**Ejabberd Configuration & Installation in Ubuntu**

**Binary Method Installation :**

Step 1:

First download the package from below link :

**wget** [**https://www.process-one.net/downloads/ejabberd/15.04/ejabberd-15.04-linux-x86\_64-installer.run**](https://www.process-one.net/downloads/ejabberd/15.04/ejabberd-15.04-linux-x86_64-installer.run)

2. Change the permission of file, make it executable :

**Chmod +x ejabberd-15.04-linux-x86\_64-installer.run**

3. Now install :

**./ejabberd-15.09-linux-x86\_64-installer.run**

Follow the instructions, it will ask for path where to install, set admin password etc.

By default it will install in **/opt/ejabberd-15.04**

**Your installation is complete. Now you are ready to start service :**

Go to : **cd /opt/ejabberd-15.04/bin**

Start service -  **./start or ./ejabberdctl start**

Stop Service**- ./stop**

Check errors-  **./ejabberdctl live**

**Configuration :**

Go to : **cd /opt/ejabberd-15.04/conf/**

conf file:  **vim ejabberd.yml**

**You can follow below link for c :** [**https://www.process-one.net/docs/ejabberd/guide\_en.html#htoc69**](https://www.process-one.net/docs/ejabberd/guide_en.html#htoc69)

**You have to search mod\_vcard in ejabberd.yml file or uncomment these lines/add below settings according to your requirement :**

**vim ejabberd.yml**

for search press : **/mod\_vcard**

Settings for **mod\_vcard :**

**mod\_vcard:  
 search: true  
 matches: 20  
 allow\_return\_all: true  
 search\_all\_hosts: false**

Settings for **mod\_muc :**

**mod\_muc:  
 access: muc\_access  
 access\_create: muc\_admin  
 access\_admin: muc\_admin**

For more example or settings you can follow the above link.

**Install XMLRPC-Erlang**

**wget http://www.ejabberd.im/files/contributions/xmlrpc-1.13-ipr2.tgz**

**tar -xzvf xmlrpc-1.13-ipr2.tgz**

**cd xmlrpc-1.13/src  
make**

**XMLRPC :**

**-**

**port: 4560**

**module: ejabberd\_xmlrpc**

**-**

**Always be careful about hyphen(-) it must be on place, otherwise you will get syntax error.**

write & quit the vi editor ejabbberd.yml file :

press ESC then **:wq**

**and start the service.**

You can also check ports are open or not on server by using below command :

**netstat -tulnp | grep 4560**

**tcp 0 0 0.0.0.0:4560 0.0.0.0:\* LISTEN 10623/beam**

**Ejabberd Installation From Source Method :**

To compile ejabberd you need to install below packages first :

1. apt-get install

* GNU Make.
* GCC.
* Libexpat 1.95 or Libxpat1-dev
* Libyaml 0.1.4 or Libyaml-dev
* Zlib 1.2.3 or higher, for Stream Compression support (XEP-0138). Optional.
* OpenSSL 1.0.0 or higher, for STARTTLS, SASL and SSL encryption.
* automake
* make
* libssl-dev
* Erlang/OTP 17.1

Installation of Erlang :

wget <https://packages.erlang-solutions.com/erlang-solutions_1.0_all.deb>

dpkg -i erlang-solutions\_1.0\_all.deb

apt-get update

apt-get install erlang

apt-get install build-essential checkinstall

**wget https://github.com/processone/ejabberd**

2. cd ejabberd-master/

3. ./autogen.sh

ignore the locale setting error

you can check the above cmd run successfully or not by using

4.echo $?

0

if result shows 0 than its ok, otherwise need to check the error

5. ./configure

6. make

7. make install

now start the service

ejabberdctl start/stop

* create a admin account using below command :

ejabberdctl register admin localhost @ppst3r@123

User admin@localhost successfully registered

**URL :** [**http://IP:5280/admin**](http://52.64.186.81:5280/admin)

* You also need to mention or allow this admin account in configuration file.

