SARK500 Embedded VoIP/TDM PBX

Administrators Guide V3.1

SARK Version 3 Release 1-135
Oct 2011

Contents

CONTENTS.....2

COPYRIGHT INFORMATION.....4

INTRODUCTION.....5

FEATURES OVERVIEW.....6

GETTING STARTED 7

SYSTEM HELP.....8

BROWSER CHOICE.....8

POWER-UP – DETERMINING THE IP ADDRESS OF THE UNIT.....9

COMMIT AND SAVE.....11

LOGGING ON TO SARK AND SETTING THE PASSWORDS.....11

ADDING TRUNKS.....14

ADDING EXTENSIONS WITH ZTP/PNP.....22

ADDING NON-PNP EXTENSIONS AND TFTP PROVISIONING.....24

ADDING A LINE-OUT EXTENSION.....26

CONFERENCING.....27

INTERACTIVE VOICE RESPONSE.....29

ADDING ROUTES.....32

CLASS OF SERVICE (COS).....33

FIREWALL.....35

CHANGING FIREWALL SETTINGS 36

FIREWALL "LOCAL" MODE 38

RESOLVING FIREWALL LOCK-OUTS.....38

NETWORK CONFIGURATION.....39

SPECIFYING ALTERNATIVE NTP SERVERS.....41

CHANGING SSH OR HTTP PORT ASSIGNMENTS.....42

SPECIFYING AN SMTP SERVER.....43

BACKUP/REGRESS & SAVE TO USB.....44

GLOBAL SETTINGS.....45

CALL RECORDING.....46

ASTERISK FILE EDIT.....47

SARK SERVICE CODES.....49

NANP SUPPORTED SERVICE CODES.....55

SARK500 HARDWARE PLATFORM.....57

PIKA TELEPHONY HARDWARE CHANNEL DRIVER.....59

CHANNEL STATUS ON THE APPLIANCE LCD.....80

DEPLOYMENT AND DIMENSIONS.....85

SARK DIRECTORY STRUCTURE ON THE WARP.....86

SARK ADDITIONAL PERL DEPENDENCIES 89

SARK ADDITIONAL BUSYBOX COMMANDS.....89

Copyright Information

TRADEMARKS

SARK UCS/MVP and SARK500 are registered trademarks of Aelintra Telecom Limited.

PIKA is a registered trademark of PIKA Technologies Inc.

All other trademarks, product names and company names and/or logos cited herein, if any, are the property of their respective holders.

DISCLAIMER

This document is provided to you for informational purposes only and is believed to be accurate as of the date of its publication, and is subject to change without notice. Aelintra Telecom Limited. assumes no responsibility for any errors or omissions in this document and shall have no obligation to you as a result of having made this document available to you or based upon the information it contains.

Introduction

The SARK500 provides a solid-state SIP/TDM PBX solution for system integrators and PBX resellers. The appliance is a purpose-designed telephony platform with its own integrated analog, digital and GSM telephony cards. The system can support up to 8 FXO (CO) lines or 8 Digital (BRI) Channels or 4 GSM channels, depending upon the configuration. It also supports a further 8 SIP or IAX2 trunks for VoIP telephony. The system can be configured with up to 20 IP handsets from any of the leading SIP manufacturers (e.g. Snom, Aastra, Cisco, Yealink etc).

One rather unique feature of the SARK500 is its ability to automatically recognize and provision PnP SIP phones (principally from Snom and Yealink) with no human intervention and no special network setup. This means that, unlike competitive VoIP offerings, which can take several hours or even days to set up, the SARK500 can be deployed into an unknown network (provided only that the network has a functioning DHCP server) with brand new factory set phones and require NO initial set-up. The system will come up; provision the phones as they come on-line and immediately be able to receive and make calls over whatever media it is given (Analogue, BRI or GSM). This makes the system as easy and quick to install as a traditional TDM PBX, an important consideration when installing into small businesses where cost is often an overriding issue.

In addition, the on-board SARK UCS/MVP software provides a feature-rich graphical workbench and decision engine designed to generate and run efficient PBX images. All adds moves and changes can be done remotely as soon as the system is installed.

Features Overview

SARK500 provides an easy to use graphical workbench and decision engine that controls and manages the PBX. The Warp appliance is a purpose-designed solid state telephony platform with its own integrated analog, digital and GSM telephony boards. The SARK Implementation on the WARP appliance uses the SARK source code directly from the SARK svn repository. SARK is also available as, a CentOS 5 .iso, a tarball or a Redhat/CentOS rpm.

SARK500 features

- Zero touch setup for PnP phones
- Fast, automated installation
- Add or change extension and voicemail accounts in seconds
- Integrated, extensible automatic provisioning service
- Supports all Asterisk-supported trunk technologies
- Reduce long distance costs with LCR and powerful pattern-based outbound routing
- Route incoming calls based on time-of-day, DID, Caller ID or caller ID pattern
- Supports BLF, call parks and directed call pickup
- Create infinite level interactive Digital Receptionist (IVR) menus
- Design sophisticated recursive call groups
- True hot-desk support (option)
- Advanced on-board voice recording feature (option)
- Manage callers using ACD and Queues
- Detect and direct incoming faxes
- Automatic Backup and one-click Restore
- Powerful on-board firewall
- easy to use network management

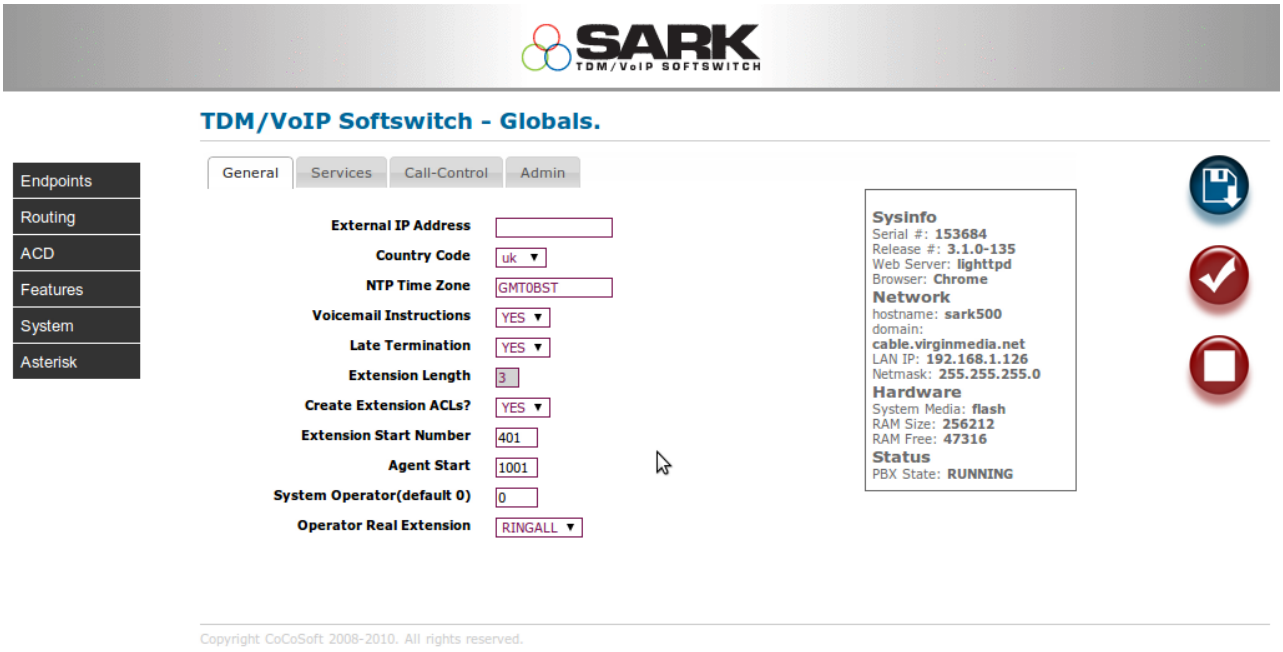
Getting Started

The SARK500 is initially set-up to allow you to install it and be up and running within just a few minutes of powering it up. It usually requires no network configuration (by default, it initially uses DHCP to obtain an IP address). Also, provided you deploy SIP PnP capable phones (such as Snom or Yealink), they will be automatically configured and registered on-the-fly when they are powered-up.

Once connected to the CO (FXO) or ISDN (ptp) lines, the SARK500 is ready to receive and make calls. As phones are powered up and configured, they are automatically added to a special ring-group called the 'RINGALL' group. Any calls arriving over the FXO, ISDN or GSM lines will be automatically sent to the RINGALL group, which will ring all of the phones, and 'first to pick-up' will get the call. The RINGALL group can be replaced when you move to more sophisticated routing and ACD configurations but initially it just allows you to get up and running very quickly. Also by default, outbound calls will initially be routed via the FXO or ISDN hunt group (depending upon whether the appliance is fitted with FXO or ISDN ports). (**N.B.** If you have no telephony cards fitted to your system then the DEFAULT route will be initially empty).

Of course, you will probably want to progress to more sophisticated call distribution and routing and you may wish to deploy VoIP trunks and perhaps remote phones. All of these things can be accomplished using the SARK500 browser-based interface. In the following sections we will cover some of these topics in more detail.

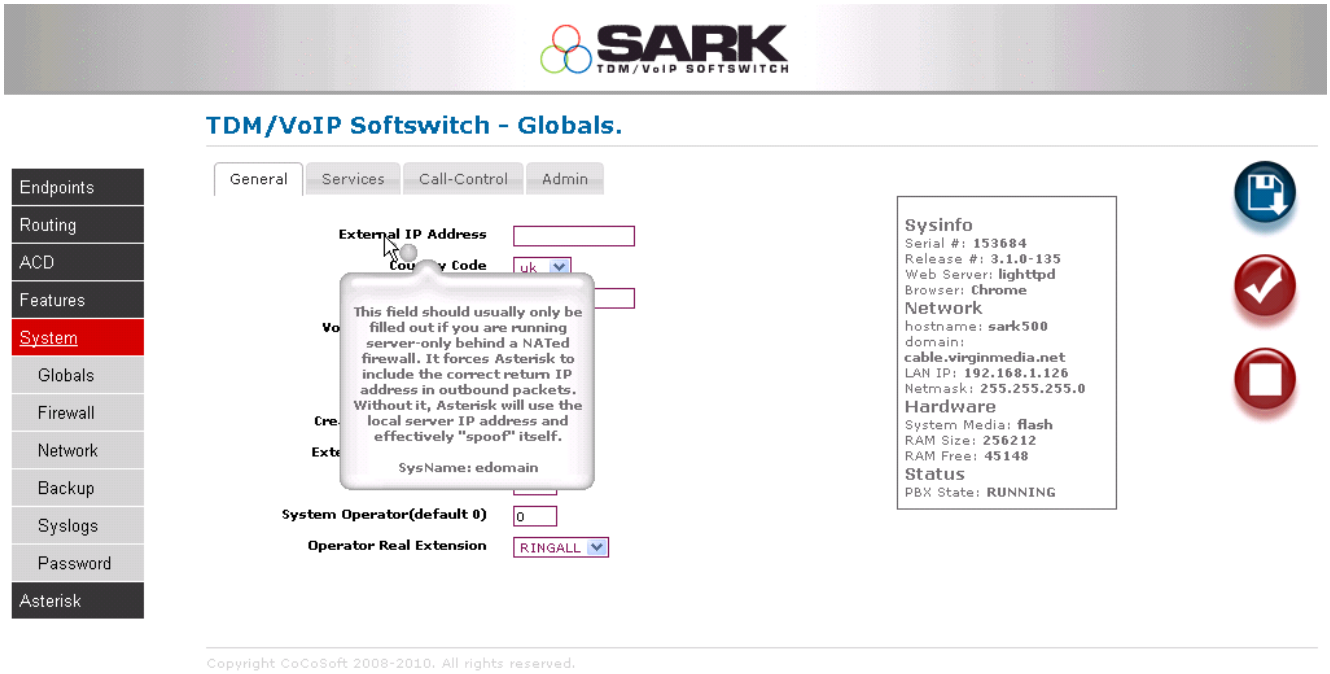
The SARK browser pages follow a standard layout. The figure below shows a typical SARK Web page.



Navigation is handled by the column to the left of the screen. Modifiable data appears in the data window in the centre and information and action buttons appear to the right. All data fields have instant help (see below). To save your changes, press the save (disk icon) button. To commit your changes (i.e. bring them into

service), click the checkmark (tick) button; this will cause SARK to regenerate the underlying Asterisk configuration files and issue a soft reload.

System Help



Every name field in the system has context sensitive help associated with it. All you need do is hover the cursor over any name field for an explanation of the field's function.

Browser choice

The SARK500 will work with most popular browsers but there are a few things to be aware of when it comes to choosing your browser software. Because the SARK500 makes extensive use of Jquery and, Jquery-ui, it performs best with modern browsers such as Chrome, Firefox and Safari. It runs noticeably more slowly with IE7 and IE8 and it will not run at all with Opera. There is also a known problem with IE8 where the sysinfo panel in the Globals section does not render.

Finally, the CPU on the Sark500 is relatively low powered so you should expect a normal response time of around 2 to 3 seconds to fully render a panel.

Power-up – determining the IP address of the unit

Once the appliance has powered up you can determine its IP address from the dual-purpose LCD display on the front panel. Touch the light-sensitive toggle-button once to show the IP address on the display. The toggle-button is located on the right hand side of the front-panel, immediately to the left of the Power LCD (see image).

System Buttons

The system buttons are clustered to the right of each data panel. The available buttons are as follows;

NEW button. Used to create a new object instance

SAVE button. Used to save a change

ACTION button. Used to perform some action (this will differ from screen to screen).

DELETE button. Used to delete an object instance.

COMMIT button. Used to “enliven” your changes by updating the PBX running instance

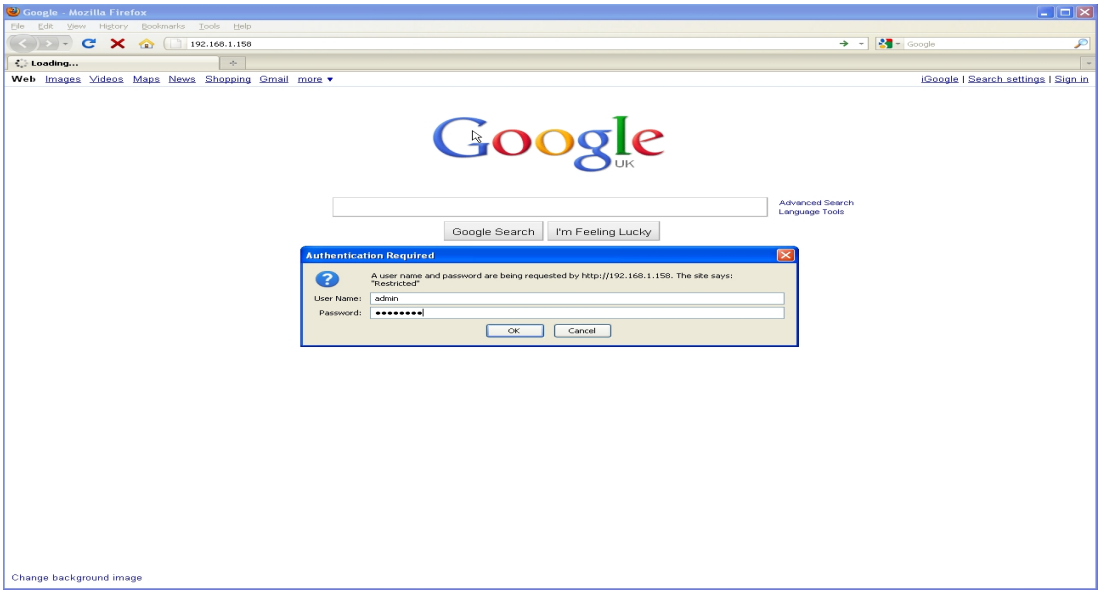
CANCEL button. Used to cancel the current operation and return to the parent screen.

Commit and Save

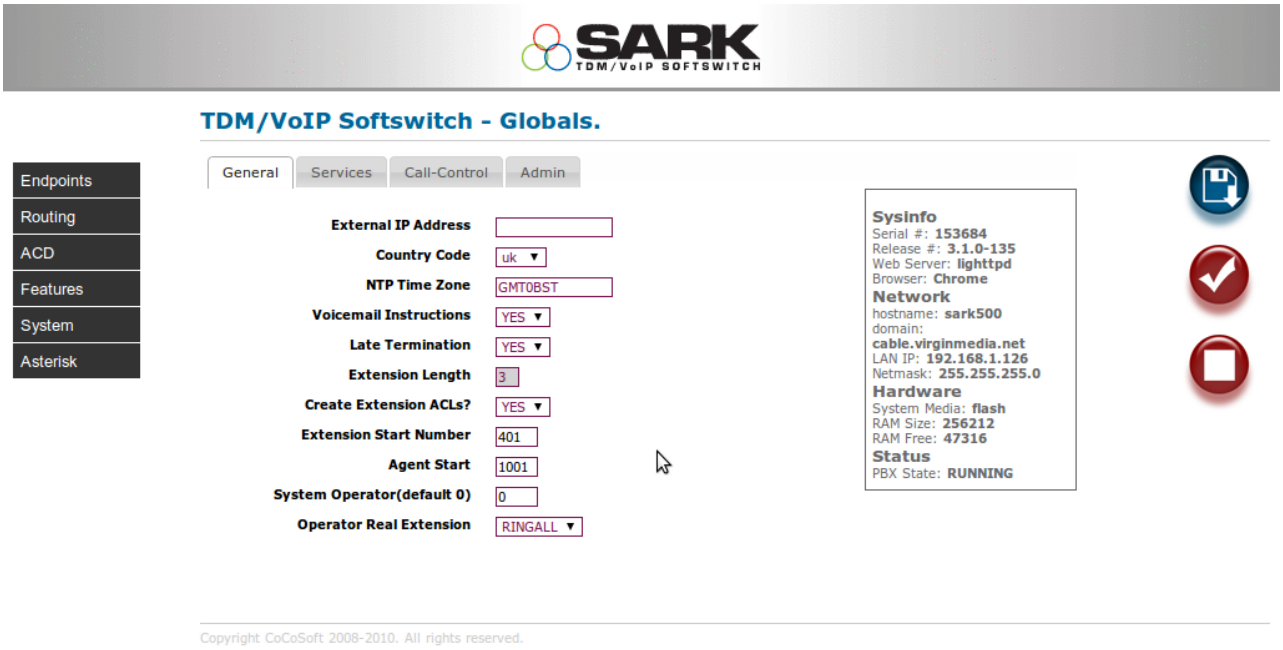
These two buttons perform similar functions however, Save will simply save your changes to the SARK database, your changes will not be *implemented* into the PBX itself until you issue a commit. This means you cam make a system change and release it all-at-once by issuing a Commit.

Logging on to SARK and setting the passwords

To log on to SARK you will need to know the IP address of the appliance. To obtain the IP address; after the appliance starts press the touch sensitive button to the right of the LCD screen. The button toggles the LCD display between the line status view and the IP address view. Type the IP address of the appliance in the address bar of any Web browser on the same subnet. The SARK login appears.



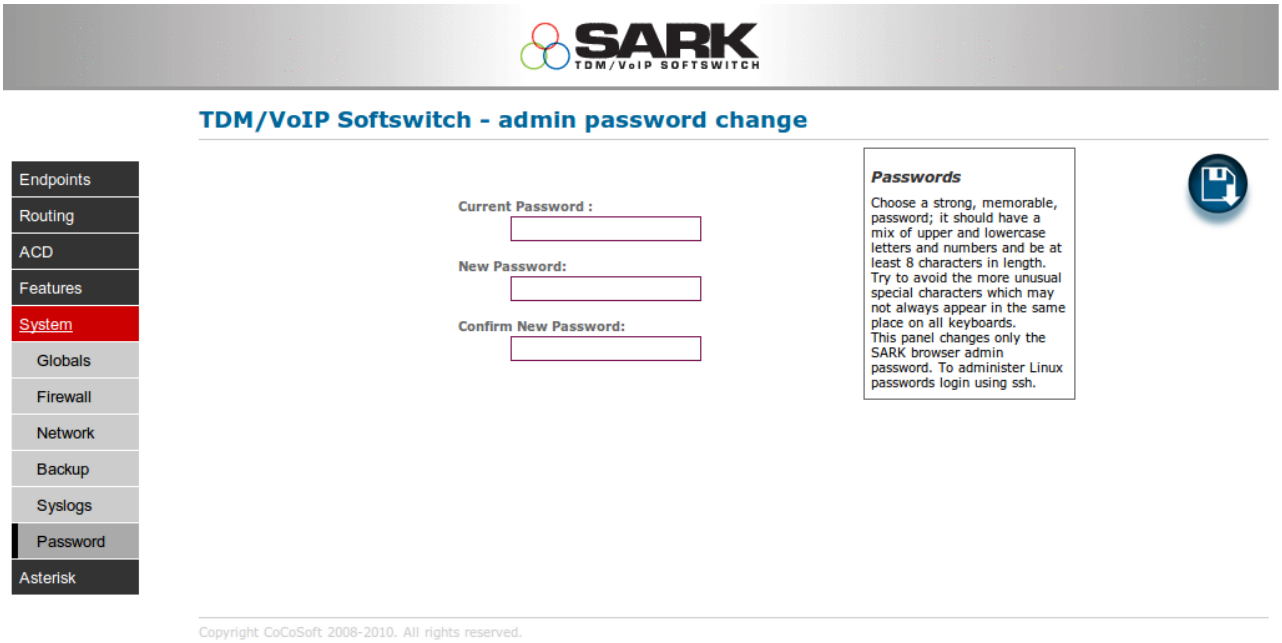
Type the user name and password in the appropriate text box. The default user name is *admin* and the default password is *sarkadmin*. Click the OK button. The SARK ‘globals’ page appears.



The System Statistics section (to the right) shows general status information.

Changing the Browser Password

For security reasons, we recommend that you change the default browser password immediately using the change password panel.



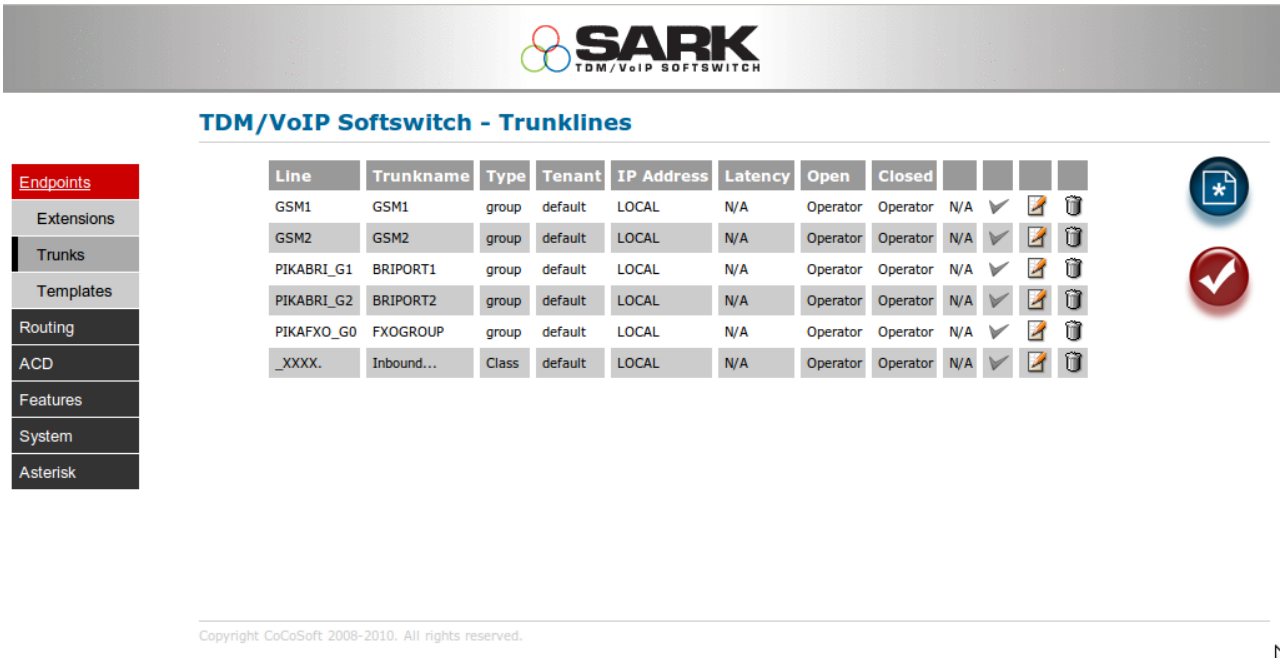
The system requires a password of at least 8 characters and you should try to use a mixture of upper and lower case and at least one numeric character. This panel is only responsible for the maintenance of the browser password. It does NOT change the root password of the box.

