

XMPP

From Wikipedia, the free encyclopedia

Extensible Messaging and Presence Protocol (XMPP) is a communications protocol for message-oriented middleware based on XML (Extensible Markup Language).^[1] It enables the near-real-time exchange of structured yet extensible data between any two or more network entities.^[2] Originally named **Jabber**,^[3] the protocol was developed by the Jabber open-source community in 1999 for near real-time instant messaging (IM), presence information, and contact list maintenance. Designed to be extensible, the protocol has been used also for publish-subscribe systems, signalling for VoIP, video, file transfer, gaming, the Internet of Things (IoT) applications such as the smart grid, and social networking services.

Unlike most instant messaging protocols, XMPP is defined in an open standard and uses an open systems approach of development and application, by which anyone may implement an XMPP service and interoperate with other organizations' implementations. Because XMPP is an open protocol, implementations can be developed using any software license; although many server, client, and library implementations are distributed as free and open-source software, numerous freeware and commercial software implementations also exist.

The Internet Engineering Task Force (IETF) formed an XMPP working group in 2002 to formalize the core protocols as an IETF instant messaging and presence technology. The XMPP Working group produced four specifications (RFC 3920, RFC 3921, RFC 3922, RFC 3923), which were approved as Proposed Standards in 2004. In 2011, RFC 3920 and RFC 3921 were superseded by RFC 6120 and RFC 6121 respectively, with RFC 6122 specifying the XMPP address format. In 2015, RFC 6122 was superseded by RFC 7622. In addition to these core protocols standardized at the IETF, the XMPP Standards Foundation (formerly the Jabber Software Foundation) is active in developing open XMPP extensions.

XMPP-based software is deployed widely across the Internet, and by 2003, was used by over ten million people worldwide, according to the XMPP Standards Foundation.^[4]



XMPP

Official logo

Contents

- 1 History
- 2 Federation
 - 2.1 Decentralization and addressing
 - 2.2 Connecting to other protocols
- 3 XMPP as an extensible Message Oriented Middleware (xMOM) platform
- 4 Implementations
 - 4.1 XMPP via HTTP and WebSocket transports
 - 4.2 Deployments
- 5 Development
 - 5.1 XSF
 - 5.2 Internet of Things
- 6 Specifications and standards
 - 6.1 Competing standards
- 7 Features
 - 7.1 Strengths
 - 7.2 Weaknesses
 - 7.3 Extensions
- 8 See also
- 9 References
- 10 External links

History

Jeremie Miller began working on the Jabber technology in 1998 and released the first version of the jabberd server on January 4, 1999.^[5] The early Jabber community focused on open-source software, mainly the jabberd server, but its major outcome proved to be the development of the XMPP protocol.

The early Jabber protocol, as developed in 1999 and 2000, formed the basis for XMPP as published in RFC 3920 and RFC 3921 (the primary changes during formalization by the IETF's XMPP Working Group were the addition of TLS for channel encryption and SASL for authentication). Note that RFC 3920 and RFC 3921 have been superseded by RFC 6120 and RFC 6121, published in 2011.

The first IM service based on XMPP was Jabber.org, which has operated continuously and offered free accounts since 1999.^[6] From 1999 until February 2006, the service used jabberd as its server software, at which time it migrated to ejabberd (both of which are free software application servers). In January 2010, the service migrated to the proprietary M-Link server software produced by Isode Ltd.^[7]

In August 2005, Google introduced Google Talk, a combination VoIP and IM system that uses XMPP for instant messaging and as a base for a voice and file transfer signaling protocol called Jingle. The initial launch did not include server-to-server communications; Google enabled that feature on January 17, 2006.^[8] Google has since

added video functionality to Google Talk, also using the Jingle protocol for signaling. In May 2013, Google announced Jabber compatibility would be dropped from Google Talk for server-to-server federation, although it would retain client-to-server support.^[9]