**vim /etc/ejabberd/ejabberd.yml**

**###' ACCESS CONTROL LISTS**

**acl:**

**##**

**## The 'admin' ACL grants administrative privileges to XMPP accounts.**

**## You can put here as many accounts as you want.**

**##**

**admin:**

**user:**

**- "admin": "localhost"**

**## - "ermine": "example.org"**

**Allow Max Offline messages :**

**## Maximum number of offline messages that users can have:**

**max\_user\_offline\_messages:**

**admin: 5000**

**all: 5000**

**## By default the frequency of account registrations from the same IP**

**## is limited to 1 account every 10 minutes. To disable, specify: infinity**

**registration\_timeout: 600**

or make changes in configuration file according to your requirements

1. **Configuration file of HomeHero Server :**

###

###' ejabberd configuration file

###

###

### The parameters used in this configuration file are explained in more detail

### in the ejabberd Installation and Operation Guide.

### Please consult the Guide in case of doubts, it is included with

### your copy of ejabberd, and is also available online at

### http://www.process-one.net/en/ejabberd/docs/

### The configuration file is written in YAML.

### Refer to http://en.wikipedia.org/wiki/YAML for the brief description.

### However, ejabberd treats different literals as different types:

###

### - unquoted or single-quoted strings. They are called "atoms".

### Example: dog, 'Jupiter', '3.14159', YELLOW

###

### - numeric literals. Example: 3, -45.0, .0

###

### - quoted or folded strings.

### Examples of quoted string: "Lizzard", "orange".

### Example of folded string:

### > Art thou not Romeo,

### and a Montague?

###. =======

###' LOGGING

##

## loglevel: Verbosity of log files generated by ejabberd.

## 0: No ejabberd log at all (not recommended)

## 1: Critical

## 2: Error

## 3: Warning

## 4: Info

## 5: Debug

##

loglevel: 5

##

## rotation: Describe how to rotate logs. Either size and/or date can trigger

## log rotation. Setting count to N keeps N rotated logs. Setting count to 0

## does not disable rotation, it instead rotates the file and keeps no previous

## versions around. Setting size to X rotate log when it reaches X bytes.

## To disable rotation set the size to 0 and the date to ""

## Date syntax is taken from the syntax newsyslog uses in newsyslog.conf.

## Some examples:

## $D0 rotate every night at midnight

## $D23 rotate every day at 23:00 hr

## $W0D23 rotate every week on Sunday at 23:00 hr

## $W5D16 rotate every week on Friday at 16:00 hr

## $M1D0 rotate on the first day of every month at midnight

## $M5D6 rotate on every 5th day of the month at 6:00 hr

##

log\_rotate\_size: 10485760

log\_rotate\_date: ""

log\_rotate\_count: 1

##

## overload protection: If you want to limit the number of messages per second

## allowed from error\_logger, which is a good idea if you want to avoid a flood

## of messages when system is overloaded, you can set a limit.

## 100 is ejabberd's default.

log\_rate\_limit: 100

##

## watchdog\_admins: Only useful for developers: if an ejabberd process

## consumes a lot of memory, send live notifications to these XMPP

## accounts.

##

## watchdog\_admins:

## - "bob@example.com"

###. ===============

###' NODE PARAMETERS

##

## net\_ticktime: Specifies net\_kernel tick time in seconds. This options must have

## identical value on all nodes, and in most cases shouldn't be changed at all from

## default value.

##

## net\_ticktime: 60

###. ================

###' SERVED HOSTNAMES

##

## hosts: Domains served by ejabberd.

## You can define one or several, for example:

## hosts:

## - "example.net"

## - "example.com"

## - "example.org"

##

hosts:

- "localhost"

##

## route\_subdomains: Delegate subdomains to other XMPP servers.

## For example, if this ejabberd serves example.org and you want

## to allow communication with an XMPP server called im.example.org.

##

## route\_subdomains: s2s

###. ===============

###' LISTENING PORTS

##

## listen: The ports ejabberd will listen on, which service each is handled

## by and what options to start it with.

##

listen:

-

port: 5222

module: ejabberd\_c2s

##

## If TLS is compiled in and you installed a SSL

## certificate, specify the full path to the

## file and uncomment these lines:

##

## certfile: "/path/to/ssl.pem"

## starttls: true

##

## To enforce TLS encryption for client connections,

## use this instead of the "starttls" option:

##

## starttls\_required: true

##

## Custom OpenSSL options

##

## protocol\_options:

## - "no\_sslv3"

## - "no\_tlsv1"

max\_stanza\_size: 65536

shaper: c2s\_shaper

access: c2s

-

port: 5269

module: ejabberd\_s2s\_in

##

## ejabberd\_service: Interact with external components (transports, ...)

##

## -

## port: 8888

## module: ejabberd\_service

## access: all

## shaper\_rule: fast

## ip: "127.0.0.1"

## hosts:

## "icq.example.org":

## password: "secret"

## "sms.example.org":

## password: "secret"

##

## ejabberd\_stun: Handles STUN Binding requests

##

## -

## port: 3478

## transport: udp

## module: ejabberd\_stun

##

## To handle XML-RPC requests that provide admin credentials:

##

-

port: 4560

module: ejabberd\_xmlrpc

access\_commands: {}

-

port: 5280

module: ejabberd\_http

request\_handlers:

