ProNu COMMUNICATIONS
INNOVATIVE VOIP DISTRIBUTION

SARK UCS/MVP TDM/IP PBX
Quick-start Installation Guide
High Availability Clusters

SARK UCS/MVP Version 2 Release 2
August 2009

Table of Contents

Introduction

SARK UCS/MVP is a hybrid TDM/IP PBX softswitch for the SMB/SME market segment. It is based upon Digium's *Asterisk* soft-switch running on SME Server Linux, a hardened variant of CentOS. SARK UCS/MVP has established a reputation for being extremely reliable, highly functional and particularly well equipped for remote support, making it very attractive to resellers. Moreover, it is well able to compete with traditional proprietary PBX platforms while offering a much lower overall cost-of-ownership to the user. SARK-HA (High Availability) is a cluster solution which delivers fully automatic failover.

Platform Options

SARK UCS/MVP is available in several different hardware models as follows: –

MODEL	DESCRIPTION	PCI Slots	MEMORY	Storage
650	Entry Level desk top	1	512Mb	4Gb Flash
850	1U rack mount	1	1Gb	80Gb SATA
1000	1U rack mount	1	1Gb	80Gb SATA
1200	2U rack mount	3	2Gb	160Gb SATA
1500	2U rack mount	3	2Gb	160Gb SATA

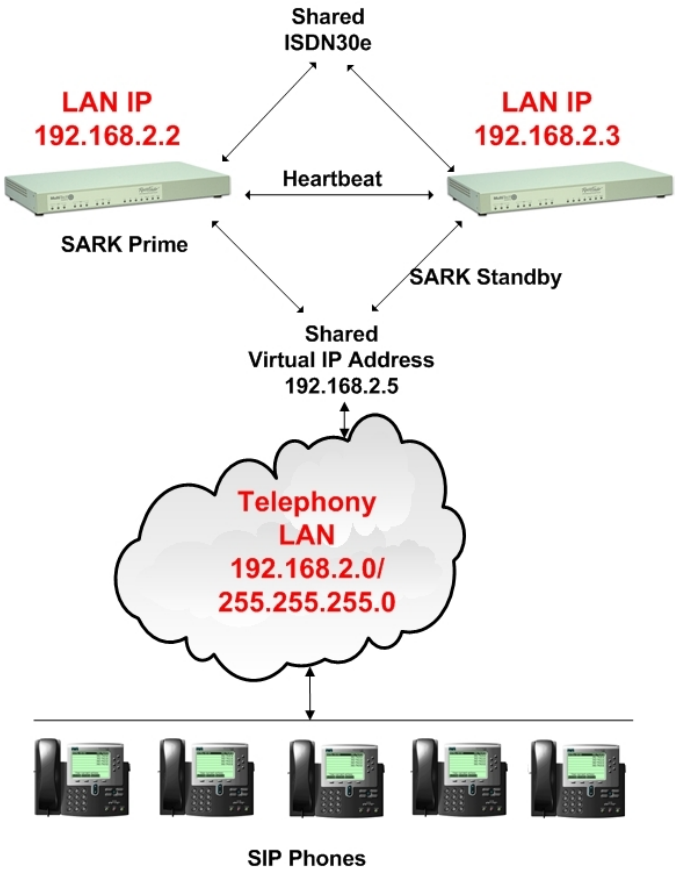
All units are capable of running their own inboard telephony gear. The 650 is restricted to 4xFXO ports or 2 ISDN2e ports. All other models can be fitted with either analog or digital cards, giving the larger systems the ability to manage up to 120 TDM voice channels (lines) per module.

SARK High Availability system defaults

Each SARK-HA node must run in server-only mode.

