XMPP

Overview

XMPP chat client

To Access

From the Community Admin page, click the **XMPP** tab.

Introduction

XMPP is a chat protocol that enables users to have instant message applications and other technologies. By using the ConverseJS XMPP client, Tiki users can integrate chat feature to their websites. See also PluginXMPP and Module XMPP

How to use XMPP in Tiki

1.1.1. install and Configure an XMPP server

Despite Tiki has ConverseJS bundled in its package, it is still necessary to have an XMPP server where clients can connect. Tiki can be very well integrated to Openfire XMPP server, but other servers also works.

1.1.2. Enable the feature and plugin

Enable the feature at "**Admin home > RTC > XMPP**" tab. Enable the plugin at "**Admin home > Editting and plugins > Plugins > XMPP plugin**"

1.1.3. Configure Tiki to talk to the XMPP server

Go to the community page on the admin panels (RTC page on Tiki 19+), select the XMPP tab, and:

1. Check the **XMPP client (ConverseJS)**.
2. On **XMPP server domain** field, example type yourserver.wikisuite.org.
3. On **XMPP http-bind URL** field, example type https://yourserver.wikisuite.org:7070/http-bind/ .
4. chose an **Authentication method** .
5. Click on Apply.

1.1.4. Use ConverseJS modes

ConverseJS can be used in 3 modes:

1. Overlayed mode
2. Embedded mode
3. Fullscreen mode

Overlayed mode

Overlayed Mode is configured as a Tiki Module. A small box will appears on bottom of the page containing a contact list. Users can click on contact list to start a chat session.

Embedded mode

The Embedded mode can be configured as a wiki_plugin. A chatbox will appear in the middle of the page. This chatbox is a group chat, where many users can send messages.

Fullscreen mode

Fullscreen mode takes all webpage space to provide the contact list and chat boxes. Users can switch

between private chats and group chats while using this mode.

Authentication mode

The ConverseJS embedded in Tiki can authenticate in XMPP server using 3 different strategies.

1. Username / password
2. Prebind
3. OAuthBearer Token

When using the username/password mechanism to authenticate, ConverseJS will ask user these information whenever is necessary. It is very easy to setup, but in other hand, it may demotivate the user to join the chat if he needs to type his password several times.

The Prebind mode is mechanism that makes XMPP server to confirm that a specific user is trying to authenticate. The user does not need to type username or password using this mode. Whenever user log into Tiki, he will automatically log into Openfire when needed. The Tiki prebind URL needs to be informed in Openfire panel.

On OAuthBearer Token, ConverseJS extract a special Token from Tiki and gives as a password to Openfire. Extra setup is needed in Openfire side in order to use this mode. Also, this mode is still beta and is available on Tiki 20.

Option	Description	Default
XMPP client (ConverseJS)	Integration with Converse.js XMPP client.	Disabled
XMPP server domain	XMPP server domain *example.org*	None
XMPP http-bind URL	Full URL to the http-bind. *https://xmpp.example.org/http-bind/*	None
XMPP MUC Domain	Required for auto-joining rooms *conference.example.org*	None
Authentication method	The authentication method to be used by XMPP client Plain | OAuth (uses Tiki as provider) | Openfire TikiToken	None
Openfire REST API endpoint	Full URL to API endpoint *https://xmpp.example.org:9091/plugins/restapi/v1/*	None
Rest API username	Username to allow Openfire API usage	None
Rest API password	Password to allow Openfire API usage	None
Always Load ConverseJS	Adds the ConverseJS Javascript and CSS files to every page.	Disabled
ConverseJS Debug Mode	Enables more logging, e.g. XML stanzas and error tracebacks to the JavaScript Console	Disabled

Option	Description	Default
ConverseJS Extra Settings	JSON format object defining extra optional settings to initialize ConverseJS	None

Option	Description	Default
XMPP client (ConverseJS)	Integration with Converse.js XMPP client.	Disabled
XMPP server domain	XMPP server domain *example.org*	None
XMPP http-bind URL	Full URL to the http-bind. *https://xmpp.example.org/http-bind/*	None
XMPP MUC Domain	Required for auto-joining rooms *conference.example.org*	None
Authentication method	The authentication method to be used by XMPP client Plain | OAuth (uses Tiki as provider) | Openfire TikiToken	None
Openfire REST API endpoint	Full URL to API endpoint *https://xmpp.example.org:9091/plugins/restapi/v1/*	None
Rest API username	Username to allow Openfire API usage	None
Rest API password	Password to allow Openfire API usage	None
Always Load ConverseJS	Adds the ConverseJS Javascript and CSS files to every page.	Disabled
ConverseJS Debug Mode	Enables more logging, e.g. XML stanzas and error tracebacks to the JavaScript Console	Disabled
ConverseJS Extra Settings	JSON format object defining extra optional settings to initialize ConverseJS	None

Option	Description	Default
XMPP client (ConverseJS)	Integration with Converse.js XMPP client.	Disabled
XMPP server domain	XMPP server domain *example.org*	None
XMPP http-bind URL	Full URL to the http-bind. *https://xmpp.example.org/http-bind/*	None

Option	Description	Default
XMPP MUC Domain	Required for auto-joining rooms *conference.example.org*	None
Authentication method	The authentication method to be used by XMPP client Plain | OAuth (uses Tiki as provider) | Openfire TikiToken	None
Openfire REST API endpoint	Full URL to API endpoint *https://xmpp.example.org:9091/plugins/restapi/v1/*	None
Rest API username	Username to allow Openfire API usage	None
Rest API password	Password to allow Openfire API usage	None
ConverseJS Debug Mode	Enables more logging, e.g. XML stanzas and error tracebacks to the JavaScript Console	Disabled
ConverseJS Extra Settings	JSON format object defining extra optional settings to initialize ConverseJS	None

New in Tiki17: the addition of the ConverseJS XMPP client. New in Tiki19: PluginXMPP to embed a chatroom in a wiki page

The reference implementation is with Openfire, but this should work with any XMPP server (perhaps with fewer features).

alias

- Converse.js
- ConverseJS
- Converse JS