"/websocket": ejabberd\_http\_ws

## "/pub/archive": mod\_http\_fileserver

web\_admin: true

http\_bind: true

## register: true

captcha: true

###. ==================

###' S2S GLOBAL OPTIONS

##

## s2s\_use\_starttls: Enable STARTTLS + Dialback for S2S connections.

## Allowed values are: false optional required required\_trusted

## You must specify a certificate file.

##

## s2s\_use\_starttls: optional

##

## s2s\_certfile: Specify a certificate file.

##

## s2s\_certfile: "/path/to/ssl.pem"

## Custom OpenSSL options

##

## s2s\_protocol\_options:

## - "no\_sslv3"

## - "no\_tlsv1"

##

## domain\_certfile: Specify a different certificate for each served hostname.

##

## host\_config:

## "example.org":

## domain\_certfile: "/path/to/example\_org.pem"

## "example.com":

## domain\_certfile: "/path/to/example\_com.pem"

##

## S2S whitelist or blacklist

##

## Default s2s policy for undefined hosts.

##

## s2s\_access: s2s

##

## Outgoing S2S options

##

## Preferred address families (which to try first) and connect timeout

## in milliseconds.

##

## outgoing\_s2s\_families:

## - ipv4

## - ipv6

## outgoing\_s2s\_timeout: 10000

###. ==============

###' AUTHENTICATION

##

## auth\_method: Method used to authenticate the users.

## The default method is the internal.

## If you want to use a different method,

## comment this line and enable the correct ones.

##

auth\_method: internal

##

## Store the plain passwords or hashed for SCRAM:

## auth\_password\_format: plain

## auth\_password\_format: scram

##

## Define the FQDN if ejabberd doesn't detect it:

## fqdn: "server3.example.com"

##

## Authentication using external script

## Make sure the script is executable by ejabberd.

##

## auth\_method: external

## extauth\_program: "/path/to/authentication/script"

##

## Authentication using ODBC

## Remember to setup a database in the next section.

##

## auth\_method: odbc

##

## Authentication using PAM

##

## auth\_method: pam

## pam\_service: "pamservicename"

##

## Authentication using LDAP

##

## auth\_method: ldap

##

## List of LDAP servers:

## ldap\_servers:

## - "localhost"

##

## Encryption of connection to LDAP servers:

## ldap\_encrypt: none

## ldap\_encrypt: tls

##

## Port to connect to on LDAP servers:

## ldap\_port: 389

## ldap\_port: 636

##

## LDAP manager:

## ldap\_rootdn: "dc=example,dc=com"

##

## Password of LDAP manager:

## ldap\_password: "\*\*\*\*\*\*"

##

## Search base of LDAP directory:

## ldap\_base: "dc=example,dc=com"

##

## LDAP attribute that holds user ID:

## ldap\_uids:

## - "mail": "%u@mail.example.org"

##

## LDAP filter:

## ldap\_filter: "(objectClass=shadowAccount)"

##

## Anonymous login support:

## auth\_method: anonymous

## anonymous\_protocol: sasl\_anon | login\_anon | both

## allow\_multiple\_connections: true | false

##

## host\_config:

## "public.example.org":

## auth\_method: anonymous

## allow\_multiple\_connections: false

## anonymous\_protocol: sasl\_anon

##

## To use both anonymous and internal authentication:

##

## host\_config:

## "public.example.org":

## auth\_method:

## - internal

## - anonymous

###. ==============

###' DATABASE SETUP

## ejabberd by default uses the internal Mnesia database,

## so you do not necessarily need this section.

## This section provides configuration examples in case

## you want to use other database backends.

## Please consult the ejabberd Guide for details on database creation.

##

## MySQL server:

##

## odbc\_type: mysql

## odbc\_server: "server"

## odbc\_database: "database"

## odbc\_username: "username"

## odbc\_password: "password"

##

## If you want to specify the port:

## odbc\_port: 1234

##

## PostgreSQL server:

##

## odbc\_type: pgsql

## odbc\_server: "server"

## odbc\_database: "database"

## odbc\_username: "username"

## odbc\_password: "password"

##

## If you want to specify the port:

## odbc\_port: 1234

##

## If you use PostgreSQL, have a large database, and need a

## faster but inexact replacement for "select count(\*) from users"

##

## pgsql\_users\_number\_estimate: true

##

## SQLite:

##

## odbc\_type: sqlite

## odbc\_database: "/path/to/database.db"

##

## ODBC compatible or MSSQL server:

##

## odbc\_type: odbc

## odbc\_server: "DSN=ejabberd;UID=ejabberd;PWD=ejabberd"

##

## Number of connections to open to the database for each virtual host

##

## odbc\_pool\_size: 10

##

## Interval to make a dummy SQL request to keep the connections to the

## database alive. Specify in seconds: for example 28800 means 8 hours

##

## odbc\_keepalive\_interval: undefined

###. ===============

###' TRAFFIC SHAPERS

shaper:

##

## The "normal" shaper limits traffic speed to 1000 B/s

##

normal: 1000

##

## The "fast" shaper limits traffic speed to 50000 B/s

##

fast: 50000

##

## This option specifies the maximum number of elements in the queue

## of the FSM. Refer to the documentation for details.

##

max\_fsm\_queue: 1000

###. ====================

###' ACCESS CONTROL LISTS

acl:

##

## The 'admin' ACL grants administrative privileges to XMPP accounts.

## You can put here as many accounts as you want.

##

admin:

user:

- "admin": "localhost"