Changing the root user Password for the Appliance

Use an SSH client to open a session to the appliance using the IP address displayed on the LCD. Type the user name ***root*** and your password at the prompts. The factory default root password is ***sarkroot***. If you want to change the password for the root user account, type ***passwd root*** at the Linux command prompt. Follow the prompts to enter and confirm the new password. Password information is preserved between restarts.

Adding Trunks

Outbound trunks for FXO, Digital (BRI) and GSM, and inbound trunks for FXO and GSM are automatically installed on the platform so there is nothing to do to activate them. You may wish/need to add/create inbound and outbound routing (see Routes below).



If you are running FXO, GSM (or BRI without DNID presentation) then inbound calls will automatically be delivered to the 'operator' entry specified in the globals panel. The system default for the operator is to point to a ring group called RINGALL which simply rings all phones; you can, of course update this to point to another call-group, queue or IVR.

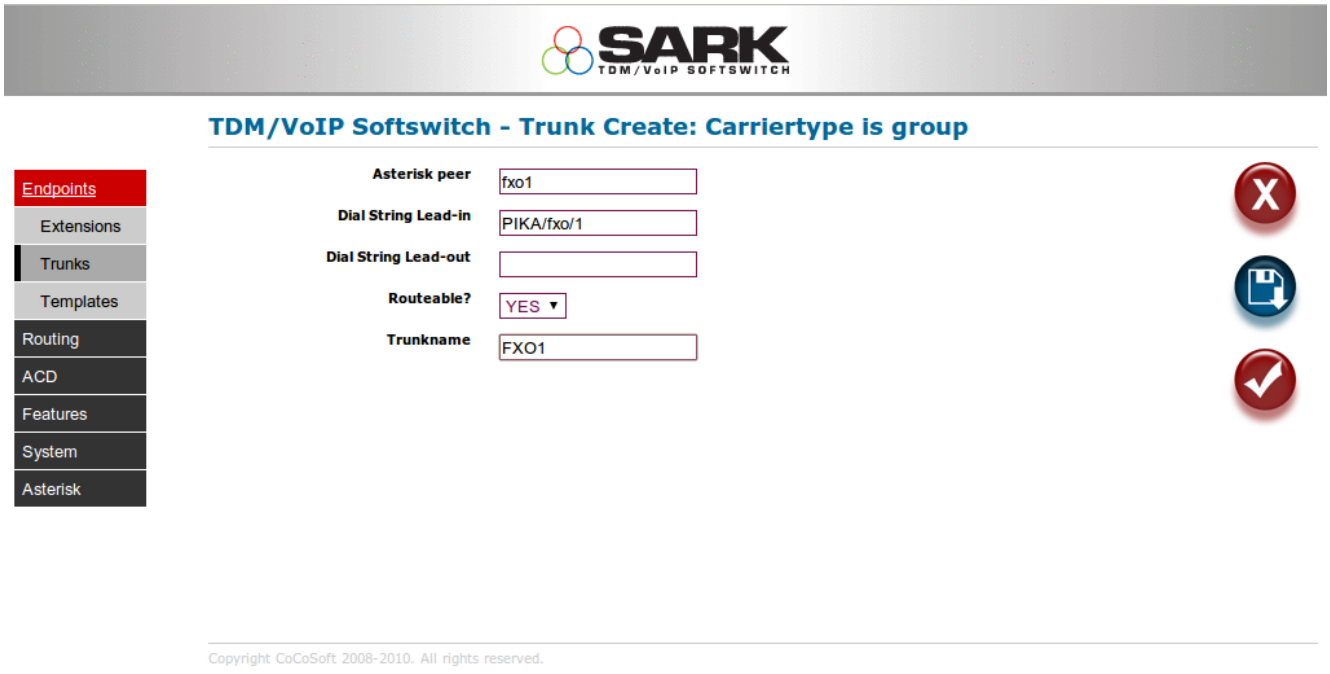
The routing of inbound calls falls into one of two categories; you can route by the Dialled Number ID (DNID) if you have digital lines with DNID enabled or you can route by channel name/number in the case of FXO and GSM lines.

To route inbound Digital (BRI) calls with DNID enabled,simply create one or more DiD (DDI) trunks to route the inbound calls. (see the next section). You may need to confirm with your telephone line supplier (PTT) how many digits will be presented in the DNID (it may be 3 or more digits depending upon your carrier).

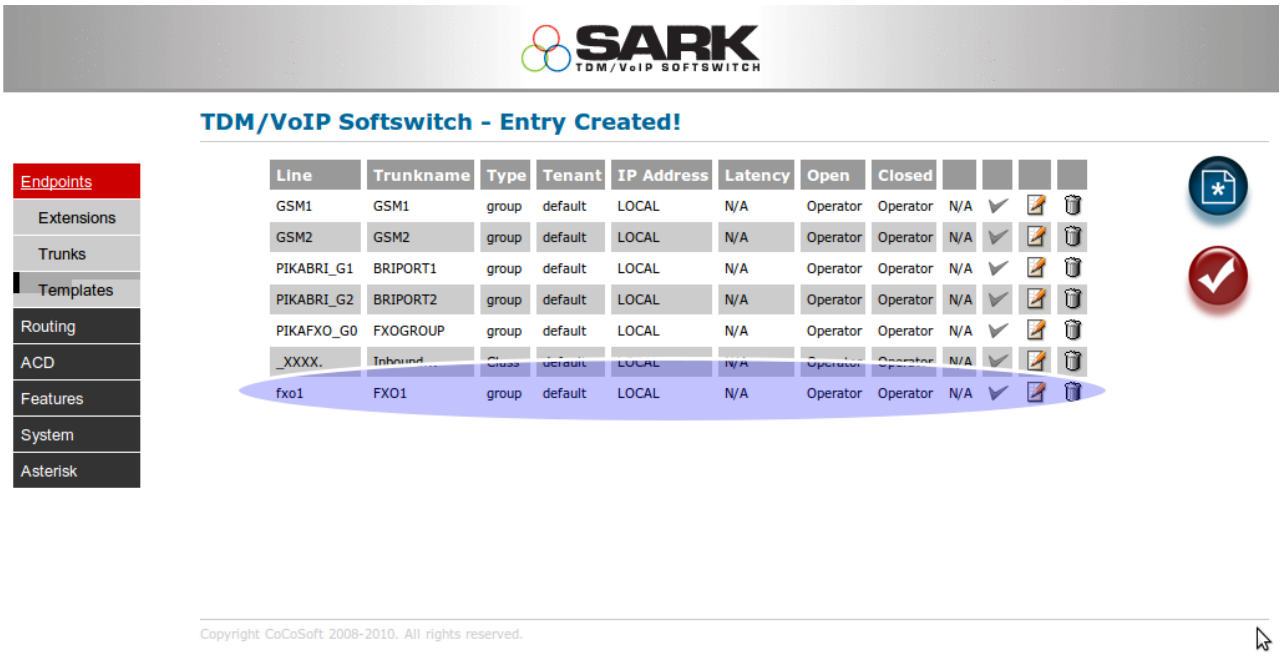
To route FXO and GSM channels you can set up a custom trunks and name them as follows:-

- for FXO, use fxo{*n*}; where {n} is the port number on the tdm card
- for GSM, use GSM{*n*}; where {n} is the port number on the gsm card.

The following shot shows the creation of a trunk to handle calls to and from channel 1 on the FXO card.



N.B. It is important that you set the *Routeable?* Variable to 'YES'. This will allow you to route calls arriving on this channel. This will create a trunk like the one highlighted below.



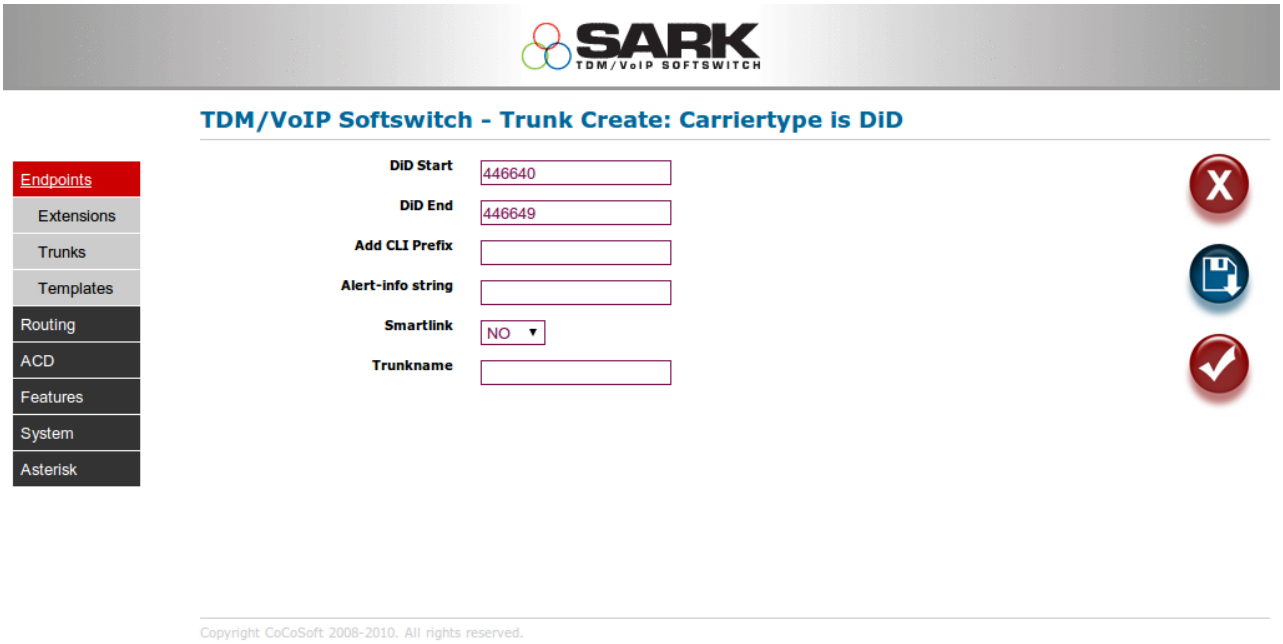
The trunk can be routed by clicking the edit button and filling out the 'open' and 'closed' route destinations in the normal way.

Adding a DDI (or DiD) Trunk

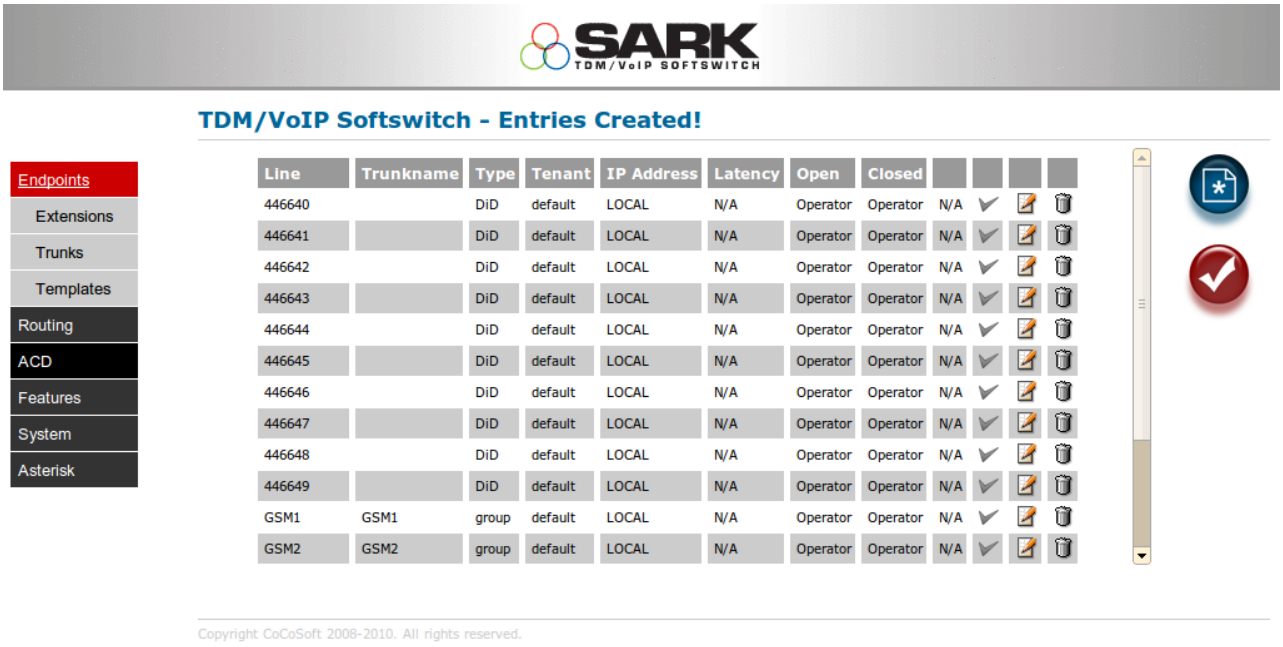
Digital (BRI) and VoIP (SIP/IAX2) inbound calls are routed by testing the 'Dialed Number ID' (DNID). To confuse matters further; in digital systems these numbers are often allocated in blocks by the telephone company and are known as DiD numbers (Direct Inward Dial) or DDI numbers (Direct Dial In). SARK has a special trunk construct to manage and route these numbers. It is called a *PTT_DiD_Group* trunk and you can define a set of one or more contiguous numbers using this trunk type.

Let's add an example DiD group with the range 446640 to 446649. To do that we can create a trunk object called a PTT_DiD_Group. Click the new button on the trunks panel and then choose "PTT_DiD_Group" from the drop down.

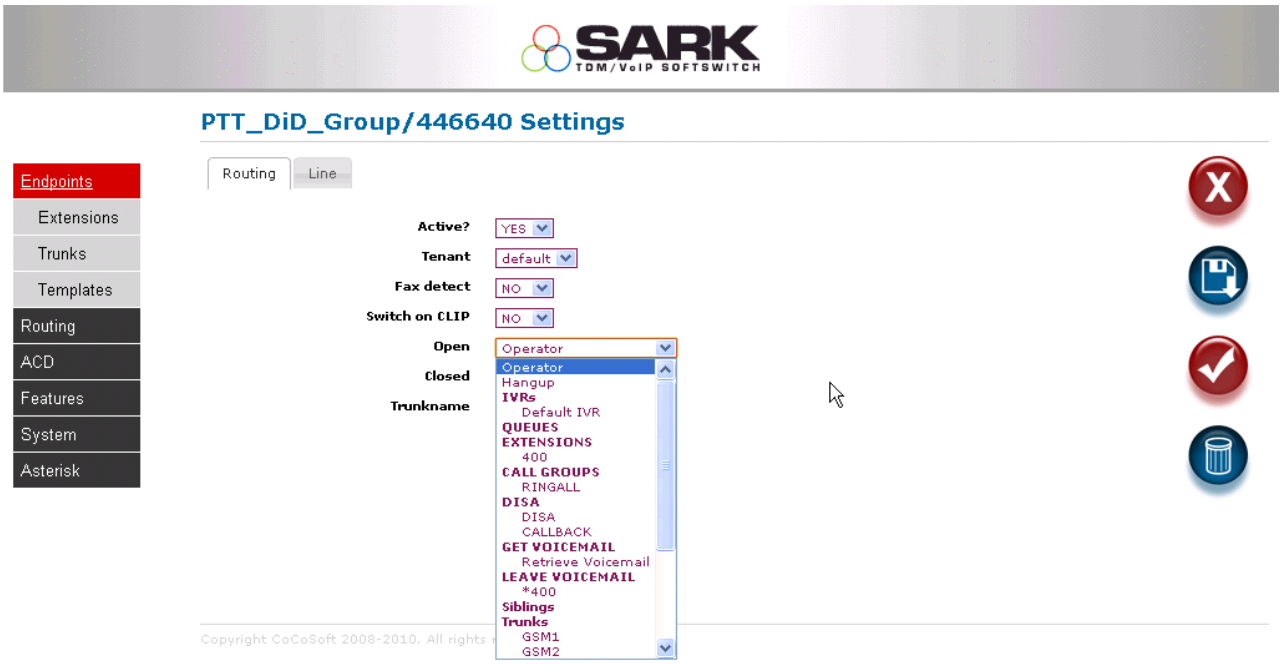
Now fill out the number range...



Click Commit to create the Trunks. Of course, the set of numbers in your Group can quite legally consist of a single number, in which case you would enter the same number in the start and end boxes. N.B. Even though the update screen requests positive confirmation of a DiD Group create, it is not unknown for a digit to be entered incorrectly causing hundreds or even thousands of DiDs to be created. Not to worry, if this happens you can simply regress SARK to the commit BEFORE you made a mistake.(see the section on Backup/Regress).



Our trunks have been created and we can now route them to their destinations by clicking the edit button on each trunk.



Route the inbound calls by clicking the open and closed inbound route drop downs and choosing a destination from the available system endpoints.

Smartlink

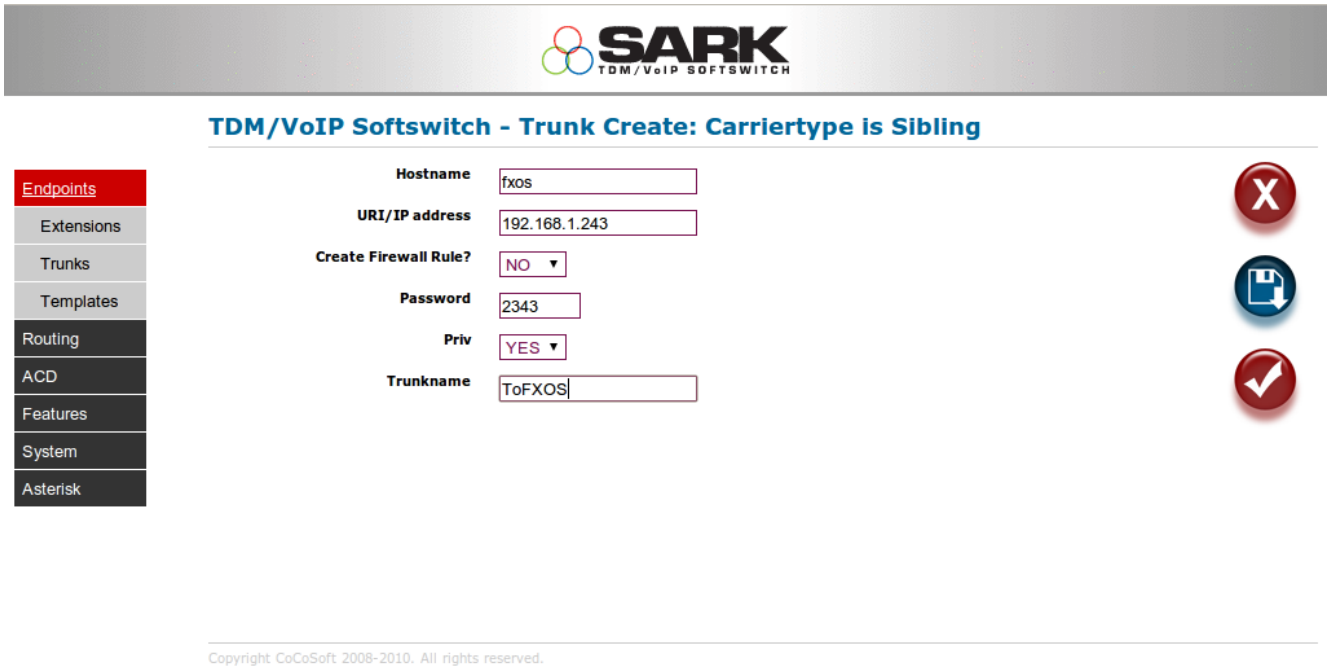
SARK has a flag called “smartlink” which you can set on when you create a DiD group. This is useful when you first create a set of extensions and then create your DiD trunks later. Smartlink will match the numbers of each extension to the right hand side numbers in the DiD group. For example, if you had an extension range of 400 to 420 and a DiD range of 668400 to 668420 then SARK will automagically route inbound calls on 668400 to extension 400; calls on 668401 to 401 and so on. This can save a lot of work when there are lots of extensions to be routed during system setup.

Adding a VoIP “Sibling” Trunk

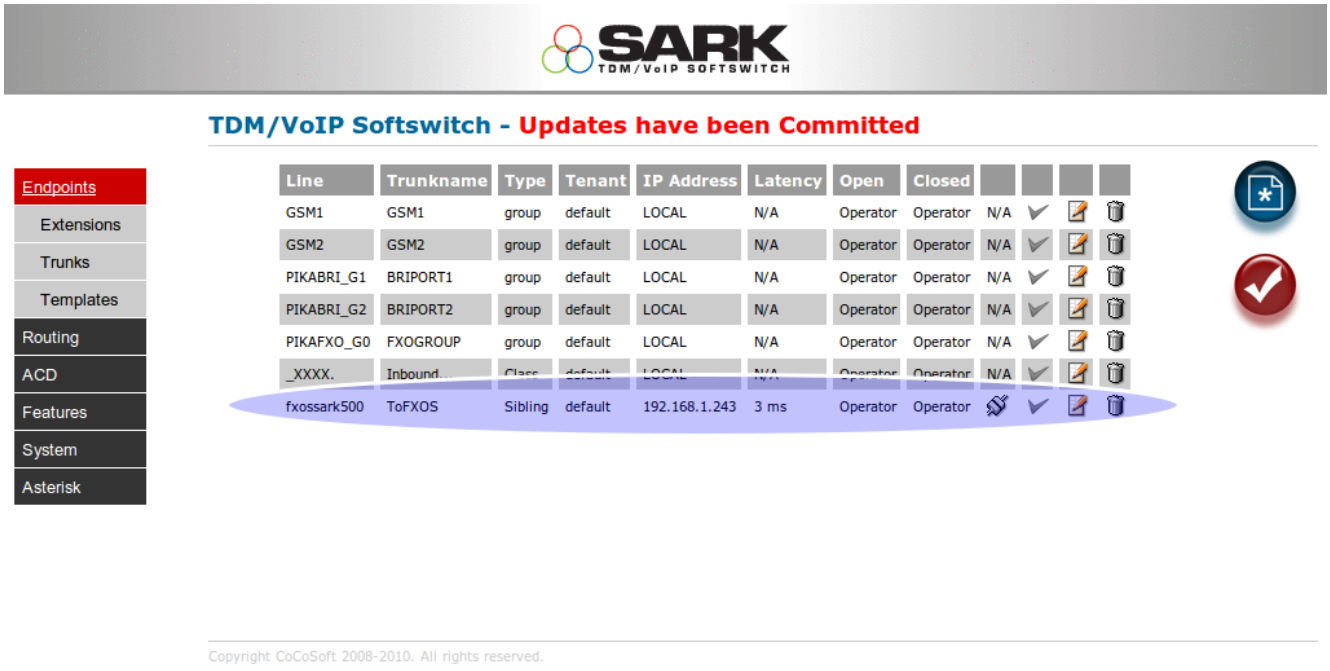
Sibling trunks provide a convenient way of routing calls between SARK instances. In this way, you can easily and quickly build a network of interconnected PBX’s which route calls to one another depending upon the number dialed.

To create a Sibling trunk choose SailToSail from the trunk type dropdown in Trunk create

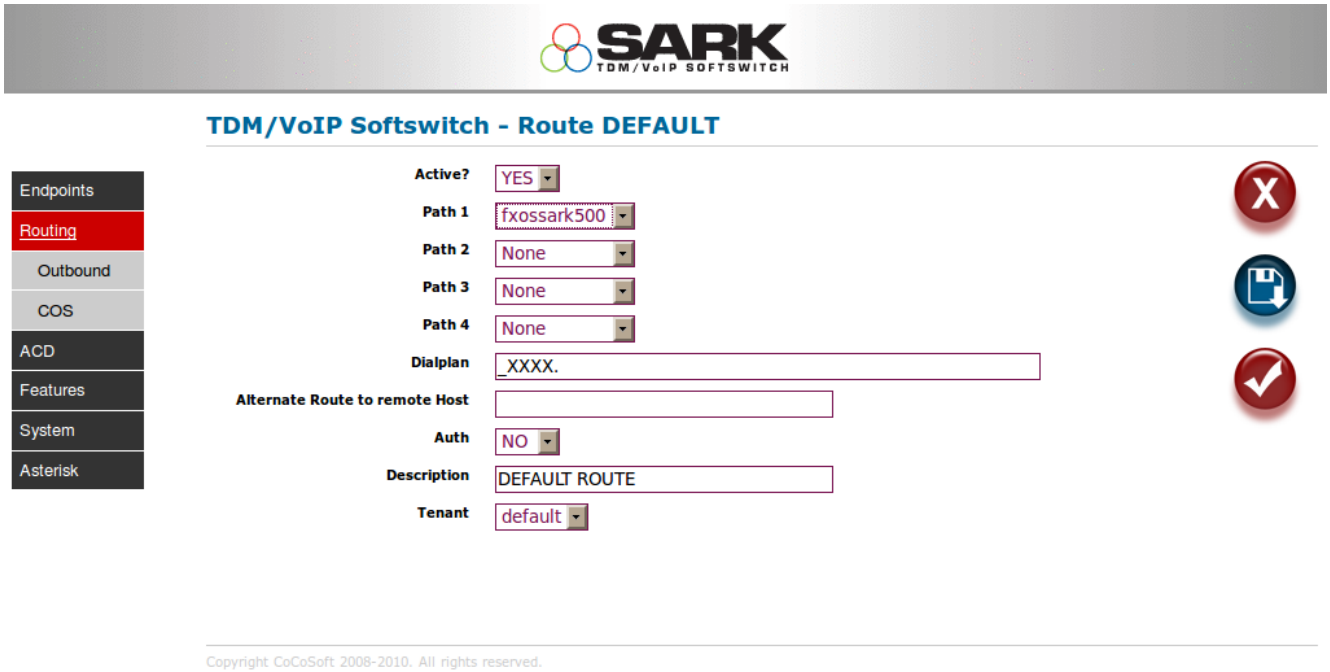
Now fill out the link information. In the Sibling name box enter the hostname of the PBX you wish to link to (just the short name not the FQDN). Fill out the IP address and include a password (this will be the same on both sides of the link). If the Sibling is remote (i.e. not in the same subnet) then you may want to have SARK create a firewall rule for it. In the example below, the other machine is in the same subnet so there is no need to create a rule.



