

# Serveur jabber

Debian GNU/Linux



Matthieu Vogelweith  
13 janvier 2009

# Résumé

Installation d'un serveur de messagerie instantannée Ejabberd 2 [1] sous Debian GNU/Linux Lenny[2].

Ce document a été rédigé en LaTeX en utilisant l'excellent Vim sous Debian GNU/Linux. Il est disponible aux formats XHTML et PDF. Les sources LaTeX sont disponibles ici : [L<sup>A</sup>T<sub>E</sub>X](#)

# Licence

Copyright ©2009 Matthieu VOGELWEITH <matthieu@vogelweith.com>.

Vous avez le droit de copier, distribuer et/ou modifier ce document selon les termes de la GNU Free Documentation License, Version 1.3 ou ultérieure publiée par la Free Software Foundation ; avec aucune section inaltérable, aucun texte de première page de couverture, et aucun texte de dernière page de couverture. Une copie de la licence est disponible dans la page [GNU Free Documentation License](#).

# Table des matières

<b>Table des matières</b>	<b>3</b>
<b>1 Présentation</b>	<b>4</b>
<b>2 Installation</b>	<b>5</b>
<b>3 Enregistrements DNS</b>	<b>6</b>
<b>4 Configuration</b>	<b>7</b>
4.1 Paramètres par défaut . . . . .	7
4.2 Configuration générale . . . . .	7
4.3 Authentification LDAP . . . . .	7
4.4 Cartes de visites LDAP . . . . .	7
<b>5 Administration Web</b>	<b>9</b>
5.1 Création d'un vhost . . . . .	9
5.2 Contacts partagés . . . . .	10
<b>6 Sécurisation</b>	<b>11</b>
6.1 Support SSL . . . . .	11
6.2 Firewall . . . . .	11
<b>7 Configuration des clients</b>	<b>13</b>
7.1 Clients Linux . . . . .	13
7.2 Clients Windows . . . . .	13
7.3 Clients MAC OS X . . . . .	13
<b>8 Références</b>	<b>14</b>

## **Chapitre 1**

# **Présentation**

## Chapitre 2

# Installation

- Installation

```
# aptitude install ejabberd
```

## Chapitre 3

# Enregistrements DNS

```
; Jabber
$ORIGIN example.org.
jabber                IN      A          xx.xx.xx.xx
_jabber._tcp         IN      SRV 5 0 5269 jabber
_xmpp-server         IN      SRV 5 0 5269 jabber
_xmpp-client._tcp    IN      SRV 5 0 5222 jabber
```

# Chapitre 4

## Configuration

### 4.1 Paramètres par défaut

Dans /etc/default/ejabberd

```
SMP=auto
```

### 4.2 Configuration générale

Dans /etc/ejabberd/ejabberd.cfg

```
%% Hostname  
{hosts, ["example.org"]}
```

Désactivation de l'enregistrement en ligne

```
{language, "fr"}.
```

### 4.3 Authentification LDAP

```
%{auth_method, internal}.  
{auth_method, ldap}.  
{ldap_servers, ["localhost"]}.  
{ldap_base, "ou=Users,dc=vogelweith,dc=com"}.  
{ldap_uids, [{"mail", "%u@vogelweith.com"}]}.  
{ldap_filter, "&(objectClass=shadowAccount)(mailalias=jabber_users@vogelweith.com)"}.
```

### 4.4 Cartes de visites LDAP

Remplissage automatique des VCARD.

```

{mod_vcard_ldap,
 [
  %% We use the same server and port, but want to bind anonymously because
  %% our LDAP server accepts anonymous requests to
  %% "ou=AddressBook,dc=example,dc=org" subtree.
  {ldap_rootdn, ""},
  {ldap_password, ""},
  %% define the addressbook's base
  {ldap_base, "ou=Users,dc=example,dc=com"},
  %% user's part of JID is located in the "mail" attribute
  {ldap_uidattr, "uid"},
  %% We have to define empty filter here, because entries in addressbook does not
  %% belong to shadowAccount object class
  {ldap_filter, ""},
  %% Now we want to define vCard pattern
  {ldap_vcard_map,
   [{"NICKNAME", "%s %s", ["givenName", "sn"]}, % just use user's part of JID as his
    nickname
    {"GIVEN", "%s", ["givenName"]},
    {"FAMILY", "%s", ["sn"]},
    {"FN", "%s %s", ["givenName", "sn"]}, % example: "Smith, John"
    {"EMAIL", "%s", ["mail"]},
    {"BDAY", "%s", ["birthDay"]}]}},
  %% Search form
  {ldap_search_fields,
   [{"User", "%u"},
    {"Name", "givenName"},
    {"Family Name", "sn"},
    {"Email", "mail"},
    {"Birthday", "birthDay"}]},
  %% vCard fields to be reported
  %% Note that JID is always returned with search results
  {ldap_search_reported,
   [{"Full Name", "FN"},
    {"Nickname", "NICKNAME"},
    {"Birthday", "BDAY"}]}
 ]}

```

## Chapitre 5

# Administration Web

Ejabberd est fournit avec un petit serveur web qui propose une interface d'administration pour le service. Par défaut, ce serveur web écoute sur le port 5280 de la machine locale. Pour simplifier l'accès à cette interface et pour éviter d'ouvrir un port supplémentaire sur le firewall, le paragraphe suivant propose de créer un virtualhost apache qui fera également proxy pour accéder à l'interface d'administration.