## - "ermine": "example.org"

##

## Blocked users

##

## blocked:

## user:

## - "baduser": "example.org"

## - "test"

## Local users: don't modify this.

##

local:

user\_regexp: ""

##

## More examples of ACLs

##

## jabberorg:

## server:

## - "jabber.org"

## aleksey:

## user:

## - "aleksey": "jabber.ru"

## test:

## user\_regexp: "^test"

## user\_glob: "test\*"

##

## Loopback network

##

loopback:

ip:

- "127.0.0.0/8"

##

## Bad XMPP servers

##

## bad\_servers:

## server:

## - "xmpp.zombie.org"

## - "xmpp.spam.com"

##

## Define specific ACLs in a virtual host.

##

## host\_config:

## "localhost":

## acl:

## admin:

## user:

## - "bob-local": "localhost"

###. ============

###' ACCESS RULES

access:

## Maximum number of simultaneous sessions allowed for a single user:

max\_user\_sessions:

all: 10

## Maximum number of offline messages that users can have:

max\_user\_offline\_messages:

admin: 5000

all: 5000

## This rule allows access only for local users:

local:

local: allow

## Only non-blocked users can use c2s connections:

c2s:

blocked: deny

all: allow

## For C2S connections, all users except admins use the "normal" shaper

c2s\_shaper:

admin: none

all: normal

## All S2S connections use the "fast" shaper

s2s\_shaper:

all: fast

## Only admins can send announcement messages:

announce:

admin: allow

## Only admins can use the configuration interface:

configure:

admin: allow

## Admins of this server are also admins of the MUC service:

muc\_admin:

admin: allow

## Only accounts of the local ejabberd server can create rooms:

muc\_create:

local: allow

## All users are allowed to use the MUC service:

muc:

all: allow

## Only accounts on the local ejabberd server can create Pubsub nodes:

pubsub\_createnode:

local: allow

## In-band registration allows registration of any possible username.

## To disable in-band registration, replace 'allow' with 'deny'.

register:

all: allow

## Only allow to register from localhost

trusted\_network:

loopback: allow

## Do not establish S2S connections with bad servers

## s2s:

## bad\_servers: deny

## all: allow

## By default the frequency of account registrations from the same IP

## is limited to 1 account every 10 minutes. To disable, specify: infinity

registration\_timeout: infinity

##

## Define specific Access Rules in a virtual host.

##

## host\_config:

## "localhost":

## access:

## c2s:

## admin: allow

## all: deny

## register:

## all: deny

###. ================

###' DEFAULT LANGUAGE

##

## language: Default language used for server messages.

##

language: "en"

##

## Set a different default language in a virtual host.

##

## host\_config:

## "localhost":

## language: "ru"

###. =======

###' CAPTCHA

##

## Full path to a script that generates the image.

##

## captcha\_cmd: "/lib/ejabberd/priv/bin/captcha.sh"

##

## Host for the URL and port where ejabberd listens for CAPTCHA requests.

##

## captcha\_host: "example.org:5280"

##

## Limit CAPTCHA calls per minute for JID/IP to avoid DoS.

##

## captcha\_limit: 5

###. =======

###' MODULES

##

## Modules enabled in all ejabberd virtual hosts.

##

modules:

mod\_adhoc: {}

mod\_mam:

access\_max\_user\_messages: 0

default\_page\_size: 25

max\_page\_size: 100

request\_activates\_archiving: true

iqdisc: parallel

default: always

## mod\_admin\_extra: {}

mod\_announce: # recommends mod\_adhoc

access: announce

mod\_blocking: {} # requires mod\_privacy

mod\_caps: {}

mod\_carboncopy: {}

mod\_client\_state: {}

mod\_configure: {} # requires mod\_adhoc

mod\_disco: {}

## mod\_echo: {}

mod\_irc: {}

mod\_http\_bind: {}

## mod\_http\_fileserver:

## docroot: "/var/www"

## accesslog: "/var/log/ejabberd/access.log"

mod\_last: {}

mod\_muc:

## host: "conference.@HOST@"

host: "conference.localhost"

access: muc

history\_size: 0

access\_create: muc\_create

access\_persistent: muc

access\_admin: muc\_admin

mam: true

default\_room\_options:

mam: true

persistent: true

## mod\_muc\_log: {}

## mod\_multicast: {}

mod\_offline:

access\_max\_user\_messages: max\_user\_offline\_messages

mod\_ping: {}

## mod\_pres\_counter:

## count: 5

## interval: 60

mod\_privacy: {}

mod\_private: {}

## mod\_proxy65: {}

mod\_pubsub:

access\_createnode: pubsub\_createnode

## reduces resource comsumption, but XEP incompliant

ignore\_pep\_from\_offline: true

## XEP compliant, but increases resource comsumption

## ignore\_pep\_from\_offline: false

last\_item\_cache: false

plugins:

- "flat"

- "hometree"

- "pep" # pep requires mod\_caps

mod\_register:

##

## Protect In-Band account registrations with CAPTCHA.

##

## captcha\_protected: true

##

## Set the minimum informational entropy for passwords.

##

## password\_strength: 32

##

## After successful registration, the user receives

## a message with this subject and body.

##

welcome\_message:

subject: "Welcome!"

body: |-

Hi.