Now click on save or commit to create your new trunk….

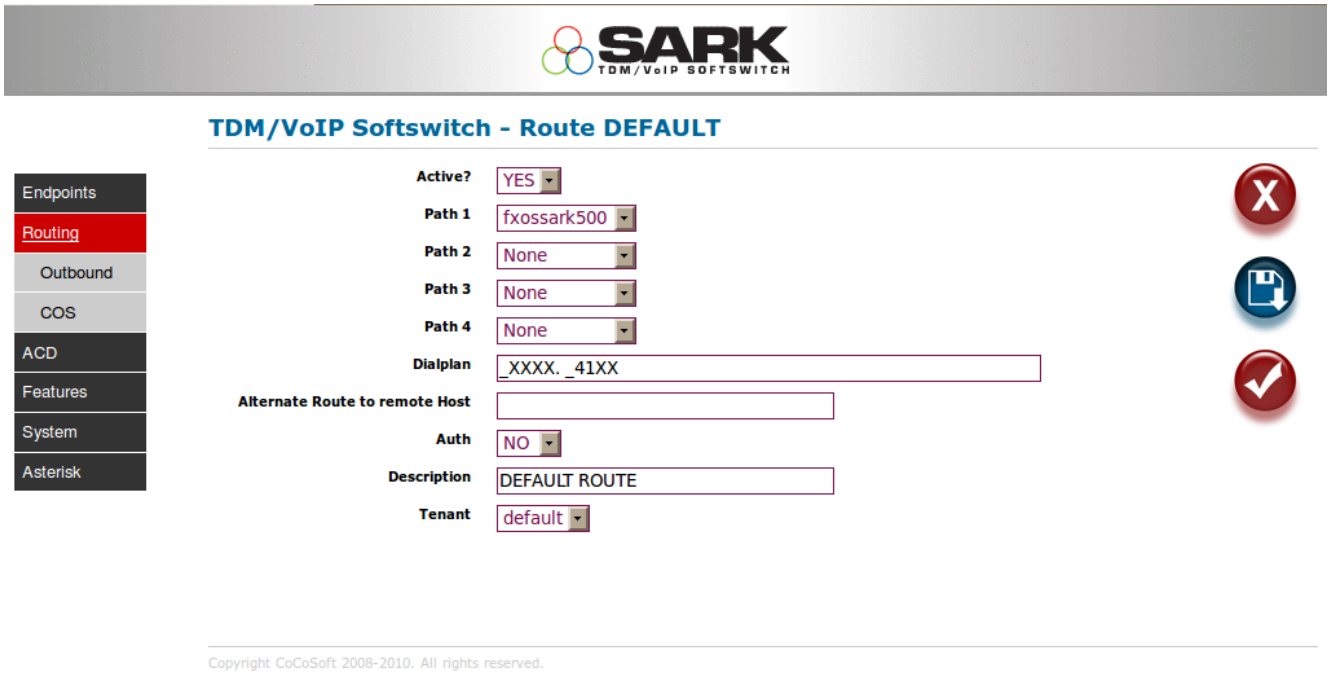


SARK has created a trunk for you with a peername which is the concatenation of your hostname and the remote hostname. You must now login to the remote PBX and create a mirror-image Sibling trunk to receive calls and send calls back. Once done, you can create a route to send calls up to the other PBX. In this example, we want to use the remote PBX as our outbound media server (perhaps for PSTN line consolidation) so we will send all of our outbound calls up to it. Here is a route that will do that...



Now, whenever we dial a PSTN number on the Sark500 it will be routed up to the other SARK PBX (perhaps a larger model) for termination onto the PSTN.

Perhaps we also want to allow extension-to-extension calling between the two PBXs. We can extend our route dial plan to include that also. Let's say, for example, that the extensions on the remote PBX are 4 digits in length and always begin with "41". We can modify our route as follows…



We've added the dialplan ***_41XX*** which will route any 4 digit dials beginning 41 up to the other PBX for termination. We would probably also create a dialplan on the remote PBX to route 3 digit numbers beginning 4 back to us (because throughout this guide, our examples use 3 digit extensions on the local box beginning at 401).

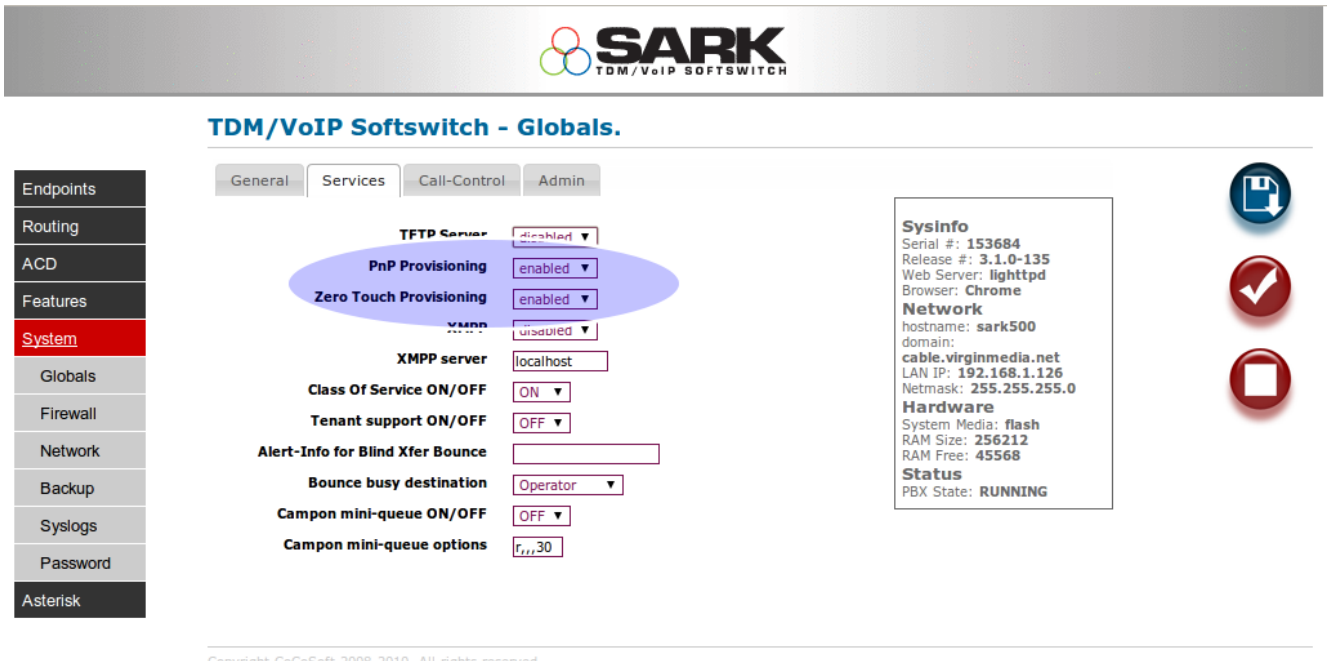
Adding Extensions with ZTP/PnP

Extensions for the FXS channels are already defined in the system database when it leaves the factory. There is nothing further to do. However, you will almost certainly want to define SIP extensions to the system. If you have SIP Multicast capable handsets (sometimes called SIP/PnP), you can use SARK500's ZTP auto-configure feature to set up your phones...

ZTP/PnP has several major benefits for users of SIP/PnP aware handsets:-

- There is no need to run DHCP option 66 and no need to run an on-board DHCP server on the SARK500.
- New phones can optionally be defined to asterisk “on-the-fly” by the listener/builder. This means that you can just connect them to the network and power them up. They will be automatically detected and provisioned by the SARK500.
- The provisioning stream is synthesized on-demand by the multicast-listener.
- No tftp or ftp files need be made available to the system.

To use ZTP and PnP you must turn the features on in the globals->services panel.



'PnP Provisioning' (or SIP Plug 'n Play) turns on the SIP Multicast listener. This allows SARK to asynchronously tell a SIP phone where its provisioning file is located *without* the need to modify the existing DHCP server or to run an on-board DHCP server or even to run the SARK unit at a static IP address.

'Zero Touch Provisioning' (ZTP) works in tandem with PnP. When ZTP is enabled SARK will not only inform a phone where its provisioning data is but it will also build a new extension on-the-fly and update Asterisk at the time the phone first requests it. What

this means is that you can simply power up a PnP aware phone and SARK will automatically create an extension for it in the SARK database, inform the phone where the provisioning data is and finally, regenerate Asterisk so that it is aware of the new extension. Thus the phone will come up, be automatically provisioned, register with Asterisk and be immediately ready to make and receive calls. This is very useful for medium to large roll-outs because it needs no prior set-up.

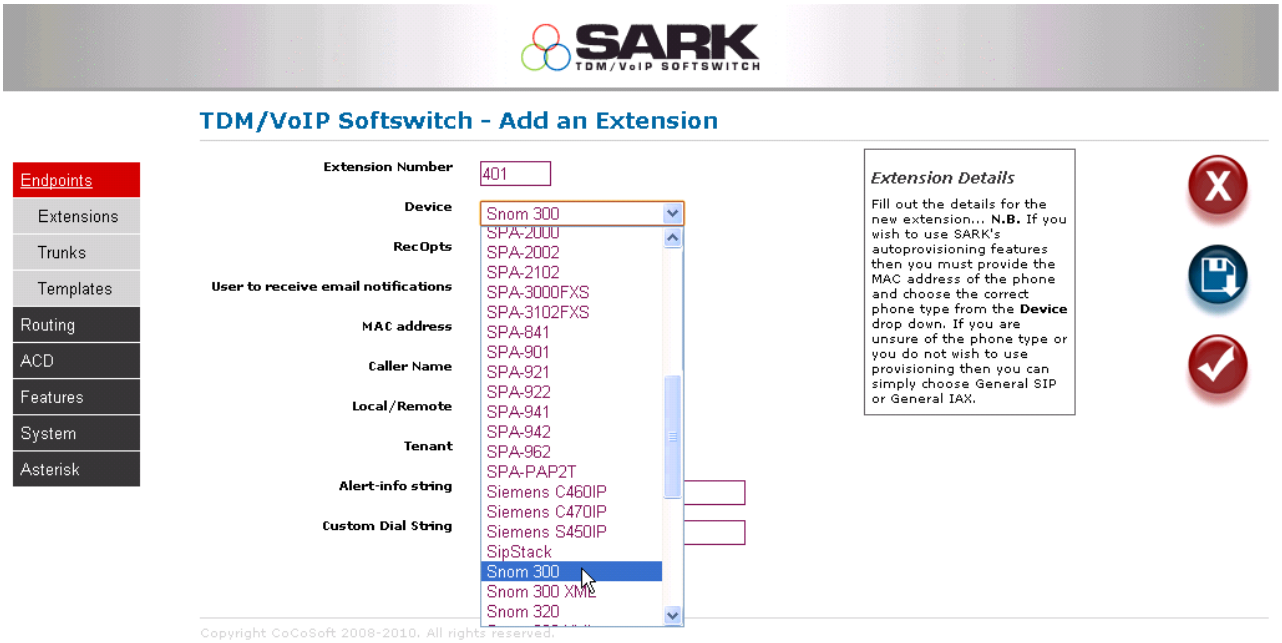
For security reasons, you should only run ZTP during initial system installation or when you have new phones to roll-out. The remainder of the time it should be switched off. You can, of course leave the PnP listener running all the time.

In order to be able to run multiple PBX servers in the same subnet, the Multicast listener must be able to selectively reply to only those PnP aware phones which it "owns" (i.e. are defined to it). For this reason, unless ZTP is enabled, SARK will only respond to provisioning requests from phones for which it already has an entry in its database. It recognizes individual phones by their MAC addresses. In this way, multiple SARK500 servers may all be listening for Multicast requests on the same subnet and each can respond to its own group of phones without disrupting other listeners.

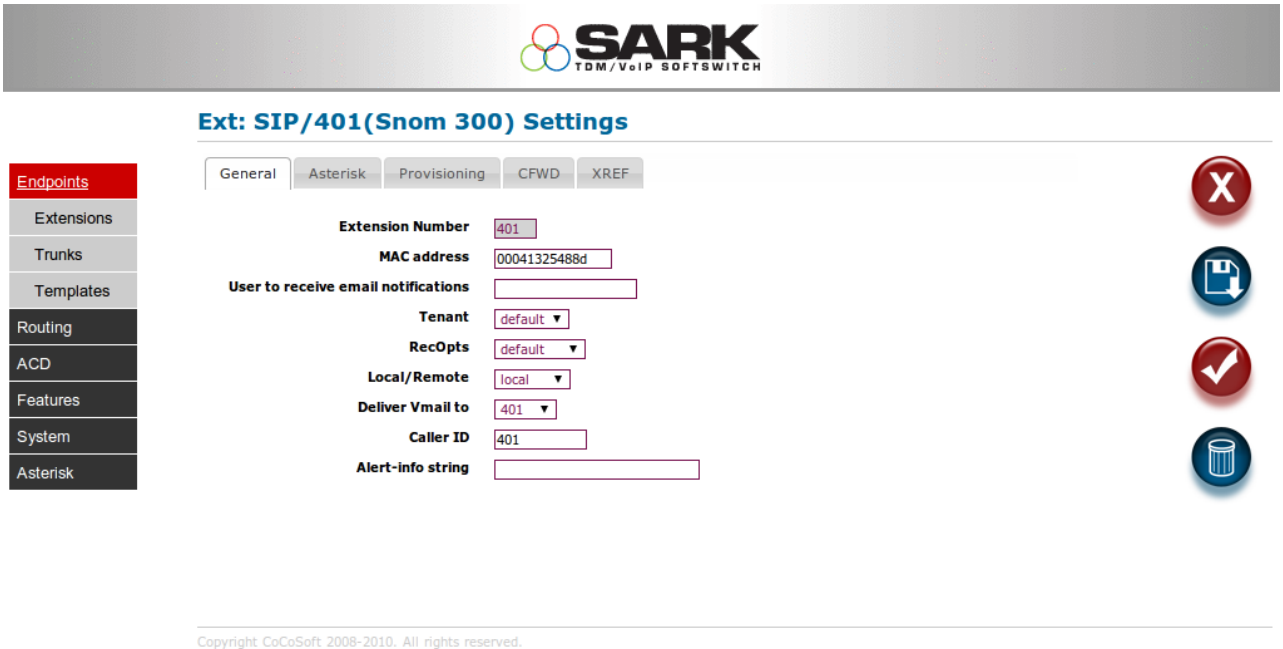
Adding non-PnP Extensions and TFTP provisioning

For non-PnP-capable handsets you can add new extensions using the SARK extensions panel …

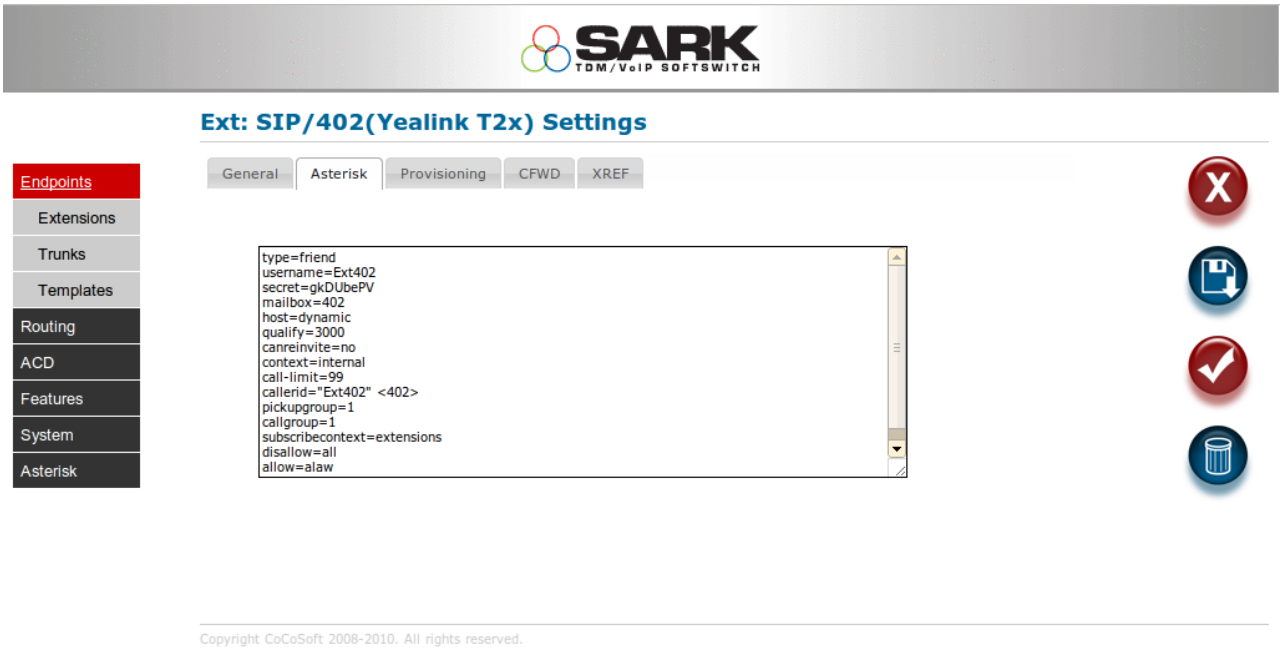
To add an extension; click extensions at the top of the navigation menu. Then click the “new” button to create an extension.



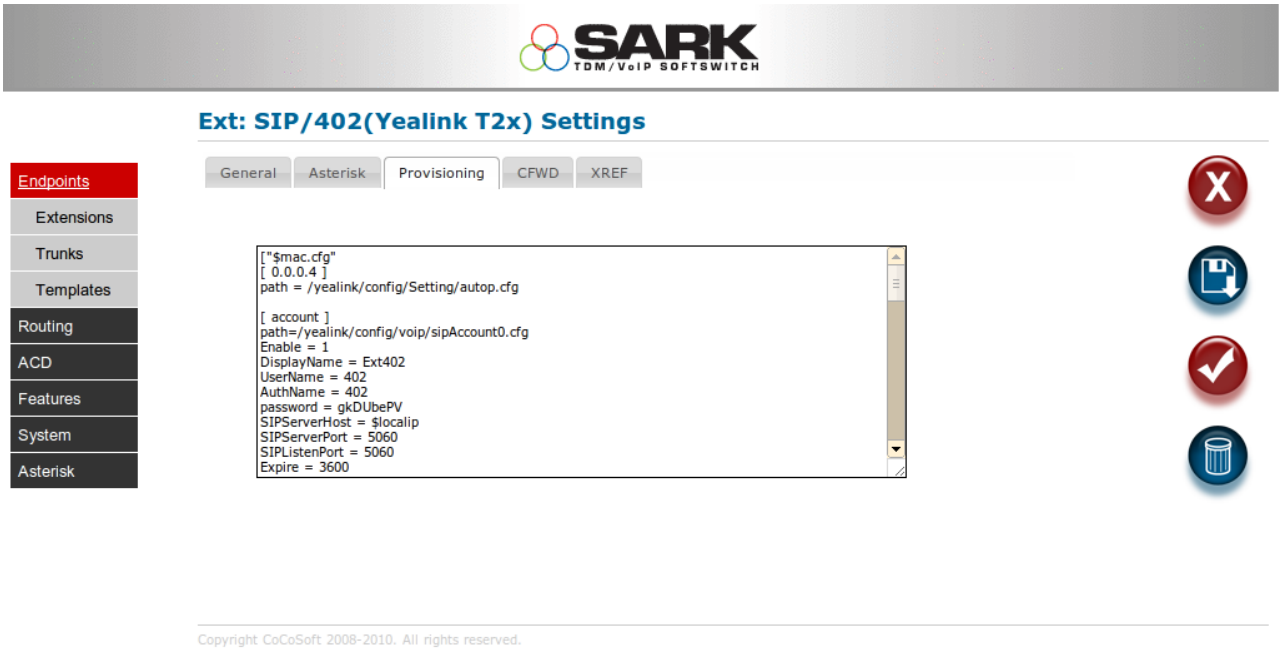
Choose the phone type from the device drop-down. Fill out the details for your phone and (optionally) enter the MAC address if you want SARK to automagically generate a TFTP provisioning file for your phone type. SARK has a fairly extensive phone database and it can provision most commercially available phone types. Click the Save button if you have several entries to create or simply click the Commit button to create this entry and regenerate the Asterisk files immediately.



SARK will create the necessary Asterisk entries with randomly generated passwords and (optionally) the correct ACL checking for your subnet…



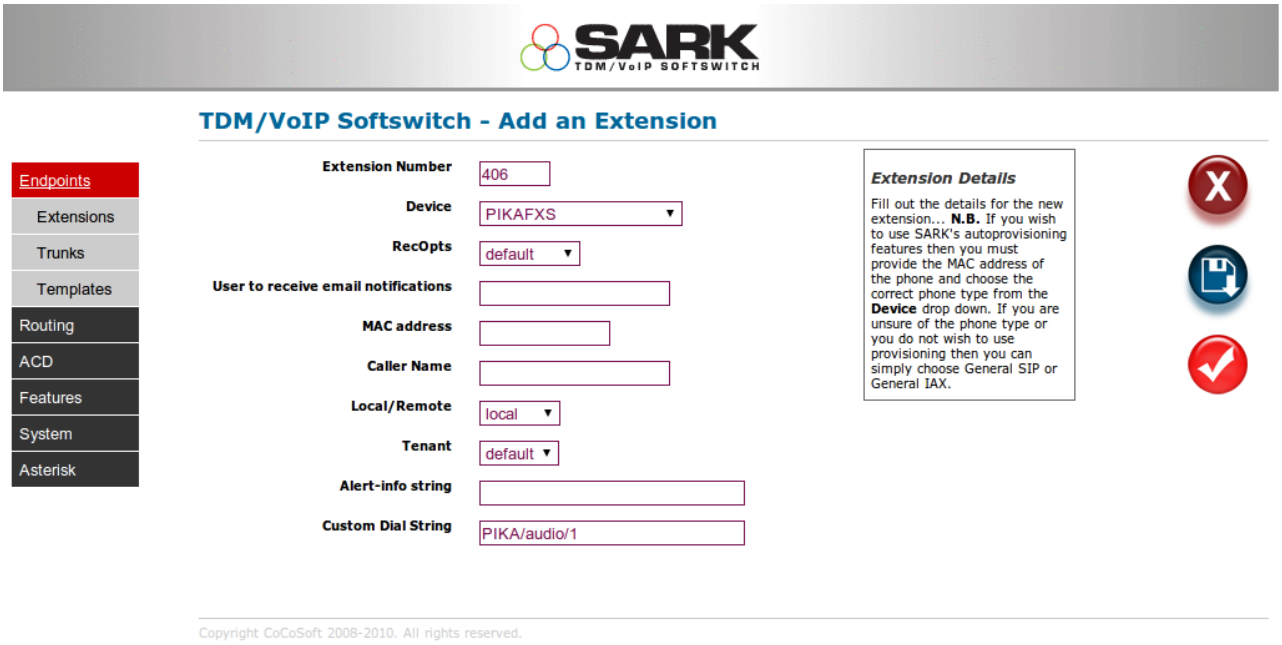
…And it will also generate the correct provisioning data for the phone type you chose…



You may freely modify any of these entries SARK has generated for you and save them back to the database. You can also freely create your own provisioning templates and profiles allowing you to tailor the SARK provisioning platform to suit your individual needs.

