

TwinStar Plus: How it Works

Document version: 1.0

Modification: May 7, 2015

Table of Contents

| | |
|--|----|
| 1. An Example of an Asterisk High Availability Cluster | 3 |
| 2. What Causes the Servers to Switch | 3 |
| 3. Replication of the Configuration Files..... | 4 |
| 4. Servers' Status Monitoring..... | 5 |
| 5. Support for Astribanks | 6 |
| 6. Servers' Network Connections..... | 7 |
| 7. Resource Management | 8 |
| 8. How to Control the TwinStar Status | 9 |
| 8.1 The <code>twinstar-ctl</code> utility..... | 9 |
| 8.1.1 Option ' <code>status</code> ' | 9 |
| 8.1.2 Option ' <code>take-over</code> ' | 9 |
| 8.1.3 Options ' <code>freeze unfreeze</code> ' | 10 |
| 8.1.4 Options ' <code>introspect</code> ' and ' <code>send-msg</code> ' | 10 |
| 8.2 The <code>twinstar</code> utility | 10 |
| 8.2.1 Option ' <code>status</code> ' | 10 |
| 8.2.2 Option ' <code>jump</code> ' | 10 |
| 8.2.3 Option ' <code>enable-wd</code> ' | 10 |
| The ' <code>twinstar enable-wd</code> ' command starts USB voltage monitoring and exchange of watchdog messages between the Astribank and the Astribank DAHDI driver. 8.2.4 Option ' <code>disable-wd</code> ' | 10 |
| 9. TwinStar Plus Configuration..... | 11 |
| 10. TwinStar Plus Configuration Files..... | 13 |
| 11. Log Files..... | 14 |

1. An Example of an Asterisk High Availability Cluster



Two Asterisk servers can comprise an Active/Passive cluster where one of the servers acts as the Master server and the other server acts as the Slave (hot-backup) server. Each server has its own private IP address. This allows server management and configuration replication to be performed between the two servers via the IP network. The VoIP PBX extensions communicate with the PBX cluster via a special public (or cluster) IP address, which is activated only on the current Master server. When the servers swap roles, due to the Master server failure, for example, then that cluster IP address will be activated automatically on the Slave server. As a result, the IP phones will continue to use the same PBX IP address.

2. What Causes the Servers to Switch

The Slave server becomes the Master when it loses the Corosync communication with the Master server. This may happen in the following situations:

- the Master server has crashed
- the network cable that is connected to the Ethernet interface (eth1 in the default configuration) that is used for Corosync communication has become disconnected
- the `twinstar-corosync-bridge` service on the Master server has stopped

Note: In TwinStar Plus configurations the Astribank's connection status has no effect on the server role (as it did in the legacy TwinStar systems).

Once a server assumes the Master role it will keep this role even when the failed server is up and working again.

3. Replication of the Configuration Files

The replication of the servers' configuration files may be performed by using DRBD (“Distributed Replicated Block Device”). Visit the <http://www.drbd.org> Web site for further details. Nowadays, the DRBD is widely used for building many different high availability clusters.

The DRBD concept assumes that there are special hard disk partitions that are replicated on the physical level via the IP network. Thus a changed hard disk block on the Master server ('primary' in DRBD terminology) will be replicated to the Slave server ('secondary' in DRBD terminology.)

All files that need to be replicated must be allocated in the DRBD partition. For example, it is usually necessary to replicate the Asterisk configuration files, MySQL database files, etc. Therefore, folders such as /etc/asterisk and /var/lib/mysql must be allocated in a DRBD partition and soft links need to be created (instead of the regular file names). For example, the /etc/asterisk will be a link that points to /replica/etc/asterisk, where “/replica” is the default mount name of the DRBD partition.

```
[root@srv-a ~]# ls -l /etc
....
-rw-r--r--  1 root    root      148 May 14  2009 asound.conf
lrwxrwxrwx  1 root    root        21 Mar 16 14:30 asterisk -> /replica/etc/asterisk
-rw-r--r--  1 root    root        1 Oct 17 23:55 at.deny
....
lrwxrwxrwx  1 root    root        21 Mar 16 14:30 fail2ban -> /replica/etc/fail2ban

[root@srv-a ~]# ls -l /var/lib
.....
lrwxrwxrwx  1 root    root        25 Mar 16 14:30 asterisk -> /replica/var/lib/asterisk
drwx-----  3 root    root     4096 Oct 15 06:26 authconfig
...
lrwxrwxrwx  1 root    root        22 Mar 16 14:30 mysql -> /replica/var/lib/mysql
...
```