Welcome to this XMPP server.

##

## When a user registers, send a notification to

## these XMPP accounts.

##

## registration\_watchers:

## - "admin1@example.org"

##

## Only clients in the server machine can register accounts

##

ip\_access: trusted\_network

##

## Local c2s or remote s2s users cannot register accounts

##

## access\_from: deny

access: register

mod\_roster: {}

mod\_shared\_roster: {}

mod\_stats: {}

mod\_time: {}

mod\_vcard:

search: false

mod\_version: {}

##

## Enable modules with custom options in a specific virtual host

##

## host\_config:

## "localhost":

## modules:

## mod\_echo:

## host: "mirror.localhost"

##

## Enable modules management via ejabberdctl for installation and

## uninstallation of public/private contributed modules

## (enabled by default)

##

allow\_contrib\_modules: true

###.

###'

### Local Variables:

### mode: yaml

### End:

### vim: set filetype=yaml tabstop=8 foldmarker=###',###. foldmethod=marker:

2.**Configuration file of bendr server :**

**###**

**### ejabberd configuration file**

**###**

**###**

**### The parameters used in this configuration file are explained in more detail**

**### in the ejabberd Installation and Operation Guide.**

**### Please consult the Guide in case of doubts, it is included with**

**### your copy of ejabberd, and is also available online at**

**### http://www.process-one.net/en/ejabberd/docs/**

**### The configuration file is written in YAML.**

**### Refer to http://en.wikipedia.org/wiki/YAML for the brief description.**

**### However, ejabberd treats different literals as different types:**

**###**

**### - unquoted or single-quoted strings. They are called "atoms".**

**### Example: dog, 'Jupiter', '3.14159', YELLOW**

**###**

**### - numeric literals. Example: 3, -45.0, .0**

**###**

**### - quoted or folded strings.**

**### Examples of quoted string: "Lizzard", "orange".**

**### Example of folded string:**

**### > Art thou not Romeo,**

**### and a Montague?**

**### =======**

**### LOGGING**

**##**

**## loglevel: Verbosity of log files generated by ejabberd.**

**## 0: No ejabberd log at all (not recommended)**

**## 1: Critical**

**## 2: Error**

**## 3: Warning**

**## 4: Info**

**## 5: Debug**

**##**

**loglevel: 4**

**##**

**## rotation: Describe how to rotate logs. Either size and/or date can trigger**

**## log rotation. Setting count to N keeps N rotated logs. Setting count to 0**

**## does not disable rotation, it instead rotates the file and keeps no previous**

**## versions around. Setting size to X rotate log when it reaches X bytes.**

**## To disable rotation set the size to 0 and the date to ""**

**## Date syntax is taken from the syntax newsyslog uses in newsyslog.conf.**

**## Some examples:**

**## $D0 rotate every night at midnight**

**## $D23 rotate every day at 23:00 hr**

**## $W0D23 rotate every week on Sunday at 23:00 hr**

**## $W5D16 rotate every week on Friday at 16:00 hr**

**## $M1D0 rotate on the first day of every month at midnight**

**## $M5D6 rotate on every 5th day of the month at 6:00 hr**

**##**

**log\_rotate\_size: 10485760**

**log\_rotate\_date: ""**

**log\_rotate\_count: 1**

**##**

**## overload protection: If you want to limit the number of messages per second**

**## allowed from error\_logger, which is a good idea if you want to avoid a flood**

**## of messages when system is overloaded, you can set a limit.**

**## 100 is ejabberd's default.**

**log\_rate\_limit: 100**

**##**

**## watchdog\_admins: Only useful for developers: if an ejabberd process**

**## consumes a lot of memory, send live notifications to these XMPP**

**## accounts.**

**##**

**## watchdog\_admins:**

**## - "admin@ip-172-31-31-239.us-west-2.compute.internal"**

**### ================**

**### SERVED HOSTNAMES**

**##**

**## hosts: Domains served by ejabberd.**

**## You can define one or several, for example:**

**## hosts:**

**## - "example.net"**

**## - "example.com"**

**## - "example.org"**

**##**

**hosts:**

**- "ip-172-31-31-239.us-west-2.compute.internal"**

**##**

**## route\_subdomains: Delegate subdomains to other XMPP servers.**

**## For example, if this ejabberd serves example.org and you want**

**## to allow communication with an XMPP server called im.example.org.**

**##**

**## route\_subdomains: s2s**

**### ===============**

**### LISTENING PORTS**

**##**

**## listen: The ports ejabberd will listen on, which service each is handled**

**## by and what options to start it with.