Adding a Line-out extension

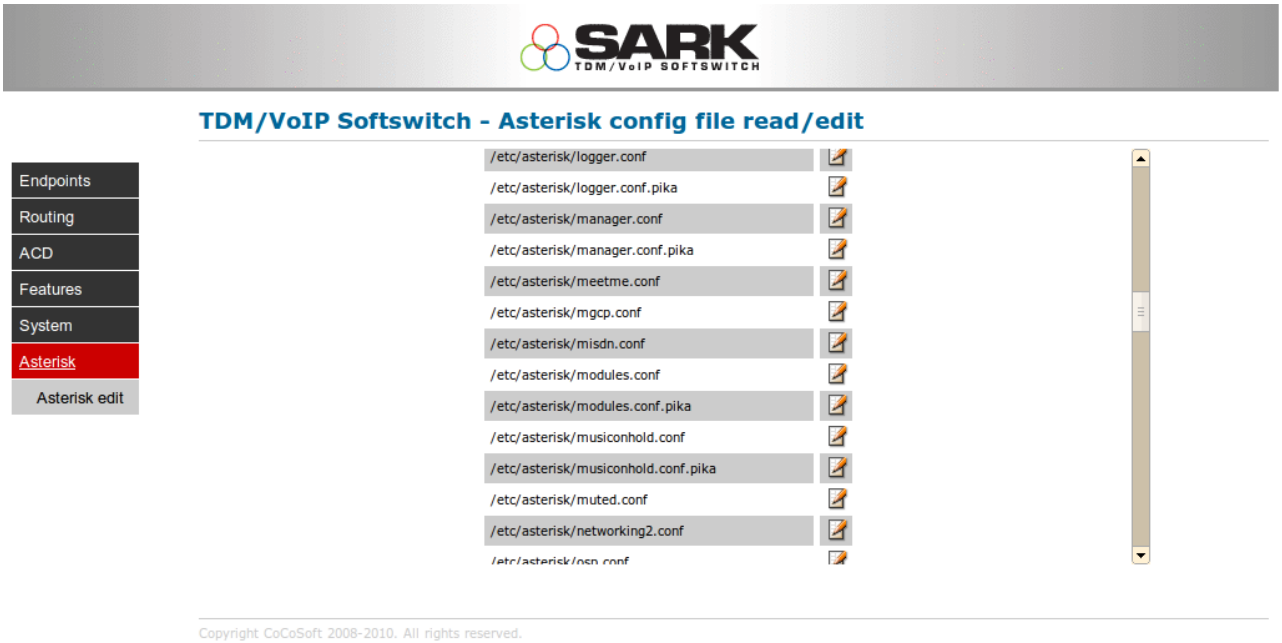
The SARK500 appliance is equipped with a line-out port that you can use to connect to a public address system. To add an extension for the port, simply create an extension; choose a device type of *PIKAFXS* and enter a Custom Dial String of *"PIKA/audio/1"*



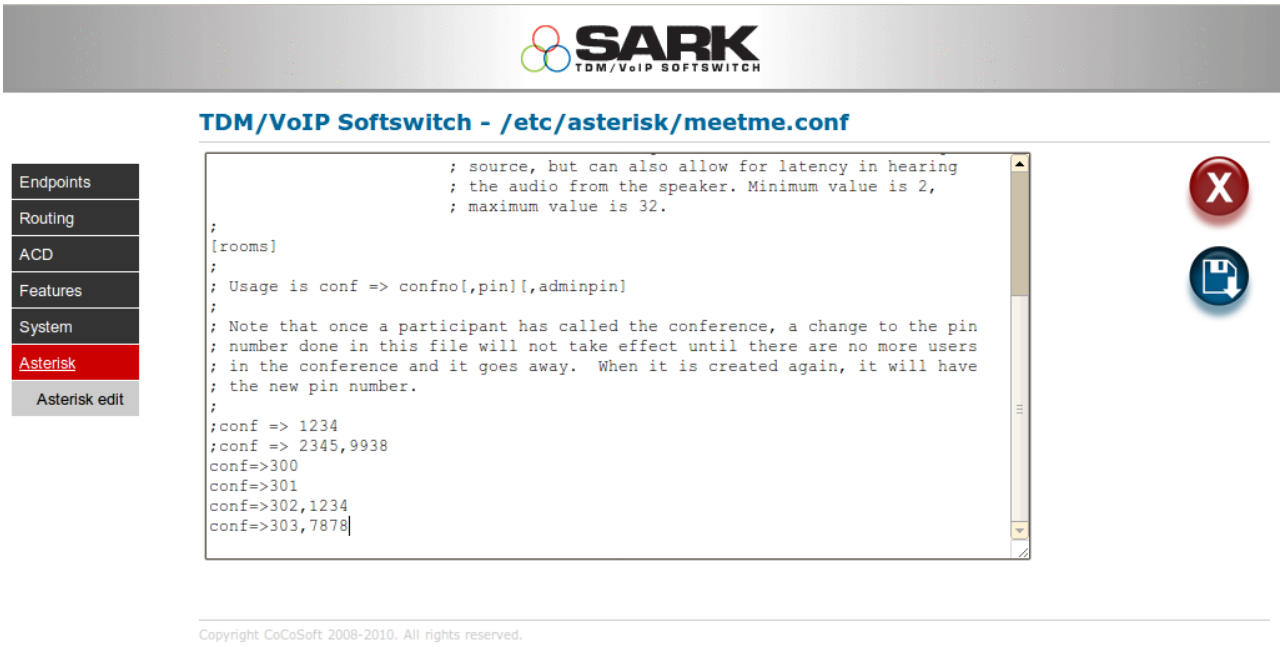
To test the line-out extension, connect your public address system to the appliance (or just plug a set of headphones into the line out jack on the back of the unit – it's a standard mini-jack plug). Dial the extension of the PIKAFXS extension from any extension. When you speak, you should hear your voice on the public address system (or in the headphones).

Conferencing

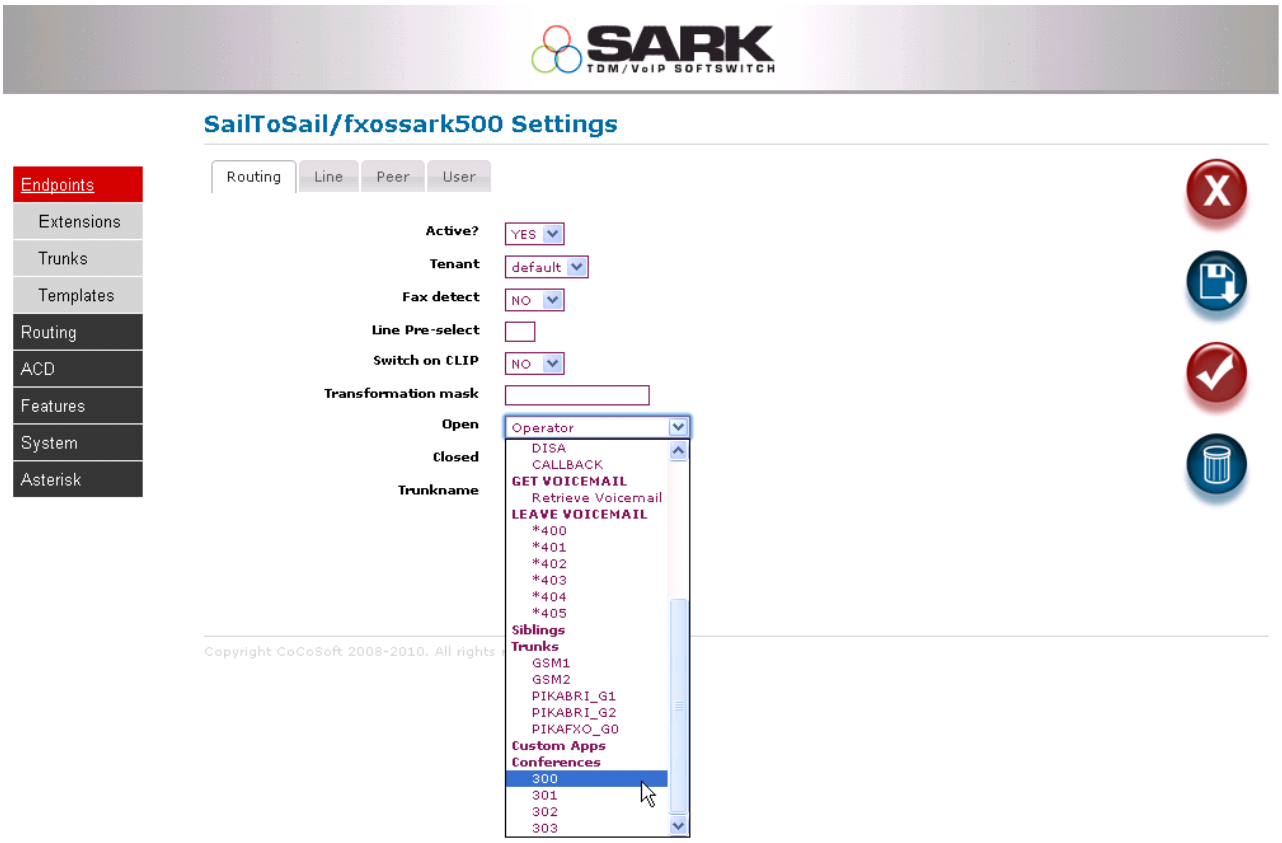
Conference room 300 is predefined to the system at the factory. If you want to add other rooms or change the conference room number you can modify the underlying Asterisk ".conf" file directly. As a general rule, SARK keeps its management of Asterisk files to a minimum. You can directly modify most asterisk files just using the Asterisk Edit panel…



To modify a conference room simply choose the meetme.conf file and make your changes directly. In the example below we've just added conference room 30,302 and 303 and now we are about to save it back.



Once the conference room has been created in this way we can either dial it directly from any extension or we can select it in a route (it will auto magically appear in all end-point drop downs)… Below we route a Sibling trunk to our new conference room so we can host a conference with an external caller.



Interactive Voice Response

To create an IVR, you must first create a system greeting.. Simply go to any phone and dial *60*+nnnn, where nnnn is a 4 digit number that we will associate with our recording. –

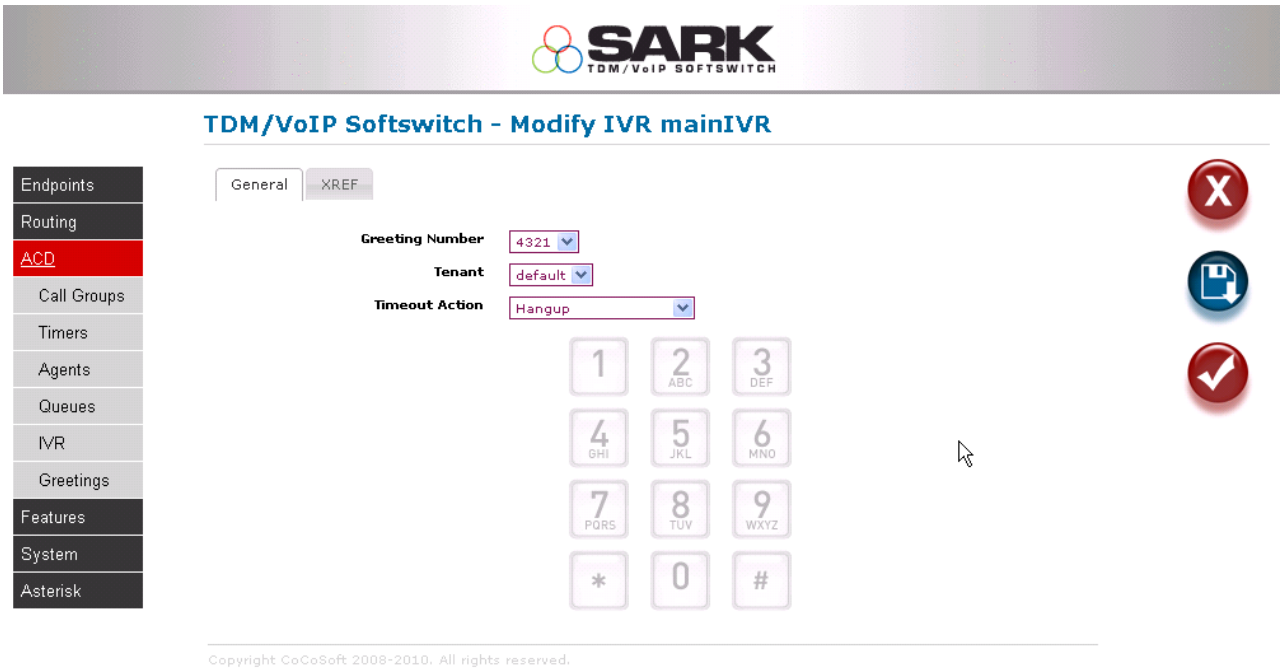
60*4321

The auto attendant will lead you through the process of creating a recording. Let's record a message like this…

'Welcome to ABC Widget Corporation. Press 1 for sales, press 2 for accounts or please hold for an operator'

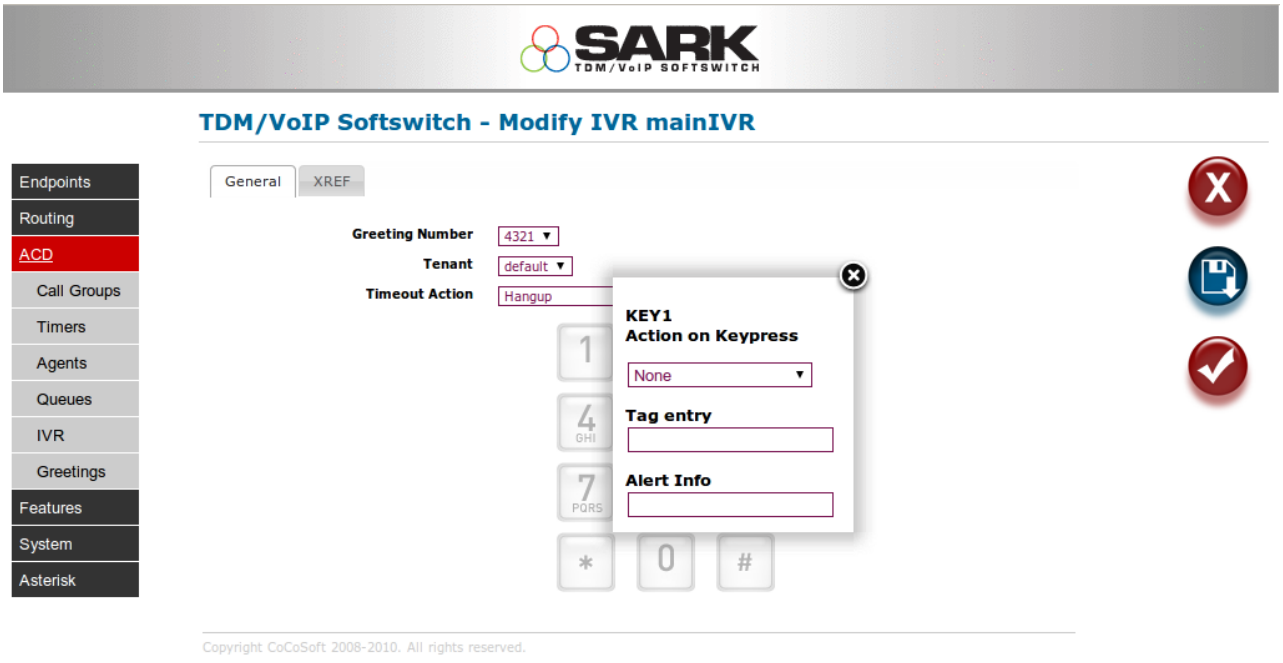
We can now go ahead and create our IVR. Choose the IVR panel from the left hand menu and then click "New" to create a new IVR. Name your IVR "mainIVR" and choose greeting 4321 from the greeting drop-down. Now click the edit icon to fill oput the details for your IVR

We choose our greeting (4321) from the drop down and then decide what to do if the caller does not press a key. Our greeting said "…or please hold for an operator", so we can choose "Operator" from the timeout drop down. Here's how it looks...

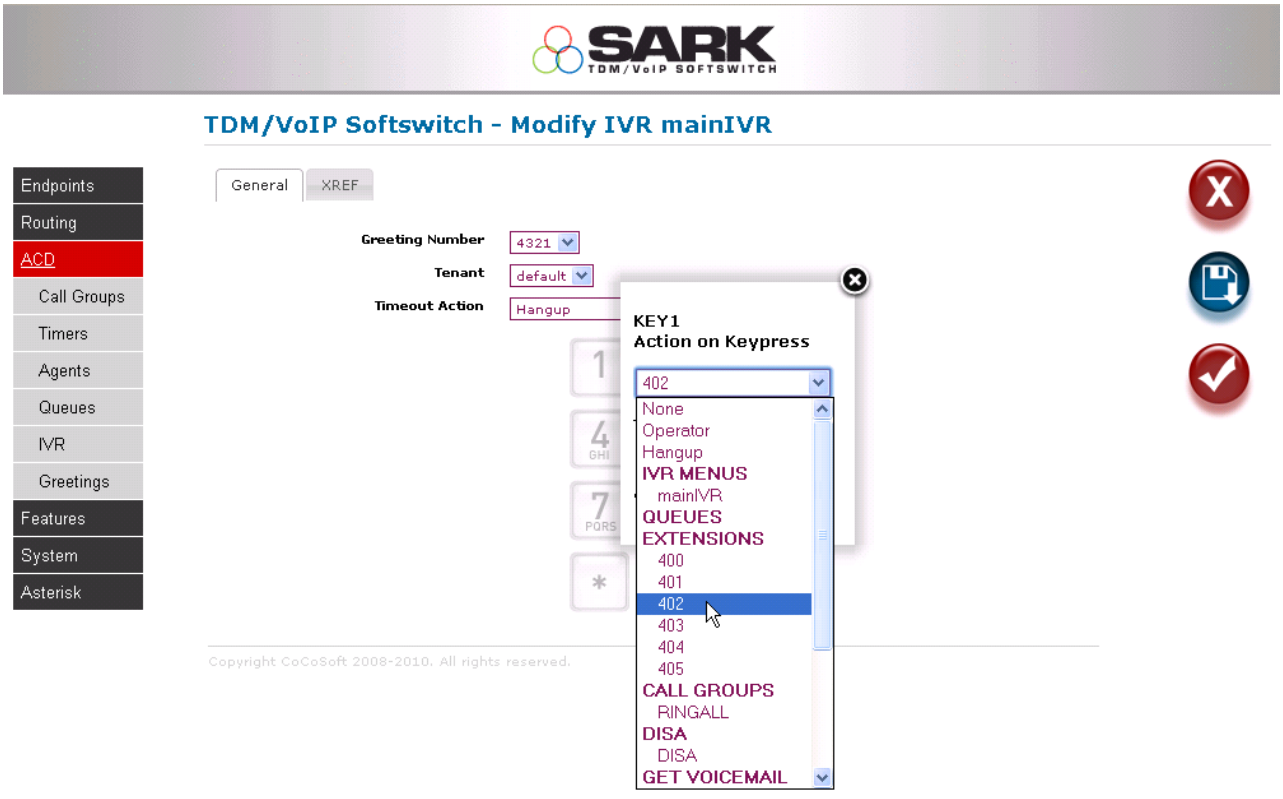


Choose a timeout action from the drop-down menu. You can timeout to any valid endpoint defined to the system.

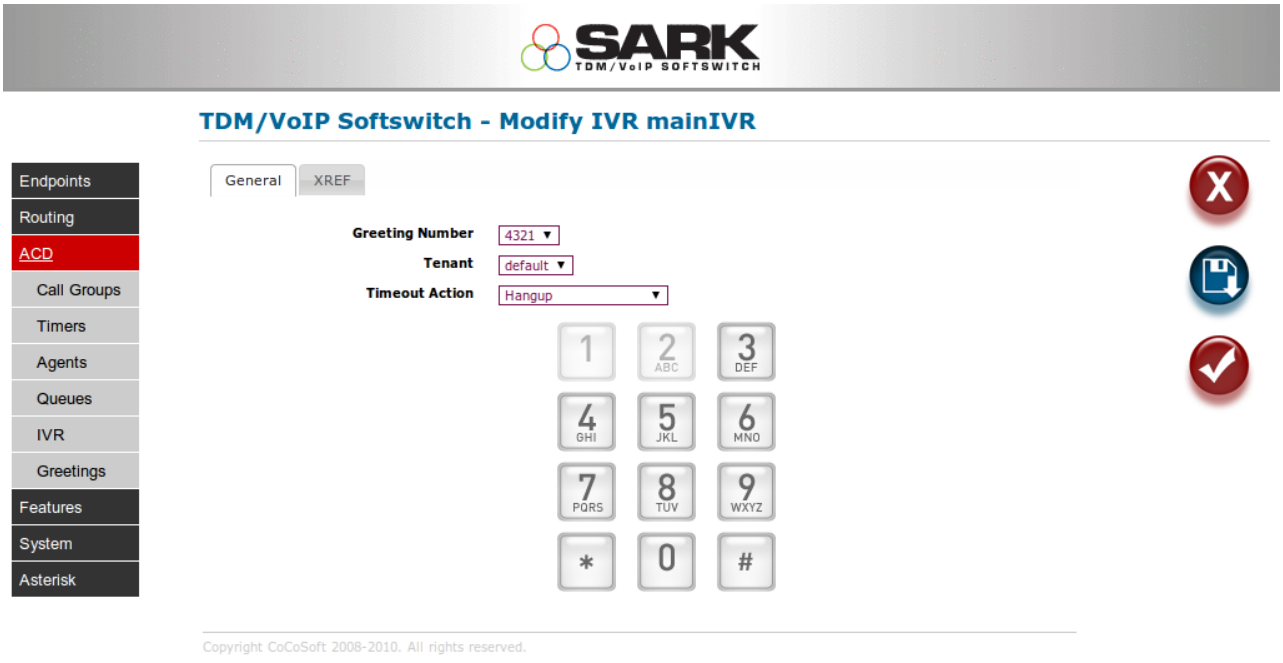
We can now choose our key-press operations by clicking on the keypad graphic and choosing an endpoint from the drop down. This will pop-up a Jquery-ui box for each key.



And….



Finally, after committing the IVR, if we return to the screen we see .that SARK has updated the graphic to show at a glance which keys are in use… You can see what each key is set to by hovering your mouse over the key.

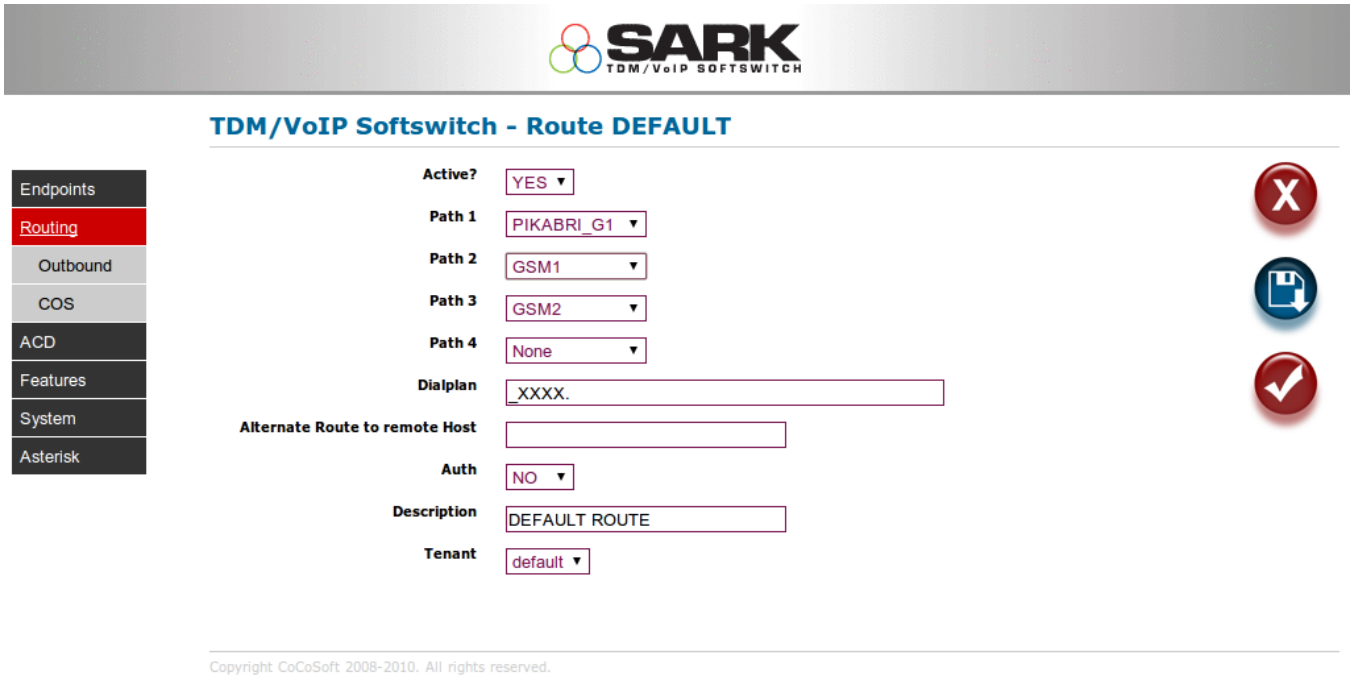


The IVR is now ready to use.

Adding Routes

Within SARK, an outbound route consists of a set of dialplans and up to four outbound pathways (Trunks). The SARK HPE processing engine will choose the first available route which matches the dialed number ID (DNID) . There is a pre-installed route on the SARK500 called DEFAULT and this route will be preset based upon the hardware fitted to the unit when it was shipped.