Please note that the DRBD partition (/replica) is mounted on the Master ('primary') server only. The programs that are running on the 'primary' server have full read/write access to the DRBD partition. On the Slave ('secondary') server, the DRBD partition is not mounted and therefore the /etc/asterisk files as well as other replicated files are not available. Services that depend on those files do not run (Asterisk, MySQL server etc) on the Slave server. This ensures that the replicated files will not be changed simultaneously on both servers.

The following folders are replicated in TwinStar Plus:

/etc/dnsmasq.d - DHCP server configuration

| | |
|---------------------|---------------------------------|
| /etc/fail2ban | - fail2ban configuration |
| /etc/asterisk | - Asterisk configuration |
| /var/log/asterisk | - Asterisk logs |
| /var/lib/asterisk | - astdb, sounds, MOH etc |
| /var/lib/dnsmasq | - DHCP server leases |
| /var/lib/mysql | - MySQL data base |
| /var/spool/asterisk | - voice mail, recordings, faxes |
| /var/www | - FreePBX stuff |
| /tftpboot | - IP phone provisioning files |

If you want a folder (e.g., /home/mydir) to be replicated it should be defined as follows:

On the Master server:

```
mv /home/mydir /replica
ln -s /replica/home/mydir /home/mydir
```

On the Slave server:

```
rm -fr /home/mydir
ln -s /replica/home/mydir /home/mydir
```

4. Servers' Status Monitoring

In addition to the data replication function, a tool that detects the Master server failure and forces the Slave server to take over the cluster services is required. TwinStar Plus uses a proprietary Xorcom protocol that works on top of Corosync Cluster Engine for "heartbeat" monitoring of the other server's health, negotiating the server role and exchanging information about the Astribanks connected to each server.

Once Corosync detects that the other server is down then the resource manager of the Slave server will activate it.

When both servers are starting up simultaneously after reboot the server that is configured to be the 'preferred' server will get role of 'Master'.

There is an option to send a notification e-mail message when a server becomes the Master.

The reliability of the Corosync communication is extremely important for the normal cluster functionality. If this communication is interrupted then both servers will become the Master. This is a so-called 'split-brain' situation. Both servers activate the cluster IP, mount the DRBD partition, run Asterisk, etc. The cluster actually stops providing the services. Intervention by the system administrator is usually required in order to solve the problem for example by restoring the

communication, checking/repairing Ethernet cable, etc. When the Corosync communication is restored, one of the servers will be rebooted automatically. This behavior is configured in the DRBD configuration file (`/etc/drbd.d/r0.res`, ref. the `pri-lost-after-sb` parameter) and helps to solve the DRBD 'split-brain' situation.

The Corosync-based communication is performed by the `twinstar-corosync-bridge` daemon. The Atribank management is performed by the `twinstar-manager` daemon. Both daemons are activated on the server startup using the `/etc/init.d/twinstar` script.

5. Support for Atribanks

It is very important that the Atribank devices automatically switch over to the Master server without human intervention. Each Atribank device has two USB interfaces (“Main” and “Backup”):



Once powered up, the Atribank will always use the “Main” USB interface, which must be connected to the preferred Master server. The “Backup” USB interface must be connected to the preferred Slave server.

After all of the services have started it is possible to activate the TwinStar Plus watch dog mechanism that monitors (a) voltage availability on the USB port and (b) continuity of communication between the Atribank device and the Atribank DAHDI driver.

