

GNU/Linux Administration - Support #183

Installing Prosody XMPP Server on Debian 7

08/21/2013 03:53 PM - Daniel Curtis

Status:	Closed	Start date:	08/21/2013
Priority:	Normal	Due date:	
Assignee:	Daniel Curtis	% Done:	100%
Category:	XMPP Server	Estimated time:	1.00 hour
Target version:		Spent time:	0.00 hour

Description

Install Prosody

- With the proper repository enabled, we're now ready to install the Prosody server. Use the following command:

```
apt-get install prosody
```

When apt finishes, the Prosody server will have been successfully installed, and will be ready for configuration. Prosody provides an init script that allows you to reload the configuration file, start, stop, or restart the XMPP server. Issue one of the following commands as appropriate:

- service prosody reload
- service prosody start
- service prosody stop
- service prosody restart

Configure Prosody Server

The configuration file for Prosody is located in `/etc/prosody/prosody.cfg.lua`, and is written in Lua syntax.

Note that in the Lua programming language, comments (lines that are ignored by the interpreter) are preceded by two hyphen characters (e.g. `--`). The default config has some basic instructions in Lua syntax, which can be helpful if you're unfamiliar with the language.

To allow Prosody to provide XMPP/jabber services for more than one domain, insert a line in the following form into the configuration file. This example defines three virtual hosts.

File excerpt:`/etc/prosody/prosody.cfg.lua`

```
Host "ducklington.org"  
Host "bucknell.net"  
Host "staff.bucknell.net"
```

Following a Host line there are generally a series of host-specific configuration options. If you want to set options for all hosts, add them below the "Host "" entry in your config file. For instance, to ensure that Prosody behaves like a proper Linux server daemon make sure that the "posix;" option is included in the "modules_enabled = { }" table.

File excerpt:`/etc/prosody/prosody.cfg.lua`

```
modules_enabled = {  
-- [...]  
"posix";  
-- [...]  
}
```

Note that there should be a number of global modules included in this table to provide basic functionality.

To disable a host without removing it from your configuration file, add the following line to its section of the file:

File excerpt:/etc/prosody/prosody.cfg.lua

```
enabled = false
```

To specify administrators for your server, add a line in the following format to your prosody.cfg.lua file.

File excerpt:/etc/prosody/prosody.cfg.lua

```
admins = { "admin1@example.com", "admin2@example.com" }
```

To add server-wide administrators, add the admins line to the "Hosts "" section. To grant specific users more granular control to administer particular hosts, you can add an admins line, or more properly tables in Lua, to specific hosts.

If you need to enable the legacy SSL/TLS support, add the following line specifying the port on which the server should listen for these connections.

File excerpt:/etc/prosody/prosody.cfg.lua

```
legacy_ssl_ports = { 5223 }
```

Do not forget to reload the configuration for the Prosody server after making any changes to your /etc/prosody/prosody.cfg.lua file, by issuing the following command:

```
service prosody reload
```

XMPP Federation and DNS

To ensure that your Prosody instance will federate properly with the rest of the XMPP network, particularly with Google's "GTalk" service (i.e. the "@gmail.com" chat tool,) we must set the SRV records for the domain to point to the server where the Prosody instance is running. We need three records, which can be created in the DNS Management tool of your choice:

```
Service: _xmpp-server Protocol: TCP Port: 5269
```

```
Service: _xmpp-client Protocol: TCP Port: 5222
```

```
Service: _jabber Protocol: TCP Port: 5269
```

The "target" of the SRV record should point to the publicly routable hostname for that machine (e.g. "squire.bucknell.net"). The priority and weight should both be set to 0.

Enabling Components

In the XMPP world, many services are provided in components, which allows for greater ease of customization within a basic framework. A common example of this is the MUC or multi-user chat functionality. To enable MUC services in Prosody you need to add a line like the following to your /etc/prosody/prosody.cfg.lua file.

File excerpt:/etc/prosody/prosody.cfg.lua

```
Component "conference.bucknell.net" "muc"
```

In this example, **conference.bucknell.net** is the domain where the MUC rooms are located, and will require an "DNS A record," that points to the IP Address where the Prosody instance is running. MUCs will be identified as JIDs (Jabber IDs) at this hostname, so for instance the "rabbits" MUC hosted by this server would be located at *rabbits@conference.bucknell.net*.

MUC, in contrast to many other common components in the XMPP world, is provided internally by Prosody. Other components, like transports to other services, run on an external interface. Each external component has its own host name, and provides a secret key which allows the central server to authenticate to it. See the following **aim.bucknell.net** component as an example.

File excerpt:/etc/prosody/prosody.cfg.lua

```
Component "aim.bucknell.net"
```

```
component_secret = "mysecretcomponentpassword"
```

Note that external components will need to be installed and configured independently of Prosody.

Typically, Prosody listens for connections from components on the localhost interface (i.e. on the 127.0.0.1 interface;). If you're connected to external resources that are running on an alternate interface, specify the following variables as appropriate in the "Host "" section of the file config file.

File excerpt:/etc/prosody/prosody.cfg.lua

```
Host ""  
  
component_interface = "192.168.0.10"  
component_ports = { 8888, 8887 }
```

Using prosodyctl

The XMPP protocol supports "in-band" registration, where users can register for accounts with your server via the XMPP interface. However, this is often an undesirable function as it doesn't permit the server administrator the ability to moderate the creation of new accounts and can lead to spam-related problems. As a result, Prosody has this functionality disabled by default. While you can enable in-band registration, we recommend using the prosodyctl interface at the terminal prompt.

If you're familiar with the ejabberdctl interface from ejabberd, prosodyctl mimics its counterpart as much as possible.

- To use prosodyctl to register a user, in this case *lollipop@*ducklington.org**, issue the following command:

```
prosodyctl adduser lollipop@ducklington.org
```

- To set the password for this account, issue the following command and enter the password as requested:

```
prosodyctl passwd lollipop@ducklington.org
```

- To remove this user, issue the following command:

```
prosodyctl deluser lollipop@ducklington.org
```

- Additionally, prosodyctl can provide a report on the status of the server in response to the following command:

```
prosodyctl status
```

Note that all of the prosodyctl commands require root privileges, unless you've logged in as the same user that Prosody runs under (not recommended).

Resources

You may wish to consult the following resources for additional information on this topic. While these are provided in the hope that they will be useful, please note that we cannot vouch for the accuracy or timeliness of externally hosted materials.

- [The official Prosody server website](#)
- [XMPP Standards Foundation](#)
- [XMPP Client Software](#)

Related issues:

Copied to FreeBSD Administration - Support #535: Install Prosody XMPP Server ...

Closed

08/21/2013

History

#1 - 01/18/2015 04:27 PM - Daniel Curtis

- Copied to Support #535: Install Prosody XMPP Server on FreeBSD added

#2 - 02/16/2015 02:35 PM - Daniel Curtis

- *Description updated*

#3 - 02/16/2015 02:36 PM - Daniel Curtis

- *Project changed from 42 to GNU/Linux Administration*

- *Category set to XMPP Server*