You can modify the DEFAULT route or add new routes of your own. To add a route, click the link Routing->Outbound in the left navigation menu and then click the “new” button. In the example below we show a DEFAULT route which has been set to send all calls out over ISDN (PIKABRI-G1). If the ISDN channel becomes unavailable then the route has been set-up to fail-over to the two GSM channels that this particular unit is fitted with (GSM1 and GSM2).

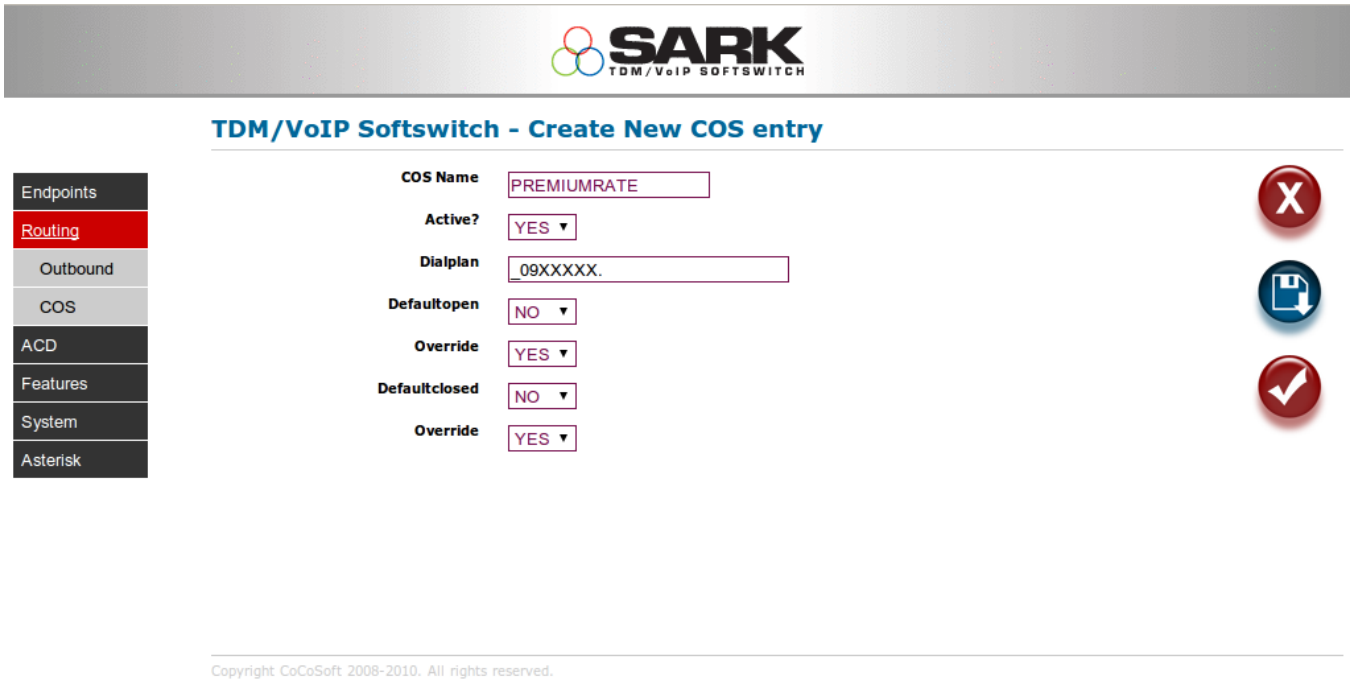


You can configure up to four trunks for each outbound route. They will be successively tried in the event that a trunk fails or refuses to honor the Dial. The dialplans are conventional Asterisk dialplans and you can code as many as you wish using white space as a separator. You can create multiple routes to send different call types over different trunks (based upon the dialed number).

Class of Service (COS)

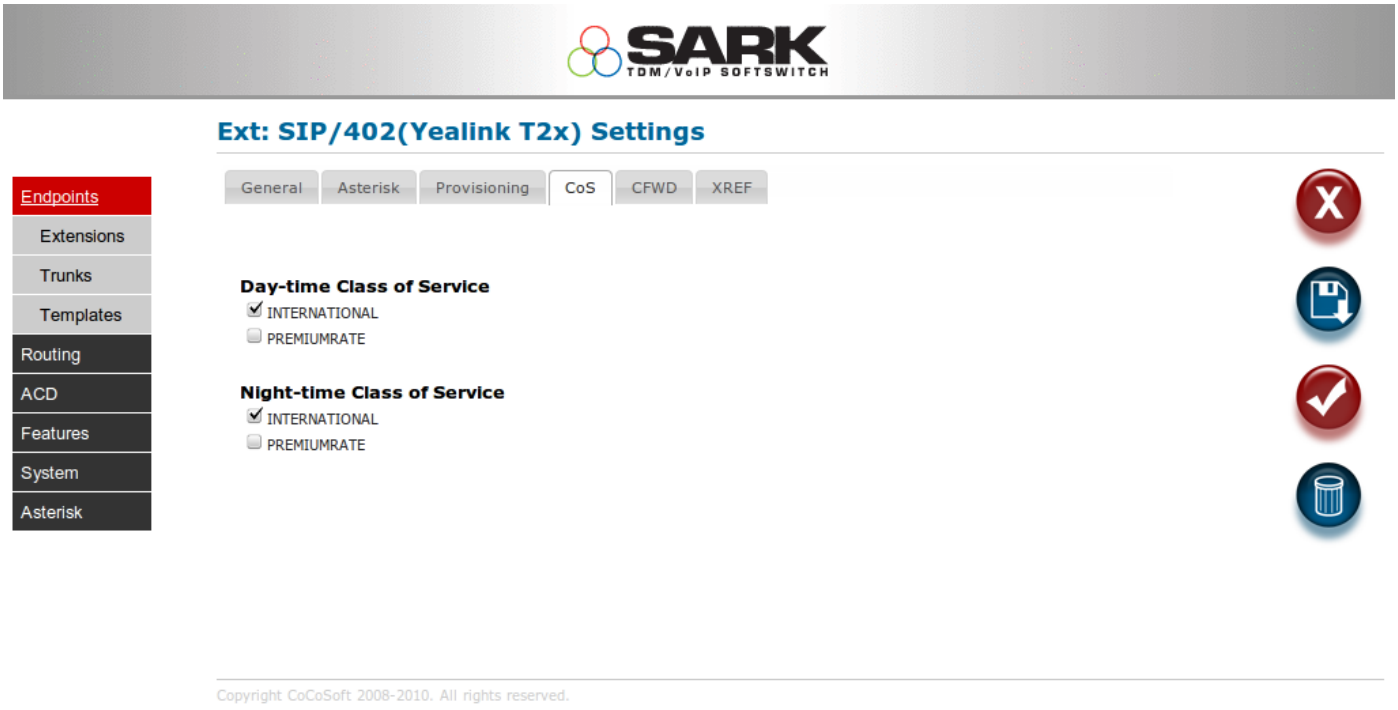
SARK's Class of Service feature allows you to select classes of number to which you would like to restrict or deny access. A simple example might be to deny all extensions the ability to call premium rate numbers (0900 in UK). Another might be to deny certain extensions the ability to call international or to call mobile numbers. You can also use it to “blacklist” individual numbers.

To use COS, you set up class rules and apply them to extensions, either selectively or globally. The rules themselves are simple number patterns in regular Asterisk format. As an example, to create a COS rule to restrict access to 0900 numbers, select the COS option from the Routing menu and click the “New Object” button...



Give the COS rule a name (PREMIUMRATE) and a pattern mask (_09XXXXX.). Finally, we need to decide if we are going to selectively or globally apply this COS. For something as important as premium rate numbers, we will apply the rule globally by setting the two Override drop downs to “YES”. Override means that the rule will be applied to all extensions irrespective of any other settings they may have. If override is not set you can set COS rules individually for each extension in the Extension->Edit->COS tab.

In this tab you can check (tick) those classes that you wish to enforce for this extension. You can also elect to have a class which is in force during normal hours or during closed hours (or both). This can be quite useful to restrict phones from making certain kinds of call during closed hours.

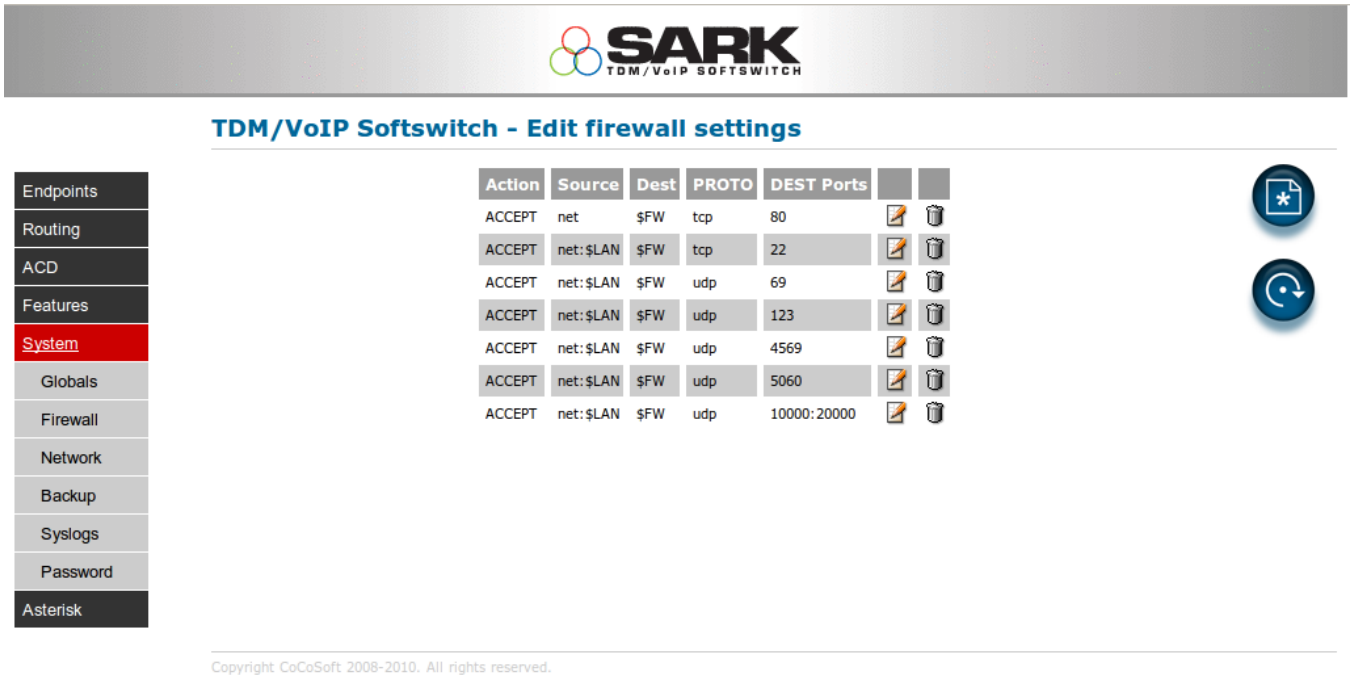


Firewall

The SARK500 uses the Shorewall firewall to manage access to the PBX. Shorewall is a very comprehensive and powerful Linux firewall which, like all firewalls, can be complex to set up. However, SARK completely automates the initial firewall creation. It also makes management relatively straightforward by exposing a set of simple rules for the user to configure. In its default configuration, the firewall is set up to only allow traffic from the local subnet (the subnet is detected automatically by SARK at system startup). . Furthermore, only the following ports are opened:-

SSH (TCP 22)
TFTP (TCP 69)
NTP (UDP 123)
IAX2 (UDP 4569)
SIP (UDP 5060)
RTP (UDP 10000:20000)

The exception is port 80 (HTTP), which is initially open without restriction. Each of the packet types (HTTP. SIP etc) has an entry in the firewall panel associated with it. You can view and modify these entries by choosing the Firewall option from the System->Firewall tab.

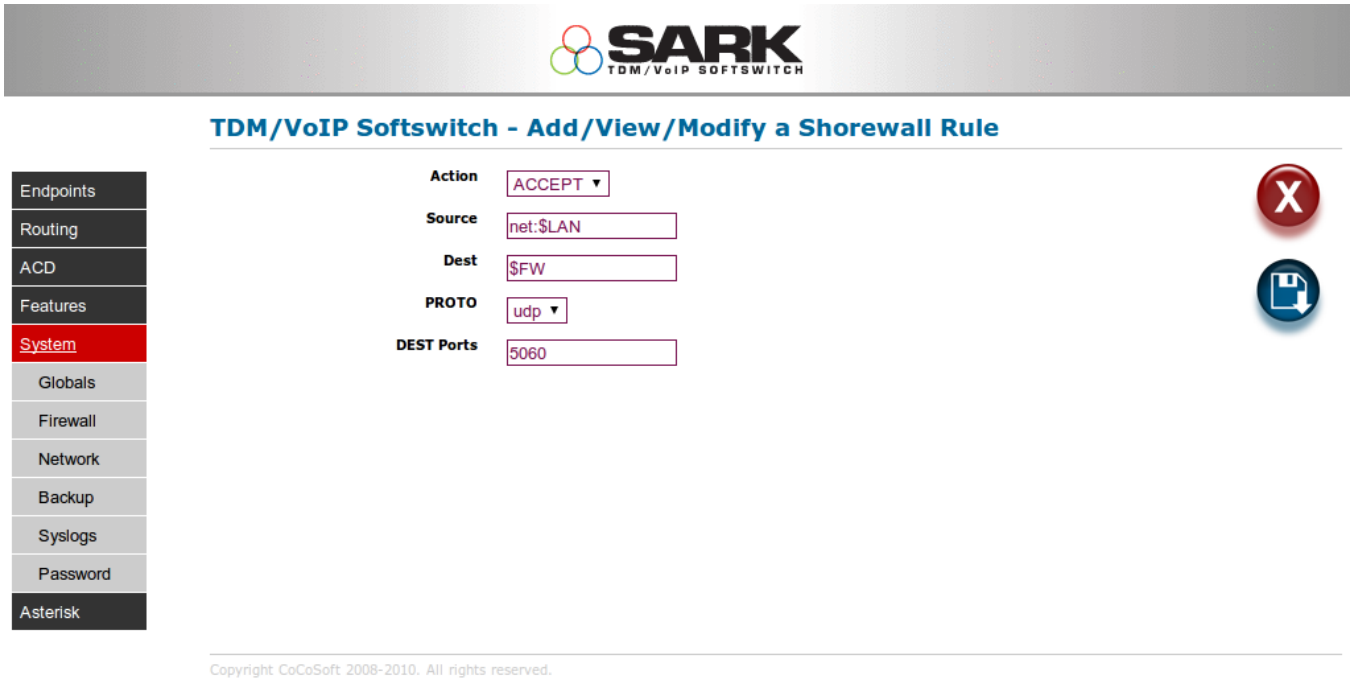


Changing firewall settings

Clicking on the edit icon for any of the rules will allow you to edit their contents. Clicking on the blue restart button will cause SARK to restart the firewall with your new settings.

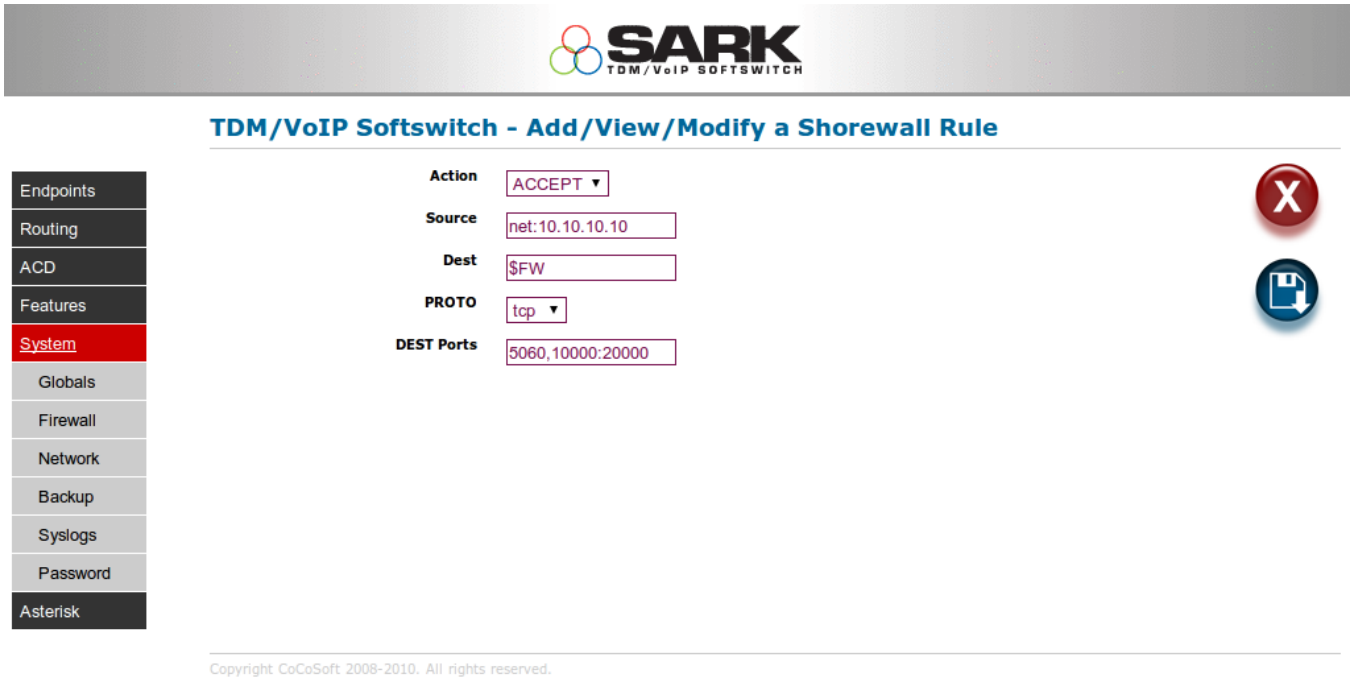
N.B. *Firewalls, by their very nature, are complex. You should familiarize yourself with Linux 2.6 iptable firewalls and the Shorewall firewall implementation before you attempt any major changes.*

As an example, let's look at the SIP settings for the firewall.



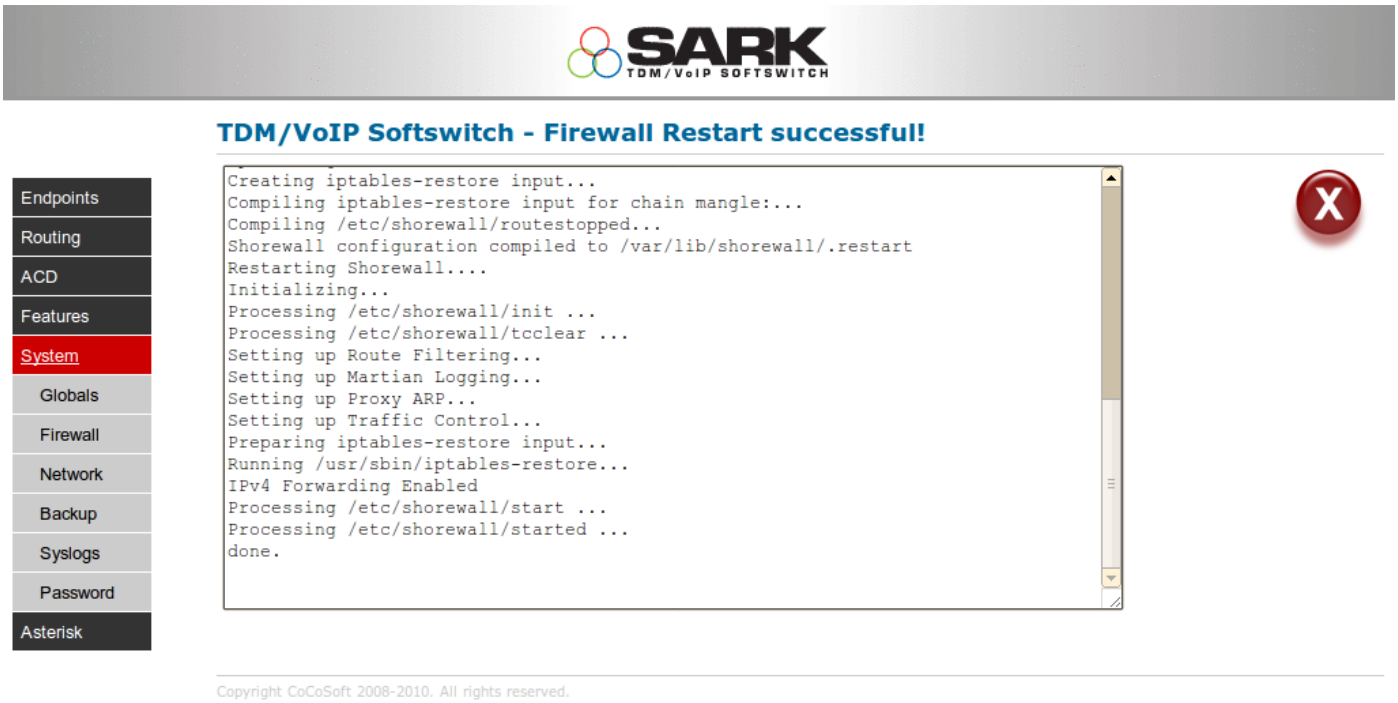
The rule consists of an action , a source, destination, protocol and port(s). This SIP rule simply says accept SIP (5060 UDP) traffic from the local LAN (defined by the symbolic parameter $LAN). SARK figures out the local LAN IP range automatically at startup.

Now, lets say we want to add a rule to accept SIP/RTP from a trusted external end-point at (for example) ip address 10.10.10.10. We can add a new ACCEPT statement like this...



This will allow a single end-point at 10.10.10.10 to send SIP/RTP packets in to the PBX (notice we specify multiple ports, separated by commas and a port range seperated by a colon. You can of course, if you wish, specify an address *range* using the standard CIDR notation (e.g. 10.10.10.0/24).

When a new rule is inserted we must manually restart the firewall by clicking the restart button (the action button to the right of the firewall panel) which will give the following report...



This indicates that all is well and you can continue.

Firewall “Local” Mode

If you make any errors in your firewall definition (if, for example, a dns name you have specified is invalid), then the firewall will not be able to start and you will see the error message “UNABLE TO RESTART THE FIREWALL”. When this happens, the firewall will revert to *LOCAL MODE.* This means that it will accept traffic from the local subnet ONLY. This should allow you to continue to access SARK via the browser and to rectify the error. Once rectified, you should restart the firewall and your rules will be loaded. **N.B.** DO NOT LEAVE THE FIREWALL IN LOCAL MODE! This is potentially unsafe and your PBX will probably not execute correctly.

Resolving Firewall lock-outs

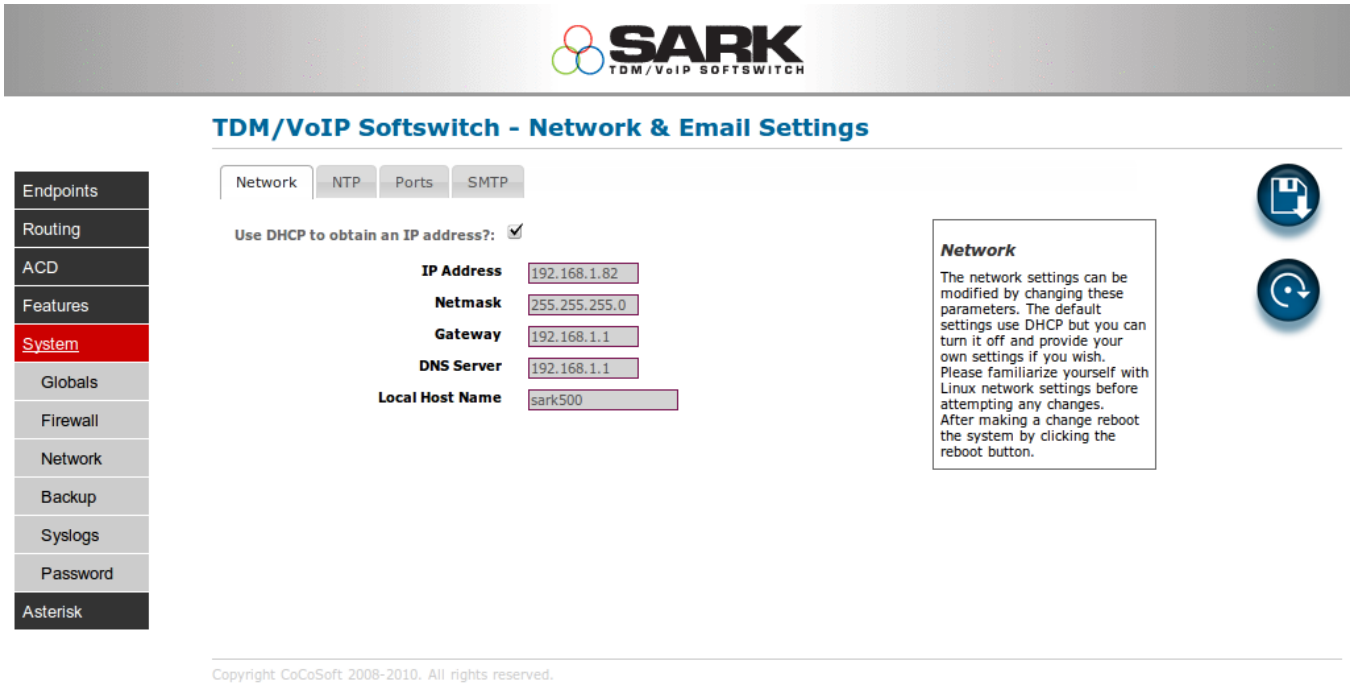
It is possible to make changes to the firewall or to the SSH and HTTP port settings such that you can “lock” yourself out from the system. You can recover from this situation by running the following procedure;

1. Connect a regular analogue (pots) phone or engineer's butt-in phone to the fxs/1 RJ11 port located to the right rear face of the unit.
2. Take the phone off-hook and dial *__36*__. You will be prompted by the auto-attendant to enter the system administrator password (pin).
3. Enter the password followed by the pound (hash) key.
4. If successful the system will set the firewall to LOCAL mode and revert the ssh and http ports to 22 and 80 respectively. The auto-attendant will respond “de-activated”.
5. You may now log in from the local sub-net using ssh and/or http to recover the firewall.