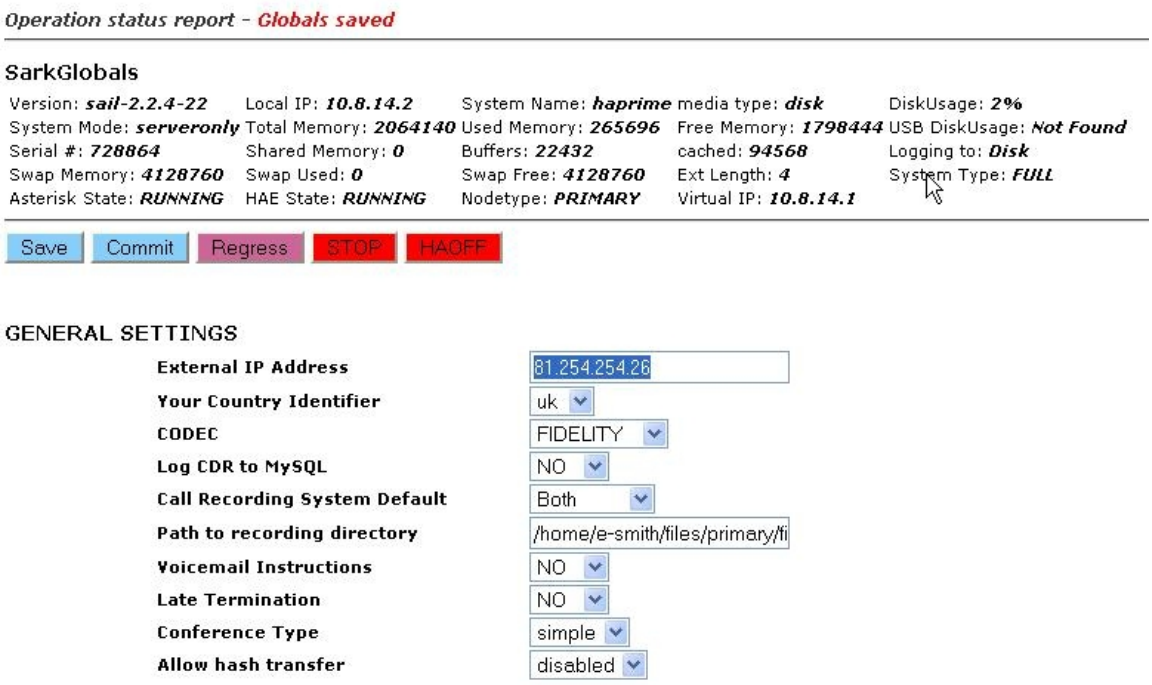
If you are running hardware with twin Ethernet circuitry then you should connect the sockets marked “LAN” to your network switches

SARK-HA requires three unassigned IP addresses; one for each of the two Cluster Servers and a third “virtual” address, which is shared. Unless you specify otherwise, your High Availability system will be delivered preconfigured to run in Server-Only mode in a regular class C network. The prime server will be configured to run at 192.168.2.3/255.255.255.0. The standby server will be configured to run at 192.168.2.4 and the virtual IP will be configured at 192.168.2.5. Your SIP and IAX endpoints should be configured to register at the virtual IP (default 192.168.2.5).



Running remote SIP phones with HA

If you intend to run remote telephones (i.e. phones which are NOT on the same subnet as the SARK cluster) then you must ensure that the *External IP* is filled out correctly in Globals panel (see the image below). The value will usually be the publicly visible IP address of your network or it may sometimes be the address at which the router is running in the upstream network.



Additional Cabling for HA

The HA "heartbeat" mechanism is preconfigured to run over either, or both, the LAN and across a serial cable link. It is STRONGLY recommended that you implement the serial pathway. If you do not, and the LAN should ever fail then it may result in a condition known as "split brain" where both nodes are still up, but each believes the other to have failed and so each starts its own Asterisk instance. As pre-configured, each node will wait for a period of 20 seconds after a loss of heartbeat before it takes any action. This should be more than sufficient to cater for short communication losses but it will not survive a "hard-down" LAN event. The simple fitment of a serial cable will prevent such outages from adversely affecting the cluster's correct operation.

HA Default assumptions

1. Wait 20 seconds after a heartbeat failure before forcing a failover
2. At system startup, wait 120 seconds for the other cluster to come on-line, after which assume that it is not coming up and proceed independently.
3. After a failover event, await manual intervention before failing back, even after the other node has come back on-line and reestablished communication.

Changing the defaults

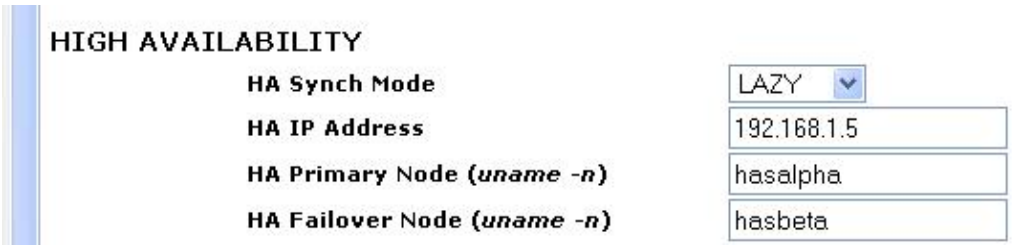
If you wish to change the factory default settings, your SARK sales & support team can arrange to have this done for you.

