

Patching Asterisk 11 for Cisco phones

Steps in this document end up modifying the version of Asterisk provided FreePBX. You assume all risks herein and will no longer be able to use upgrade scripts without breaking functionality. You will have to get upgrades manually and apply them manually.

Some people like Cisco handsets, some people don't. If you fall into the first category then you may be pleased to hear that they can work very well with an Asterisk system, no Cisco Unified Call Manager required. They actually work *better* in some areas when driven by Asterisk. And of course, in other areas... not so well.

There's an amount of work to be done to get here, and hopefully there is enough information here to help you on your way.

To get the most out of these phones, Asterisk needs a patch applied from <http://usecallmanager.nz>

This is the old version of this document, and is for use with Asterisk 11 only.

Everything in this document has been tested on a clean install of FreePBX and confirmed working with the following specific versions:

- Cisco firmware 11-0-1-11 for the 8800 series
- Cisco firmware 9.4.2-SR2-2 for the 9900 series
- Asterisk 11.21-2
- cisco-usecallmanager-patch 11.21-2
- FreePBX 13 distribution

-
- [About this document](#)
 - [Prep-work](#)
 - [Patching Asterisk](#)
 - [Getting the firmware](#)
 - [Configuration files](#)
 - [Device](#)
 - [featurepolicy.xml](#)
 - [softkeys.xml](#)
 - [Asterisk settings](#)
 - [extensions_custom.conf](#)
 - [notes](#)
 - [Adhoc call recording](#)
 - [blocking video](#)
 - [sip_custom_post.conf](#)
 - [extensions_override_freepbx.conf](#)
 - [sip_notify_custom.conf](#)
 - [very optional extras](#)
 - [beeps](#)
 - [directory](#)
 - [Unsolved](#)
 - [Thanks](#)

About this document

All of the notes here are based on my research, drawing on the works of others and many (many) hours of trial and error. Everything here has been tested by building a new FreePBX server and following these instructions to end up with a working solution.

Where files are provided, the first line in each is a commented out line in the relevant language, showing the name apportioned to that file.

This is not intended to be an exhaustive investigation into the workings of these devices, it does not answer every question that could be asked, or deal with every possible configuration of the systems; it provides a working template from which further work can be done by the user if desired. When I started working with these devices I was unable to find a source of information that provided that, so I made one.

Prep-work

To start, you will need a FreePBX installation, working and updated. Deploy the current version and select Asterisk 11 at installation. Once it's running, register and update all the modules (if necessary).

Important: The Cisco phones use TCP only, so add `tcpenable=yes` to the SIP Settings of FreePBX. When you add Cisco extensions, set the Transport to TCP-Only.

Make an extension for one of your cisco handsets, and run into the first gotcha. By default, FreePBX will create a random SIP password for you, those passwords are too long for the Cisco phones, I didn't yet trial-and-error out the maximum length, I just deleted half of the password and that worked. Even at half the length of the auto one, it's still plenty secure.

With the install running, and an extension ready, it's time to patch asterisk itself, and then create the config files for the phone.

Once the configuration files have been made, plug in a handset and let it boot, it will pick up all the settings and should register correctly. Ensure the DHCP server is offering a 150 response pointing to the FreePBX server so the phone can find the tftpboot service.

Patching Asterisk

Time to get your hands dirty. FreePBX does not come with the Asterisk source files but they do have source RPMs available that contain pretty much everything you need. The patch process is simple enough:

SSH into the FreePBX box as whatever user you use for this purpose, I'll select root for this document.

First, configure the build environment:

```
yum install epel-release
yum install mISDN-devel
yum install rpm-build
yum groupinstall "Development Tools"
yum remove epel-release
mkdir -p ~/rpmbuild/{BUILD,RPMS,SOURCES,SPECS,SRPMS}
echo '%_topdir %(echo $HOME)/rpmbuild' > ~/.rpmmacros
```

Now download the FreePBX SRPM and the patch. We also download mISDNuser, it makes building Asterisk easier if this is available. The filenames here are current at time of writing but you should check for newer ones as appropriate. The Asterisk version and the patch version *must* match.

```
cd ~/rpmbuild/SOURCES
wget http://download.opensuse.org/tumbleweed/repo/src-oss/suse/src/mISDNuser-2.0.19-1.5.src.rpm
wget https://issues.asterisk.org/jira/secure/attachment/53565/cisco-usecallmanager-11.21-2.patch
wget http://yum.freepbxdistro.org/pbx/SRPMS/asterisk/11/asterisk11-11.21.2-1.shmz65.1.104.src.rpm
```