When at least one of the conditions is not met, or a special command is issued ('twinstar jump'), the Atribank will deactivate the “Main” USB interface and automatically activate the “Backup” USB interface.

The DAHDI/Asterisk “plug-n-play” feature allows Atribank activation at any moment.

It is not necessary to wait until all configured Atribanks are active before a server can be activated as the Master and Asterisk starts.

In case of an Atribank failure the system will continue to use the surviving devices.

It is very important that the `XPP_HOTPLUG_DAHDI=yes` and `ASTERISK_SUPPORTS_DAHDI_HOTPLUG=yes` will be defined in the `/etc/dahdi/init.conf` file, as follows:

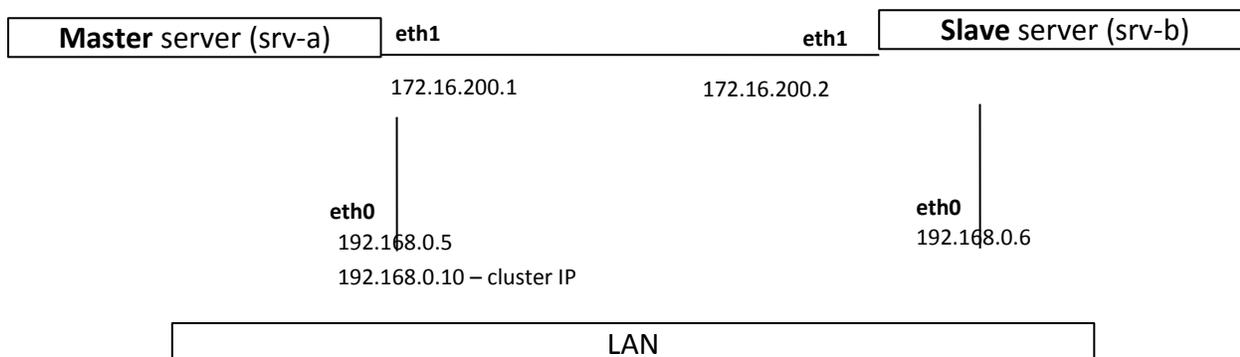
```
#  
# Shell settings for Dahdi initialization scripts.
```

```
# This replaces the old/per-platform files (/etc/sysconfig/zaptel,
# /etc/defaults/zaptel)
#
# The maximal timeout (seconds) to wait for udevd to finish generating
# device nodes after the modules have loaded and before running dahdi_cfg.
#DAHDI_DEV_TIMEOUT=40
#
# A list of modules to unload when stopping.
# All of their dependencies will be unloaded as well.
#DAHDI_UNLOAD_MODULES=""           # Disable module unloading
#DAHDI_UNLOAD_MODULES="dahdi echo" # If you use OSLEC
#
# Override settings for xpp_fxloader
#XPP_FIRMWARE_DIR=/usr/share/dahdi
#XPP_HOTPLUG_DISABLED=yes
XPP_HOTPLUG_DAHDH=yes
ASTERISK_SUPPORTS_DAHDH_HOTPLUG=yes
#
# Disable udev handling:
#DAHDI_UDEV_DISABLE_DEVICES=yes
#DAHDI_UDEV_DISABLE_SPANS=yes
```

6. Servers' Network Connections

The Corosync and DRBD protocols require very reliable communication channels between the two servers. For example, if the Corosync communication is distorted then each server will assume that the other has malfunctioned and will take on the Master role. This is a so-called “split-brain” situation.