Don't forget to restart the firewall before you bring the system back into production.

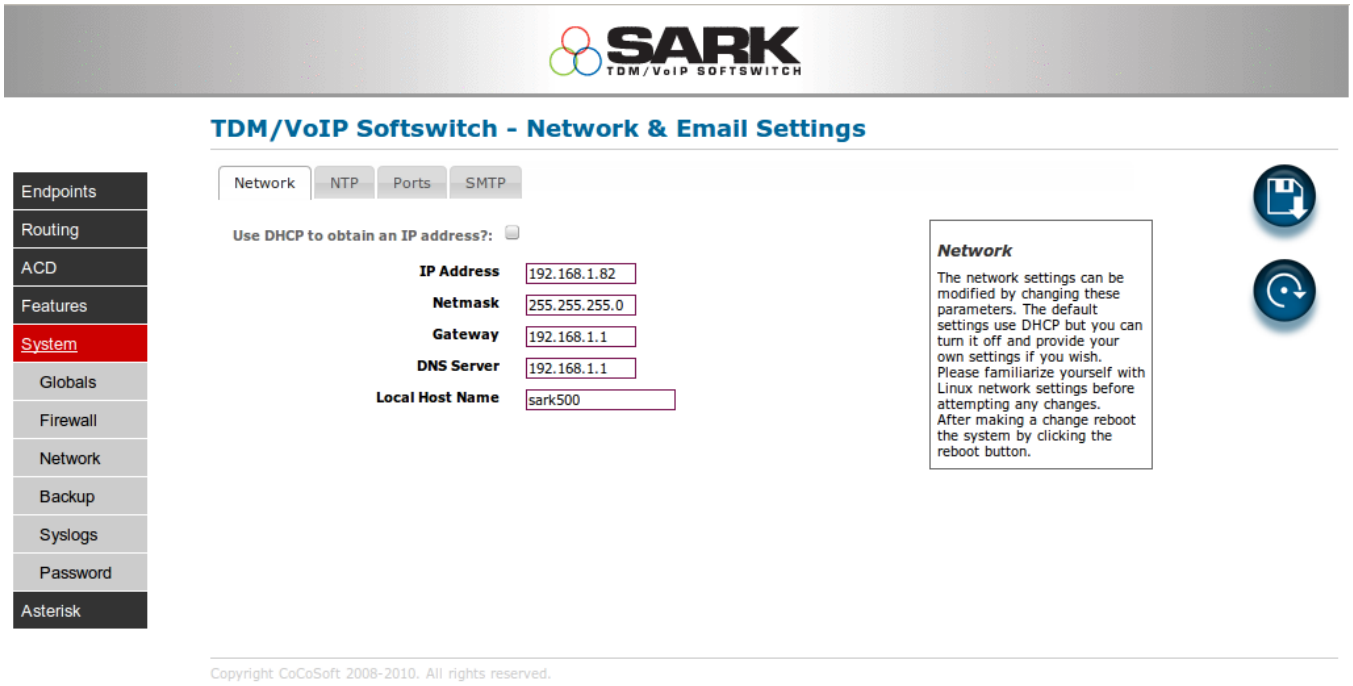
Network Configuration

To configure the appliance on your network, click the Network link in the navigation menu. The following screen appears. It allows you to set various network values.



DHCP is enabled by default on the appliance. When you connect the appliance to your network, the DHCP server on your network issues a valid IP address to the appliance during the startup sequence. If your network does not have a DHCP server, the appliance has a default IP address in the range 192.168.1.x.

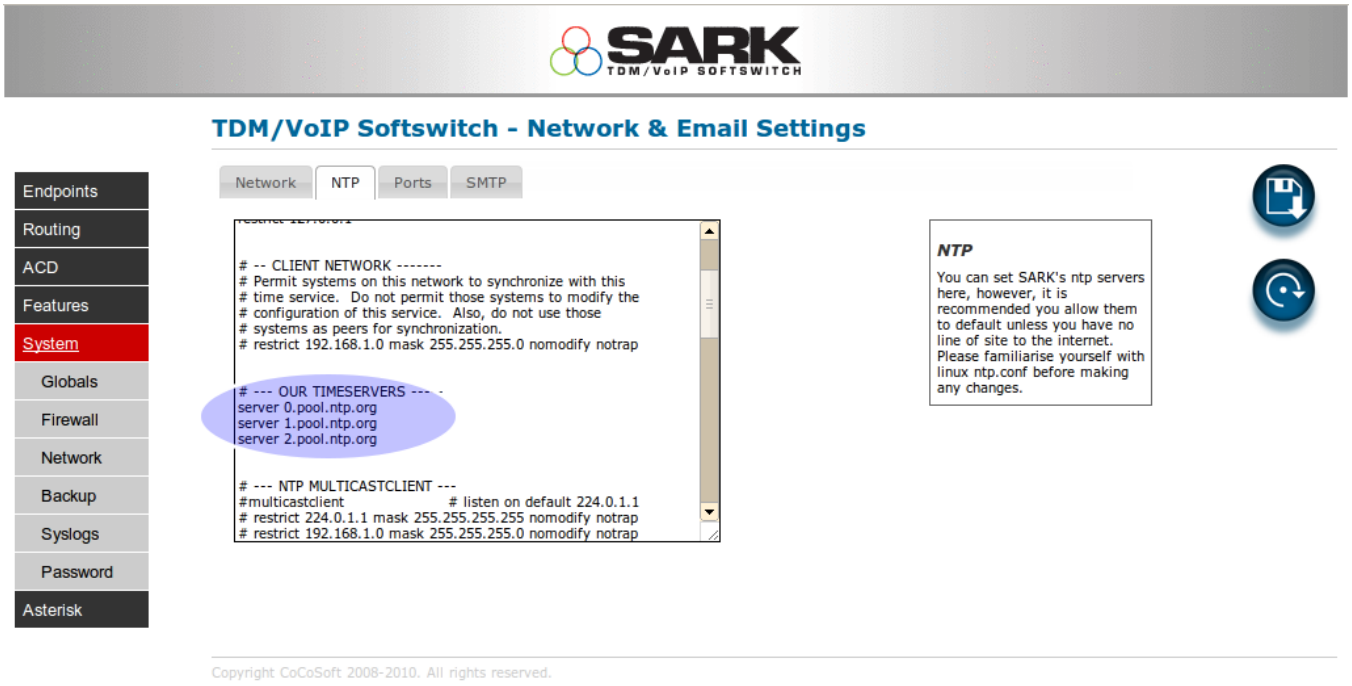
If you want to use a static IP address, you can use the network configuration page to change the network settings. Click the DHCP on/off checkbox to enable the static settings. Once done, you can reboot the unit by clicking the action button to the right of the Network settings panel.



- In the IP Address text box put the IP address you want to use
- Change the values of the other fields as required for your network.
- You must enter values for all fields if DHCP is no.
- Click the Save button.
- Click the Reboot button to restart the appliance.

Specifying alternative NTP servers

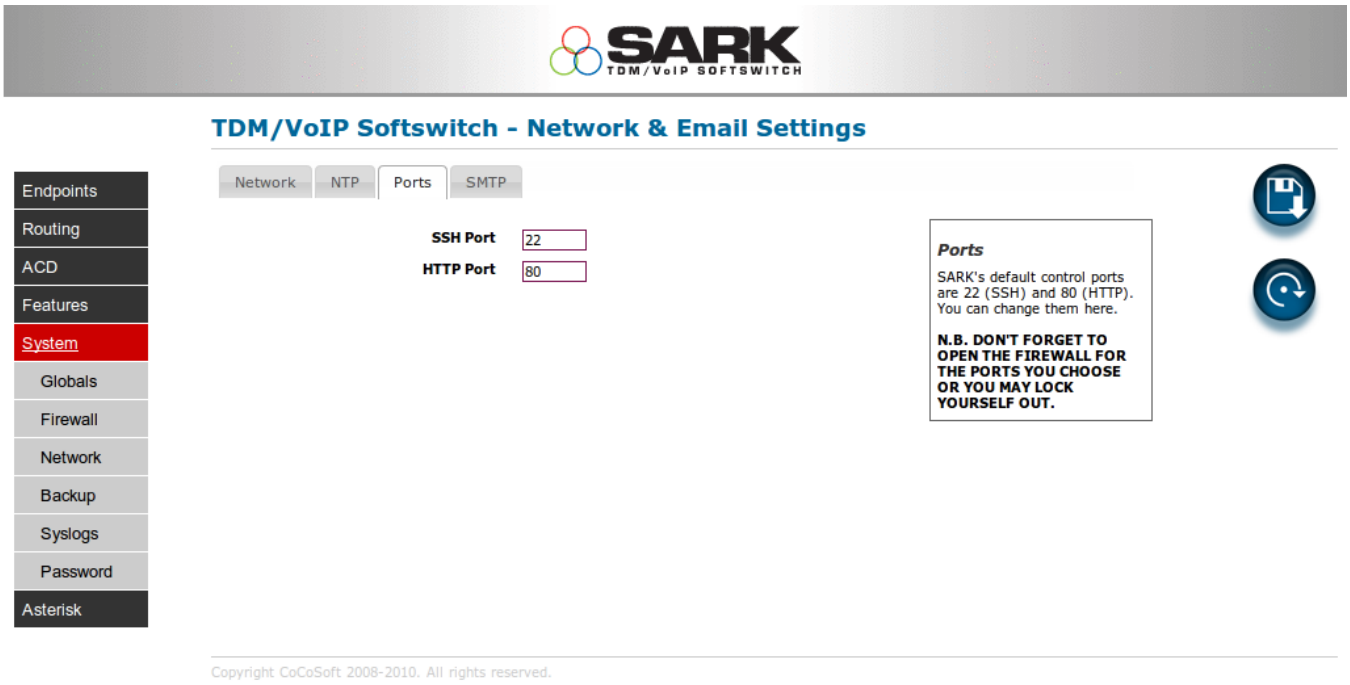
You can also specify which time servers the appliance should use. By default it will use pool.ntp.org but you can change this in the NTP section by clicking the NTP tab.



Action your changes by rebooting the server.

Changing SSH or HTTP port assignments

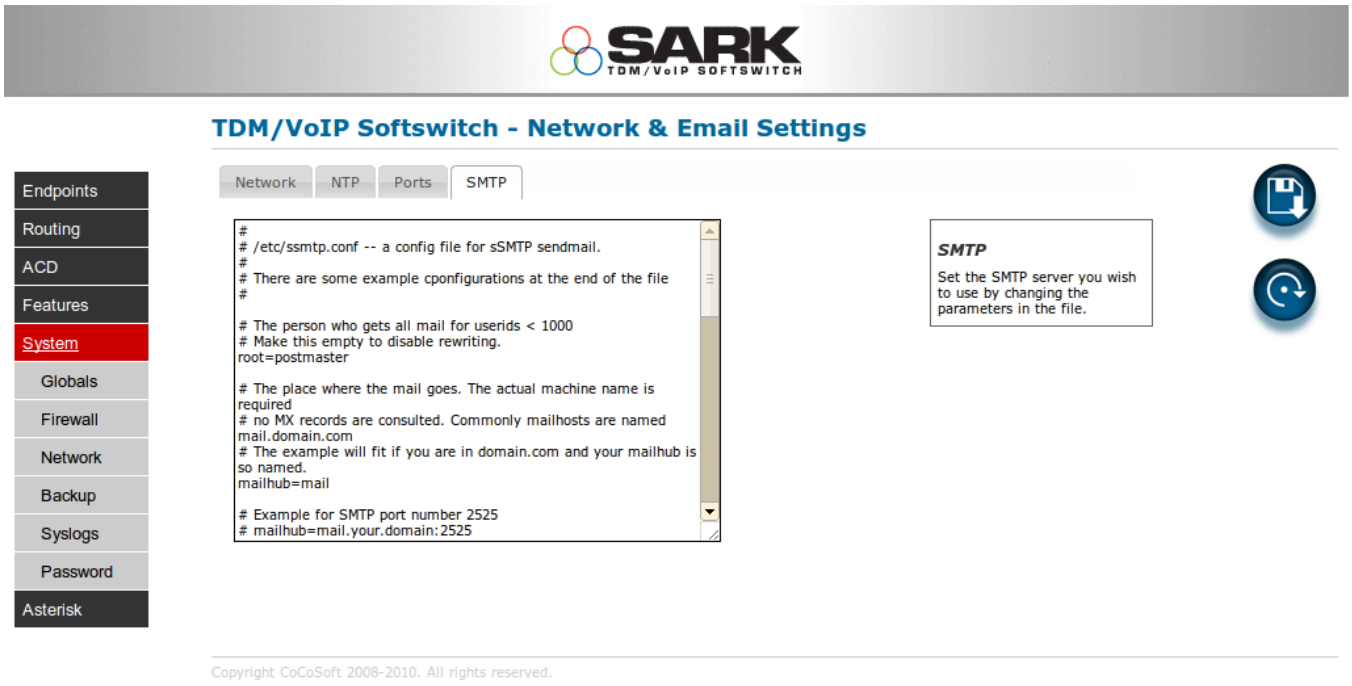
You can change the SSH and HTTP ports by clicking the ports tab.



To action your changes reboot the server.
N.B.! Make sure that you open your new port choices in the firewall *before* you reboot. Otherwise you may inadvertently lock yourself out from the appliance.

Specifying an SMTP server

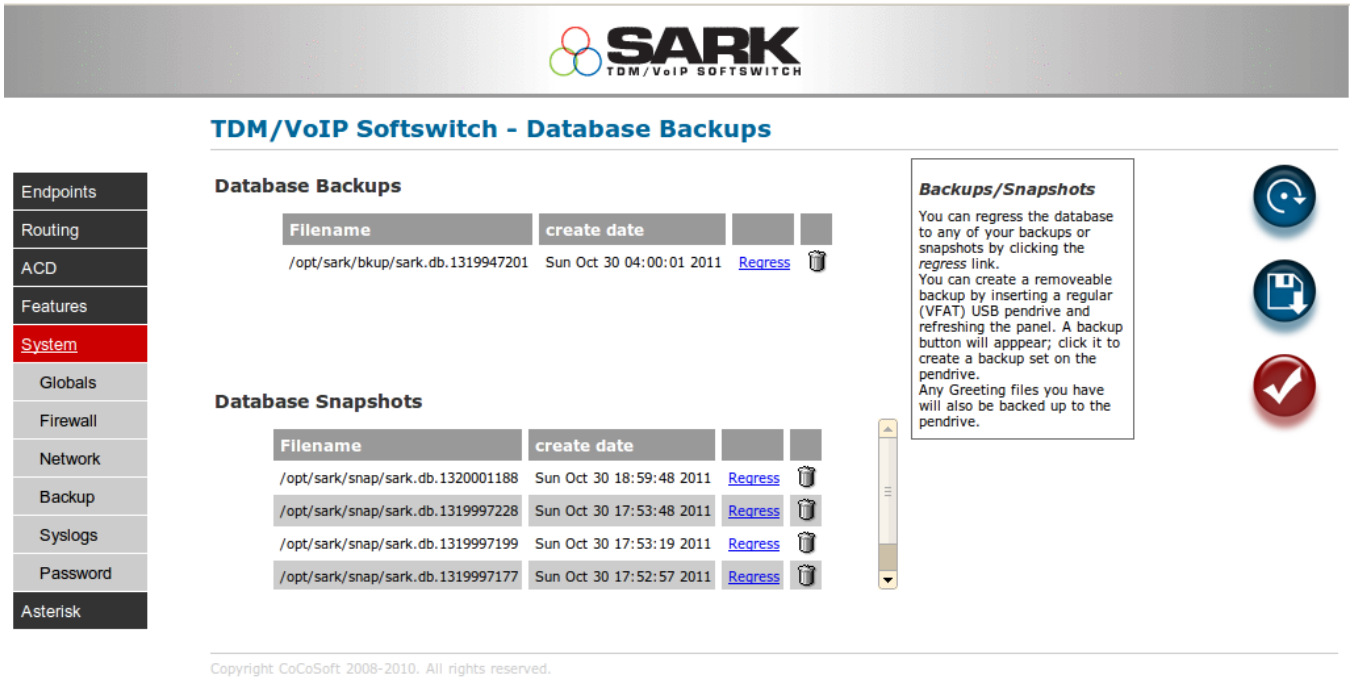
If you wish to have the PBX send voicemail as email attachments then you will need to nominate an SMTP server to deliver the mail. The SARK500 appliance uses a lightweight MUA called ssmtp. The configuration file for ssmtp is relatively simple and it can be accessed directly from the SMTP tab in the network panel. There are set-up examples in the default conf file.



Fill out your changes and save the file. It will be immediately available to ssmtp so there is no need to restart any services.

Backup/Regress & Save to USB

SARK will automatically spawn backups every night. It will keep up to 7 on-board backups and you can "regress" to a backup simply by clicking on it. It will also take an "Instant Backup" (called a snapshot) every time you issue a commit. You can take a commit (or issue an explicit backup) before you begin a piece of work or make a change and then perhaps regress to the start version if you are not satisfied with the end result or if you make a mistake during configuration. You can view your current snapshots & backups by choosing Backup/Regress from the Navigation menu.

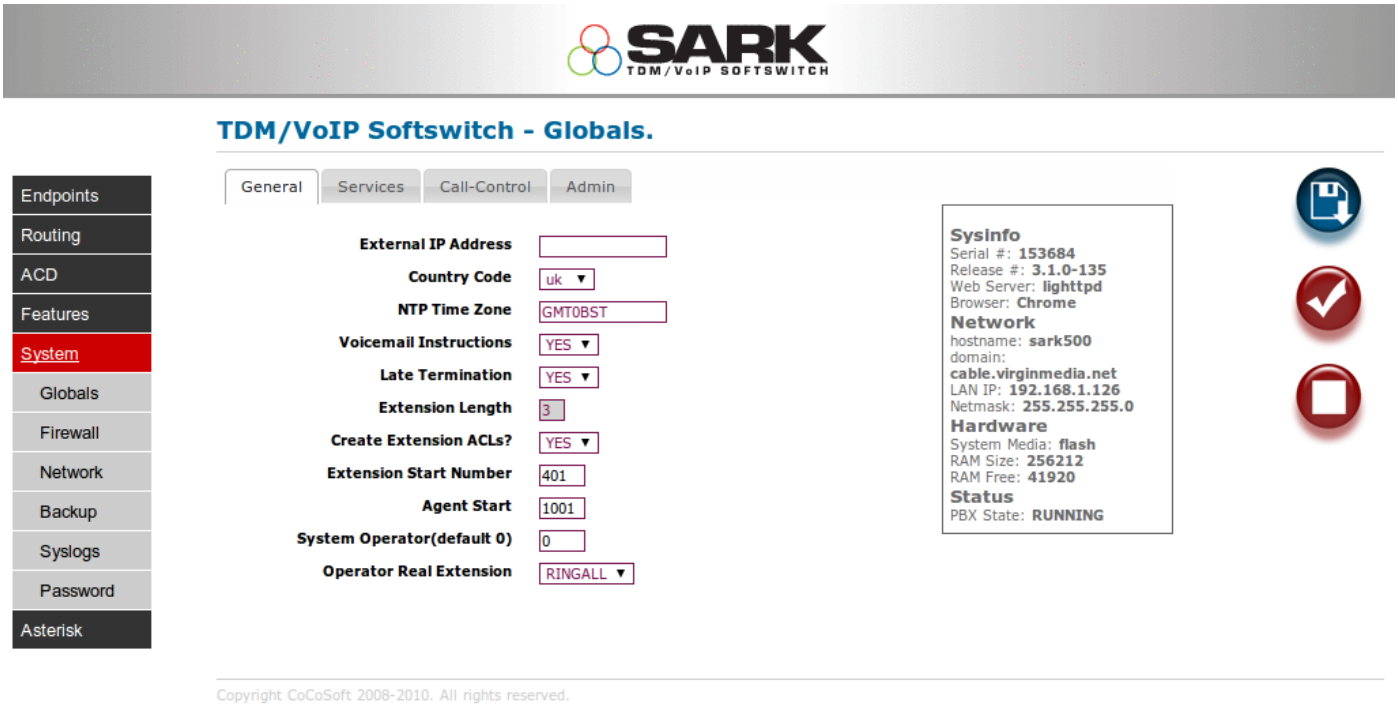


To regress to an earlier database version simply click the regress link next to the backup or snapshot you wish to use. Regression is non-destructive because SARK will automatically take a backup of the current database immediately before the regression. Thus you can freely "regress the regression" if required. Regressions and snapshots which are identical to the current working copy are highlighted in yellow.

You can save all of your backups and snapshots to an external USB drive. Insert the USB drive into the USB slot on the rear of the appliance and click Backup/Regress. An additional "save to disk" button will appear (see the image above). Clicking on the button will save all of your snaps and backups into a pair of time-stamped tar files on the USB drive. You can now remove your backup and store it in a safe place.

Global Settings

The Global Settings Web page allows you to configure global PBX settings. In the left navigation menu, click Global Settings. The following Web page appears.



You can click through the various tabs to set the system up to your requirements.

Call Recording

SARK has several different settings for call recording which you can choose to suit your requirements. However, before you begin call recording, you should be aware that in most countries and jurisdictions there are legal implications whenever you record a call and that neither Aelintra Telecom nor its distributors or resellers will be responsible in the event that you use the recording features in an unlawful manner. If you are in any doubt regarding the legality of your proposed recordings then you should seek legal advice *before* you begin.

Call recording will only be enabled if you have an SD card fitted to your SARK500. The internal flash memory is limited and quite slow to update. Recording to it can overwhelm the unit's ability to keep pace with the data throughput.

You can specify the type of call recording you want to make at the system level and you can override this setting for each individual endpoint. The system setting is in the *Globals* panel under the *General* tab.