DHCP considerations

Your SARK-HA cluster can provide automatic provisioning services for most popular phone types. In order to facilitate this you will need to provide DHCP option 66 information from your DHCP server (and option 150 if you are planning to run Cisco 79xx SIP phones). Option 66 should point to the FQDN of the SARK-HA *Virtual* IP address. It is a good idea to set up a resolvable name for this, perhaps using one of the free dynamic dns services.

Defining HA settings to the server-manager

There are only four settings to define to the server-manager (these will have been preset for you if you have pre-ordered an HA system from the factory)…



HA Synch Mode:

Description - Dropdown.
Permissable values - {LAZY|LOOSE}. LAZY means periodic synchronization of the two system images in the cluster. LOOSE means no synchronization.
Default - LAZY.

HA IP Address:

Description - This is the *Virtual Address* that the cluster will run at. You should allocate a free static address on your subnet for this .
Permissable values - standard dotted quadrant
Default - None.

HA Primary Node Name:

Description - Nodename of this server as given by *uname -n*.
Permissable values - The *uname -n* node name.
Default - None.

HA Secondary Node Name:

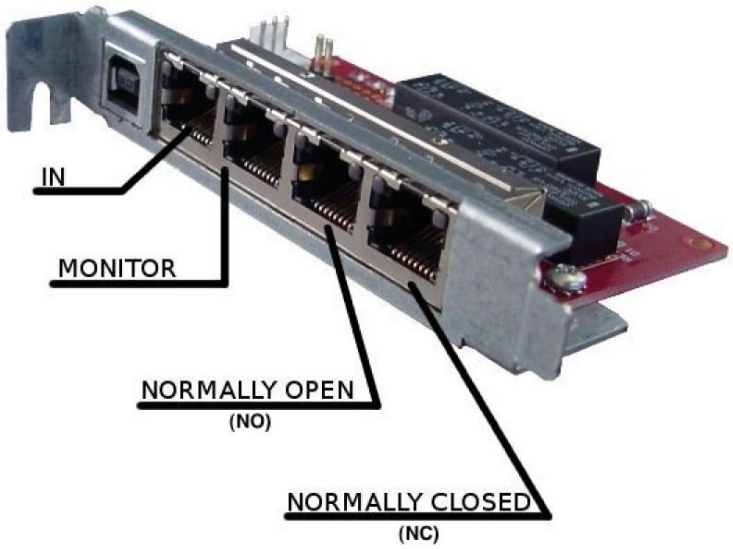
Description - Nodename of the failover server as given by *uname -n*.
Permissable values - The *uname -n* node name of the *failover* server.
Default - None.

IMPORTANT NOTE

These four fields must be *IDENTICAL* on *BOTH* HA nodes in the cluster.

Implementing shared T1/E1 PRI with the Rhino Failover Card

SARK-HA has full on-board support for the Rhino single port failover card. The card allows ISDN PRI circuits to be shared between HA cluster nodes. Access to the ISDN circuit is passed back and forth between nodes during failover and failback. In keeping with the rest of SARK, operation of the card is entirely automatic. All the user need do is to inform SARK that the card is present in the configuration and cable out the card correctly.



Card Installation

The software drivers for the card are shipped with the SARK-HA rpm. However, we need to make a small modification to /etc/ld.so.conf to reflect the load directory...

Make the following changes on BOTH machines (for pre-ordered HA systems this will normally have been done at the factory during system assembly, however if you are upgrading an existing system then you may have to perform this step)...

[root@hasalpha ~]# cat /etc/ld.so.conf
include ld.so.conf.d/*.conf
/usr/local/bglibs/lib
/usr/local/lib

Add the last line (/usr/local/lib) to etc/ld.so.conf and run ldconfig.

You must also inform SARK of the card's presence by running the following commands at the Linux console

db selintra setprop global RHINOSPF YES
db selintra-work setprop global RHINOSPF YES

DON'T FORGET THAT THESE CHANGES NEED TO BE MADE ON BOTH MACHINES IN THE CLUSTER!

Reboot both of your systems and the card is now ready to use.

Card Cable-out

Cable-out the card as follows... With the card's USB port on the left and proceeding left to right...

1. Connect Socket 1 (marked "IN" on the diagram) to the PTT NTE.
2. Connect Socket 3 (marked "NO" on the diagram) to the T1/E1 card in the PRIMARY node.
3. Connect Socket 4 (marked "NC" on the diagram) to the T1/E1 card in the STANDBY node.

Operating Sequence - catastrophic failure

With power OFF, the card will bridge IN and NC. With power ON, the card will bridge IN and NO. In this way, the system will feed ISDN30 signal to the PRIMARY node when power is on and to the secondary node when power is off. Thus if the PRIMARY node fails (loses power) then the ISDN30 signal will be transferred automatically to the STANDBY node. For this reason it is vitally important that the card is powered from the PRIMARY node.

