

How to Configure FreeSWITCH To Use EZCNAM

This document was adapted with permission from Luis Daniel Lucio Quiroz

The screenshot shows the FusionPBX Dialplan configuration interface. At the top, it says "Dialplan" and "Dialplan include general settings." There are buttons for "BACK", "COPY", and "SAVE". The configuration form includes fields for Name (CNAM), Order (002), Number (empty), Domain (Global), Context (domain_name), Enabled (True), and Continue (True). There is also a Description field. Below the form is a table of dialplan actions:

Tag	Type	Data	Break	Inline	Group	Order	
condition	context	public			0	10	X
condition	destination_number	\d{11,}			0	20	X
action	set	call_direction=inbound			0	50	X
action	set	effective_caller_id_name=\${cidlookup(\${caller_id_number})}		true	0	100	X

FusionPBX CNAM Dialplan

This document describes how to use EZCNAM with your current FreeSWITCH installation, by using either the FusionPBX interface or directly via the FreeSWITCH dialplan.

CIDLookup is the FreeSWITCH module who handles the Name Caller automatic setting. This means that when an incoming call happens, mod_cidlookup will handle and set the correct name. This way, when the extension rings, you won't see only a number, but a name as well.

You will need the following:

- A FreeSWITCH installation
- Mod_cidlookup must be enabled:
https://freeswitch.org/confluence/display/FREESWITCH/mod_cidlookup
- An EZCNAM account. Take note of the API key associated with your EZCNAM account, because you'll need to use it below.

Configuring FreeSWITCH's CidLookup Module

Edit your cidlookup.conf.xml file and do the following:

1. Add or edit this line:

```
<param name="url"
value="https://api.ezcnam.com/v1?key=XXXXXXX&out=text&phone=${caller_id_number}" /
>
```

2. If you are not using the database capability, comment out the odbc-dsn line to avoid annoying warnings.

After that you need to load or reload the module: load mod_cidlookup

Using CidLookup

To activate it, you may need to add this line in your dialplan configuration:

```
<action
application="set" data="effective_caller_id_name=${cidlookup(${caller_id_number})}"/>
```

The next step is to set-up the dialplan.

Dialplan Configuration without FusionPBX

The dialplan needs to be processed only when an incoming call is received, and it needs to be the first one (or at least before the transfer). Here is an example configuration:

```
<extension name="cnam" continue="true">
  <condition field="context" expression="public">
    <action application="set" data="caller_id_name=${cidlookup(${caller_id_number})}"/>
  </condition>
</extension>
```

Note that this dialplan must be run in the public context.

Dialplan Configuration with FusionPBX

Best way to do this is by creating an incoming route with the following characteristics (as shown in the screenshot above):

- condition: is context public?
- condition: is caller number longer than 11 digits?
- action: set call_direction=inbound
- action: set effective_caller_id_name=\${cidlookup(\${caller_id_number})}

Side Effect

Using this method forces a synchronous call. This means that FreeSWITCH will wait until the CID lookup is finished. If for a reason there is a problem there and connection does not answer, you will be stuck.

Good Luck!