Then it's just a case of unpack the files, tell FreePBX about the new patch, compile the files and install them:

```
rpm --nomd5 -ivh ~/rpmbuild/SOURCES/asterisk11-11.21.2-1.shmz65.1.104.src.rpm
rpm --nomd5 -ivh ~/rpmbuild/SOURCES/mISDNuser-2.0.19-1.5.src.rpm
yum-builddep ~/rpmbuild/SPECS/mISDNuser.spec
rpmbuild -bp ~/rpmbuild/SPECS/mISDNuser.spec
rpmbuild -ba ~/rpmbuild/SPECS/mISDNuser.spec
rpm -Uvh ~/rpmbuild/RPMS/x86_64/libmisdn1-*.rpm
rpm -Uvh ~/rpmbuild/RPMS/x86_64/mISDNuser*.rpm
yum-builddep ~/rpmbuild/SPECS/asterisk11.spec
rpmbuild -bp ~/rpmbuild/SPECS/asterisk11.spec
```

```
nano ~/rpmbuild/SPECS/asterisk11.spec
```

```
[locate the first instance of the word "Patch12" and add this exact line directly below it:]
Patch13: cisco-usecallmanager-11.21.2.patch
```

```
[then search for "patch12" (note the change in case) and add this exact line directly below it:]
%patch13 -pl
```

```
[save the file and close the editor]
```

```
rpmbuild -bp ~/rpmbuild/SPECS/asterisk11.spec
rpmbuild -ba ~/rpmbuild/SPECS/asterisk11.spec
rpm -Uvh ~/rpmbuild/RPMS/x86_64/asterisk11-[a-l]*.rpm --force
rpm -Uvh ~/rpmbuild/RPMS/x86_64/asterisk11-[n-r]*.rpm --force
rpm -Uvh ~/rpmbuild/RPMS/x86_64/asterisk11-t*.rpm --force
rpm -Uvh ~/rpmbuild/RPMS/x86_64/asterisk11-voicemail-11*.rpm --force
```

And that's it! Reboot the server to make sure everything is settled in and you should be left with a working version of Asterisk 11 and FreePBX with the Cisco patches included.

Getting the firmware

Three files are needed directly from Cisco. To download them, you need a Cisco login, which for basic levels, is free.

You need the firmware files for your phone, the user locale and the network locale. Check the website linked earlier for some decent instructions on how to get and use those files. In fact, spend some time reading everything else on that site while you're at it.

Put the newly extracted files into /tftpboot

Configuration files

Device

Each phone must have a complete configuration file, they don't use templates in the way other SIP phones do. The file is lengthy, so I have added comments inline rather than explaining here. Don't simply copy/paste the file, look for the comments and adjust values as necessary. The shown file is for a phone with a MAC address of 1234567890 and configured as extension 102 for a server at pbx.domain.internal with a password of StrongPassword. It provides a BLF linekey, a DND linekey and an ad-hoc recording button.

You can use a file named XMLDefault.cnf.xml and provide a minimal version of the config so as to get a firmware onto the phone, but I've not bothered.

The file is named with the prefix of the type of phone, these devices use 'SEP' followed by the mac address of the phone with no separators and in uppercase, and has a .cnf.xml extension. As such, these files are known as SEPMAC config files, or variations on that theme.

Each config file has a <versionStamp> key, it contains a version 1 UUID. The phone uses this to recognise a changed file and it will extract the timestamp from the one in the main file to show a 'last configured' value.

The config files should be made for each phone, and placed in the /tftpboot folder of the FreePBX server.

