Server Technologies Core Use Case Comparison

The two most promising Chat Server technologies are presented in this document with analysis of how well they promise to address the core use cases that have been determined in consultation with the client.

# Choice of protocol

[Wikipedia lists a large number of messaging protocols.](https://en.wikipedia.org/wiki/Comparison_of_instant_messaging_protocols) In order for a chat server to be considered for this project, a number of criteria were established:

1. Open Source (non-proprietary) - Pinnacle wishes to remain in control of the source code.
2. Offers end-to-end encryption - enabling secure communications
3. Server-based (centralised database) - for legal reasons, Pinnacle needs to be able to access all chat data

These factors limited the available chat protocols to those listed in Table 1. N.B. Matrix is server-based and fulfills (1) and (2) however Matrix servers are decentralised.

### Table 1 - Protocols Considered

|  |  |
| --- | --- |
| **Protocol** | **Comments** |
| MTProto (Telegram) | Relatively new protocol. [Developers have implemented their own cryptography.](https://www.theatlantic.com/technology/archive/2016/01/isiss-favorite-messaging-app-has-a-security-problem/422460/) This approach can be risky. |
| Signal | [TextSecure Server requires sms service (e.g. Twilio)](http://stackoverflow.com/questions/33699970/run-custom-textsecure-signal-server/34471995#34471995) to register users on the network - an unnecessary limitation for this project |
| XMPP | Well established protocol (started in 1999). Standards-based with a number of optional encryption protocols. Of the three, it also supports bulletins to all contacts which although not a core use case is a useful feature in a corporate environment e.g. to advise of a planned outage |

This analysis has led us to conclude that XMPP is the most suitable protocol for use in our project.

# XMPP Servers

There are a large range of XMPP servers. While Team Orange’s project brief is centred on development of the client, the choice of server is important as well. In order to support future changes to the instant messaging system, the following characteristics are **desirable** from a technical/architectural standpoint:

* Web server technology stack to host the XMPP server has a reputation for reliability and security (minimise back doors into the chats)
* XMPP server implementation uses a recognised, common development language (for future updates and maintenance)
* The code for the server is open source
* The project is actively maintained

There are also a non-technical requirement. In order to speed development and offer easy access to the team which is spread geographically, a web interface to the server is required.

The following **mandatory** requirements for the XMPP server are based on the core use cases:

* User management (create/update/delete)
* Access control (authentication and authorisation)
* Maintain contact list for each user
* Store and retrieve chats
* Maintain subscriptions to channels

The core of XMPP is defined in the RFC3920 standard. It covers the mandatory requirements listed above in technical, non-implementation specific terms. Most XMPP servers have significantly more functionality than this core specification. This functionality is defined in technical documents known as XEPs. When examining the status of XEP implementation, two servers are front runners:

* OpenFire
* Prosody

Both of these projects are actively maintained and are good candidates for selection. OpenFire is developed in Java while Prosody is written in Lua. At present, none of the team has experience with Lua. While the majority (if not all) development will take place on the client, it is reassuring (and convenient from a future development viewpoint) to have the entire technology stack implemented in Java. In addition, OpenFire has a web interface as standard for the administration console. In Prosody, this is currently implemented by a community module which is in the beta stage.

# Server Chosen

In conclusion, from a review of the documentation on the two servers and internet research, OpenFire XMPP server appears to meet our needs well. We have chosen to use OpenFire as our XMPP server.