EnduroX Administration Manual

REVISION HISTORY

NUMBER	DATE	DESCRIPTION	NAME
1.0	2012-12	Initial draft	MV

Contents

Chapter 1

How to configure EnduroX

To configure EnduroX you have to finish several steps.

- Have a seperate system user for each EnduroX instance.
- Setup System Environment (mount mq file system, configure mq params)
- Setup environment configuration
- Setup *ndrxdconfig.xml*
- Setup *ndrxdebug.conf*
- Startup the application

Chapter 2

Setup System Environment

In this chapter will be descirbed typical GNU/Linux system configuration required for EnduroX. Server Posix queue paramters must be set-up:

Typical configuration would look like (normally you can put this in /etc/rc.local):

```
# Mount the /dev/mqueue
mkdir /dev/mqueue
mount -t mqueue none /dev/mqueue
echo 32000 > /proc/sys/fs/mqueue/msg_max
echo 10000 > /proc/sys/fs/mqueue/msgsize_max
echo 10000 > /proc/sys/fs/mqueue/queues_max
```

Which means:

- *msg_max* - Maximum number of messages in queue (32K)
- *msgsize_max* - Maximum message size, ~10KB (can be set up up to 32K)
- *queues_max* - Maximum number of queues (10K)

See [MQ_OVERVIEW] for more details about Posix queues on GNU/Linux.

Also do not remember to update EnduroX environment variables to reflect these settings. Variabales like:

- *NDRX_MSGMAX*
- *NDRX_MSGSIZEMAX*

See [EX_ENV] for more configuration flags.

Update also should be done to system security configuration. I.e. update to */etc/security/limits.conf* is required, following lines must be added:

```
*               soft    msgqueue        -1
*               hard    msgqueue        -1
```

Chapter 3

Setup environment configuration

EnduroX depends lot of Environment variables. See manpage of *ex_env* ([EX_ENV]) to see all paramters that must be setup. There is also sample configuration provided. Normally it is expected that seperate shell script file is setup containing all paramters. Then to load the environment, login with EnduroX user in, and run following command in your app dir, for example:

```
$ cd /endurox/app/conf
$ . setapp
```

Chapter 4

Setup *ndrxdconfig.xml*

Vital part of EnduroX is *ndrxdconfig.xml* config. Full path to it is configured in *NDRX_CONFIG* environment variable. To see how to setup see manpage of it [NDRXCONFIG].

Chapter 5

Setup *ndrxdebug.conf*

Environment variable *NDRX_DEBUG_CONF* contains full path to debug file. See [DEBUGCONF] for more details about *debug.conf* setup.

Chapter 6

Start EnduroX application

To start the EnduroX application, login with EnduroX instance user, and issue following command *xadmin start*, for example:

```
$ xadmin start -y
ndrxd PID (from PID file): 25037
exec tpevsrv -k 0myWI5nu -i 14 -e /tmp/TPEVSRV -r --  :
        process id=26793 ... Started.
exec tpbridge -k 0myWI5nu -i 101 -e /tmp/BRIDGE002 -r -- -n2 -r -i 0.0.0.0 -p 4433 -tP -z30 ↩
    :
        process id=26794 ... Started.
exec tpbridge -k 0myWI5nu -i 102 -e /tmp/BRIDGE012 -r -- -n12 -r -i 10.10.10.2 -p 14433 -tA ↩
    -z30 :
        process id=26795 ... Started.
Startup finished. 3 processes started.
```

To see all commands available by *xadmin* (xa) see manpage of it [XADMIN].

Chapter 7

Recovery from crashed local ATMI monitor

EnduroX is capable to continue with out restart in case if *ndrxd* local ATMI monitor is crashed or contains invalid data structures for some reason. In this case administrator can do following:

- $ kill -9 <ndrxd PID>
- remove ndrxd queue, for example: $ rm /dev/mq/n00b,sys,bg,ndrxd
- restart *ndrxd* in learning mode, by: $ nohup ndrxd -k $NDRX_RNDK -r

Chapter 8

Cluster configuration

To setup cluster see you have to setup bridge ATMI processes on each of the machines. See [TPBRIDGE] documentation to have understanding of clustering. Sample setup of cluster node which activetly connects to Node 2 and waits call from Node 12 could look like:

```
<?xml version="1.0" ?>
<endurox>
        <appconfig>
                <sanity>10</sanity>
                <brrefresh>6</brrefresh>
                <restart_min>1</restart_min>
                <restart_step>1</restart_step>
                <restart_max>5</restart_max>
                <restart_to_check>20</restart_to_check>
        </appconfig>
        <defaults>
                <min>1</min>
                <max>2</max>
                <autokill>1</autokill>
                <respawn>1<respawn>
                <start_max>2</start_max>
                <pingtime>1</pingtime>
                <ping_max>4</ping_max>
                <end_max>3</end_max>
                <killtime>1</killtime>
        </defaults>
        <servers>
                <!-- Connect to cluster node 2, we will wait for call -->
                <server name="tpbridge">
                        <max>1</max>
                        <srvid>101</srvid>
                        <sysopt>-e /tmp/BRIDGE002 -r</sysopt>
                        <appopt>-n2 -r -i 0.0.0.0 -p 4433 -tP -z30</appopt>
                </server>
                <!-- Connect to cluster node 12, we try to connect activetly to it -->
                <server name="tpbridge">
                        <max>1</max>
                        <srvid>102</srvid>
                        <sysopt>-e /tmp/BRIDGE012 -r</sysopt>
                        <appopt>-n12 -r -i 195.122.24.13 -p 14433 -tA -z30</appopt>
                </server>
        </servers>
</endurox>
```

Chapter 9

Additional documentation

9.1 Internet resources

[1] [ATMI-API] http://docs.oracle.com/cd/E13203_01/tuxedo/tux71/html/pgint6.htm

[2] [FML-API] http://docs.oracle.com/cd/E13203_01/tuxedo/tux91/fml/index.htm

[3] [EX_OVERVIEW] ex_overview.pdf

[4] [MQ_OVERVIEW] man 7 mq_overview

[5] [EX_ENV] man 5 ex_env or ex_env.pdf

[6] [NDRXCONFIG] man 5 ndrxconfig.xml or ndrxconfig.xml.pdf

[7] [DEBUGCONF] man 5 ndrxdebug.conf or ndrxdebug.conf.pdf

[8] [XADMIN] man 8 xadmin or xadmin.pdf

[9] [TPBRIDGE] man 8 tpbridge or tpbridge.pdf

Chapter 10

Glossary

This section lists

ATMI
: Application Transaction Monitor Interface

UBF
: Unified Buffer Format it is similar API as Tuxedo's FML