When the phone loads the config file, it is checked for errors and the phone will refuse the file if errors are found, it will not however give any indication of where those errors may be. The file shown here *works* but there is many reasons why it may not be *perfect*. An excellent breakdown of the various sections is available, again at <http://docs.acsdata.co.nz/asterisk-cisco>

```
<!-- SEP1234567890.cnf.xml -->
<?xml version="1.0" encoding="utf-8"?>
<device>
  <versionStamp>c1b05fde-ef64-11e5-9ce9-5e5517507c66</versionStamp>
  <fullConfig>>true</fullConfig>
  <deviceProtocol>SIP</deviceProtocol>
  <sshUserId>cisco</sshUserId>
  <sshPassword>cisco</sshPassword>
  <transportLayerProtocol>1</transportLayerProtocol>
  <dndCallAlert>5</dndCallAlert>
  <featurePolicyFile>featurepolicy.xml</featurePolicyFile>
  <missedCallLoggingOption>0</missedCallLoggingOption>
  <commonProfile>
    <backgroundImageAccess>>true</backgroundImageAccess>
    <phonePassword></phonePassword>
    <callLogBlfEnabled>0</callLogBlfEnabled>
  </commonProfile>
  <advertiseG722Codec>1</advertiseG722Codec>
  <authenticationURL></authenticationURL>
  <messagesURL></messagesURL>
  <informationURL></informationURL>
  <servicesURL></servicesURL>
  <directoryURL></directoryURL>

  <!-- ** Firmware file -->
  <loadInformation>sip8845_65.11-0-1-11</loadInformation>
  <devicePool>
    <dateTimeSetting>
      <dateTemplate>YA.M.D</dateTemplate>
      <timeZone>GMT Standard/Daylight Time</timeZone>
      <ntp>
        <ntp>
```

```

        <name>0.us.pool.ntp.org</name>
        <ntpMode>Unicast</ntpMode>
    </ntp>
</ntps>
</dateTimeSetting>
<callManagerGroup>
    <members>
        <member>
            <callManager>
                <processNodeName>**SERVER-IP-OR-URL**</processNodeName>
                <ports>
                    <sipPort>5060</sipPort>
                </ports>
            </callManager>
        </member>
    </members>
</callManagerGroup>
</devicePool>
<sipProfile>
    <alwaysUsePrimeLine>true</alwaysUsePrimeLine>
    <phoneLabel>**EXTENSION-NUMBER**</phoneLabel>
    <sipProxies>
        <registerWithProxy>true</registerWithProxy>
    </sipProxies>
    <sipCallFeatures>
        <cnfJoinEnabled>true</cnfJoinEnabled>
        <callForwardURI>x-cisco-serviceuri-cfwdall</callForwardURI>
        <callPickupURI>x-cisco-serviceuri-pickup</callPickupURI>
        <callPickupListURI>x-cisco-serviceuri-opickup</callPickupListURI>
        <callPickupGroupURI>x-cisco-serviceuri-gpickup</callPickupGroupURI>
        <meetMeServiceURI>x-cisco-serviceuri-meetme</meetMeServiceURI>
        <abbreviatedDialURI>x-cisco-serviceuri-abbrdial</abbreviatedDialURI>
        <rfc2543Hold>false</rfc2543Hold>
        <callHoldRingback>0</callHoldRingback>
        <localCfwdEnable>true</localCfwdEnable>
        <semiAttendedTransfer>true</semiAttendedTransfer>
        <anonymousCallBlock>2</anonymousCallBlock>
        <callerIdBlocking>2</callerIdBlocking>
        <dndControl>0</dndControl>
        <remoteCcEnable>true</remoteCcEnable>
    </sipCallFeatures>
    <sipStack>
        <sipInviteRetx>6</sipInviteRetx>
        <sipRetx>10</sipRetx>
        <timerInviteExpires>180</timerInviteExpires>
        <timerRegisterExpires>3600</timerRegisterExpires>
        <timerRegisterDelta>5</timerRegisterDelta>
        <timerKeepAliveExpires>120</timerKeepAliveExpires>
        <timerSubscribeExpires>120</timerSubscribeExpires>
        <timerSubscribeDelta>5</timerSubscribeDelta>
        <timerT1>500</timerT1>
        <timerT2>4000</timerT2>
        <maxRedirects>70</maxRedirects>
        <remotePartyID>true</remotePartyID>
        <userInfo>Phone</userInfo>
    </sipStack>
    <transferOnhookEnabled>true</transferOnhookEnabled>
    <enableVad>false</enableVad>
    <preferredCodec>none</preferredCodec>
    <dtmfAvtPayload>101</dtmfAvtPayload>
    <dtmfDbLevel>3</dtmfDbLevel>
    <dtmfOutOfBand>avt</dtmfOutOfBand>
    <stutterMsgWaiting>0</stutterMsgWaiting>
    <callStats>true</callStats>
    <offhookToFirstDigitTimer>15000</offhookToFirstDigitTimer>
    <silentPeriodBetweenCallWaitingBursts>10</silentPeriodBetweenCallWaitingBursts>
    <startMediaPort>10000</startMediaPort>
    <stopMediaPort>20000</stopMediaPort>
    <natEnabled>false</natEnabled>
    <natReceivedProcessing>false</natReceivedProcessing>
    <natAddress></natAddress>
    <kpml>0</kpml>
    <voipControlPort>5060</voipControlPort>
    <dscpForAudio>184</dscpForAudio>
    <dscpVideo>136</dscpVideo>
    <ringSettingBusyStationPolicy>0</ringSettingBusyStationPolicy>
    <dialTemplate>dialtemplate.xml</dialTemplate>
    <softKeyFile>softkeys.xml</softKeyFile>
</sipLines>

