Installing and running IRRd

Source code and documentation for IRRd is available online at:
http://www.irrd.net, current version is 2.2.3

Also, a user guide is included as part of the distribution as irrd-user.pdf.

IRRd software is used to run Merit's RADB routing registry which can be queried at whois.radb.net. For more info on the RADB, see www.radb.net.

To build and install the distribution, execute the following commands:

```
cd src
./configure
make
make install
```

Binaries are installed in /usr/local/sbin by default.

Create the directory for databases

```
# cd /var
#mkdir irr
#cd irr
#mkdir databases
```

Running IRRd

gaurab@ktm# irrd

gaurab@ktm# telnet localhost 5674

Database Creation and Interactive Configuration

```
# telnet localhost 5674
Trying 127.0.0.1...
Connected to localhost.localdomain (127.0.0.1).
Escape character is '^]'.
IRRd version 2.2.3 [16Dec2004]

No Access Verification

host0 Password: *
```

```
host0 IRRd#
host0 IRRd#            ?
  config
  dbclean
  delete
  enable                          Enable
  exit                            Quit from the current level
  export
  irrdcacher
  kill                            Kill off the IRRd daemon
  mirror
  quit                            Quit from the current level
  read
  reboot                          Reboot IRRd
  reload
  set
  show
  write
host0 IRRd#
host0 IRRd# sh ?
  config
  connections                     Show current IRR connections
  database                        Show database status
  ip
  mirror-status
  statusfile                      Show statusfile location.
  threads                         Show status of threads
  timers
  version                         Show version and uptime
host0 IRRd# sh
host0 IRRd# sh database
Listening on port 43 (fd=8)
Memory-only indexing
Default Database Query Order:

  Database       Size (kb)    Rt Obj    AutNum Obj
Serial #    Last Export #
-------------    ---------    ------    ----------
--------    -------------
 TOTAL                 0.0         0             0

Creating Databases

host0 IRRd# enable
host0 IRRd# config
host0 Config> irr_direcotry /var/irr/databases
host0 Config> irr_database local authoritative
host0 Config> irr_database local access 10
host0 Config> irr_database local write-access 10
host0 Config> irr_database local mirror-access 20
```

```
host0 Config> access-list 10 permit 10.1.0.1/20 refine
host0 Config> access-list 20 permit 10.1.0.0/24 refine
host0 Config> access-list 99 deny all
host0 Config>
 host0 Config> debug server file-name /var/log/irrd.log
 host0 Config> debug server syslog
 host0 Config> debug submission file-name /var/log/irr-
email.log
host0 Config> irr_max_connections 100
host0 Config> ^Z
host0 IRRd# write
Writing configuration file to /etc/irrd.conf
host0 IRRd# sh config
##############################################################
##########
# MRTd -- MRT version 2.2.3 [16Dec2004]
##############################################################
##########
#
debug all /tmp/irrd.log 0
#
line vty
  login
  password ********
!
irr_database local authoritative
irr_database local access 10
irr_database local write-access 10
irr_database local mirror-access 20
irr_database local clean 172800
access-list 10 permit 10.1.0.1/20 refine
access-list 20 permit 10.1.0.0/24 refine
access-list 99 deny all
irr_max_connections 100
host0 IRRd#
```

Now exit the configuration mode and restart irrd. When you log in again, after providing the password, do a 'show database', you will see one database.

```
[root@host0 gaurab]# telnet localhost 5674
Trying 127.0.0.1...
Connected to localhost.localdomain (127.0.0.1).
Escape character is '^]'.
IRRd version 2.2.3 [16Dec2004]

User Access Verification

host0 Password: ********
host0 IRRd# sh database
Listening on port 43 (fd=10)
Memory-only indexing
Default Database Query Order: local
```

```
  Database       Size (kb)    Rt Obj    AutNum Obj    Serial #    Last
Export #
-------------    ---------    ------    ----------    --------
-------------
 local                0.0         0             0           0
 TOTAL                0.0         0             0

local  AUTHORITATIVE
   Last email/tcp update Never
   Last loaded 07:06:34 02/19/2005
   Next dbclean in 56:26:47
host0 IRRd#
```

