

Providing Great Debug

Many times a support technician or a developer will ask you to provide logs or "debug". You may think to yourself, "I have no idea what logs they want?" or "I've barely used this PBX because everything has been working fine up until this point so I have no idea how to help!"

That's the main reason this post was created.

There are a couple different types of debug:

- [FreePBX CLI Debug](#)
- [FreePBX Debug](#)
- [Browser Debug \(Console Log\)](#)
 - [CHROME:](#)
 - [SAFARI:](#)
 - [IE9:](#)
 - [FIREFOX:](#)
 - [OPERA:](#)
- [Asterisk Logs](#)
- [Asterisk Logs - Part II](#)
- [Backtraces \(Segfaults/Core Dumps/Asterisk Crashing\)](#)
- [Uploading Logs](#)

FreePBX CLI Debug

If you are running FreePBX 13 or higher and are executing a command through fwconsole you can use the `--verbose` option to output a stack trace that is especially helpful for developers to be able to fix problems

FreePBX Debug

If you think the error you are facing is a FreePBX GUI issue (white screen or blank page or something not working right in the GUI) then you will want to provide developers the debug log of the gui.

To do this simply run (from the SSH CLI)

```
amportal a debug
```

This will generate output similar to what is shown below:

```
[root@dahdidev1 ~]# amportal a debug
Fetching FreePBX settings with gen_amp_conf.php..
trying to run as user asterisk:
[FPBXDEBUGDISABLE] already set to [0]

==> /var/log/asterisk/freepbx_debug <==

==> /var/log/httpd/error_log <==
mv: cannot stat `/var/lib/asterisk/agi-bin/license-*.z1': No such file or
directory
[Wed Dec 03 00:05:09 2014] [error] [client 125.64.35.67] File does not
exist: /var/www/html/zc
[Wed Dec 03 01:14:26 2014] [error] [client 125.64.35.67] File does not
exist: /var/www/html/ucp/zc
[Wed Dec 03 01:14:26 2014] [error] [client 125.64.35.67] File does not
exist: /var/www/html/restapi
[Wed Dec 03 01:14:26 2014] [error] [client 125.64.35.67] File does not
exist: /tftpboot/zc
[Wed Dec 03 01:14:26 2014] [error] [client 125.64.35.67] File does not
```

```

exist: /var/www/html/aastra
[Wed Dec 03 01:14:26 2014] [error] [client 125.64.35.67] File does not
exist: /var/www/html/restapps
[Wed Dec 03 01:59:48 2014] [error] [client 94.102.49.82] script '/var/www
/html/recordings/index.php' not found or unable to stat
[Wed Dec 03 01:59:50 2014] [error] [client 94.102.49.82] File does not
exist: /var/www/html/admin/modules/admindashboard
[Wed Dec 03 01:59:51 2014] [error] [client 94.102.49.82] client denied by
server configuration: /var/www/html/admin/page.framework.php

==> /var/log/asterisk/freepbx.log <==
[2014-Dec-03 09:44:02] [PHP-WARNING] (/var/www/html/admin/modules/endpoint
/functions.inc/functions_epm_ucp.php:0) - No license for this product
(PBXact) - make sure zend_loader.license_path is properly configured in
your ini file!
[2014-Dec-03 09:44:02] [PHP-WARNING] (/var/www/html/admin/modules/endpoint
/functions.inc/functions_epm_ucp.php:0) - License check failed!
[2014-Dec-03 09:57:01] [WARNING] (libraries/modulefunctions.legacy.php:7)
- Deprecated Function _module_distro_id detected in /var/www/html/admin
/modules/sysadmin/functions.inc/general.php on line 107
[2014-Dec-03 09:57:01] [PHP-NOTICE] (/var/www/html/admin/modules/sysadmin
/functions.inc/license.php:53) - Undefined index: register
[2014-Dec-03 09:57:02] [WARNING] (libraries/modulefunctions.legacy.php:7)
- Deprecated Function _module_distro_id detected in /var/www/html/admin
/modules/sysadmin/functions.inc/general.php on line 107
[2014-Dec-03 09:57:02] [PHP-NOTICE] (/var/www/html/admin/modules/sysadmin
/functions.inc/license.php:53) - Undefined index: register
[2014-Dec-03 09:58:20] [INFO] (bin/freepbx_setting:39) - trying to run as
user asterisk:
[2014-Dec-03 09:58:20] [INFO] (bin/freepbx_setting:40) -
[2014-Dec-03 09:58:21] [INFO] (bin/freepbx_setting:61) - [FPBXDEBUGDISABLE]
already set to [0]
[2014-Dec-03 09:58:21] [INFO] (bin/freepbx_setting:62) -

```

Now the the initial interface has started up it's time to go to whatever GUI issue you are having and replicate it. Once you have done that copy and past what is shown to you in the output of this command and send it to a developer or support technician.