```

```

    <!-- Main line, register with asterisk using your credentials as in FreePBX -->
    <line button="1" lineIndex="1">
        <!-- the 'lineIndex' here is ESSENTIAL, without it, the phone will 'lock' missed calls on
the display, causing the phone to be basically broken. Do not put a line index on the other keys, that
will also break things. -->
        <featureID>9</featureID>
        <featureLabel>**USER-FULL-NAME**</featureLabel>
        <callWaiting>3</callWaiting>
        <proxy>USECALLMANAGER</proxy>
        <port>5060</port>
        <name>**EXTENSION-NUMBER**</name>
        <authName>**EXTENSION-NUMBER**</authName>
        <authPassword>**SIP PASSWORD**</authPassword>
        <sharedLine>>false</sharedLine>
        <messageWaitingLampPolicy>3</messageWaitingLampPolicy>
        <messageWaitingAMWI>0</messageWaitingAMWI>
        <messagesNumber>*97</messagesNumber>
        <ringSettingIdle>4</ringSettingIdle>
        <ringSettingActive>2</ringSettingActive>
        <autoAnswer>
            <autoAnswerEnabled>2</autoAnswerEnabled>
        </autoAnswer>
        <forwardCallInfoDisplay>
            <callerName>>true</callerName>
            <callerNumber>>true</callerNumber>
            <redirectedNumber>>false</redirectedNumber>
            <dialNumber>>true</dialNumber>
        </forwardCallInfoDisplay>
        <maxNumCalls>3</maxNumCalls>
        <busyTrigger>3</busyTrigger>
    </line>

    <!-- I like a clear view of the phones Do Not Disturb state, this linekey provides that -->
    <line button="2">
        <featureID>130</featureID>
        <featureLabel>DND</featureLabel>
    </line>

    <!-- BLF KEY, set to the extension number of a station you want this phone to monitor and be
able to pick up calls for etc -->
    <line button="3">
        <featureID>21</featureID>
        <featureLabel>**NAME**</featureLabel>
        <speedDialNumber>**BLF-EXTENSION-NUMBER**</speedDialNumber>
        <featureOptionMask>1</featureOptionMask>
    </line>

    <line button="5">
        <featureID>159</featureID>
        <featureLabel>Record</featureLabel>
    </line>

</sipLines>
</sipProfile>
<vendorConfig>
    <disableSpeaker>>false</disableSpeaker>
    <disableSpeakerAndHeadset>>false</disableSpeakerAndHeadset>
    <pcPort>0</pcPort>
    <settingsAccess>1</settingsAccess>
    <garp>1</garp>
    <voiceVlanAccess>0</voiceVlanAccess>
    <videoCapability>1</videoCapability>
    <ciscoCamera>1</ciscoCamera>
    <autoSelectLineEnable>1</autoSelectLineEnable>
    <webAccess>0</webAccess>
    <spanToPCPort>1</spanToPCPort>
    <daysDisplayNotActive>1,7</daysDisplayNotActive>

    <!-- the next four lines: At 8am, activate phone and keep the screen on for an hour gives the
office a good 'awake' look. Then after 9am, turn the display on if there is an incoming call. Turn off
after 10 minutes of inactivity. -->
    <displayOnTime>08:00</displayOnTime>
    <displayOnDuration>01:00</displayOnDuration>
    <displayIdleTimeout>00:30</displayIdleTimeout>
    <displayOnWhenIncomingCall>1</displayOnWhenIncomingCall>
    <simplifiedNewCall>0</simplifiedNewCall>
    <phoneBookWebAccess>1</phoneBookWebAccess>

```

```

<handsetWidebandEnable>2</handsetWidebandEnable>
<headsetWidebandEnable>2</headsetWidebandEnable>
<headsetWidebandUIControl>1</headsetWidebandUIControl>
<handsetWidebandUIControl>1</handsetWidebandUIControl>
<separateMute>0</separateMute>
<sshAccess>1</sshAccess>
<sshPort>22</sshPort>
<autoCallSelect>1</autoCallSelect>
<incomingCallToastTimer>5</incomingCallToastTimer>
<joinAndDirectTransferPolicy>1</joinAndDirectTransferPolicy>
<bluetooth>1</bluetooth>
<bluetoothProfile>0,1</bluetoothProfile>
<btspbap>1</btspbap>
<usb1>1</usb1>
<usb2>1</usb2>
<usbClasses>0,1,2</usbClasses>
<g722CodecSupport>2</g722CodecSupport>

<!-- set to 1 to use the softkey.xml file, set to 0 to use the featurePolicy.xml file -->
<softkeyControl>1</softkeyControl>
</vendorConfig>

<userLocale>
  <name>English_United_Kingdom</name>
  <!-- the name above is the exact name of the folder inside /tftpboot where the locale files are
stored -->
  <uid>1</uid>
  <langCode>en_GB</langCode>
  <version>1.0.0.0-1</version>
  <winCharSet>iso-8859-1</winCharSet>
</userLocale>

<networkLocaleInfo>
  <name>English_United_Kingdom</name>
  <uid>1</uid>
  <langCode>en_GB</langCode>
  <version>1.0.0.0-1</version>
</networkLocaleInfo>

<phoneServices>
  <provisioning>0</provisioning>
  <phoneService type="1" category="0">
    <name>Missed Calls</name>
    <url>Application:Cisco/MissedCalls</url>
    <vendor></vendor>
    <version></version>
  </phoneService>
  <phoneService type="2" category="0">
    <name>Voicemail</name>
    <url>Application:Cisco/Voicemail</url>
    <vendor></vendor>
    <version></version>
  </phoneService>
  <phoneService type="1" category="0">
    <name>Received Calls</name>
    <url>Application:Cisco/ReceivedCalls</url>
    <vendor></vendor>
    <version></version>
  </phoneService>
  <phoneService type="1" category="0">
    <name>Placed Calls</name>
    <url>Application:Cisco/PlacedCalls</url>
    <vendor></vendor>
    <version></version>
  </phoneService>
  <phoneService type="1" category="0">
    <name>Contacts</name>
    <!-- set the URL below to match your server DNS name, and edit the extension to match the
correct extension for this file -->
    <url>http://freepbx.server.internal/cisco/directory.php?xtn=101</url>
    <vendor></vendor>
    <version></version>
  </phoneService>
  <phoneService type="0" category="0">
    <name>Visual Voicemail</name>
    <url>**url of the visual voicemail application**</url>
    <vendor></vendor>
    <version></version>
  </phoneService>