**

**##**

**listen:**

**-**

**port: 5222**

**module: ejabberd\_c2s**

**max\_stanza\_size: 65536**

**shaper: c2s\_shaper**

**access: c2s**

**starttls: true**

**certfile: "/opt/ejabberd-15.04/conf/server.pem"**

**## Custom OpenSSL options**

**##**

**## protocol\_options:**

**## - "no\_sslv3"**

**## - "no\_tlsv1"**

**-**

**port: 5269**

**module: ejabberd\_s2s\_in**

**max\_stanza\_size: 131072**

**shaper: s2s\_shaper**

**##**

**## ejabberd\_service: Interact with external components (transports, ...)**

**##**

**## -**

**## port: 8888**

**## module: ejabberd\_service**

**## access: all**

**## shaper\_rule: fast**

**## ip: "127.0.0.1"**

**## hosts:**

**## "icq.example.org":**

**## password: "secret"**

**## "sms.example.org":**

**## password: "secret"**

**##**

**## ejabberd\_stun: Handles STUN Binding requests**

**##**

**## -**

**## port: 3478**

**## transport: udp**

**## module: ejabberd\_stun**

**##**

**## To handle XML-RPC requests that provide admin credentials:**

**##**

**-**

**port: 4560**

**module: ejabberd\_xmlrpc**

**-**

**port: 5280**

**module: ejabberd\_http**

**request\_handlers:**

**"/websocket": ejabberd\_http\_ws**

**# "/pub/archive": mod\_http\_fileserver**

**web\_admin: true**

**http\_poll: false**

**http\_bind: true**

**## register: true**

**captcha: false**

**##**

**## s2s\_use\_starttls: Enable STARTTLS + Dialback for S2S connections.**

**## Allowed values are: false optional required required\_trusted**

**## You must specify a certificate file.**

**##**

**## s2s\_use\_starttls: optional**

**##**

**## s2s\_certfile: Specify a certificate file.**

**##**

**## s2s\_certfile: "/opt/ejabberd-15.04/conf/server.pem"**

**## Custom OpenSSL options**

**##**

**## s2s\_protocol\_options:**

**## - "no\_sslv3"**

**## - "no\_tlsv1"**

**##**

**## domain\_certfile: Specify a different certificate for each served hostname.**

**##**

**## host\_config:**

**## "example.org":**

**## domain\_certfile: "/opt/ejabberd-15.04/conf/example\_org.pem"**

**## "example.com":**

**## domain\_certfile: "/opt/ejabberd-15.04/conf/example\_com.pem"**

**##**

**## S2S whitelist or blacklist**

**##**

**## Default s2s policy for undefined hosts.**

**##**

**## s2s\_policy: s2s**

**##**

**## Outgoing S2S options**

**##**

**## Preferred address families (which to try first) and connect timeout**

**## in milliseconds.**

**##**

**## outgoing\_s2s\_families:**

**## - ipv4**

**## - ipv6**

**## outgoing\_s2s\_timeout: 10000**

**### ==============**

**### AUTHENTICATION**

**##**

**## auth\_method: Method used to authenticate the users.**

**## The default method is the internal.**

**## If you want to use a different method,**

**## comment this line and enable the correct ones.**

**##**

**auth\_method: internal**

**##**

**## Store the plain passwords or hashed for SCRAM:**

**## auth\_password\_format: plain**

**## auth\_password\_format: scram**

**##**

**## Define the FQDN if ejabberd doesn't detect it:**

**## fqdn: "server3.example.com"**

**##**

**## Authentication using external script**

**## Make sure the script is executable by ejabberd.**

**##**

**## auth\_method: external**

**## extauth\_program: "/path/to/authentication/script"**

**##**

**## Authentication using ODBC**

**## Remember to setup a database in the next section.**

**##**

**## auth\_method: odbc**

**##**

**## Authentication using PAM**

**##**

**## auth\_method: pam**

**## pam\_service: "pamservicename"**

**##**

**## Authentication using LDAP**

**##**

**## auth\_method: ldap**

**##**

**## List of LDAP servers:**

**## ldap\_servers:**

**## - "loalhost"**

**##**

**## Encryption of connection to LDAP servers:**

**## ldap\_encrypt: none**

**## ldap\_encrypt: tls**

**##**

**## Port to connect to on LDAP servers:**

**## ldap\_port: 389**

**## ldap\_port: 636**

**##**

**## LDAP manager:**

**## ldap\_rootdn: "dc=example,dc=com"**

**##**

**## Password of LDAP manager:**

**## ldap\_password: "\*\*\*\*\*\*"**

**##**

**## Search base of LDAP directory:**

**## ldap\_base: "dc=example,dc=com"**

**##**

**## LDAP attribute that holds user ID:**

**## ldap\_uids:**

**## - "mail": "%u@mail.example.org"**

**##**

**## LDAP filter:**

**## ldap\_filter: "(objectClass=shadowAccount)"**

**##**

**## Anonymous login support:**

**## auth\_method: anonymous**

**## anonymous\_protocol: sasl\_anon | login\_anon | both**

**## allow\_multiple\_connections: true | false**

**##**

**## host\_config:**

**## "public.example.org":**

**## auth\_method: anonymous**

**## allow\_multiple\_connections: false**

**## anonymous\_protocol: sasl\_anon**

**##**

**## To use both anonymous and internal authentication:**

**##**

**## host\_config:**

**## "public.example.org":**

**## auth\_method:**

**## - internal**

**## - anonymous**

**### ==============**

**### DATABASE SETUP**

**## ejabberd by default uses the internal Mnesia database,**

**## so you do not necessarily need this section.