

Flash Operator Panel 2

Anyone that has been involved with Asterisk based systems, will generally know about the product called Flash Operator Panel. It has been around for years, and is included in many of the popular Asterisk distributions including Elastix. Its Author, Nicolas Gudino, has, after many years released a major revamp of this popular tool in the guise of FOP2 or Flash Operator Panel 2.

Whilst this new version has a cost, and a very reasonable one (\$US40), it can be used on many home, micro-business PBX systems without the need for a licence (upto 15 buttons which include trunks, extension, queues, conference as buttons).

So this tutorial is to assist Elastix users in implementing FOP2 onto their systems.

First step is to disable the old Flash Operator panel

To do this , type

```
nano /var/www/html/panel/op_server.cfg (note nano is an editor that I use and installed, you can use VI or similar )
```

Look for the following line

Your config file may show the listen_port line commented out. Remove the semicolon if it is there and change the port over to 4446 (not worried what port, just need it to stop listening on 4445). So the line should look like this

Save and exit.

Now the first thing we need to do is to setup a user in the Asterisk Management Interface (AMI). This is done via the unembedded Freepbx. Click on the tools table and then click on Asterisk API.

Add a Manager Name called fop2 and give it a Manager Secret of fop2secret

Now Download the latest FOP2 to your workstation

Use WinSCP or similar to move the file to /usr/src

Use SSH or console

```
cd /usr/src
```

```
tar -zxvf fop2-001-i386-glibc2.5-centos5.tgz
```

This will extract all the files into /usr/src/fop2 directory

```
cd /usr/src/fop2
```

```
make install
```

```
cd /usr/local/fop2
```

```
nano fop2.cfg
```


Your config file would be different in that your manager_host would be your Elastix system IP address.

You will also note that we are using the manager name and manager secret that we set up right at the start of this tutorial. You can change it as you wish, just make sure you change it in both the Asterisk API page and also in this config file

You will also note that I have uncommented and changed the master key. This is basically an override password for the system admin, so that he can login on the FOP2 page without having to remember the password for each user

Leave the user list commented. We are going to use the automatically generated one

If it isn't already, comment out buttonflie=buttons.cfg

Leave the line that as is that says

```
#exec autoconfig-users-freepbx.sh
```

This line tells the server to get it its user login settings from the Freepbx configurations.

Once you have completed these changes, reboot your system will pickup the changes so far.

Once you have saved this then issue the following command

```
/usr/local/fop2/fop2_server --test
```

You should see the following screen come up

Don't panic on the license issue, we mainly want to confirm that it is talking to the AMI Interface.

If you haven't got this screen, then go back over your settings.

Next lets add the bits that improve the functionality of FOP2

```
nano /etc/asterisk/sip_custom.conf
```

and add the following line (at the end if you have something already in your sip_custom.conf

You may note that FOP2 instructions on www.fop2.com say to add it to sip.conf, and as most Elastix users know, this file is overwritten during upgrades, so we add it to the sip_custom.conf file so that it remains in place during upgrades.

Also if you want your queues to have full functionality in FOP2, you also need to make sure that the following EVENT WHEN CALLED is set to yes under each of the Queues (under Queue Options) in Freepbx or Embedded Freepbx, similar to the following

Now just so that we know that all the configurations have been read, just do a final reboot before we finish.

Before we login to the FOP2 Web Interface, we need to point out a passage from the FOP2 documentation. The reason why I point this out, is that in a rush to get things up an going, I missed it, and spent several hours, thinking that things weren't working as I was using the phone login and secret.

"Instead of setting up each user for fop2, by using this script it will read asterisk's voicemail.conf file and generate users from there, with the same voicemail password and granting all permissions."

What this line means (especially as you probably read it before you understand about the Freepbx auto scripts - which is what we are using), is that the login and password for the FOP2 Web Panel is your extension number and the password that is setup on voicemail.

So what this means also, is that if your user does not have Voicemail enabled, they will not be setup with a login into FOP2.

Now if this is your setup, and many of your users do not have voicemail, then you will need to read the FOP2 documentation, and learn how to setup manually configured configuration files, so it is not impossible, just a little more work.

So with that in hand.....

Go to your browser and type in

`http://{yourElastixIPAddress}/fop2`

You will be greeted by the following screen

Enter in your extension and voicemail password and you should be presented with the following screen

or something similar

For the basic interface, that's about it.

Now a couple of other points before you spend a few more hours wondering why it is not working (e.g. your queues don't show up, or the conferences don't show up, or you are missing some trunks, or you are missing a few users extensions. The demo licence allows up to 15 buttons. The buttons shown in the above example number 14. So buttons refer to extensions, queues, trunks and conferences.

If you go over this number, it does not place the buttons on your panel. If you need more than 14 buttons then you need to purchase a licence. At \$US40 dollars, to remove this limitation, it is a no-brainer. I would have to say that it is the best value for \$40, and for any system that needs more than 15 buttons is generally going to be a business system (as opposed to a home system or test system), and \$40 is affordable.

I would have to say that Nicolas has done a great job on the Documentation, far more than any of his other products, which shows the commitment that he has made to FOP2. The more that we support this product (by purchasing licences) the more Nicolas will put into this product in terms of development.

If you want to implement the Phone Directory, I recommend reading the documentation at www.fop2.com which will take you through that next step.

Manual Setup (complete all the above setup first)

Now as many may find, the current implementation of FOP2 with its automatic scripts to locate user ID's for logins and/or for the setup of buttons, is limited. For instance, if most of your users do not have voicemail, then they will not have an automatic login setup. Similarly, you may not wish to display all the extensions in FOP2 that you have setup, e.g. Test Extensions, Test Queues, the Bosses home extension etc.

This is where you may need to change the way the scripts work.

so edit fop2.cfg under /usr/local/fop2

as you can see, we have manually added some users, and we have given them certain access to functions. 201 has access to all functions, 202 also has full access, just written in long form, 203 has access only to dial , ability to transfer a call, hangup, call pickup and ability to record.

The first number is the extension, and the second is the password. In the case above we are keeping it simple an their extension number is also their password.

Note that we have now uncommented the line `buttonfile=buttons.cfg` and commented the `exec` line underneath. This tells it to use the `button.cfg` file for the buttons config, and commenting the `exec` file (basically the `freepbx` autoconfig), stops it performing an autoconfiguration from `Freepbx`. If you leave the `exec` line uncommented, it will ignore your `buttons.cfg`, and it will basically ignore the users that you have setup.

Now before you go racing ahead, you need to setup a `buttons.cfg`. There is a quick way to do this (to save you going line by line building a `buttons.cfg`).

at the console under the `/usr/local/fop2` directory type the following line

```
./autoconfig-buttons-freepbx.sh >buttons.cfg
```

this will create the `buttons.cfg`. You can manually edit this file, take out extensions or queues, or whatever you wish. Thats it. Now restart FOP2 by the following command

service fop2 restart

Thats it, enter FOP2 via the browser and test.

Now that you are using the manual configuration, it means new users need to be set manually, but for many, the functionality that comes with FOP2, a little manual configuration is not a big deal. However it is understood that FOP2 will have a freepbx module in the near future allowing configuration from the Freepbx GUI, turning on and off functions, without the need for the manual editing.

Also you will need manual configuration if you are going to use Multi Tenant and/or monitor more than one PBX.