```

```
</phoneServices>
</device>
```

...

featurepolicy.xml

The 9900 series prefers to use the feature policy file to control the softkeys at the bottom of the display, but can be instructed to use a softkeys.xml file instead

Some of the available button options are rather Cisco in nature, and I did not find much value in having them available to the users. Fewer options seemed to be the path to success here.

As such, this file is small and provides only a few buttons:

- pickup: answer a ringing phone in your ringgroup
- redial: dial the last completed number
- divert: used for incoming calls, rejects the call and sends to voicemail (if voicemail is configured for that phone, otherwise it terminates the call)
- QRT: Quality Reporting Tool. If pressed during a call, this will send some debugging info back to Asterisk when the call completes, this can be seen on the console or in the logs.

Again, more details on what this file can provide can be found at the site linked in the introduction.

```
<!-- featurepolicy.xml -->
<featurePolicy name="Feature Policy">
  <versionStamp>c1b05fde-ef64-11e5-9ce9-5e5517507c66</versionStamp>
  <featureDef name="Redial">
    <id>7</id>
    <enable>true</enable>
  </featureDef>
  <featureDef name="Call PickUp">
    <id>10</id>
    <enable>true</enable>
  </featureDef>
  <featureDef name="Divert (Alerting)">
    <id>3</id>
    <enable>true</enable>
  </featureDef>
  <featureDef name="Quality Reporting Tool">
    <id>15</id>
    <enable>true</enable>
  </featureDef>
</featurePolicy>
```

softkeys.xml

The 8800 series did not read the featurepolicy.xml file I had, and would only respond correctly to the softkeys.xml.

Here is the file I use, again I have removed the more 'Cisco' centric buttons, but this file does allow greater control of the phone, so it is longer and does offer a few more options.

Errors in this file will result in the phone ignoring it and simply displaying the default set of all buttons.

```
<!-- softkeys.xml -->
<softKeyCfg>
  <versionStamp>c1b05fde-ef64-11e5-9ce9-5e5517507c66</versionStamp>
  <typeSoftKey>
    <softKeyDef keyID="Undefined">
      <tag>0</tag>
      <eventID>0</eventID>
      <helpID>0</helpID>
    </softKeyDef>
    <softKeyDef keyID="Redial">
      <tag>1</tag>
      <eventID>1</eventID>
      <helpID>301</helpID>
    </softKeyDef>
    <softKeyDef keyID="NewCall">
      <tag>2</tag>
      <eventID>2</eventID>
      <helpID>302</helpID>
    </softKeyDef>
  </typeSoftKey>
</softKeyCfg>
```

```

</softKeyDef>
<softKeyDef keyID="Hold">
  <tag>3</tag>
  <eventID>3</eventID>
  <helpID>303</helpID>
</softKeyDef>
<softKeyDef keyID="&lt;&lt;">
  <tag>8</tag>
  <eventID>8</eventID>
  <helpID>308</helpID>
</softKeyDef>
<softKeyDef keyID="CFwdAll">
  <tag>5</tag>
  <eventID>5</eventID>
  <helpID>305</helpID>
</softKeyDef>
<softKeyDef keyID="EndCall">
  <tag>9</tag>
  <eventID>9</eventID>
  <helpID>309</helpID>
</softKeyDef>
<softKeyDef keyID="Resume">
  <tag>10</tag>
  <eventID>10</eventID>
  <helpID>310</helpID>
</softKeyDef>
<softKeyDef keyID="Answer">
  <tag>11</tag>
  <eventID>11</eventID>
  <helpID>311</helpID>
</softKeyDef>
<softKeyDef keyID="PickUp">
  <tag>17</tag>
  <eventID>17</eventID>
  <helpID>317</helpID>
</softKeyDef>
<softKeyDef keyID="QRT">
  <tag>75</tag>
  <eventID>22</eventID>
  <helpID>322</helpID>
</softKeyDef>
<softKeyDef keyID="Select">
  <tag>78</tag>
  <eventID>29</eventID>
  <helpID>329</helpID>
</softKeyDef>
<softKeyDef keyID="iDivert">
  <tag>80</tag>
  <eventID>31</eventID>
  <helpID>331</helpID>
</softKeyDef>
<softKeyDef keyID="Record">
  <tag>7747</tag>
  <eventID>74</eventID>
  <helpID>374</helpID>
</softKeyDef>
</typeSoftKey>

<softKeySets>
  <softKeySet id="On Hook">
    <softKey keyID="Redial" />
    <softKey keyID="NewCall" />
    <softKey keyID="PickUp" />
    <softKey keyID="CFwdAll" />
  </softKeySet>
  <softKeySet id="Off Hook">
    <softKey keyID="Redial" />
    <softKey keyID="EndCall" />
  </softKeySet>
  <softKeySet id="Off Hook With Feature">
    <softKey keyID="Redial" />
    <softKey keyID="EndCall" />
  </softKeySet>
  <softKeySet id="Digits After First">
    <softKey keyID="&lt;&lt;" />
    <softKey keyID="EndCall" />
  </softKeySet>
  <softKeySet id="Ring Out">
    <softKey keyID="Undefined" />
  </softKeySet>