**

**## This section provides configuration examples in case**

**## you want to use other database backends.**

**## Please consult the ejabberd Guide for details on database creation.**

**##**

**## MySQL server:**

**##**

**## odbc\_type: mysql**

**## odbc\_server: "server"**

**## odbc\_database: "database"**

**## odbc\_username: "username"**

**## odbc\_password: "password"**

**##**

**## If you want to specify the port:**

**## odbc\_port: 1234**

**##**

**## PostgreSQL server:**

**##**

**## odbc\_type: pgsql**

**## odbc\_server: "server"**

**## odbc\_database: "database"**

**## odbc\_username: "username"**

**## odbc\_password: "password"**

**##**

**## If you want to specify the port:**

**## odbc\_port: 1234**

**##**

**## If you use PostgreSQL, have a large database, and need a**

**## faster but inexact replacement for "select count(\*) from users"**

**##**

**## pgsql\_users\_number\_estimate: true**

**##**

**## ODBC compatible or MSSQL server:**

**##**

**## odbc\_type: odbc**

**## odbc\_server: "DSN=ejabberd;UID=ejabberd;PWD=ejabberd"**

**##**

**## Number of connections to open to the database for each virtual host**

**##**

**## odbc\_pool\_size: 10**

**##**

**## Interval to make a dummy SQL request to keep the connections to the**

**## database alive. Specify in seconds: for example 28800 means 8 hours**

**##**

**## odbc\_keepalive\_interval: undefined**

**### ===============**

**### TRAFFIC SHAPERS**

**shaper:**

**##**

**## The "normal" shaper limits traffic speed to 1000 B/s**

**##**

**normal: 1000**

**##**

**## The "fast" shaper limits traffic speed to 50000 B/s**

**##**

**fast: 50000**

**##**

**## This option specifies the maximum number of elements in the queue**

**## of the FSM. Refer to the documentation for details.**

**##**

**max\_fsm\_queue: 1000**

**###. ====================**

**###' ACCESS CONTROL LISTS**

**acl:**

**##**

**## The 'admin' ACL grants administrative privileges to XMPP accounts.**

**## You can put here as many accounts as you want.**

**##**

**admin:**

**user:**

**- "admin": "ip-172-31-31-239.us-west-2.compute.internal"**

**##**

**## Blocked users**

**##**

**## blocked:**

**## user:**

**## - "baduser": "example.org"**

**## - "test"**

**## Local users: don't modify this.**

**##**

**local:**

**user\_regexp: ""**

**##**

**## More examples of ACLs**

**##**

**## jabberorg:**

**## server:**

**## - "jabber.org"**

**## aleksey:**

**## user:**

**## - "aleksey": "jabber.ru"**

**## test:**

**## user\_regexp: "^test"**

**## user\_glob: "test\*"**

**##**

**## Loopback network**

**##**

**loopback:**

**ip:**

**- "127.0.0.0/8"**

**##**

**## Bad XMPP servers**

**##**

**## bad\_servers:**

**## server:**

**## - "xmpp.zombie.org"**

**## - "xmpp.spam.com"**

**##**

**## Define specific ACLs in a virtual host.**

**##**

**## host\_config:**

**## "localhost":**

**## acl:**

**## admin:**

**## user:**

**## - "bob-local": "localhost"**

**### ============**

**### ACCESS RULES**

**access:**

**## Maximum number of simultaneous sessions allowed for a single user:**

**max\_user\_sessions:**

**all: 10**

**## Maximum number of offline messages that users can have:**

**max\_user\_offline\_messages:**

**admin: 5000**

**all: 100**

**## This rule allows access only for local users:**

**local:**

**local: allow**

**## Only non-blocked users can use c2s connections:**

**c2s:**

**blocked: deny**

**all: allow**

**## For C2S connections, all users except admins use the "normal" shaper**

**c2s\_shaper:**

**admin: none**

**all: normal**

**## All S2S connections use the "fast" shaper**

**s2s\_shaper:**

**all: fast**

**## Only admins can send announcement messages:**

**announce:**

**admin: allow**

**## Only admins can use the configuration interface:**

**configure:**

**admin: allow**

**## Admins of this server are also admins of the MUC service:**

**muc\_admin:**

**admin: allow**

**## Only accounts of the local ejabberd server can create rooms:**

**muc\_create:**

**local: allow**

**## All users are allowed to use the MUC service:**

**muc:**

**all: allow**

**## Only accounts on the local ejabberd server can create Pubsub nodes:**

**pubsub\_createnode:**

**local: allow**

**## In-band registration allows registration of any possible username.