The Xorcom TwinStar Plus servers have two Ethernet interfaces. It is strongly recommended to connect the servers each to other directly with Ethernet CAT5e or CAT6 cable. In the standard TwinStar Plus configuration the `eth1` interfaces are used for this, and the `eth0` interfaces are connected to the company LAN. Below is an example of a typical connections schema:



Note that the `eth0` interface of the Master server has two IP addresses: 192.168.0.5 and 192.168.0.10. The second address is the cluster IP address that is activated automatically on the current 'Master' server. In other words, when the `srv-b` server becomes to be the 'Master' server then the 192.168.0.10 address will be activated on `srv-b eth0` interface as the second address. It means that the SIP phones must be configured to communicate with the PBX via the cluster IP address (192.168.0.10). In this case, the SIP phones will be unaware which server is currently has the 'Master' role.

7. Resource Management

There are two important services that run on each server:

- `twinstar-corosync-bridge` – communicates with Corosync and decides about the server switching and resources activation/deactivation
- `twinstar-manager` – monitors Astribanks on both servers and reconnects the Astribanks to the Master server if they were accidentally connected to the Slave server

Resources are the services that are provided by cluster. For example: Asterisk, MySQL, Web server etc. When `twinstar-corosync-bridge` on the Slave server detects that the communication with the Master server is lost then it assumes the role of the Master server and activates the resources in the following order:

1. the cluster IP address activation
2. the DRBD partition mounting
3. MySQL server start
4. Asterisk start
5. the Web server start
6. DNSMASQ server start
7. Any other configured resources

The content and order of the services to be activated is defined by alphabetical order of the service script names as they appear in the `/usr/share/twinstar/run.d` folder.

For example:

```
# ls -l /usr/share/twinstar/run.d/
total 0
lrwxrwxrwx 1 root root 27 Mar 23 08:15 08-ipaddr.run -> ../services.d/08-ipaddr.run
lrwxrwxrwx 1 root root 28 Mar 23 08:15 09-srcaddr.run -> ../services.d/09-srcaddr.run
lrwxrwxrwx 1 root root 25 Mar 23 08:15 10-drbd.run -> ../services.d/10-drbd.run
lrwxrwxrwx 1 root root 31 Mar 23 08:15 15-filesystem.run -> ../services.d/15-filesystem.run
lrwxrwxrwx 1 root root 26 Mar 23 08:15 20-mysql.run -> ../services.d/20-mysql.run
lrwxrwxrwx 1 root root 29 Mar 23 08:15 25-asterisk.run -> ../services.d/25-asterisk.run
lrwxrwxrwx 1 root root 31 Mar 23 08:15 28-astribanks.run -> ../services.d/28-astribanks.run
lrwxrwxrwx 1 root root 27 Mar 23 08:15 30-apache.run -> ../services.d/30-apache.run
lrwxrwxrwx 1 root root 25 Mar 23 08:15 40-cpbx.run -> ../services.d/40-cpbx.run
lrwxrwxrwx 1 root root 28 Mar 23 08:15 40-dnsmasq.run -> ../services.d/40-dnsmasq.run
lrwxrwxrwx 1 root root 34 Mar 23 08:15 40-f2ban-starter.run -> ../services.d/40-f2ban-starter.run
lrwxrwxrwx 1 root root 25 Mar 23 08:15 40-fop2.run -> ../services.d/40-fop2.run
```

Thus the `08-ipaddr.run` script with parameter `'start'` will be run first. It will be followed by the `09-srcaddr.run` script, etc.

During the server deactivation the scripts are called in the opposite order using the `'stop'` parameter.

Note that the scripts in the `run.d` folder are actually symbolic links to the real files that are located in the `services.d` folder. The `twinstar_config` utility creates these symbolic links during the servers configuration.

8. How to Control the TwinStar Plus Status

There are two utilities that are used for TwinStar Plus cluster management:

- `twinstar-ctl` - used for the cluster management
- `twinstar` - used for Astribank management

8.1 The `twinstar-ctl` utility

The `twinstar-ctl` has several command line options.