```



```

        <softKey keyID="EndCall" />
    </softKeySet>
    <softKeySet id="Connected">
        <softKey keyID="Hold" />
        <softKey keyID="EndCall" />
        <softKey keyID="Record" />
        <softKey keyID="QRT" />
    </softKeySet>
    <softKeySet id="Connected No Feature">
        <softKey keyID="Undefined" />
        <softKey keyID="EndCall" />
    </softKeySet>
    <softKeySet id="Connected Transfer">
        <softKey keyID="Undefined" />
        <softKey keyID="EndCall" />
        <softKey keyID="Transfer" />
    </softKeySet>
    <softKeySet id="On Hold">
        <softKey keyID="Resume" />
        <softKey keyID="NewCall" />
    </softKeySet>
    <softKeySet id="Ring In">
        <softKey keyID="Answer" />
        <softKey keyID="iDivert" />
    </softKeySet>
</softKeySets>
</softKeyCfg>

```

Asterisk settings

The patch provides documentation in the form of sample files, feel free to read them. Some things are certainly needed though, and are covered here.

extensions_custom.conf

A few additions to this file. We need to add some code to enable the 'pickup' softkey. It transmits a message to the server that would go unhandled otherwise. Also, if you are using video handsets, you may notice that a call to (for example) your mobile phone will result in a black screen on the deskphone as it tries to show you a video stream that will never arrive, a quick dialplan modification in this file can prevent that.

```

;extensions_custom.conf
; ----- External calls, lock codec to prevent video
exten => _0..1,Set(__SIP_CODEC=alaw)
same => next,Goto(outbound-allroutes,${EXTEN},1)

; ----- Feature codes, lock codec to prevent video
exten => _*.1,Set(__SIP_CODEC=g722)
same => next,Goto(from-internal,${EXTEN},4)

; ----- CISCO custom
[from-internal-custom]
; ----- Strip the x-cisco-serviceuri- prefix to make handling the cisco events in the dialplan a little
simpler
exten => _[x]-cisco-serviceuri-.1,Goto(${EXTEN:19},1)

; ----- Group Pickup softkey
exten => pickup,1,Pickup()
same => next,Hangup(normal_circuit_congestion)

; ----- BLF Pickup softkey
exten => _blfpickup-.1,PickupChan(SIP/${EXTEN:10})
same => next,Hangup(normal_circuit_congestion)

; ----- AdHoc call recording softkey
exten => record,1,Answer()
same => next,Wait(0.5)
same => next,Set(CONFBRIDGE(bridge,record_conference)=yes)
same => next,Set(CONFBRIDGE(bridge,record_file)=${RECORD_PEERNAME})
same => next,Set(CONFBRIDGE(user,quiet)=yes)
same => next,ConfBridge(${RECORD_UNIQUEID})
same => next,Hangup(normal_clearing)

```

```

; Cisco SIP phones support displaying diversion information
; ** this part not currently working **
; exten => forward,SIPAddHeader(Diversion: "${SIPPEER(${PEERNAME},callerid_name)}" <sip:${PEERNAME}
@localhost>\;screen=yes\;privacy=off)
; same => Goto(extensions,${CALLFORWARD},1)

; Enable forwarding
exten => _cfwdall-. ,1,Answer
same => next,Set(SIPPEER(${CHANNEL(peername)}),callforward)=${EXTEN:8})
same => next,Hangup(normal_clearing)

; Disable forwarding
exten => cfwdall,1,Answer
same => next,Set(SIPPEER(${CHANNEL(peername)}),callforward=)
same => next,Hangup(normal_clearing)

```

notes

Adhoc call recording

If the recording softkey has been enabled for a linekey in the sepmac.cnf.xml (9900 only) then the word 'Recording' will appear next to it. This does not mean that the call is currently recording, or being recorded. Press the button and a message will appear telling you that the recording has started. The state of the button will also change to indicate this.

