de Xinetd permettent d'avoir un grand niveau d'information et de gérer chaque service de maniére indépendant et fin sur le numéro IP, les horaires et de garder une trace des différentes opérations, sans avoir à développer des outils autour de inetd.

Les choses particuliérement interressantes sont le temps d'accés, la restriction sur les horaires et la limitation du nombre de processus pour un service donné, l'accés étant contolé par tcp_wrapper.

Un ensemble de ces produits Tcp_Wrapper, Xinetd, Secure C2 permet de retracer relativement facilement les opérateurs sur un service.
Xinetd cohabite trés bien avec Tcpd. Facile à installer et à debugger, il existe un programme (itox) pour transcrire le fichier de configuration de inetd (inetd.conf) en un fichier pour xinetd (xinetd.conf).

Références

[1] [RFC862] *Echo Protocol*

[2] [RFC863] *Discard Protocol*

[3] [RFC864] *Character Generator Protocol*

[4] [RFC867] *Character Generator Protocol*

[5] [RFC868] *Time Protocol*

[6] [RFC1413] *Identification Protocol*

[7] [RFC931] *Authentification Server*

[8] [Man] *Xinetd(8) Xinetd.log(5) Xinetd.conf(5)*