In January 2008, AOL introduced experimental XMPP support for its AOL Instant Messenger (AIM) service,^[10] allowing AIM users to communicate using XMPP. However, in March 2008, this service was discontinued. As of May 2011, AOL offers limited XMPP support.^[11]

In September 2008, Cisco Systems acquired Jabber, Inc., the creators of the commercial product Jabber XCP.^[12]

In February 2010, the social-networking site Facebook opened up its chat feature to third-party applications via XMPP.^[13] Some functionality was unavailable through XMPP, and support was dropped in April 2014.^[14]

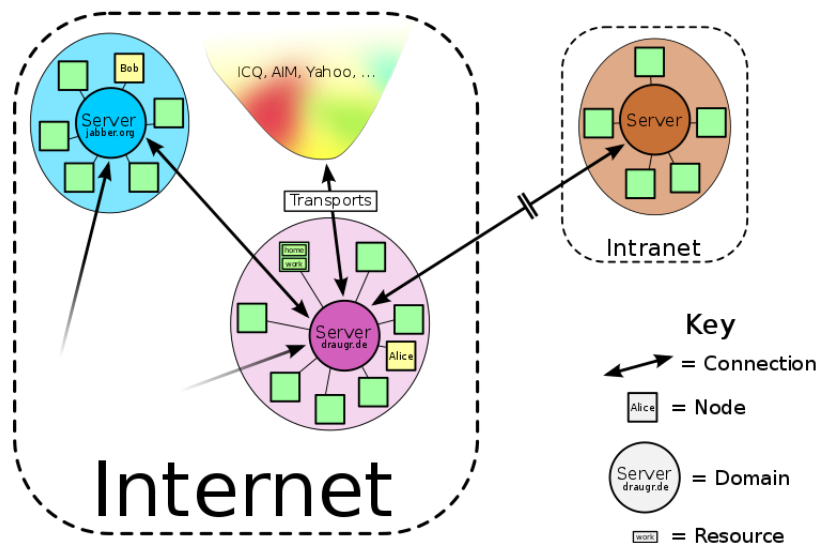
Similarly, in December 2011, Microsoft released an XMPP interface to its Microsoft Messenger service.^[15] Skype, its de facto successor, also provides limited XMPP support.^[16] However, these are not native XMPP implementations.

Federation

Decentralization and addressing

The XMPP network uses a client–server architecture; clients do not talk directly to one another. The model is decentralized - anyone can run a server. By design, there is no central authoritative server as there is with services such as AOL Instant Messenger or Windows Live Messenger. Some confusion often arises on this point as there is a public XMPP server being run at `jabber.org`, to which a large number of users subscribe. However, anyone may run their own XMPP server on their own domain.

Every user on the network has a unique XMPP address, called *JID* (for historical reasons, XMPP addresses are often called *Jabber IDs*). The JID is structured like an email address with a username and a



A simple XMPP network with the servers `jabber.org` and `draugr.de`. Green clients are online, yellow clients are writing each other and small green *subclients* are the resources of one user. The brown network is not connected to the internet. The server `draugr.de` is connected to other IM services (ICQ, AIM and other) via *XMPP transports*.

domain name (or IP address^[17]) for the server where that user resides, separated by an at sign (@), such as `username@example.com`.

Since a user may wish to log in from multiple locations, they may specify a **resource**. A resource identifies a particular client belonging to the user (for example home, work, or mobile). This may be included in the JID by appending a slash followed by the name of the resource. For example, the full JID of a user's mobile account could be `username@example.com/mobile`.