The 8800 does not support 'record' as a linekey, and it must be added as a softkey instead.

blocking video

This works by instructing asterisk to use only the codec provided in the command. Asterisk 13 is happy with multiple codecs, such as *ulaw&alaw&g729*, Asterisk 11 on the other hand, is not. As a result, only a single codec can be selected, choose carefully.

sip_custom_post.conf

This one is interesting, it sets up a template for Cisco extensions and then applies those settings to each device. If a handset needs to subscribe for BLF hints, the subscription is added here.

For this example, we have three phones:

- extension 101
- extension 102, which has a BLF linekey for extension 103
- extension 103, which has a BLF linekey for extension 102

```

;sip_custom_post.conf
[cisco](!)
cisco_usecallmanager=yes
cisco_pickupnotify_alert=from,to
cisco_pickupnotify_timer=5
dndbusy=yes
rpid_update=yes
rpid_immediate=yes
allowsubscribe=yes
notifyhold=yes

[101](+,cisco)
[102](+,cisco)
subscribe=103
[103](+,cisco)
subscribe=102

```

extensions_override_freepbx.conf

Enabling those BLF buttons needs slightly different hints to be set up for the phones, add lines such as these for the example above.

```

;extensions_override_freepbx.conf
[ext-local]
exten => 102,hint,SIP/${EXTEN},SIP/${EXTEN}
exten => 103,hint,SIP/${EXTEN},SIP/${EXTEN}

```

sip_notify_custom.conf

It is very useful to be able to reboot the phones remotely. The patch allows this, so long as you have the required sip-notify commands available.

sip notify cisco-restart instructs the phone to reload the config files, allowing changes such as new lines keys to take affect. Takes just a few seconds.

sip notify cisco-reset performs a warm reboot of the phone, will pick up the full config and takes a bit longer than the restart.

```
;sip_notify_custom.conf
[cisco-restart]
Event=>service-control
Subscription-State=>active
Content-Type=>text/plain
Content=>action=restart
Content=>RegisterCallId=${SIPPEER(${PEERNAME},regcallid)}
Content=>ConfigVersionStamp={00000000-0000-0000-0000-000000000000}
Content=>DialplanVersionStamp={00000000-0000-0000-0000-000000000000}
Content=>SoftkeyVersionStamp={00000000-0000-0000-0000-000000000000}
Content=>FeatureControlVersionStamp={00000000-0000-0000-0000-000000000000}

[cisco-reset]
Event=>service-control
Subscription-State=>active
Content-Type=>text/plain
Content=>action=reset
Content=>RegisterCallId=${SIPPEER(${PEERNAME},regcallid)}
Content=>ConfigVersionStamp={00000000-0000-0000-0000-000000000000}
Content=>DialplanVersionStamp={00000000-0000-0000-0000-000000000000}
Content=>SoftkeyVersionStamp={00000000-0000-0000-0000-000000000000}
Content=>FeatureControlVersionStamp={00000000-0000-0000-0000-000000000000}
```

very optional extras

beeps

We found that transferring a call was nicer if a beep was sounded as the connection switched, this isn't default behaviour, turn it on with this:

(not currently working correctly)

```
;features_general_custom.conf
xfersound = beep
xferfailsound = beeperr
pickupsound = beep
pickupfailsound = beeperr
```

directory

In the SEPMAC config file, near the bottom, you may have noticed the entry for the directory. This points to a script that you can put anywhere.

I include here a script I wrote that will present three options to the user:

- Internal Directory
- Ring groups
- Personal contacts

Put this script into `/var/www/html/cisco` and chown it to `asterisk:asterisk`

The script is called when you press the physical 'directory' button on the phone.