Browser Debug (Console Log)

CHROME:

1. Press either CTRL + SHIFT + J to open the "console" tab of the **Developer Tools**.

Alternative method:

1. Press either CTRL + SHIFT + I or F12 to open the **Developer Tools**.
2. Press ESC (or click on "Show console" in the bottom right corner) to slide the console up.

Note: In Chrome's dev tools, there is a "console" tab. However, a smaller "slide-up" console can be opened while any of the other tabs is active.

SAFARI:

1. Press CTRL + ALT + I to open the **Web Inspector**.
2. See Chrome's step 2. (Chrome and Safari have pretty much identical dev tools.)

Note: Step 1 only works if the "Show Develop menu in menu bar" check box in the Advanced tab of the Preferences menu is checked!

IE9:

1. Press F12 to open the developer tools.
2. Click the "console" tab.

FIREFOX:

1. Press CTRL + SHIFT + K to open the **Web console** (COMMAND + SHIFT + K on Macs).

or, if Firebug is installed:

1. Press F12 to open **Firebug**.
2. Click on the "console" tab.

OPERA:

1. Press CTRL + SHIFT + I to open **Dragonfly**.
2. Click on the "console" tab.

Asterisk Logs

If you think this is an Asterisk bug or a FreePBX dial plan generation bug then you need to get the Asterisk Logs

To get these logs you will need to run (from SSH CLI):

```
tail -f /var/log/asterisk/full
```

This will then give you output like:

```
root@freepbxdev1 ivr]# tail -f /var/log/asterisk/full
[2014-12-03 13:20:53] WARNING[6227] res_pjsip_mwi.c: AOR 1000 has no
configured mailboxes. MWI subscription failed
[2014-12-03 13:22:41] WARNING[6227] res_pjsip_mwi.c: AOR 1000 has no
configured mailboxes. MWI subscription failed
[2014-12-03 13:23:22] WARNING[6227] res_pjsip_mwi.c: AOR 1000 has no
configured mailboxes. MWI subscription failed
[2014-12-03 13:25:51] WARNING[6227] res_pjsip_mwi.c: AOR 1000 has no
configured mailboxes. MWI subscription failed
[2014-12-03 13:27:41] WARNING[6227] res_pjsip_mwi.c: AOR 1000 has no
configured mailboxes. MWI subscription failed
[2014-12-03 13:27:59] WARNING[6287] res_calendar_caldav.c: Unknown
response to CalDAV calendar myGoogleCal, request REPORT to /calendar/dav
/en.usa%23holiday%40group.v.calendar.google.com/events: SSL handshake
failed: SSL error: GnuTLS internal error.
[2014-12-03 13:28:20] WARNING[6227] res_pjsip_mwi.c: AOR 1000 has no
configured mailboxes. MWI subscription failed
[2014-12-03 13:30:49] WARNING[6227] res_pjsip_mwi.c: AOR 1000 has no
configured mailboxes. MWI subscription failed
[2014-12-03 13:32:41] WARNING[6227] res_pjsip_mwi.c: AOR 1000 has no
configured mailboxes. MWI subscription failed
[2014-12-03 13:33:18] WARNING[6227] res_pjsip_mwi.c: AOR 1000 has no
configured mailboxes. MWI subscription failed
```

Now make a phone call or cause the disruption to happen and copy and paste that output to a developer or support technician.

Asterisk Logs - Part II

Suppose you want a call trace from a specific call that has already happened, so it's too late to see it in the console live. First locate the call in the CDR, and get the uniqueid from the system column for the call in question:

Call Date	Recording	System	CallerID	Outbound CallerID	DID	App	Destination	Disposition	Duration	Userfield	Account
Tue, Feb 13 2018 8:59 AM		1518526777.67	"Yate Softphone" <5002>			Queue	700	ANSWERED	00:29		
Tue, Feb 13 2018 8:56 AM		1518526587.44	"Yate Softphone" <5002>			Queue	700	ANSWERED	00:10		
Thu, Feb 8 2018 12:54 PM		1518108843.46	"Yate Softphone" <5002>			Dial	5004	NO ANSWER	00:04		
Thu, Feb 8 2018 12:51 PM		1518108682.27	"Yate Softphone" <5002>			VoiceMail	5004	ANSWERED	00:02		
Thu, Feb 8 2018 12:50 PM		1518108624.26	"Yate Softphone" <5002>			VoiceMail	5004	ANSWERED	00:02		

Then grep the Asterisk full logs for that value:

```
[root@34693894 ~]# grep 1518526777.67 /var/log/asterisk/full*
/var/log/asterisk/full-20180214:[2018-02-13 08:59:37] VERBOSE[24184][C-00000001] pbx.c: Executing [s@macro-user-callerid:1] Set("SIP/5002-00000001", "TOUCH_MONITOR=1518526777.67") in new stack
/var/log/asterisk/full-20180214:[2018-02-13 08:59:37] VERBOSE[24184][C-00000001] pbx.c: Executing [700@from-internal:37] QueueLog("SIP/5002-00000001", "700,1518526777.67,NONE,DID,") in new stack
```