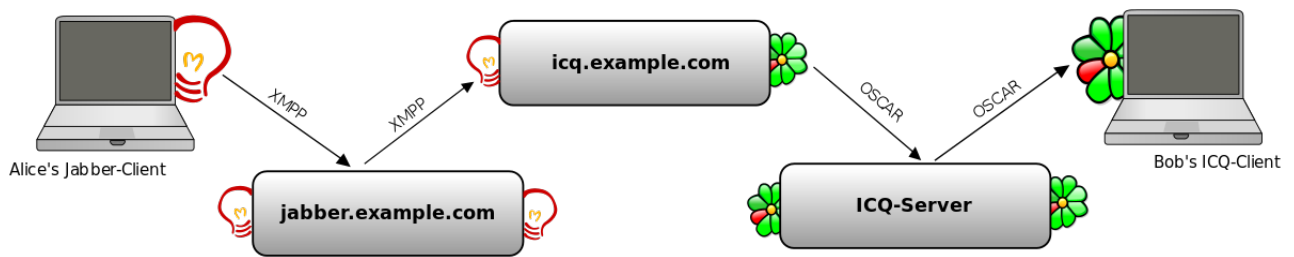
Each resource may have specified a numerical value called **priority**. Messages simply sent to `username@example.com` will go to the client with highest priority, but those sent to `username@example.com/mobile` will go *only* to the mobile client. The highest priority is the one with largest numerical value.

JIDs without a username part are also valid, and may be used for system messages and control of special features on the server. A resource remains optional for these JIDs as well.

The means to route messages based on a logical endpoint identifier - the JID, instead of by an explicit IP Address present opportunities to use XMPP as an Overlay network implementation on top of different underlay networks.

Connecting to other protocols

One of the original design goals of the early Jabber open-source community was enabling users to connect to multiple



Alice sends a message through the XMPP net to the ICQ transport. The message is next routed to Bob via the ICQ network.

instant messaging systems (especially non-XMPP systems) through a single client application. This was done through entities called *transports* or *gateways* to other instant messaging protocols, but also to protocols such as SMS or email. Unlike multi-protocol clients, XMPP provides this access at the server level by communicating via special gateway services running alongside an XMPP server. Any user can "register" with one of these gateways by providing the information needed to log on to that network, and can then communicate with users of that network as though they were XMPP users. Thus, such gateways function as client proxies (the gateway authenticates on the user's behalf on the non-XMPP service). As a result, any client that fully supports XMPP can access any network with a gateway without extra code in the client, and without the need for the client to have direct access to the Internet. However, the client proxy model may violate terms of service on the protocol used (although such terms of service are not legally enforceable in several countries) and also requires the user to send their IM username and password to the third-party site that operates the transport (which may raise privacy and security concerns).

Another type of gateway is a server-to-server gateway, which enables a non-XMPP server deployment to connect to native XMPP servers using the built in interdomain federation features of XMPP. Such server-to-server gateways are offered by several enterprise IM software products, including:

- IBM Lotus Sametime^{[18][19]}
- Skype for Business Server (formerly named Microsoft Lync Server and Microsoft Office Communications Server – OCS)^[20]

XMPP as an extensible Message Oriented Middleware (xMOM) platform

XMPP provides a general framework for messaging across a network, which offers a multitude of applications beyond traditional Instant Messaging (IM) and the distribution of Presence data. While several service discovery protocols exist today (such as zeroconf or the Service Location Protocol), XMPP provides a solid base for the discovery of services residing locally or across a network, and the availability of these services (via presence information), as specified by XEP-0030 DISCO.^[21]

Building on its capability to support discovery across local network domains, XMPP is well-suited for cloud computing where virtual machines, networks, and firewalls would otherwise present obstacles to alternative service discovery and presence-based solutions. Cloud computing and storage systems rely on various forms of communication over multiple levels, including not only messaging between systems to relay state but also the migration or distribution of larger objects, such as storage or virtual machines. Along with authentication and in-transit data protection, XMPP can be applied at a variety of levels and may prove ideal as an extensible middleware or Message Oriented Middleware (MOM) protocol. Widely known for its ability to exchange XML-based content natively, it has become an open platform for the exchange of other forms of content including proprietary binary streams, Full Motion Video (FMV) content, and the transport of file-based content, via for example the Jingle series of extensions. Here the majority of the applications have nothing to do with human communications (i.e., IM) but instead provide an open means to support machine-to-machine or peer-to-peer communications across a diverse set of networks.

Implementations

XMPP is implemented by a large number of clients, servers, and code libraries.^[22] These implementations are provided under a variety of software licenses.