8.1.1 Option `'status'`

Command `twinstar-ctl status` shows the current server status. For example:

```
[root@srv-a ~]# twinstar-ctl status
AstribanksCount:2      <- Number of Astribanks connected to the server
FrozenMaster:  0       <- Whether the server is 'frozen'. Ref. Option 'freeze'.
InTransition:  false   <- Whether the resources activation/deactivation process is
                        performing.
IsMaster:      true    <- The server role
LocalNodeId:   1       <- The local Corosync node id
MasterNodeId:  1       <- The Master node id
NodeCount:    2        <- How many nodes present in Corosync cluster
PreferredMaster:true   <- Whether this server is the 'preferred' server.
Version:      1.0
```

8.1.2 Option `'take-over'`

When the `twinstar-ctl take-over` command is run on the Slave server then it will assume the role of Master. The exception is when the current Master server is 'frozen' (ref. option 'freeze' below).

8.1.3 Options 'freeze|unfreeze'

The `twinstar-ctl freeze` command causes the current Master server to “freeze”. It means that the Slave server will never be Master even when the `twinstar-ctl take-over` command is issued on it, or communication with the Master server is broken. The `unfreeze` option cancels “freezing” of the Master.

These options are useful during maintenance procedures when connection between servers can be disrupted but role switching of the servers is not desired.

8.1.4 Options 'introspect' and 'send-msg'

These options can be used for development of additional TwinStar Plus features. The 'introspect' option shows the available d-Bus interfaces (ref. <http://en.wikipedia.org/wiki/D-Bus>) provided by the `twinstar-corosync-bridge` service.

The 'send-msg' option allows sending an arbitrary text message from one server to another.

8.2 The `twinstar` utility

This utility is used for the Astribank management and has several command line options. The utility can be run on the server where the Astribank(s) are currently connected.

8.2.1 Option 'status'

Command '`twinstar status`' shows that current Astribanks status. For example:

```
#twinstar status
DEVICE          PORT      WATCHDOG    POWER0     POWER1
usb:001/006     0         on           yes        yes
usb:001/007     0         on           yes        yes
```

where,

- PORT – shows the currently active port: 0- for Main; 1- for Backup
- WATCHDOG – shows whether the Astribank watchdog process is enabled
- POWER0, POWER1 – indicates whether Astribank senses voltage on the Main and Backup USB connectors correspondingly.

8.2.2 Option 'jump'

The '`twinstar jump`' command forces the Astribank to switch to another USB interface.

8.2.3 Option 'enable-wd'

The '`twinstar enable-wd`' command starts USB voltage monitoring and exchange of watchdog messages between the Astribank and the Astribank DAHDI driver.

8.2.4 Option 'disable-wd'

The 'twinstar disable-wd' command stops USB voltage monitoring and exchange of watchdog messages between the Astribank and the Astribank DAHDI driver.

Before stopping the DAHDI for maintenance purposes the watchdog should be disabled. Otherwise, the Astribanks will be switched over unexpectedly to the second server.

9. TwinStar Plus Configuration

Note! The detailed step-by-step TwinStar Plus configuration instructions can be found in the "Installation Instructions for Xorcom TwinStar Plus Servers" document on the [Xorcom Web site](#).

Following is an overview of the configuration process:

In order to simplify the configuration procedure a special utility called 'twinstar_config' is provided. The utility should be run only on the preferred Master server (srv-a in the default configuration). The utility is capable of configuring the following components on both servers:

- Corosync
- DRBD
- cluster IP address
- set of optional resources to be automatically started/stopped (e.g., dnsmasq, fop2, etc)
- e-mail notification about server activation

Input parameters for twinstar_config are concentrated in the /usr/share/twinstar/twinstar_config.conf file.