This will return a few lines, which will include the Asterisk CALL-ID (not to be confused with CallerID), the second number in the square brackets. It will also return a file name, as full logs are rotated daily and purged weekly. Perform a second grep on the CALL-ID and filename like:

```
[root@34693894 ~]# grep C-00000001 /var/log/asterisk/full-20180214
[2018-02-13 08:59:37] VERBOSE[29432][C-00000001] netsock2.c: Using SIP RTP TOS bits 184
[2018-02-13 08:59:37] VERBOSE[29432][C-00000001] netsock2.c: Using SIP RTP CoS mark 5
[2018-02-13 08:59:37] VERBOSE[24184][C-00000001] pbx.c: Executing [700@from-internal:1] Macro("SIP/5002-00000001", "user-callerid,") in new stack
[2018-02-13 08:59:37] VERBOSE[24184][C-00000001] pbx.c: Executing [s@macro-user-callerid:1] Set("SIP/5002-00000001", "TOUCH_MONITOR=1518526777.67") in new stack
[2018-02-13 08:59:37] VERBOSE[24184][C-00000001] pbx.c: Executing [s@macro-user-callerid:2] Set("SIP/5002-00000001", "AMPUSER=5002") in new stack
[2018-02-13 08:59:37] VERBOSE[24184][C-00000001] pbx.c: Executing [s@macro-user-callerid:3] GotoIf("SIP/5002-00000001", "0?report") in new stack

*snip*
```

This will return the full call trace from beginning to end of a specific call. You can then search through the trace and send to pastebin for a shareable URL:

```
[root@34693894 ~]# grep C-00000001 /var/log/asterisk/full-20180214 |
pastebin
title: Automatic Pastebin from Sangoma OS 7
name: root@34693894.deployments.pbxact.com
private: 1
expire: 0
http://pastebin.freepbx.org/view/<obfuscated>
```

If you wish to remove sensitive information before sending to pastebin you can use sed. Suppose you had a caller ID (or DID) of 3145551212 and you want to replace it with NPANXXXXXX, you can do so using this command:

```
grep C-00000001 /var/log/asterisk/full | sed 's/3145551212/NPANXXXXXX/g' |
pastebin
```

Using grep to get a call trace in this way, requires that every line in the trace contain the string that is being grepped. If you have SIP debug enabled, and you also need SIP trace information for debug purposes, it will be filtered out by the above grep commands.

Backtraces (Segfaults/Core Dumps/Asterisk Crashing)

You must run this against the version of Asterisk that is crashing. EG if you had Asterisk 13.17 and it core dumped and then you rolled back to 13.16 you can't run GDB against this as the debug tools will be on 13.16 and won't be able to correctly parse the core dump from 13.17

If you've gotten this far, you hit a rare occurrence with Asterisk in that it's randomly dying on you. The first thing to do is to determine if you are able to obtain a backtrace. To do this look for a file named core.* in your tmp directory. Please note the date and time of the core file you use as to not use an old core dump.

```
ls -l /tmp | grep core.
```

If you're running FreePBX Distro 7, you can simply install the 'sangoma-devel' package and then use the new 'debuginfo-install' command to download all the required debug packages. Note that you should replace the asterisk version in the command with the actual asterisk version that you are running ('asterisk11' or 'asterisk14', for example)

```
# Only for FreePBX Distro 7!
yum install -y sangoma-devel
debuginfo-install --enablerepo=centos7-debuginfo asterisk14
```

If you're running an older version of FreePBX Distro, you will need to install these packages manually, again replacing the asterisk version in this example with the actual version of asterisk you are running.

```
yum install pjproject-debuginfo asterisk13-debuginfo
```

If the asterisk version is 13.14.0 and 14.3.0 above then use the ast_coredumper tool to obtain the backtrace.

```
/var/lib/asterisk/scripts/ast_coredumper /tmp/[name of the core file]
```

If you want a backtrace from a running instance of Asterisk, possibly it's hung but has not crashed and not generated a core dump, you can do so using:

```
/var/lib/asterisk/scripts/ast_coredumper --running --tarball-coredumps
```

Otherwise you can then run the command below to obtain a backtrace.txt file which can be investigated and uploaded to the ticket

```
gdb -se "asterisk" -ex "bt full" -ex "thread apply all bt" --batch -c  
[name of the core file] > /tmp/backtrace.txt
```

If you are unable to locate the core file, we recommend you check out the [Asterisk Wiki on Obtaining Backtraces](#).

Uploading Logs

You can then upload your logs to pastebin.freepbx.org or by piping to the pastebin command in the distro:

```
cat /etc/asterisk/modules.conf | pastebin
```