**

**## To disable in-band registration, replace 'allow' with 'deny'.**

**register:**

**all: allow**

**## Only allow to register from localhost**

**trusted\_network:**

**loopback: allow**

**## Do not establish S2S connections with bad servers**

**## s2s:**

**## bad\_servers: deny**

**## all: allow**

**## By default the frequency of account registrations from the same IP**

**## is limited to 1 account every 10 minutes. To disable, specify: infinity**

**## registration\_timeout: 600**

**##**

**## Define specific Access Rules in a virtual host.**

**##**

**## host\_config:**

**## "localhost":**

**## access:**

**## c2s:**

**## admin: allow**

**## all: deny**

**## register:**

**## all: deny**

**### ================**

**### DEFAULT LANGUAGE**

**##**

**## language: Default language used for server messages.**

**##**

**language: "en"**

**##**

**## Set a different default language in a virtual host.**

**##**

**## host\_config:**

**## "localhost":**

**## language: "ru"**

**### =======**

**### CAPTCHA**

**##**

**## Full path to a script that generates the image.**

**##**

**## captcha\_cmd: "/opt/ejabberd-15.04/lib/ejabberd-15.04/priv/bin/captcha.sh"**

**##**

**## Host for the URL and port where ejabberd listens for CAPTCHA requests.**

**##**

**## captcha\_host: "example.org:5280"**

**##**

**## Limit CAPTCHA calls per minute for JID/IP to avoid DoS.**

**##**

**## captcha\_limit: 5**

**### =======**

**### MODULES**

**##**

**## Modules enabled in all ejabberd virtual hosts.**

**##**

**modules:**

**mod\_adhoc: []**

**mod\_announce: # recommends mod\_adhoc**

**access: announce**

**## mod\_blocking: [] # requires mod\_privacy**

**mod\_caps: []**

**mod\_carboncopy: []**

**mod\_configure: [] # requires mod\_adhoc**

**mod\_disco: []**

**## mod\_echo: []**

**## mod\_irc: []**

**mod\_http\_bind: []**

**## mod\_http\_fileserver:**

**## docroot: "/var/www"**

**## accesslog: "/opt/ejabberd-15.04/logs/access.log"**

**mod\_last: []**

**mod\_muc:**

**## host: "conference.@HOST@"**

**access: muc**

**access\_create: muc\_create**

**access\_persistent: muc\_create**

**access\_admin: muc\_admin**

**## mod\_muc\_log: []**

**mod\_offline:**

**access\_max\_user\_messages: max\_user\_offline\_messages**

**## mod\_ping: []**

**## mod\_pres\_counter:**

**## count: 5**

**## interval: 60**

**mod\_privacy: []**

**mod\_private: []**

**## mod\_proxy65: []**

**mod\_pubsub:**

**access\_createnode: pubsub\_createnode**

**## reduces resource comsumption, but XEP incompliant**

**ignore\_pep\_from\_offline: true**

**## XEP compliant, but increases resource comsumption**

**## ignore\_pep\_from\_offline: false**

**last\_item\_cache: false**

**plugins:**

**- "flat"**

**- "hometree"**

**- "pep" # pep requires mod\_caps**

**mod\_register:**

**##**

**## Protect In-Band account registrations with CAPTCHA.**

**##**

**## captcha\_protected: true**

**##**

**## Set the minimum informational entropy for passwords.**

**##**

**## password\_strength: 32**

**##**

**## After successful registration, the user receives**

**## a message with this subject and body.**

**##**

**welcome\_message:**

**subject: "Welcome!"**

**body: |-**

**Hi.**

**Welcome to this XMPP server.**

**##**

**## When a user registers, send a notification to**

**## these XMPP accounts.**

**##**

**## registration\_watchers:**

**## - "admin1@example.org"**

**##**

**## Only clients in the server machine can register accounts**

**##**

**ip\_access: trusted\_network**

**##**

**## Local c2s or remote s2s users cannot register accounts**

**##**

**## access\_from: deny**

**access: register**

**mod\_roster: []**

**mod\_shared\_roster: []**

**## mod\_time: []**

**mod\_vcard:**

**search: true**

**matches: 20**

**allow\_return\_all: true**

**search\_all\_hosts: false**

**mod\_version: []**

**##**

**## Enable modules with custom options in a specific virtual host**

**##**

**## append\_host\_config:**

**## "localhost":**

**## modules:**

**## mod\_echo:**

**## host: "mirror.localhost"**

**##**

**## Enable modules management via ejabberdctl for installation and**

**## uninstallation of public/private contributed modules**

**## (enabled by default)**

**##**

**allow\_contrib\_modules: true**

**### Local Variables:**

**### mode: yaml**

**### End:**

**### vim: set filetype=yaml tabstop=8**