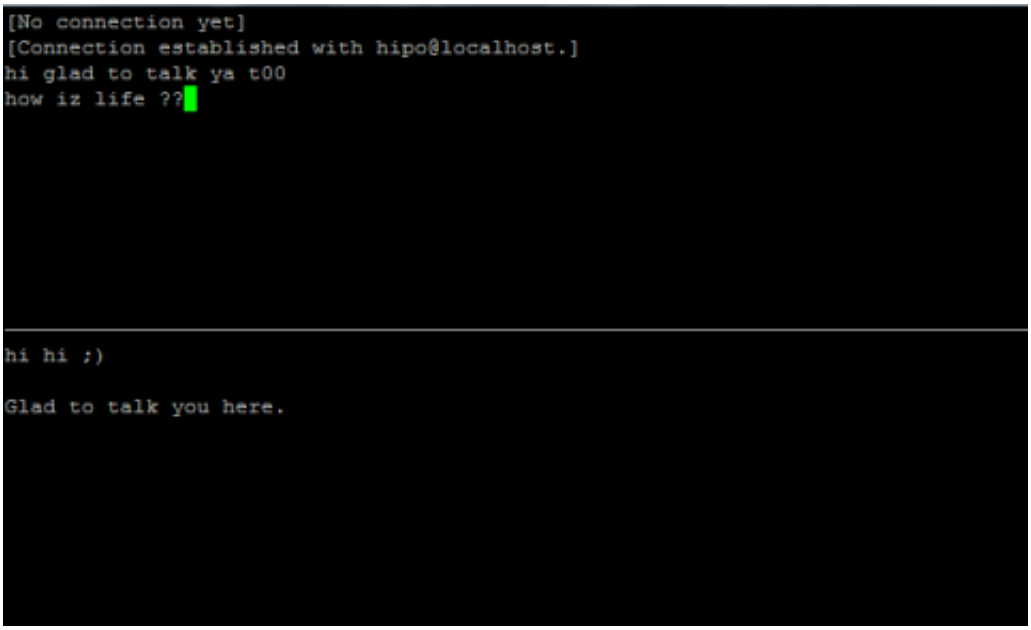
Installing and making work interactive peer to peer chat throught talkd using talk on Debian GNU / Linux

Author : admin



After writting in previous article on how **talk** be used to handle interactive chat console sessions on FreeBSD, I thought of dropping a few lines on how same is done on Debian, so here is how:

1.; Install *talk* and *talkd*

noah:/home/hipo# apt-get --yes install talk talkd
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following NEW packages will be installed:
talk talkd
0 upgraded, 2 newly installed, 0 to remove and 93 not upgraded.
Need to get 19.0 kB/42.3 kB of archives.
After this operation, 201 kB of additional disk space will be used.
Get:1 http://ftp.nl.debian.org/debian/ stable/main talkd amd64 0.17-14 [19.0 kB]
Fetched 19.0 kB in 0s (67.1 kB/s)
Selecting previously deselected package talk.
(Reading database ... 90%
Unpacking talk (from .../talk_0.17-14_amd64.deb) ...
Selecting previously deselected package talkd.

```
Unpacking talkd (from .../talkd_0.17-14_amd64.deb) ...
Processing triggers for man-db ...
Setting up talk (0.17-14) ...
update-alternatives: using /usr/bin/netkit-ntalk to provide /usr/bin/talk (talk) in auto mode.
Setting up talkd (0.17-14) ...
```

2.;; Check and make sure talk and ntalkd lines are present in */etc/inetd.conf*

```
noah:/home/hipo# grep -i talk /etc/inetd.conf
#:BSD: Shell, login, exec and talk are BSD protocols.
talk dgram udp wait nobody.tty /usr/sbin/in.talkd in.talkd
ntalk dgram udp wait nobody.tty /usr/sbin/in.ntalkd in.ntalkd
```

Now you probably wonder why are there two lines in */etc/inetd.conf* for))

in.talkd and **in.ntald**

in.talkd daemon's aim is to deliver talk sessions between logged in users on one Linux host with few logged in users willing to talk to each other locally;;
Wheter *in.ntalkd* is designed to serve interactive user talks between the host where *in.ntalkd* is installed and remote systems ruwhich have the **talk** client program installed. Of course in order for remote talks to work properly the firewall (if such has to be modified to allow in.ntalkd chats. I've never used in.ntalkd and on most machines having in.ntald hanging around from inetd, could be a potential security hole so, for people not planning to initiate remote *TALK*s between Unix / Linux / BSD hosts on a network it is a good practice the ntalkd line seen above in *inetd.conf* to be commented out ::;

```
noah:/home/hipo# grep -i talk /etc/inetd.conf
#:BSD: Shell, login, exec and talk are BSD protocols.
talk dgram udp wait nobody.tty /usr/sbin/in.talkd in.talkd
#ntalk dgram udp wait nobody.tty /usr/sbin/in.ntalkd in.ntalkd
```

3.;;; Restart openbsd-inetd init script and talk is ready to use

```
noah:~# /etc/init.d/openbsd-inetd restart
* Restarting internet superserver inetd
```

Onwards to use talk between two users the syntax is same like on other BSD, as a matter of fact **TALK** - *console / terminal interactive chat* originally was developed for the **4.2BSD UNIX** release ;; the Linux code is a port of this BSD talk and not rewrite from scratch.

Using talk between two logged in users on **pts/1** (lets say user test) and **tty1** (user logged as root) is done with:

```
noah:~$ tty
noah:~$ talk root@localhost tty1
/dev/pts/1
```

On **tty1** the user has to have enabled *Talk* sessions request, by default this behaviour in Debian and probably other Debian based Linuxes (Ubuntu) for instance is configured to have talks disabled, i,e ,,,

```
root@noah:~# mesg
is n
```

Enabling it on root console is done with:

```
root@noah:~# mesg y
```

Once enabled the root will be able to see the TALK service requests on tty1 otherwise, the user gets nothing. With enabled messaging the root user will get on his tty:

```
Message from TalkDaemon@his_machine...
talk: connection requested by your_name@your_machine.
talk: respond with: talk your_name@your_machine
```

So on the root console to reply back to talk chat request:

```
noah:~$ talk hipo@localhost
```