Whatever you choose in this section will be inherited by any endpoint which has its call recording preferences set to *default.* You can also override this directive by visiting the edit panel for an endpoint and setting its preference to some value other than *default.*

Recording process

All voice capture is done directly to /var/spool/asterisk/monitor. Immediately after a recorded call ends, a task is called to off-load the recording from *monitor* to a staging dataset (MONITOROUT). The default is /var/spool/asterisk/monout and you can "over-mount" this with a NFS/NAS type device if required or use a cron updater to move the recordings over the network. Recordings are named as follows:-

{Linux Epoch}-{DNID}-{CLID}.wav

A real example calling from extension 5099 to a test number; 01924 566170, looks like this

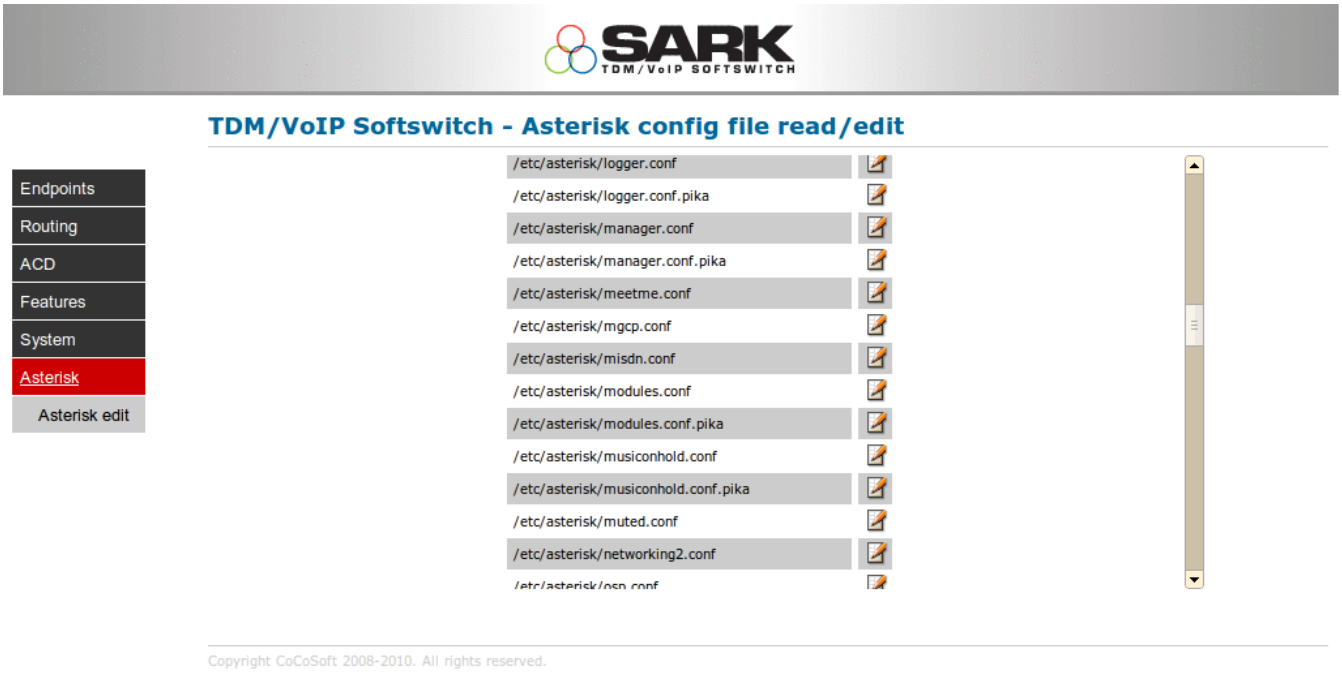
1263728106-01924566170-5099.wav

Limiting the recording space

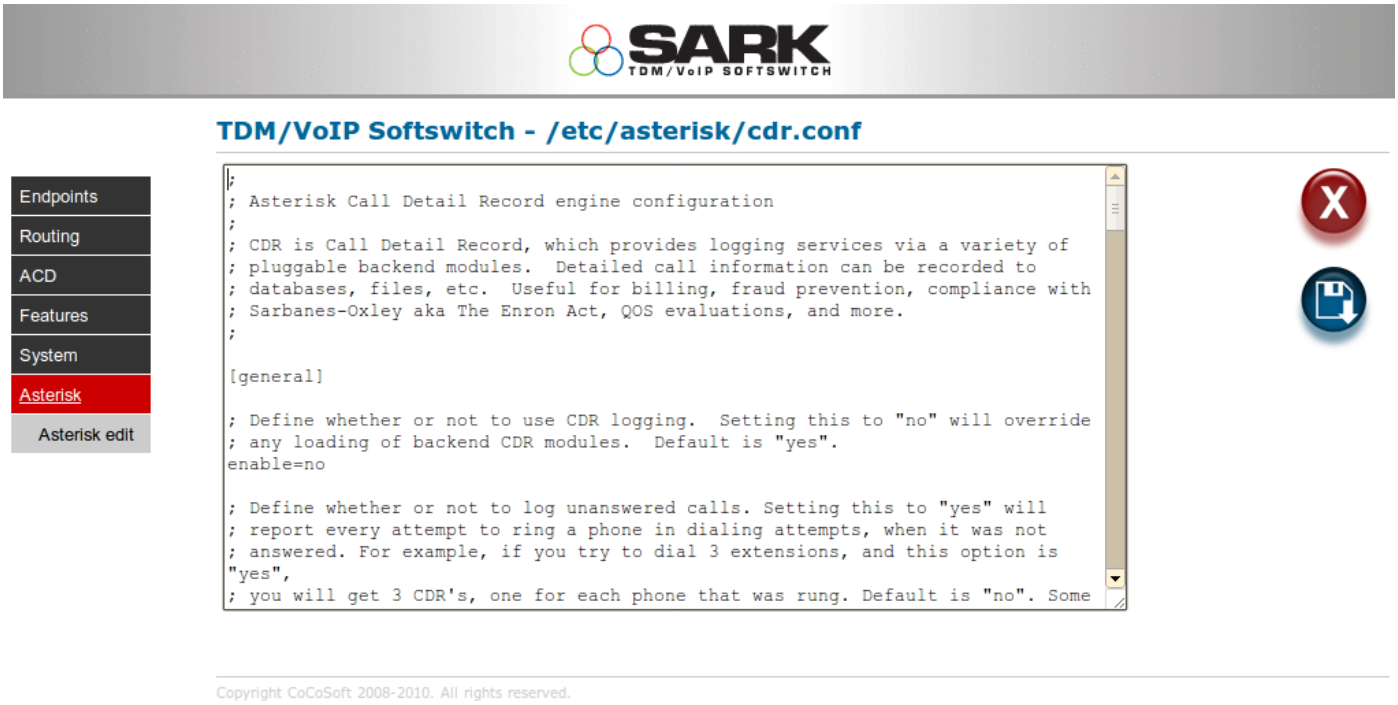
The SARK500 automatically manages the storage set aside for call recording. By default this is limited to about 16 hours of talk time. Once the storage has been used up, SARK will make space for new recordings by deleting the oldest recordings as it goes thus continually keeping the last 16 hours worth of recordings.

Asterisk File Edit

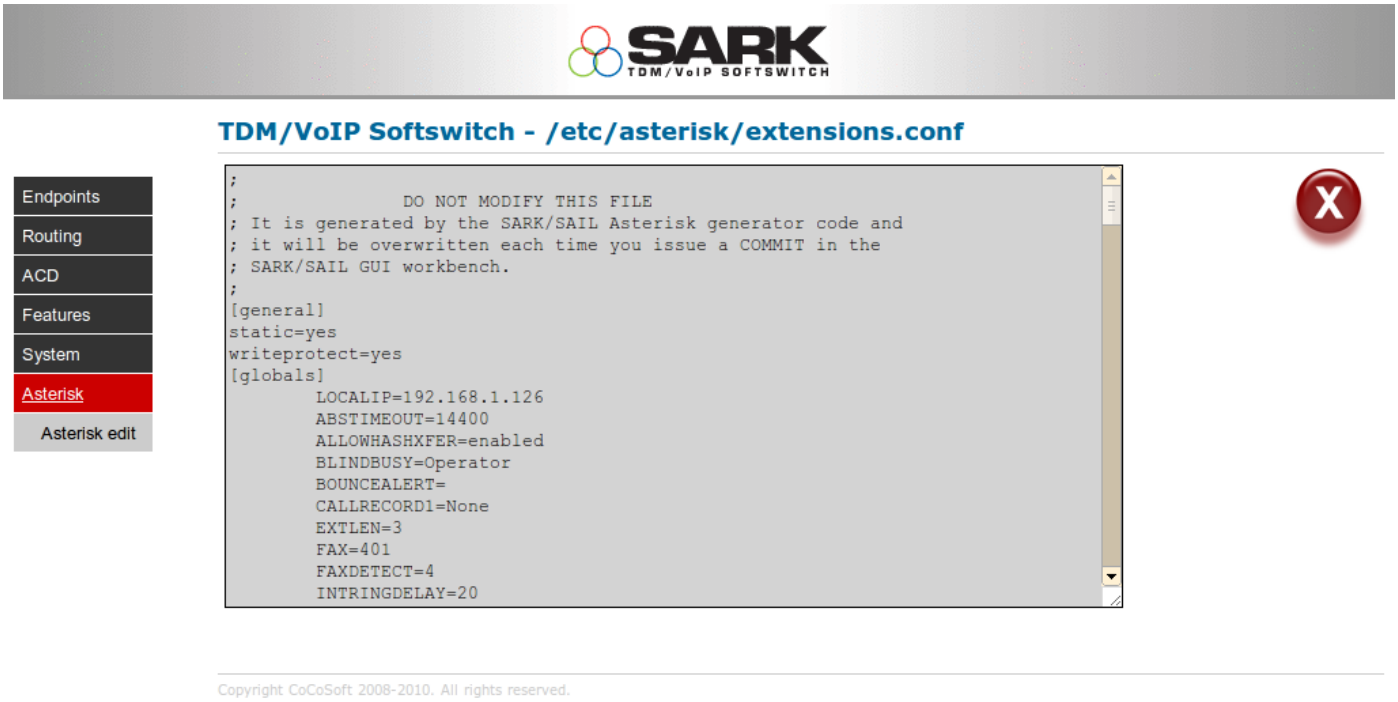
You can freely modify most of the Asterisk ".conf" files to suit your needs. There are a some files which SARK maintains and it won't allow you to modify them directly but even those files have user include files that you can use to create and maintain your own settings. To change an Asterisk file simply select it using the Asterisk file edit panel.



You can freely modify it and save it back...



If you choose a file that SARK maintains, then the content will still be viewable but it will be “grayed out” and you won't be able to save or update it. One of the files which SARK manages is called extensions.conf...



As you can see from the above, when you display a system managed file, the “Save” button is not presented and the content is set to read-only.

SARK Service Codes

TRANSFERS, PARKING & GROUP PICKUP				
Key	**data**	**function**	**Availability/Terminal Type**	**Comment**
# #	extension	Blind Transfer	internal	Immediately transfer a call - use with analogue phones - usually turned off
*1	none	One Touch Record(OTR) or One Touch Retrospective Record(OTRR)*	internal	Activates call recording This is an in-band request. Actual key sequence may vary - refer to site administrator
*2#	extension	Attended Transfer	internal	Dial callee then transfer - use with analogue phones - usually turned off
Transfer Key	extension	Transfer	All SIP phones	Either blind or attended depending upon the phone type
*##*5#*	none	Call Park	internal	Parks the call into the first free park - usually used with BLF parks
*8	none	Call Pickup	internal	Pick up ringing call on same group
*8	extension	Directed Call Pickup	internal	Pick up specific ringing call - usually used with BLF and Visual Call Pickup on Snom or Aastra
Hold	none	Hold	SIP phones which have a hold function or key	Hold call and play MOH
*Message Light/Mail Icon *50**	none	Pick up voicemail	internal	Connect to Voicemail

300 - 307	none	Conference Rooms	internal	Actual key sequence may vary depending upon extension number choice and other factors
499	none	Broadcast directly to the overhead page amps using line out	internal	Check with your administrator to see if this feature is activated – Actual key sequence may vary depending upon extension number choice
700 - 702	none	Parked call pickup	internal	Actual key range may vary depending upon extension number choice- refer to site administrator
* +*extension*	none	dial or transfer to an extension's Voicemail	internal	Useful to transfer a call direct to an extension's voicemail box.
**+ *extension*	none	dial or transfer to an extension's MiniQueue (camp-on)	internal	Used to transfer a call to a busy extensions “camp-on” queue.

*OTRR Requires Advanced recording feature to be installed.

Diverts

DIVERTS				
Key	**data**	**function**	**Availability/Terminal Type**	**Comment**
20	none	Call Forward Immediate to Voicemail Toggle (DND)	internal	sets feature on/off
21+ext	none or extension	Call Forward Immediate (CFIM) Toggle	internal	Use with no extension to turn off
22+ext	none or extension	Call Forward on Busy/No Answer (CFBS) Toggle	internal	Use with no extension to turn off
23	none	Clear All Call Forward Instructions (CLACF)	internal	clear all call forward instructions for this extension
27+ext	extension	Follow Me (RCFIM)	internal	Direct a remote extension's calls to the local station - requests your vmail passwd
12+ext	extension	Temporarily forward operator calls to this extension	master user	Useful in small office environments during operator rest/break periods

Permissions

PERMISSIONS				
Key	**data**	**function**	**Availability/Terminal Type**	**Comment**
24	none	Allow Call Forward to External Numbers	master user - requires password	
25	none	Deny Call Forward to External Numbers	master user - requires password	
30	none	Set Master Timers to Automatic (default)	master user - requires password	Incoming calls will be checked against time of day
31	none	Set Master Timers to global "CLOSED"	master user - requires password	Incoming calls will be routed to closed handler
32	none	Set Master Timers to global "OPEN"	master user - requires password	Incoming calls will be routed to open handler

Voicemail

VOICEMAIL				
Key	**data**	**function**	**Availability/Terminal Type**	**Comment**
50	password	Local Voicemail	internal	connect to voicemail from own station
51	Extension & password	Remote Voicemail	internal/external	Connect to voicemail from another station

Miscellaneous operations

Miscellaneous				
Key	**data**	**function**	**Availability/Terminal Type**	**Comment**
26+n	none or ring delay	Set ring delay (seconds)	internal	if no ring delay is specified the ring will not timeout
40+group	none, pagegroup (call group) or extension	Page	internal	Use with no extension to page all phones
52	none	Echo test	internal/external	
55	none	Time and date	internal	
56	none	Check my extension number	internal	

System Greetings

System Greetings				
Key	**data**	**function**	**Availability/Terminal Type**	**Comment**
60+greet	4-digit greeting number	Record a system greeting	internal	Record a system greeting
61+greet	4-digit greeting number	Listen to a system greeting	internal	

Agents and Queues

Agents/Queues				
Key	**data**	**function**	**Availability/Terminal Type**	**Comment**
65	Agent Number/Password	Log on as an agent	internal	
66	Agent Number/Password followed by #	Log off as an agent	internal	
67	Agent Number	ChanSpy Function	master user - requires password	Allows a supervisor to anonymously listen to agents servicing Queues
68	extension	ChanSpy Function	master user - requires password	Allows a supervisor to anonymously listen to an extension

NANP Supported Service Codes

Key	data	function	Availability/Terminal Type	Comment
*60	none	Time and date	internal	
*65	none	Check my extension number	internal	
*72	Extension	Call Forward Immediate (CFIM)	internal	
*73	none	Cancel Call Forward Immediate (CFIM)	internal	
*78	none	DND ON	internal	
*79	None	DND OFF		
*90	extension	Call Forward on Busy/No Answer (CFBS)	internal	
*91	extension	Cancel Call Forward on Busy/No Answer (CFBS)	internal	
*97	password	Local Voicemail	internal	connect to voicemail from own station
*98	extension/password	Remote Voicemail	internal/external	Connect to voicemail from another station
*99	Greeting Number	Listen to a	internal	

greeting

SARK500 hardware platform

Overview

Unlike its bigger siblings, which use Intel ‘x86’ hardware, the SARK500 runs on the PIKA technologies’ WARP platform. Surprisingly few changes were necessary to deploy SARK onto Warp. Over 90% of the code is exactly the same as it is on any other SARK platform. The C code has been recompiled for the WARP's PPC CPU and there are some other changes due to the absence of a regular System V initiator, however they are relatively minor.
The majority of the SARK code all resides in a single tree in /opt/sark. This has been placed into persistent storage using conventional symlinks. A very few SARK features have been disabled on the Warp platform due to missing platform functionality

PIKA infrastructure changes

- IPtable support added by Mark Recoskie (PIKA technologies)
- Shorewall firewall added
- Daemon startup sequence changed and a firewall starter added (S70shorewall)
- Daemontools starter added for the SARK helper
- Daemontools starter added for the Multicast listener
- Cron root added for the SARK500 timer, backup and ageing routines
- ntp startup (S32ntpd) modified to provide a local ntp server
- lighttpd recompiled with PCRE support (for mod rewrite)
- Busybox recompiled with additional commands

SARK disabled features

- Dynamic Proxy has been disabled because the Lighttpd rewrite engine doesn’t support it.
- Network scanning to discover “orphan” SIP devices is disabled

SARK new features for PIKA

- sarkpnet.pl has been created to manage ip changes in the PIKA environment
- setntp.pl creates the ntp server restrictions for the local subnet
- sarkedsw.pl (firewall) has been modified for PIKA from the EL5 project

SARK Storage layout on the Warp

- SARK program storage is almost all on */persistent/opt/sark*
- The only exception is the HPE (high performance engine) which is in /var/lib/asterisk/agi-bin.
- SARK customer data storage (the database, backups and various customer managed files) is held on */persistent2/opt/sark*. This gives a clean separation between data and logic and makes firmware upgrade simpler.

Platform Options

The SARK 500 is available in several different hardware configurations as follows: −

MODEL	DESCRIPTION	Slot1	Slot2
500	VOIP	-	-
500F4	VoIP/FXO	4xFXO	-
500F8	VoIP/FXO	4xFXO	4xFXO
500B2	VoIP/BRI	2xBRI	-
500B4	VoIP/BRI	2xBRI	2xBRI
500G2	VoIP/GSM	2xGSM	-
500G4	VoIP/GSM	2xGSM	2xGSM
500F4G2	VoIP/FXO/GSM	4xFXO	2xGSM
500B2G2	VoIP/BRI/GSM	2xBRI	2xGSM

As you will see from the above FXO and BRI channels cannot be mixed in the same frame but both BRI and FXO can be mixed with GSM.

PIKA Telephony Hardware Channel Driver

Configuration

When the appliance starts, the hardware discovery software generates the configuration files for the channel driver and underlying PIKA software, based on the type of telephony modules installed on the appliance. Three configuration files are generated:

- /etc/asterisk/pika.conf - channel driver configuration
- /etc/pika/pikagp.cfg - PIKA GrandPrix configuration
- /etc/pika/pikagp_aoh.cfg- PIKA GrandPrix configuration

In most cases, the default values are sufficient and it is not necessary to manually edit the configuration files. If you need to edit the files, you must log on to the appliance using an SSH client such as “Putty”, which is a free SSH client for Windows. The user name is *root* and the default password is *sarkroot*. Configuration files contain sections of associated key assignments, presented in an INI-file fashion. Sections are denoted in the form [section name]. Key assignments are denoted by *key=value*.

This section describes the configuration settings for the channel driver. For details about PIKA GrandPrix configuration files, refer to the PIKA GrandPrix Configuration Guide on the PIKA Support web page.

The configuration parameters are grouped in sections. The valid parameters are described below.

Global Parameters

The parameters in the [general] section of the file /etc/asterisk/pika.conf apply to all BRI, FXO and FXS channels.

usenativebridge
Specifies whether to use native bridging for calls between PIKA channels. The voice path of the channels connects directly without going through Asterisk. In most cases, the value should remain yes for the best performance and audio quality.
Default value = yes.

echocancelwhenbridged
Specifies if echo cancellation should be used on bridged calls.
Default_value = yes.

busycount
The number of busy tones to wait for before hanging up. A higher number will lower the probability of random hangups, but will increase the time required to hangup a channel. The default is 2.

Busypattern

Specifies the cadence of the busy signal. The format is tone_length,silence_length. In many countries, it is 500 ms on, 500 ms off.

The following shows the default configuration entries for this section.

```
[general]
usenativebridge=yes
echocancelwhenbridged=yes
busycount=2
busypattern=500,500
```

The parameters in the section [audio] configure the appliance audio line-in and line-out ports.

txgain
Changes the outgoing volume level on the channels. Valid values are -18 to 6.
Default_value = 0

rxgain
Changes the incoming volume level on the channels. Valid values are -18 to 6.
Default_value = 0

mute
Mutes the audio ports. Valid values are 0 (false) or 1 (true).
Default value = 0

compandmode
Audio format to use when connecting to other channels through Asterisk. Valid values are 0 (mulaw) and 1 (alaw).
Default value = 0

The following shows the default configuration entries for this section.

```
[audio]
rxgain=0
txgain=0
mute=0
compandmode=0
```

Group parameters

The following parameters are common across all channel types.

gp_group
A reference between the pika.conf file and a configuration file used internally by the channel driver. It is not used to configure any channel driver behavior and should not be changed by the user.

conf_ref
A reference between the pika.conf file and a configuration file used internally by the channel driver. It is not used to configure any channel driver behavior and should not be changed by the user.

context
The initial default context for the dial plan. Default value = from-internal for FXS lines and from-pstn for FXO
and BRI trunks

extension
The initial default extension for the dial plan (used for FXO interfaces and FXS interfaces if immediate=yes is set).
For digital interfaces, leave this keyword empty to receive DID in the ${EXTEN} variable. Default value = s.

accountcode
This field, if defined, is used to populate the accountcode field of the CDR. It can be any string up to 20 characters.
Default=fxs_grp for FXS lines, fxo_group for FXO trunks and digital_grp for BRI trunks

immediate
Applies to FXS and BRI channels. Specifies whether to go immediately to the default context, exten right after the
FXS port goes offhook. Default_value = no.

mailbox
If a mailbox is specified, a stutter dialtone will be heard when the phone is taken off-hook if the mailbox contains new messages. By default, this keyword is not set.

echocancel
Specifies whether to enable echo cancellation. Default value = yes.

echotaillength
The tail length value, in milliseconds, when echo cancellation is enabled. The range is 1 ms - 128 ms. The default value is 64 ms. This value is usually sufficient for connection between any channels on the appliance. This parameter should not be changed unless required because increasing it can adversely affect voice quality due to extra CPU use.

echosuppression
The echo suppressor is designed to eliminate any residual echo and noise that exists after echo cancellation has removed the predicted echo using the echo path model and reference signal input. For human listener applications, this keyword must remain enabled. Default value = yes.

comfortnoise
Enable comfort noise insertion. Default value = yes.

echologging
Enables per channel echo cancellation logging. This should only be used when debugging audio path issues as it can affect system performance. A message is generated at the Asterisk CLI when echo cancellation is activated as a reminder that logging is on. Log files are stored either in the local directory from which Asterisk was started or in /tmp. The log file name formats are ec_ref-mmmmmmmm-nnnm.raw and ec_sin-mmmmmmmm-nnnm.raw where mmmm is the media stream ID and nnnn is the file count.
Default value = no

usecallerid
Specifies whether to detect callerid on incoming calls for FXO ports and whether to generate callerid for FXS ports.
Default value = yes.

faxdetection
Enable fax detection.
Default value = yes.

musiconhold
The music on hold class associated with this group.
Default value = default.

language
Language for prompts purposes mostly.
Default value = default

amaflags
Sets the AMA (Automated Message Accounting) flags for billing. Valid values are: default, omit, billing and documentation.
Default value = default.

canpark
Specifies whether to allow parking of active calls (check features.conf for the parking extension, the default is 700). In order to unpark the call later, the parking context must be included within the default context (check features.conf) also; the default context name is parkedcalls.
Default value = no.

txgain/rxgain
These keywords change the volume levels on the channels. Valid values are from -300 (-30 db) to +300 (+30db).
Default_value = 0.

clippingratio
Specifies the amount of clipping that will performed to the RTP input signal to prevent overload distortion. The units are dB relative to the maximum signal level that can be represented as a mu-law or a-law value. This value should not normally be changed. – Default=-2

useagc
Specifies whether AGC (Automatic Gain Control) should be used. This feature does not work during native bridging

usenativebridge=yes
Default value = no.

agc.in.enable
Specifies whether AGC should be used for the receive side of the call.

agc.out.enable
Specifies whether AGC should be used for the transmit side of the call.

agc.out.attackRate
Attack rate for the transmit side of the call in ms.

agc.out.decayRate
Decay rate for the transmit side of the call in ms.

agc.in.attackRate
Attack rate for the receive side of the call in ms.

agc.in.decayRate
Decay rate for the receive side of the call in ms.

agc_in.targetLevel
Target power level for the receive side of the call in dBm0.

agc_in.minGain
Minimum attenuation level for the receive side of the call in 1/10 dB.

agc_in.maxGain
Maximum amplification level for the receive side of the call in 1/10 dB.

agc_in.speechLevel

Threshold in dBm0 for speech detection for the receive side of the call.

agc_out.targetLevel
Target power level for the transmit side of the call in dBm0.

agc_out.minGain
Minimum attenuation level for the transmit side of the call in 1/10 dB.

agc_out.maxGain
Maximum amplification level for the transmit side of the call in 1/10 dB.

agc_out.speechLevel
Threshold in dBm0 for speech detection for the transmit side of the call.

group
Value from 0 to 31. This group is used to indicate which group the Dial command should use. E.g.: Dial(PIKA/FXO/g0/1234). There is no default value. The group number is unique for different interface types (FXS, FXO, BRI). The following can be used to specify the search algorithm to use when selecting a free channel in the group:

g - choose the first free channel, searching from lowest to highest channel number
G - choose the first free channel, searching from highest to lowest channel number
r - use round-robin channel selection, starting with the lowest channel number
R - use round-robin channel selection, starting with the highest channel number

channels
One or more channels included in this group. Channels must always be the last configuration
statement for a group.