Below is an excerpt from the twinstar_config.conf file. Each parameter is accompanied by an explanation:

```
# IP address that is used for cluster access. SIP phones etc will access the PBX by
# using this address. The address is automatically activated on the server that
# takes control of the cluster (the Master server). In order to ensure correct
# IP routing table building, in addition to the IP address the system must know
# the subnet mask. Therefore, the address must be defined in
# the CIDR (Classless Inter-domain Routing) notation.
CLUSTER_IP=192.168.4.161/20

# Servers' IP addresses. These addresses will be used for DRBD and during the
# servers' configuration process by the twinstar_config utility.
SRV_A_IP=172.16.200.1
SRV_B_IP=172.16.200.2
```

```
# Corosync parameters. By default, the COROSYNC_BINDNETADDR is configured for the
# same Ethernet interface that is used for DRBD.
COROSYNC_RINGNUMBER='0'
COROSYNC_BINDNETADDR='172.16.200.0'
COROSYNC_MCASTADDR='239.192.244.6'
COROSYNC_MCASTPORT='5405'

# Whether or not the dhcpd service must be activated on the active server. The
# default value is "No".
#DHCPD_ENABLE=Yes

# Whether or not the dnsmasq service must be activated on the active server. The
# default value is "No".
DNSMASQ_ENABLE=Yes

# Whether or not the fail2ban must be activated on the active server. The default
# value is "No".
FAIL2BAN_ENABLE=Yes
.....
# E-mail notification about server activation
#EMAIL_NOTIFY=yes
#EMAIL_ADDR=admin@example.com
#EMAIL_ADDR_FROM=pbx@example.com
```

In order to send an automatic notification e-mail message when a server becomes the Master, uncomment the `EMAIL_NOTIFY`, `EMAIL_ADDR` and `EMAIL_ADDR_FROM` parameters and define the desired e-mail addresses. Also, the CompletePBX e-mail settings must be configured via the CompletePBX Web interface (Admin|System Settings|Email Settings).

The `twinstar_config` utility has the following command line parameters:

```
[root@srv-a twinstar]# /usr/share/twinstar/twinstar_config
Usage: /usr/share/twinstar/twinstar_config <action> [opts]

Actions:

    keys                                -- Generate SSH and Corosync keys.
    reconfigure [-i] [-a]               -- Change IP and/or Asterisk/DAHDI
                                        configuration.
    system-prep                          -- Prepare a new system after the Rapid Recovery
                                        images were installed on both servers.
```

In most cases, upon receiving a new TwinStar Plus system it is necessary to do the following:

- configure the servers' network settings
- configure the parameters in the `/usr/share/twinstar_config.conf` file on the preferred master server (srv-a)
- generate the SSH and Corosync keys:


```
cd /usr/share/twinstar
./twinstar_config keys
```
- generate the cluster configuration files related to the IP addresses


```
./twinstar_config reconfigure -i
```
- reboot both servers

10. TwinStar Plus Configuration Files

| Configuration File | Purpose |
|--|---|
| <code>/usr/share/twinstar_config.conf</code> | Configuration parameters for the <code>twinstar_config</code> utility |
| <code>/etc/drbd.d/r0.res</code> | DRBD configuration file. Built by <code>twinstar_config</code> utility |
| <code>/etc/corosync/corosync.conf</code> | Corosync configuration file (built by <code>twinstar_config</code> utility) |
| <code>/etc/twinstar/preferred_master</code> | If the file exists and contains 1 then the server is the preferred Master server. The file usually does not exist on the srv-b server. |
| <code>/etc/twinstar/runconf.d/*</code> | Configuration files for the resource activation/deactivation scripts. For example, <code>ifcfg-eth0.conf</code> and <code>srcaddr.conf</code> contain the cluster IP address. The <code>notify.conf</code> contains e-mail addresses for notification purposes. Some of the files are modified by the <code>twinstar_config</code> utility. |
| <code>/etc/twinstar/frozen_master</code> | The file is created by <code>twinstar-ctl</code> utility when the <code>twinstar-ctl freeze</code> command is executed. The file contains the node ID (1 or 2) of the 'frozen' Master. The file is deleted on both servers when the <code>twinstar-ctl unfreeze</code> command is executed. |

11. Log Files

The TwinStar Plus related log files are located in the `/var/log/cluster` folder.