Setting up the contact e-mail

```
host0 IRRd# config
host0 Config> db_admin gaurab@localhost
host0 Config>
host0 IRRd# write
```

Setting up submission mechanism

The irr_rpsl_submit program is used for populating the database. To accetp e-mails submissions, you need to make the right entry into the /etc/aliases file. On RedHat and Derived Systems (like Fedora), you will also need to make an entry into /etc/smrsh.

```
#vi /etc/aliases

db-submit: "|/usr/local/sbin/irr_rpsl_submit -f /
etc/irrd.conf"
```

Save and exit, then create a symlink for irr_rpsl_submit (if you are using Fedora Linux)

```
#cd /etc/smrsh
#ln -s /usr/local/sbin/irr_rpsl_submit irr_rpsl_submit
#newaliases

To allow for TCP connections, add the following to /
etc/services

irr_rpsl_submit  8888/tcp              # irrd submission port

Now create a entry in /etc/xinetd.d

#cd /etc/xinetd.d
#vi irr_rpsl_submit
service irr_rpsl_submit
{
        disable = no
        socket_type             = stream
        protocol                = tcp
        wait                    = no
        user                    = root
```

```
        server                  = /
usr/local/sbin/irr_rpsl_submit
        server_args             = -D
}
```

Save and exit
restart xinetd.d. Netstat should now show the rpsl submit daemon running

```
#/etc/init.d/xinetd restart
#netstat -l
```

You are done :-)

Populating the Database
We'll limit ourselves to create three object types in our Routing Registry

```
Maintainer object   Specifies authorization for objects
AS object           Administration and routing policy of an AS
Route object        A single route to be added to the registry
```

Setting up the Maintainer Objects
Maintainer objects needs to be setup manually. The automated mechanisms work only after the maintainer objects have been setup. The maintainer object is self-referenced, as every object needs a mnt-by attribute.

Edit /var/irr/databases/local.db and add the following objects.

```
mntner:         MAINT-POP1
descr:          Maintainer for POP1
admin-c:        MYRR-GRU
tech-c:         MYRR-GRU
upd-to:         gaurab@localhost
mnt-nfy:        gaurab@localhost
mnt-by:         MAINT-POP1
auth:           MAIL-FROM gaurab@localhost
auth:           MAIL-FROM gaurab@localhost
changed:        gaurab@localhost <mailto:gaurab@localhost> 100000
source:         MYRR

aut-num:        AS3856
as-name:        PCH-AS
descr:          Packet Clearing House Global AS
admin-c:        MYRR-GRU
tech-c:         MYRR-GRU
import:         from AS-ANY  accept ANY
export:         to AS-ANY  announce ANY
notify:         gaurab@localhost
mnt-by:         MAINT-POP1
changed:        gaurab@localhost 10000
source:         MYRR
```

```
route:          10.1.0.0/24
descr:          prefix for me
origin:         AS3856
mnt-by:         MAINT-POP1
changed:        gaurab@localhost 10000
source:         MYRR
```

The autnum and route objects were added to test the install. Now login into the irrd and do the following

```
host0 IRRd# reload local
host0 IRRd# sh database
```

You should be happy now :-)

Now try

```
$ whois -h localhost 10.1.0.1
```

If you see what you expect, you are done.

Populating the Database by e-mail
send a e-mail to "db-submit@localhost <mailto:db-submit@localhost>" with the exact object you want to add. Send it from an e-mail address that is authorized in the MAIL-FROM attribute.

IRRToolSet Installation

To install, you need g++ 2.8.0 or higher, libstdc++-2.8.0 or higher, bison, flex and gmake. Please note that the toolset is not gcc3 compliant (yet).

You can download the source and install it. (Since the compile failed on the lab PCs, we'll use the binaries for Linux)

You just need to untar and copy the binaries to the appropriate directory

$ tar -xzf IRRToolSet-4.7.3-pc-linux-gnu.tar.gz
$ cd IRRToolSet-4.7.3-pc-linux-gnu/bin
# cp * /usr/local/bin/*
# cd ../man/man1
# cp * /usr/local/man/man1/
# cd ../man3
# cp * /usr/local/man/man3/

Running RtConfig
$ RtConfig -h localhost
RtConfig >