Operating Sequence - Asterisk failure (SMITH event)

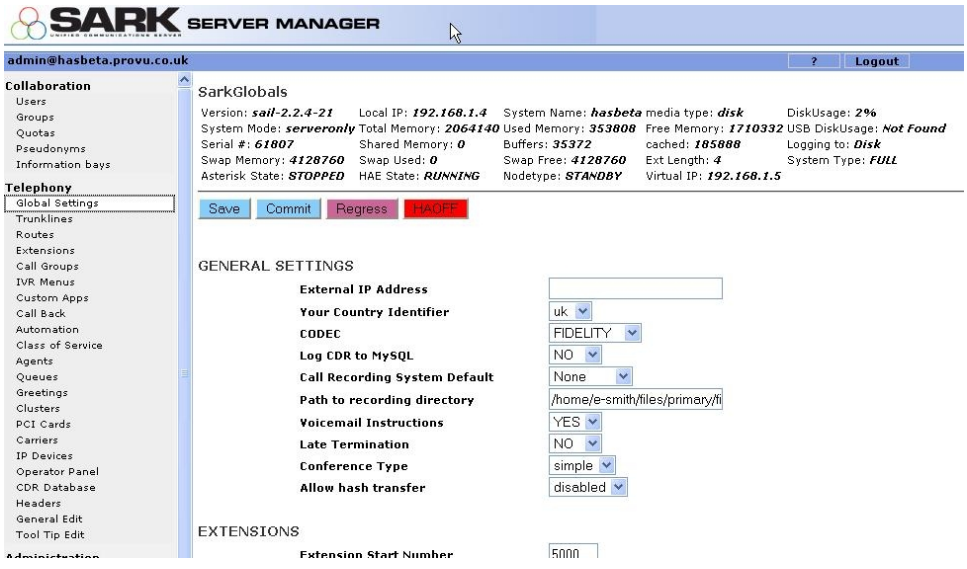
A watchdog daemon runs on both the PRIMARY and STANDBY nodes. Should Asterisk fail (upon whichever node it is currently running), then the daemon will "force" a failover event. It will also send the necessary commands to the Rhino card to failover the ISDN30 signal. If you are familiar with failover theory then this is essentially an inversion of a so-called “STONITH” event (Shoot The Other Node In The Head); effectively a SMITH event (Shoot Myself In The Head),

System Factory Startup

After initial power up your systems will come on-line and automatically take up their respective rolls. You can see the condition of each node by bringing up the *"Gobals"* panel on each of the server-manager browsers.

Here is how your Primary node should look…

The High Availability engine is running and Asterisk has been started.
Similarly, here is how your Standby node should look…



The High Availability engine is running and Asterisk is currently down (which is as it should be).

System Shutdown sequences

1. Full shutdown

To fully shutdown the entire system execute the following sequence (assumes normal running state – i.e. Asterisk up on the primary node).

1. Stop the High Availability Engine on the standby node
2. Power down the standby node.
3. Stop Asterisk on the primary node (press Red STOP in Globals)
4. Stop the High Availability Engine on the primary node.
5. Power down the primary node

2. Shutdown Primary Node

1. Force failover by pressing Red STOP on the primary node (in Globals).
2. Stop the High Availability Engine on the primary node.
3. Power down the primary node

3. Shutdown Standby Node

1. Stop the High Availability Engine on the standby node
2. Power down the standby node.

System Startup sequences

1. Full startup from cold with High Availability turned OFF on both Nodes

1. On the Standby server, in Globals, press Blue HA-ON.
2. On the Primary server, in Globals, press Blue HA-ON.

Initially you will see that both panels go to Red HA-OFF indicating that the HA Engine is running. Press refresh on the primary Globals panel and after a few moments you should see the Red STOP button displayed. This tells you that Asterisk is up and running on the Primary.

2. Force failover or fail-back of Asterisk

1. On the active node, in “Globals” press Red STOP.

3. Startup Failures

At cluster power-up, the two nodes will wait for each other to come on-line. There is an absolute timeout of 120 seconds (from the heartbeat start-up) after which the powered up cluster will assume the other node is not coming up and it will independently bring its copy of Asterisk on-line and seize the virtual address.

Asterisk Restart

An Asterisk restart may occasionally be needed after system maintenance.

1. Stop the High Availability Engine on the standby node
2. Stop the High Availability Engine on the primary node.
3. Asterisk should normally stop after the HA engine is stopped, however if it does not then force it down by pressing the red STOP in Globals.
4. Start the High Availability Engine on the primary node.(this will also start Asterisk after a minute or so)
5. Start the High Availability Engine on the standby node