### 5.1 Création d'un vhost

- Activation du module proxy\_http de apache

```
# a2enmod proxy_http
```

- Dans /etc/apache2/sites-available/xx\_jabber :

```
<VirtualHost *:80>

    ServerAdmin webmaster@example.org
    ServerName jabber.example.org
    ServerAlias jabber
    ServerSignature Off

    # Redirection vers le vhost en HTTPS
    RewriteEngine on
    RewriteCond %{SERVER_PORT} !^443$
    RewriteRule ^/(.*)$ https://%{SERVER_NAME}$1 [R=301,L]

</VirtualHost>

<VirtualHost *:443>

    ServerAdmin webmaster@example.org
    ServerName jabber.example.org
    ServerAlias jabber
    ServerSignature Off

    RedirectMatch ^/$ /admin/

    SSLEngine on
    SSLCertificateFile /etc/ssl/certs/ejabberd.pem

    <Proxy http://127.0.0.1:5280>
        Allow from all
    </Proxy>
```

```
SSLProxyEngine on
ProxyPreserveHost on
ProxyPass /admin/ http://127.0.0.1:5280/admin/
ProxyPassReverse /admin/ http://127.0.0.1:5280/admin/

ErrorLog /var/log/apache2/jabber_error.log
CustomLog /var/log/apache2/jabber_access.log combined

</VirtualHost>
```

## 5.2 Contacts partagés

Affichage automatique de tous les contacts Dé-commenter la ligne :

```
{mod\_shared\_rosters, []}
```

Puis aller dans l'interface web <http://jabber.example.org> puis dans Virtual Hosts -> monde-maine.com -> Shared Roster

Ajouter alors un groupe en remplissant tous les champs et mettre @all@ dans les "group members".

## Chapitre 6

# Sécurisation

### 6.1 Support SSL

- Edition de /etc/ssl/ejabberd.cnf :

```
[ req ]
default_bits           = 2048
default_keyfile        = privkey.pem
distinguished_name     = req_distinguished_name
prompt                = no
string_mask            = nombstr
x509_extensions        = server_cert

[ req_distinguished_name ]
countryName            = FR
stateOrProvinceName   = France
localityName           = Strasbourg
organizationName       = Example
organizationalUnitName = Jabber Server
commonName             = jabber.example.org
emailAddress           = webmaster@example.org

[ server_cert ]
basicConstraints       = critical, CA:FALSE
subjectKeyIdentifier   = hash
keyUsage               = digitalSignature, keyEncipherment
extendedKeyUsage       = serverAuth, clientAuth
nsCertType             = server
nsComment              = "Jabber Certificate"
```

- Création du certificat :

```
# openssl req -x509 -new \
  -config /etc/ssl/ejabberd.cnf \
  -out /etc/ssl/certs/ejabberd.pem \
  -keyout /etc/ssl/certs/ejabberd.pem \
  -days 730 -nodes -batch
```

### 6.2 Firewall

```
# Jabber
```

ACCEPT	all	\$FW	tcp	5222
ACCEPT	all	\$FW	tcp	5269

# Chapitre 7

## Configuration des clients

### 7.1 Clients Linux

- pidgin [3]
- empathy [4]
- kopete [5]

### 7.2 Clients Windows

- pidgin [3]
- spark [6]

### 7.3 Clients MAC OS X

- ichat [7]

## Chapitre 8

# Références

- [1] Site officiel de ejabberd. [ejabberd.jabber.ru](http://ejabberd.jabber.ru).
- [2] Site officiel du projet debian. [www.debian.org](http://www.debian.org).
- [3] Site officiel de pidgin. [pidgin.im](http://pidgin.im).
- [4] Site officiel de pidgin. [live.gnome.org/Empathy](http://live.gnome.org/Empathy).
- [5] Site officiel de kopete. [kopete.kde.org](http://kopete.kde.org).
- [6] Site officiel de spark. [www.igniterealtime.org/projects/spark/](http://www.igniterealtime.org/projects/spark/).
- [7] Site officiel de ichat. [www.apple.com/macosx/features/ichat.html](http://www.apple.com/macosx/features/ichat.html).