`corosync.log` – contains the Corosync service messages:

```
Mar 19 09:31:08 corosync [MAIN ] Corosync Cluster Engine ('1.4.7'): started and ready to
provide service.
Mar 19 09:31:08 corosync [MAIN ] Corosync built-in features: nss dbus rdma snmp
Mar 19 09:31:08 corosync [MAIN ] Successfully read main configuration file
'/etc/corosync/corosync.conf'.
Mar 19 09:31:08 corosync [TOTEM ] waiting_trans_ack changed to 1
Mar 19 09:31:08 corosync [TOTEM ] Token Timeout (1000 ms) retransmit timeout (238 ms)
Mar 19 09:31:08 corosync [TOTEM ] token hold (180 ms) retransmits before loss (4 retrans)
Mar 19 09:31:08 corosync [TOTEM ] join (50 ms) send_join (0 ms) consensus (1200 ms) merge
(200 ms)
Mar 19 09:31:08 corosync [TOTEM ] downcheck (1000 ms) fail to recv const (2500 msg)
Mar 19 09:31:08 corosync [TOTEM ] seqno unchanged const (30 rotations) Maximum network MTU
1402
.....
```

`twinstar.log` – contains messages generated by the `twinstar-corosync-bridge` and by the `twinstar-manager` services:

```
Mar 19 09:31:12 srv-a twinstar-corosync-bridge[1678]: INFO: Starting
Mar 19 09:31:12 srv-a twinstar-corosync-bridge[1678-stderr]: redirected stderr
Mar 19 09:31:12 srv-a twinstar-corosync-bridge[1678]: INFO: Registered on dbus
Mar 19 09:31:12 srv-a twinstar-corosync-bridge[1678]: INFO: MASTER: (nodeid = 1 priority = 1
sync_counter = 20) -- local sync_counter = 20
Mar 19 09:31:13 srv-a twinstar-corosync-bridge[1678]: INFO: ----- started
/usr/share/twinstar/run.d/08-ipaddr.run (is_master=1) (pid = 1709)
Mar 19 09:31:14 srv-a twinstar-corosync-bridge[1678-stderr]: 2015/03/19_09:31:14 INFO:
Adding inet address 192.168.0.253/20 to device eth0
Mar 19 09:31:14 srv-a twinstar-corosync-bridge[1678-stderr]: 2015/03/19_09:31:14 INFO:
Bringing device eth0 up
Mar 19 09:31:14 srv-a twinstar-manager[1686]: INFO: nodeid: 1
Mar 19 09:31:14 srv-a twinstar-manager[1686]: INFO: master: 1 [me]
Mar 19 09:31:14 srv-a twinstar-manager[1686]: INFO: Start.
Mar 19 09:31:14 srv-a twinstar-corosync-bridge[1678-stderr]: 2015/03/19_09:31:14 INFO:
/usr/libexec/heartbeat/send_arp -i 200 -r 5 -p /var/run/resource-agents/send_arp-
192.168.0.253 eth0 192.168.0.253 auto not_u
sed not_used
Mar 19 09:31:14 srv-a twinstar-corosync-bridge[1678]: INFO: ----- ended
/usr/share/twinstar/run.d/08-ipaddr.run (is_master=1) (pid = 1709, status = 0)
Mar 19 09:31:14 srv-a twinstar-manager[1686]: INFO: run_step_end:
/usr/share/twinstar/run.d/08-ipaddr.run start [0]
```



```
Mar 19 09:31:14 srv-a twinstar-corosync-bridge[1678]: INFO: ----- started
/usr/share/twinstar/run.d/09-srcaddr.run (is_master=1) (pid = 1808)
Mar 19 09:31:14 srv-a twinstar-manager[1686]: INFO: run_step_begin:
/usr/share/twinstar/run.d/09-srcaddr.run start
.....
```

Asterisk is a registered trademark of Digium, Inc. TwinStar, TwinStar Plus, and Astribank are trademarks of Xorcom, Ltd. All other marks are the property of their respective holders.