The Personal directory works like this: If your users use UCP, instruct them to make a group in their contacts, and name that group by their extension number. So whoever has extension 101 should make a group called 101. In that group, they put their contacts. The script will locate the numbers in that group, and show them on the phone.

```
// directory.php
<?php
header ("content-type: text/xml");
$xtn = array_key_exists('xtn', $_GET) ? $_GET['xtn'] : NULL;
$mode = array_key_exists('mode', $_GET) ? $_GET['mode'] : NULL;
$url = "http://".$_SERVER['HTTP_HOST'].$_SERVER['PHP_SELF'];
```

```

if (!@include_once(getenv('FREEPBX_CONF') ? getenv('FREEPBX_CONF') : '/etc/freepbx.conf')) {
    include_once('/etc/asterisk/freepbx.conf');
}

// -----

switch ($mode) {
    case "extensions":
        echo directoryShow("extensions", $url, $xtn);
        break;
    case "groups":
        echo directoryShow("groups", $url, $xtn);
        break;
    case "personal":
        echo directoryShow("personal", $url, $xtn);
        break;
    default:
        echo directoryMenu($url, $xtn);
}

// -----

function directoryMenu($url, $xtn) {
    $menudata = array(
        array('Internal Phonebook', "$url?mode=extensions")
        ,array('Ring groups', "$url?mode=groups")
        ,array('Personal contacts', "$url?mode=personal&xtn=$xtn")
    );
    $xml = new SimpleXMLElement('<CiscoIPPhoneMenu/>');
    $xml -> addChild('Prompt', 'Select a directory');

    foreach ($menudata as $menuitem) {
        $menuItem = $xml -> addChild('MenuItem');
        $menuItem -> addChild('Name', $menuitem[0]);
        $menuItem -> addChild('URL', $menuitem[1]);
    }
    return ($xml->asXML());
}

// -----

function directoryShow($mode, $url, $xtn) {
    global $db;
    switch ($mode) {
        case "extensions":
            $sql = "SELECT name, extension FROM users WHERE name NOT LIKE '%FAX%' AND extension < 400
ORDER BY name";
            $title = "Internal directory";
            $prompt = "Select a name";
            break;
        case "groups":
            $sql = "SELECT description, grpnum FROM ringgroups WHERE grpnum < 599 ORDER BY description";
            $title = "Department directory";
            $prompt = "Select a group";
            break;
        case "personal":
            $sql = "
                SELECT
                    CONCAT(contactmanager_group_entries.displayname, ' (' , contactmanager_entry_numbers.
type,')') AS 'name'
                    ,contactmanager_entry_numbers.number AS 'extension'
                FROM contactmanager_groups
                LEFT JOIN contactmanager_group_entries ON contactmanager_groups.id =
contactmanager_group_entries.groupid
                LEFT JOIN contactmanager_entry_numbers ON contactmanager_group_entries.id =
contactmanager_entry_numbers.entryid
                WHERE contactmanager_groups.name = '$xtn'
                ORDER BY contactmanager_group_entries.displayname
            ";
            $title = "Personal contacts";
            $prompt = "Select a name";
            break;
        default:
            return;
    }

    $results = $db->getAll($sql, DB_FETCHMODE_ORDERED);
    $numrows = count($results);
    $sendoflist = false;
}

```

```

$xml = new SimpleXMLElement('<CiscoIPPhoneDirectory/>');
$xml -> addChild('Title', $title);
$xml -> addChild('Prompt', $prompt);

$page = array_key_exists('page', $_GET) ? $_GET['page'] : 0;
$count = $page * 32;

for ($row=$count; $row <= $count+31; $row++) {
    if (is_null($results[$row][0])) {
        $endoflist = true;
    } else {
        $endoflist = false;
        $directoryEntry = $xml -> addChild('DirectoryEntry');
        $directoryEntry -> addChild('Name', $results[$row][0]);
        $directoryEntry -> addChild('Telephone', $results[$row][1]);
    }
}
$softkey = $xml -> addChild('SoftKeyItem');
$softkey -> addChild('Name', 'Exit');
$softkey -> addChild('URL', 'SoftKey:Exit');
$softkey -> addChild('Position', '1');

$softkey = $xml -> addChild('SoftKeyItem');
$softkey -> addChild('Name', 'Dial');
$softkey -> addChild('URL', 'SoftKey:Dial');
$softkey -> addChild('Position', '2');

if ($page > 0) {
    $softkey = $xml -> addChild('SoftKeyItem');
    $softkey -> addChild('Name', 'Prev');
    $softkey -> addChild('URL', 'SoftKey:Exit');
    $softkey -> addChild('Position', '3');
}

if (!$endoflist) {
    $softkey = $xml -> addChild('SoftKeyItem');
    $softkey -> addChild('Name', "Next");
    $softkey -> addChild('URL', "$url?mode=$mode&page=" . ++$page);
    $softkey -> addChild('Position', '4');
}
return ($xml->asXML());
}
?>

```

Unsolved

Not everything is done. If you happen to know how to resolve these points, please do get in touch!

- The 8800 series of phones show an outbound call on the display as "101@server.domain.com" instead of "101" as it is shown on the 9900 phones when configured in the same manner. I have not yet been able to resolve this
- Recording a call using the linekey/sofkey does not make the recording available to the user in UCP
- The recording key cannot be used to pause/unpause the Asterisk recorder (if configured) so *1 is still needed for this.
- Setting the Bellcore style ring cadences in the sip headers works, but I've not found a way to instruct the phone to play a specific ringtone, so the selection of the tones is at the users discretion, rather than controlled by admin.

Thanks

Major props to Gareth for writing the cisco-usecallmanager patch in the first place. It goes without saying that without his work, this document wouldn't exist and the Cisco phone on my desk... wouldn't be there.