

How to debug incoming calls

Applies to VoipNow 3 and higher!

This knowledge base article presents several debug methods useful when confronting with problems regarding the incoming calls.

Check if the allowed IP/network addresses for incoming calls are set on the channel

The IP/network address the channel accepts calls from should be set to the IP or network address of the provider. This applies only to incoming calls made through that channel.

To do so, take the following next steps:

STEP 1: Click the **Channels** link available in the left navigation panel.

STEP 2: Select the desired channel from the list and click the **Edit channel** icon.

STEP 3: Verify the value specified for the **Accept calls from IPs/network** option.

If a domain name is added, then you will have to change it to the provider's IP/network address.

If you do not know this IP/network address, please ask the provider or use a ping command with the provider's domain name.

You can add multiple IP/network addresses if the provider uses multiple addresses to route calls through.

Make sure the channel registers to the provider

If the provider requires a registered channel, then the provider should give you the necessary credentials (username and password), as well as the channel mandatory settings (codecs, in-band signaling, etc.).

If you have set up a channel and it does not show up in the interface as **Registered**, then you need to make sure that:

- The given username and password were correctly specified.
- The **Do not register** option is not selected (active).
- You have not selected any unnecessary options (for example, the **Authorized username** field is usually left empty).
- The provider's IP address/domain name was correctly specified.

If the channel options appear to be set up correctly, yet the channel still fails to register, the best tool to find out the cause is the Asterisk Console:

1. Connect to the terminal as `root` using a SSH client (e.g. `Putty`).
2. Enter the following command to start the Asterisk Console:

```
asterisk -rvvvvvvvvvvvvvvvvvvvvvvvvvvvvv
```

3. Check if Asterisk is able to connect to the provider's remote host by running the command below. This will print a list with all the hosts currently connected to Asterisk. If your channel is on the list and its status is OK, then Asterisk is able to communicate with it.

```
sip show peers
```

4. Next, you need to find out what the provider returns for the REGISTER SIP packet the server is sending when trying to register the channel.

In order to gain access to the SIP traffic, you need to enable SIP debugging from the Asterisk Console using the command below:

```
sip set debug on
```

To disable debugging, use:

```
sip set debug off
```

- Now look for the `REGISTER` packets in the console window and the responses you get from the provider's IP address. This information can be very useful to the support team if you post a ticket regarding a channel issue. If the provider requires special channel settings, the response will most likely contain 4xx packets. In such a case, you need to check with the provider to see the settings needed for the channel to be able to register on their server.

Check if the calls actually reach the server

Let us assume the channel registers correctly to the provider or makes use of an IP-based authentication and you have already configured the DIDs assigned to you by the provider. If the extension still does not ring when you place a call, it is recommended that you use the Asterisk console to find out the problem. The call may not reach your server on account of three possible reasons:

1. You have not filled in the **Accept calls from IPs/network** field or, if you have, you did not enter the correct IP address.
2. The provider is not routing calls to your server. If so, you need to check this with your provider.
3. There is a network/connection problem either between the provider and your server (usually caused by a router/firewall), or between the phone you are calling from and the provider.

To solve the issue, you need to connect to the console as described above, enable SIP debugging and then try calling the number again.

If the call reaches the server, then you should be able to see a lot of SIP packets and messages in the Asterisk Console. If you can't see anything at all, it means the call cannot reach Asterisk. In this case, the problem might be caused by the Kamailio service. For helpful hints, you have to take a look at the Kamailio log:

1. Log on to the server console and increase the level of the debugging output:

```
kamctl fifo debug 3
```

2. At this point, you need to run the following command:

```
tail -f /var/log/kamailio/kamailio.log
```

Make a call and look at the output. If the DID or provider IP is not correctly set up on the channel, you should be able to see a message similar to the one below:

```
NOTICE: IP address 1.2.3.4 is not allowed to call DID number 12345678. Unauthorized request.
```

In this case, you must set up the IP to 1.2.3.4 on the same channel for which you have defined the DID as 12345678.

If you get SIP traffic from the provider during the call, this means the call reaches the server and the problem has a different cause (please see the next section).

Ensure you have correctly set up the DIDs

If the channel is fully functional, the calls reach the server, yet the extension you have assigned to the DID still fails to ring, then you should take the following steps:

STEP 1: Diagnose this issue. To do this, connect to the server and open an Asterisk Console. Then enable SIP debugging and call the DID. Upon calling the DID, the provider will send an `INVITE` packet to your server to establish a connection. Here is an `INVITE` SIP packet example:

```
INVITE sip:1234567@192.168.1.2:5050 SIP/2.0
Record-Route: <sip:192.168.1.2;lr=on;ftag=be3c3d58;did=bfb.0832fc56>
Via: SIP/2.0/UDP 192.168.1.4:49648;branch=z9hG4bK-d8754z-701948696f7f21-1---d8754z-;rport=49648
Max-Forwards: 69
Contact: <sip:0003*001@192.168.1.4:49648>
To: "1234567"<sip:1234567@192.168.1.2>
From: "some_user"<sip:0003*001@192.168.1.4>;tag=be3c3d58
Call-ID: OTVmYTJmMDQ1ZWU1YTExNmQ3ZjE2N2U0ZDJkYTY2MDM.
CSeq: 1 INVITE
Allow: INVITE, ACK, CANCEL, OPTIONS, BYE, REFER, NOTIFY, MESSAGE, SUBSCRIBE, INFO
Content-Type: application/sdp
```

STEP 2: Note the `INVITE` header (the first line). This is a proper header for a DID sent by the provider.

The provider needs to specify the DID getting the call in the `INVITE` header. This is usually done except when:

- Some providers do not send the DID in the `INVITE` header at all. They either send it in the `TO` header or in a custom header. If it gets sent in the `TO` header, VoipNow can use it. If the provider sends the DID in a custom header, you need to configure the channel to get the DID using the **Get DID from custom header** option available in the **Channel Settings** page.
- Some providers add a prefix to the DID or send a completely different number instead. In such cases, you need to modify the DID of the phone number in VoipNow to match the DID sent by the provider, otherwise the system will not be able to determine the extension the call should be routed to.

Related articles

- [Troubleshooting CallNow calls that drop](#)
- [How to debug 504 gateway timeout](#)

- [How to set up a SIP channel to interconnect with Skype forBusiness account](#)
- [How to detect abnormal traffic using Pike](#)
- [How to transfer incoming calls to another extension voicemail](#)