The following parameters are specific to FXS channels.

cancallforward
Specifies whether call forwarding is enabled, enabled by *72 and disabled by *73, unless overridden by callforwardon and/or callforwardoff. Default value = no.

callforwardon
Customizes the number to dial to enable call forwarding.
Default value = *72.

callforwardoff
Customizes the number to dial to disable call forwarding.
Default value = *73.

lcterm_fxs
If enabled, the channel driver issues an LC Term signal (Loop Current Termination) to the line. This signal is typically used to allow automated dialing equipment to go on-hook when the remote end has ended the call. Default value = yes.

callwaiting
Enables or disables call waiting.
Default value = no.

calltransfer
Enables or disables call transfer.
Default value = yes.
The following shows the default settings in section [fxs]. The settings in this section apply to all FXS channels on the appliance. This section is always present because there is always at least one FXS channel on the appliance, the built-in FXS channel. The include entry at the bottom of the section indicates which of the caller ID files Asterisk uses. Refer to section Adding Extensions (pg. 17) for information about caller ID files.

```
[fxs]
gp_group=0
conf_ref=PHONE_0
context=from-internal
extension=s
accountcode=fxs_grp
cancallforward=no
callforwardon=*72
callforwardoff=*73
lcterm_fxs=yes
callwaiting=no
calltransfer=yes
immediate=no
mailbox=
echocancel=yes
echotaillength=64
echosuppression=yes
comfortnoise=yes
echologging=no
usecallerid=yes
faxdetection=yes
musiconhold=default
language=default a
maflags=default
canpark=no
txgain=0
rxgain=0
clippingratio=-2
useagc=no
agc.in.enable=no
agc.out.enable=no
agc.in.targetLevel=-15.0
agc.out.targetLevel=-15.0
```

agc.in.minGain=-6
agc.out.minGain=-6
agc.in.maxGain=18
agc.out.maxGain=18
agc.in.attackRate=170
agc.out.attackRate=170
agc.in.decayRate=750
agc.out.decayRate=750
agc.in.speechLevel=-36
agc.out.speechLevel=-36
group=0
#include pika_fxs_5.conf

The following parameters are specific to FXO channels.

waitfordialtone
If enabled, the channel driver will wait for the dial tone before dialing. Default value = yes.

hookflashco
Specifies whether to allow hook flash to be sent to the CO. Applicable only to Centrex lines and/or lines with call waiting service. Default value = no.

flashstring
String to use to invoke hook flash on the trunk. The default is ##. Asterisk processes only the first 2 characters.

callerid
Default=asreceived

The following shows the default settings in section [fxo].

```
[fxo]
gp_group=0
conf_ref=TRUNK_0
context=from-pstn
extension=s
accountcode=fxo_grp
waitfordialtone=yes
hookflashco=no
flashstring=##
callerid=asreceived
echocancel=yes
echotaillength=64
echosuppression=yes
comfortnoise=yes
echologging=no
usecallerid=yes
faxdetection=yes
musiconhold=default
language=default
amaflags=default
canpark=no
txgain=0
rxgain=0
clippingratio=-2
useagc=no
agc.in.enable=no
agc.out.enable=no
agc.in.targetLevel=-15.0
agc.out.targetLevel=-15.0
agc.in.minGain=-6
agc.out.minGain=-6
agc.in.maxGain=18
agc.out.maxGain=18
agc.in.attackRate=170
agc.out.attackRate=170
agc.in.decayRate=750
agc.out.decayRate=750
agc.in.speechLevel=-36
agc.out.speechLevel=-36
group=0
channels=1-4
```

The following parameters are specific to BRI channels

callerid
Default=asreceived

nationalprefix
Default=0

internationalprefix
Default=00

The following shows the default settings in section [digital]. Each span has its own set of configuration parameters.

```
[digital]
gp_group=1
conf_ref=BRI_1_0
briphone=false
context=from-pstn
extension=
accountcode=digital_grp
callerid=asreceived
nationalprefix=0
internationalprefix=00
immediate=no
echocancel=yes
echotaillength=64
echosuppression=yes
comfortnoise=yes
echologging=no
usecallerid=yes
faxdetection=yes
musiconhold=default
language=default
amaflags=default
canpark=no
txgain=0
rxgain=0
clippingratio=-2
useagc=no
agc.in.enable=no
agc.out.enable=no
agc.in.targetLevel=-15.0
agc.out.targetLevel=-15.0
agc.in.minGain=-6
agc.out.minGain=-6
agc.in.maxGain=18
agc.out.maxGain=18
agc.in.attackRate=170
agc.out.attackRate=170
agc.in.decayRate=750
agc.out.decayRate=750
agc.in.speechLevel=-36
agc.out.speechLevel=-36
group=1
channels=1-2
conf_ref=BRI_1_1
briphone=false
context=from-pstn
```

extension=
accountcode=digital_grp

callerid=asreceived

nationalprefix=0
internationalprefix=00
immediate=no
echocancel=yes
echotaillength=64
echosuppression=yes
comfortnoise=yes
echologging=no
usecallerid=yes
faxdetection=yes
musiconhold=default
language=default
amaflags=default
canpark=no
txgain=0
rxgain=0
clippingratio=-2
useagc=no
agc.in.enable=no
agc.out.enable=no
agc.in.targetLevel=-15.0
agc.out.targetLevel=-15.0
agc.in.minGain=-6
agc.out.minGain=-6
agc.in.maxGain=18
agc.out.maxGain=18
agc.in.attackRate=170
agc.out.attackRate=170
agc.in.decayRate=750
agc.out.decayRate=750
agc.in.speechLevel=-36
agc.out.speechLevel=-36
group=2
channels=3-4

Examples

The following shows configuration examples for the appliance.

Configuring the Appliance for BRI

If there are one or more BRI modules present, the file contains configuration entries for the BRI interfaces. Depending on your network configuration, you may need to change some settings. By default, each BRI span is configured as node type client (TE) and networking configuration point-to-multi-point (PMP). PMP was chosen as the default for ease of compatibility. For example, the terminal endpoint identifier (TEI) does not need to be pre-configured in a PMP configuration.

In the configuration file /etc/pika/pikagp_aoh.cfg, there is a section entry for a BRI board. Regardless of the number of BRI modules present (one or two), there is a single entry plus associated keys for the board. There is also a group entry plus associated keys for each span. The file contains two or four span sections, depending on the number of modules present.

The following example shows BRI board group [DIGITAL_1]. The only key you may need to change is clockmode. In most cases, you should use the default value, slave. This means that timing for the board is derived from the network. The setting master applies only in loop back mode, which is typically used for testing.

```
[DIGITAL_1]
id=1
;serial=25800717
interfacetype=bri_eu
clockmode=slave
```

Examples of BRI span groups are shown further below. The important keys for BRI span groups are:

networkconfig
specifies whether the appliance is in a point-to-point or point-to-multi-point configuration
values:
pp - indicates point-to-point
pmp - default value, indicates point-to-multi-point

node
specifies whether this span on the appliance is functioning as a TE or NT device
values:
client - default value, indicates a TE device
network - indicates an NT device

Overlap Sending and Receiving

The appliance supports both overlap receiving and overlap sending.

Overlap receiving means that the Setup message for incoming calls does not contain the complete set of digits required to route the call. Subsequent Info messages contain the remaining digits. Overlap receiving is enabled for the appliance and may not be disabled. The appliance transparently handles both the enbloc or overlap methods for receiving digits.

Overlap sending means that the Setup message for an outgoing call does not contain all the digits necessary to route the call. Info messages contain digits and the final Info message contains a "sending complete" flag to indicate that all the digits have been provided. Overlap sending is disabled by default. To enable it, set the values as required for the
following keys on the appropriate span in /etc/pika/pikagp_aoh.cfg.

addsendingcomplete
specifies whether to add a sending complete information element to all outgoing setup messages
values:
yes - default value, all digits will be sent in the Setup message, indicates enbloc digit collection
no - digits may be sent in Info messages after the Setup message, enables overlap sending

maxdigitsconnrq
indicates the maximum number of called digits that may be placed in the outgoing Setup message.
values:
0 - default value, all digits will be sent in the Setup message, used for enbloc digit collection
a numeric value between 1 and 32 - Specifies the number of digits to send in the Setup message, subsequent digits will be sent in an Info message, used with overlap sending enabled

Depending on the Asterisk dial plan used, you may need to change the value of the following parameter in the /etc/asterisk/pika.conf.

immediate
indicates whether to start the dial plan before digit collection is complete
values:
yes - start the dial plan and send the digits directly to Asterisk as DTMF digits
no - the PIKA channel driver will attempt to match the digits with a existing extension before starting the dial plan

The following example shows the settings for the first span on a board. It is configured as a TE device in a point-to-point configuration. Overlap sending is disabled.

```
[BRI_1_0]
interface=board1,0
networkconfig=pp
switchtype=etsi
countrycode=Europe
line_compandmode=alaw
node=client
numbertype=unknown
umberplan=unknown
channelselect=exclusive
endpoint=true
autoreconnect=false
addsotoend=false
resetchidextensionbit=false
enablet309=false
permanentactivation=false
dropteiondeact=false
;default_tei=0
;num_digits=0
;local_number=
;servicetype=voice
;in_calls_behavior=TRANSPARENT_OVERLAP_RCV
;out_calls_behavior=NONE
;general_calls_behavior=CC_UUI_RECOGNITION
;addsendingcomplete=yes
;maxdigitsconnrq=0
;taillength=64
```

The following example shows the settings for the second span on a board. It is configured as an NT in a point-to-multi-point configuration. Overlap sending is enabled.

```
[BRI_1_1]
interface=board1,1
networkconfig=pmp
switchtype=etsi
countrycode=europe
line_compandmode=alaw
node=network
numbertype=unknown
numberplan=unknown
channelselect=exclusive
endpoint=true
autoreconnect=false
addsotoend=false
resetchidextensionbit=false
enablet309=false
permanentactivation=false
dropteiondeact=false
;default_tei=0
;num_digits=0
;local_number=
;servicetype=voice
;in_calls_behavior=TRANSPARENT_OVERLAP_RCV
;out_calls_behavior=NONE
;general_calls_behavior=CC_UUI_RECOGNITION
addsendingcomplete=no
maxdigitsconnrq=1
;taillength=64
```

You must restart Asterisk for the configuration changes to take effect. Click STOP and START on the SARK Globals panel to restart asterisk.

Configuring Asterisk on the Appliance for use Outside North America

The Asterisk channel driver on the appliance uses a North America configuration unless there is a BRI module installed, in which case it uses a European configuration. You can use the application pikacf to change the configuration. After you finish using pikacf, you must restart the appliance.

To configure the appliance for other countries, perform the following steps:

- Click STOP in SARK Gobals panel.
- Run the PIKA configuration program to generate new configuration files. At the Linux command prompt on the appliance, type pikacf.

NOTE: BRI is not supported for North America, therefore, even if you select the country code US, settings chosen for BRI reflect a European configuration.
Enter two letter country code, for example 'US' for United States. (Enter '?' to display the list of supported country codes or 'c' to customize individual parameters) Country code: [US]
Choose the appropriate companding mode and caller id mode values for your country.

```
Compand mode, ulaw or alaw: (Enter '?' for help) [ulaw] alaw

Select one of the following caller id modes
FSK BELLCORE
2. FSK ETSI
3. FSK BT
4. FSK NTT
5. DTMF ETSI
6. DTMF DK
7. DTMF SE FI
8. None
Selection: (Enter '?' for help) [8]
```

After the configuration program completes, the configuration files contain the entries for any FXO, FXS and BRI ports on the appliance.

NOTE: Any country for which there is a predefined configuration will have the correct line impedance value set. Configuration for other countries will use the North American value. There is no configuration file entry to change the line impedance.

Change the indication tones for Asterisk in the file /etc/asterisk/indications.conf
In the section [general], if your country is in the list of the default countries, modify the line "country=" as appropriate.

Restart the appliance:
At the Linux command prompt on the appliance, type reboot.

Call Progress

Call progress uses inband tone patterns (such as ringing, busy or fast busy) received from remote devices to determine why a call has been disconnected. Enabling call progress ensures that billing does not start until a two-way speech connection has been established. Call progress is enabled by default for the analog ports on the appliance. Only North American call progress tones have been defined and therefore, call progress may not work outside of North America. Two options are available for this situation.

1. Disable call progress (recommended)
   - In the file /etc/pika/pikagp.cfg, change all entries in the file that match callpa=callpa_settings to ;callpa=callpa_settings.
   - In the file /etc/pika/pikagp_aoh.cfg, change all entries in the file matching ;answer=speech or answer=speech
     to answer=none.

2. Modify the tone definitions in /etc/pika/inccpa.cfg to correspond to the values for your country. The comments in the file and the following websites provide information about defining the tones.
- www.3amsystems.com/wireline/tone-search.htm
- www.itu.int/dms_pub/itu-t/oth/02/06/T02060000040002PDFE.pdf

Using the PIKA CLI

The following lists the Asterisk CLI commands for the PIKA channel driver.

pika debug {on| off}
Enables or disables debug messages

pika rec {on | off} {fromasterisk | toasterisk} gpgroup channel [nocli]
Enables or disables audio recording for debugging purposes, either from Asterisk to the card (fromasterisk) or to Asterisk from the card (toasterisk) where gpgroup is the GrandPrix group number and channel is the channel or channel range to record. The option nocli disables CLI messages. We recommend that you do not use nocli.

pika reload
Reloads the PIKA channel driver to activate configuration changes.

pika show audio
Shows debugging information for the line-in and line-out ports. Used for internal PIKA debugging only.

pika show channels [gpgroup] [channel]
Shows the status of the PIKA channels where gpgroup is the GrandPrix group number and channel is one or more channels in gpgroup.

pika show info
Shows the PIKA board and driver information

pika show version
Shows the PIKA channel driver version information

Channel Status on the appliance LCD

The LCD displays the status for each channel when Asterisk runs. The display reflects only PIKA channels, not other Asterisk channels types, such as SIP channels. During the startup sequence, the LCD shows the message WARP appliance...loading. After approximately two minutes, the startup sequence finishes. When Asterisk runs on the system, the LCD on the appliance shows system information and the line status of the channels. The LCD shows the IP address and when Asterisk starts, changes to show the channel states. A single press of the LCD toggle button (beside the LED and highlighted below) switches the display to show the IP address. Press the button again to return to the channel state display. The icons on the display reflect different channel states.

FXS Channels

Icon	Description
	on-hook or disconnected
	channel in use
	channel disconnected

BRI Channels and Spans

Note that the icons for BRI represent spans, not channels.

Icon	Description
	one or both channels on the span are in use
	span disconnected, both channels are unavailable

GSM Channels

	channel in use
	channel disconnected, the radio is not registered with the network

Examples

The following example shows the display where channel one (FXS) has a call in progress. A channel on BRI span two and a channel on BRI span four have calls in progress. BRI spans three and five are active but currently idle. BRI spans six through nine are not connected.

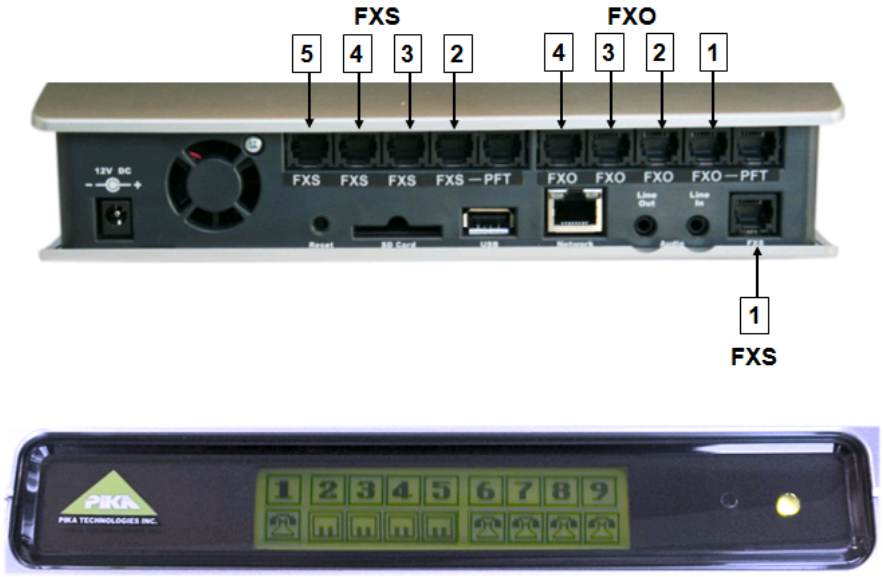
The following example shows the display where channel one (FXS), three (FXS) and eight (FXO) have calls in progress. Channel nine (FXO) is active but is currently idle. Channels six and seven (FXO) are not connected.

The following example shows the display where channel one (FXS) is idle, channel two (GSM) has a call in progress, and channel three (GSM) is disconnected (no SIM card plugged in).

Port Numbering - Provisioning versus LCD

On the LCD, channels are numbered sequentially, one through nine, regardless of the module type inserted into the module bay. When phones and trunks are provisioned in Asterisk configuration files, channels are numbered sequentially based on the module type. FXS channels are always numbered one through five, where channel one is always the built-in FXS port. FXO channels are always numbered one through four. BRI channels are numbered one through eight in the dial plan (note that the LCD shows spans, not channels).

The following diagram shows an example of this mapping. The FXO module is installed in the first bay, which is closest to the built-in FXS channel. The FXS module is in bay two. When you configure Asterisk (e.g. fxs/2 in the dial plan), use the numbers as shown on the back view below. The bottom view of the LCD shows the same module installation and the corresponding channel numbering.



If the appliance is wall mounted or installed in such a way that the display is upside down, press the toggle button on the LCD face plate three times in succession to reverse the display.

BRI Link State Display

Due to limitations of the BRI hardware and the link protocol, the LCD may not accurately represent the channel state under certain conditions. This section attempts to summarize the expected behavior. A span or channel is considered connected when both ends of the link are connected within the telephony network and exchanging low-level signaling information. The LCD reflects the state of the channel in Asterisk.

Appliance Startup

If a span is connected when Asterisk starts, the LCD shows that the channels are connected and idle.
If a span is not connected, the LCD shows the empty icon box, which indicates that the channels are not connected. When the span is connected, Asterisk updates the display to show that the channels are idle.

Call Processing

When there is a call in progress, the display shows that the channel is in use.
If the span is disconnected (the cable is unplugged either locally or at the far-end) during a call, Asterisk updates the LCD immediately to show the empty icon box. When the channel is reconnected, Asterisk updates the LCD immediately to reflect the change to the idle state.
If the LCD shows that a channel is connected and Asterisk attempts a call on that channel, if the channel is actually disconnected, the LCD will first show the channel in use, and then after a few seconds Asterisk updates the display to show that the channel is disconnected.

Power Saving

The network side (NT) of a BRI span initiates power saving mode at a pre-determined time (usually ninety seconds) following the completion of the last call on the span. Low-level signaling information between the NT and TE stops and the channels on the span are deactivated. Note that the appliance cannot initiate power saving, even in NT mode. If Asterisk attempts a call on a deactivated channel, the channel automatically exits power saving mode, the call progresses normally, and the LCD accurately reflects the line state.

If a channel is considered connected when power saving mode is initiated, the LCD indicates that the channel is connected even if it is subsequently disconnected (the cable is unplugged either locally or at the far-end). If Asterisk then attempts a call on a disconnected channel, the call will fail and Asterisk update the LCD to indicate that the channel is disconnected.

Deployment and dimensions

The appliance can be wall or bench mounted. The dimensions are 34mmx172mmx234mm(HxLxW).

The external PSU delivers 12V DC to the Appliance. You should NOT use any PSU other than the one supplied with the appliance or by your SARK500 reseller or distributor.

FXO/FXS ports are standard RJ11 while Digital ports are RJ45.

The external audio in/out ports are terminated to minijack (3.5mm). There is also a USB socket for engineering use and a semi-permanent CFCard mounting slot. If a CFCard is fitted to your system then you should NOT attempt to remove it; it is necessary for correct system function.

SARK Directory Structure on the WARP

Most of SARK's files are resident in */opt*. SARK customer files (that we will not directly change during a system update) are resident in persisten2/opt. For PIKA-ppc /opt is 'symlinked' to /persistent and /opt customer files are symlinked again into persistent2. SARK itself is not very large; the entire package in uncompressed form is only 9.8Mb. Below we list all of the SARK datasets (and dependencies) as they appear when installed onto the WARP platform.

```
persistent/opt
`-- sark
    |-- amacs                                  Adds/Moves/Changes
    |-- bkup → /persistent2/opt/sark/bkup      Daily backups
    |-- db → /persistent2/opt/sark/db          Database
    |-- generator                              code generator
    |-- passwd → /persistent2/opt/sark/passwd
    |-- perl                                   Perl Support Code
    |   `-- modules
    |       `-- sark
    |-- scripts                                script lib
    |-- snap → /persistent2/opt/sark/snap      snapshots
    `-- www                                    http doc root
        |-- cgi-bin                            http code
        `-- sark-common                        Common Elements (CSS,JS images etc)


persistent/autorun/                            Modified autorun sequence
|-- S01udev
|-- S20sarksdtransfer                          SD Card Manager (new)
|-- S30networking
|-- S32ntpd
|-- S39hmp
|-- S42gsm
|-- S42lcdlib
|-- S60crond                                   Moved
|-- S65sarkupdates                             Database updater(new)
|-- S70shorewall                               Shorewall starter (new)
|-- S89lighttpd
|-- S93zaptel
|-- S94chan_pika
|-- S95tftpd                                   Moved
|-- S96astmanproxy
|-- S98asterisk                                Moved
persistent/etc/
```

|-- asterisk **Asterisk SARK includes**
| |-- sark_agents_header.conf
| |-- sark_agents_main.conf
| |-- sark_cdr_mysql.conf
| |-- sark_customer_agents_main.conf → persistent2/etc/asterisk
| |-- sark_customer_iax_devices.conf → persistent2/etc/asterisk
| |-- sark_customer_iax_headers.conf → persistent2/etc/asterisk
| |-- sark_customer_iax_main.conf → persistent2/etc/asterisk
| |-- sark_customer_iax_registrations.conf → persistent2/etc/asterisk
| |-- sark_customer_manager.conf → persistent2/etc/asterisk
| |-- sark_customer_queues.conf → persistent2/etc/asterisk
| |-- sark_customer_queues_main.conf → persistent2/etc/asterisk
| |-- sark_customer_sip_devices.conf → persistent2/etc/asterisk
| |-- sark_customer_sip_headers.conf → persistent2/etc/asterisk
| |-- sark_customer_sip_main.conf → persistent2/etc/asterisk
| |-- sark_customer_sip_registrations.conf → persistent2/etc/asterisk
| |-- sark_customer_vmail_header.conf → persistent2/etc/asterisk
| |-- sark_customer_vmail_main.conf → persistent2/etc/asterisk
| |-- sark_features_applicationmap.conf
| |-- sark_features_featuremap.conf
| |-- sark_features_general.conf
| |-- sark_iax_header.conf
| |-- sark_iax_localnet_header.conf
| |-- sark_iax_main.conf
| |-- sark_iax_registrations.conf
| |-- sark_manager.conf
| |-- sark_queues_header.conf
| |-- sark_queues_main.conf
| |-- sark_sip_header.conf
| |-- sark_sip_localnet_header.conf
| |-- sark_sip_main.conf
| |-- sark_sip_registrations.conf
| |-- sark_vmail_header.conf
| |-- sark_vmail_layout.conf
| |-- sark_vmail_main.conf
| |-- sarkdir.conf

persistent/etc/lighttpd.conf **http configuration**

|-- shorewall **Shorewall SARK includes**
| |-- sark_rules.cfg → persistent2/opt/sark/Shorewall/sark_rules.cfg

persistent/var/lib/asterisk/agi-bin/
|-- sarkhpe **HPE**

service/
`-- srkhelpd **SARK helper Daemon**

/var/spool/cron/crontabs/root **SARK cron jobs**

persistent2 **Customer data**

`--etc
|-- HOSTNAME
|-- hosts
|-- http_port
|-- networking.conf
|-- passwd
|-- resolv.conf
|-- ssh_port
|-- ssmtp

`-- asterisk **Asterisk #include Files**
|-- sark_customer_agents_main.conf
|-- sark_customer_iax_devices.conf
|-- sark_customer_iax_headers.conf
|-- sark_customer_iax_main.conf
|-- sark_customer_iax_registrations.conf
|-- sark_customer_manager.conf
|-- sark_customer_queues.conf
|-- sark_customer_queues_main.conf
|-- sark_customer_sip_devices.conf
|-- sark_customer_sip_headers.conf
|-- sark_customer_sip_main.conf
|-- sark_customer_sip_registrations.conf
|-- sark_customer_vmail_header.conf
`-- sark_customer_vmail_main.conf
`-- opt
`-- sark
|-- bkup **Daily backups**
|-- db **Database files**
| |-- DBSQL
| |-- emptyTheDB
| |-- lineIOdump
| |-- sark.db
| |-- sarkstart.db
| `-- startDBSQL
|-- passwd
|-- shorewall **Shorewall Customer files**
| |-- sark_rules.cfg
`-- snap **Rolling Snapshots**

SARK Additional Perl Dependencies

SARK requires additional Perl modules over and above those installed as part of the Warp platform. The modules are taken from CPAN.

AnyEvent
AsteriskAMI
DBD-SQLite
parent
IO-Interface
IO-Socket-Multicast

SARK Additional BusyBox commands

SARK requires additional BusyBox commands over and above those installed as part of the Warp platform. The commands are added by regenerating BusyBox.

arp
arping
ip
install
printf