XMPP via HTTP and WebSocket transports

The original and "native" transport protocol for XMPP is Transmission Control Protocol (TCP), using open-ended XML streams over long-lived TCP connections.

As an alternative to the TCP transport, the XMPP community has also developed an HTTP transport for web clients as well as users behind restricted firewalls. In the original specification, XMPP could use HTTP in two ways: *polling*^[23] and *binding*. The polling method, now deprecated, essentially implies messages stored on a

server-side database are being fetched (and posted) regularly by an XMPP client by way of HTTP 'GET' and 'POST' requests. The binding method, implemented using *Bidirectional-streams Over Synchronous HTTP* (BOSH),^[24] allows servers to push messages to clients as soon as they are sent. This push model of notification is more efficient than polling, where many of the polls return no new data.

Because the client uses HTTP, most firewalls allow clients to fetch and post messages without any hindrances. Thus, in scenarios where the TCP port used by XMPP is blocked, a server can listen on the normal HTTP port and the traffic should pass without problems. Various websites let people sign into XMPP via a browser. Furthermore, there are open public servers that listen on standard http (port 80) and https (port 443) ports, and hence allow connections from behind most firewalls. However, the IANA-registered port for BOSH is actually 5280, not 80.

A perhaps more efficient transport for real-time messaging is WebSocket, a web technology providing for bi-directional, full-duplex communications channels over a single TCP connection. XMPP over WebSocket binding is defined in the IETF proposed standard RFC 7395.

Deployments

Several large public IM services natively use XMPP, including LiveJournal's "LJ Talk",^[25] Nimbuzz, and HipChat. Various hosting services, such as DreamHost, enable hosting customers to choose XMPP services alongside more traditional web and email services. Specialized XMPP hosting services also exist in form of cloud so that domain owners need not directly run their own XMPP servers, including Cisco WebEx Connect, Chrome.pl, Flosoft.biz (<https://flosoft.biz/products/jabber/>), i-pobox.net, and hosted.im.

Some of the largest messaging providers use, or have been using, various forms of XMPP based protocols in their backend systems without necessarily exposing this fact to their end users. This includes WhatsApp, Gtalk and Facebook Chat^{[26][27]} (the deprecated Facebook messaging system). Most of these deployments are built on the free-software, Erlang-based XMPP server called ejabberd.

XMPP is also used in deployments of non-IM services, including smart grid systems such as demand response applications, message-oriented middleware, and as a replacement for SMS to provide text messaging on many smartphone clients.

XMPP is the de facto standard for private chat in gaming related platforms such as Origin,^[28] Raptr, PlayStation, and the now discontinued Xfire. Two notable exceptions are Steam^[29] and Xbox LIVE; both use their own proprietary messaging protocols.

Development

XSF

The XMPP Standards Foundation (XSF) develops and publishes extensions to XMPP through a standards process centered on *XMPP Extension Protocols* (XEPs, previously known as Jabber Enhancement Proposals - JEPs). The following extensions are in especially wide use:

- Data Forms^[30]
- Service Discovery^[31]
- Multi-User Chat^[32]
- Publish-Subscribe^[33] and Personal Eventing Protocol^[34]
- XHTML-IM^[35]
- File Transfer^[36]
- Entity Capabilities^[37]
- HTTP Binding^[24]
- Jingle for voice and video

Internet of Things

XMPP features such as federation across domains, publish/subscribe, authentication and its security even for mobile endpoints are being used to implement the Internet of Things. Several XMPP extensions are part of the experimental implementation: Efficient XML Interchange (EXI) Format;^[38] Sensor Data;^[39] Provisioning;^[40] Control;^[41] Concentrators;^[42] Discovery.^[43]

These efforts are documented on a page in the XMPP wiki dedicated to Internet of Things^[44] and the XMPP IoT mailing list.^[45]

Specifications and standards

The IETF XMPP working group has produced a series of Request for Comments (RFC) documents:

- RFC 3920 (superseded by RFC 6120)
- RFC 3921 (superseded by RFC 6121)
- RFC 3922
- RFC 3923
- RFC 4622 (superseded by RFC 5122)
- RFC 4854
- RFC 4979
- RFC 6122 (superseded by RFC 7622)

The most important and most widely implemented of these specifications are:

- RFC 6120, *Extensible Messaging and Presence Protocol (XMPP): Core*, which describes client–server messaging using two open-ended XML streams. XML streams consist of <presence/>, <message/> and <iq/> (info/query). A connection is authenticated with Simple Authentication and Security Layer (SASL) and encrypted with Transport Layer Security (TLS).
- RFC 6121, *Extensible Messaging and Presence Protocol (XMPP): Instant Messaging and Presence* describes instant messaging (IM), the most common application of XMPP.
- RFC 7622, *Extensible Messaging and Presence Protocol (XMPP): Address Format* describes the rules for XMPP addresses, also called JabberIDs or JIDs. Currently JIDs use PRECIS (as defined in RFC 7564) for handling of Unicode characters outside the ASCII range.

Competing standards

XMPP has often been regarded as a competitor to SIMPLE, based on Session Initiation Protocol (SIP), as the standard protocol for instant messaging and presence notification.^{[46][47]}

The XMPP extension for multi-user chat^[32] can be seen as a competitor to Internet Relay Chat (IRC), although IRC is far simpler, has far fewer features, and is far more widely used.

The XMPP extensions for publish-subscribe^[33] provide many of the same features as the Advanced Message Queuing Protocol (AMQP).

Features

Strengths

Decentralization

The architecture of the XMPP network is similar to email; anyone can run their own XMPP server and there is no central master server.

Open standards

The Internet Engineering Task Force formalized XMPP as an approved instant messaging and presence technology under the name of XMPP (the latest specifications are RFC 6120 and RFC 6121). No royalties or granted permissions are required to implement these specifications.

History

XMPP technologies have been in use since 1999. Many implementations of the XMPP standards exist for clients, servers, components, and code libraries.

Security

XMPP servers can be isolated (e.g., on a company intranet), and secure authentication (SASL) and encryption (TLS) have been built into the core XMPP specifications.

Flexibility

Custom functionality can be built on top of XMPP. To maintain interoperability, common extensions are managed by the XMPP Standards Foundation. XMPP applications beyond IM include chatrooms, network management, content syndication, collaboration tools, file sharing, gaming, remote systems control and monitoring, geolocation, middleware and cloud computing, VoIP, and identity services.

Weaknesses

Does not support Quality of Service (QoS)

Assured delivery of messages has to be built on-top of the XMPP layer. There are two XEPs proposed to deal with this issue, XEP-0184 Message delivery receipts (<http://www.xmpp.org/extensions/xep-0184.html>) which is a draft standard, and XEP-0333 Chat Markers (<http://xmpp.org/extensions/xep-0333.html>) which is considered experimental.

Text-based communication

Since XML is text based, normal XMPP has a higher network overhead compared to purely binary solutions. This issue is being addressed by the experimental XEP-0322: Efficient XML Interchange (EXI) Format (<http://xmpp.org/extensions/xep-0322.html>), where XML is serialized in a very efficient binary manner, especially in schema-informed mode.

In-band binary data transfer is limited

Binary data must be first base64 encoded before it can be transmitted in-band. Therefore, any significant amount of binary data (e.g., file transfers) is best transmitted out-of-band, using in-band messages to coordinate. The best example of this is the Jingle XMPP Extension Protocol, XEP-0166 (<http://xmpp.org/extensions/xep-0166.html>).

Extensions

The XMPP Standards Foundation or XSF (formerly the Jabber Software Foundation) is active in developing open XMPP extensions, so called *XEP*.^[48] However, extensions can also be defined by any individual, software project, or organization. Another example is the federation protocol in Apache Wave, which is based on XMPP.^[49]

See also

- Comparison of instant messaging clients
- Comparison of instant messaging protocols
- Comparison of XMPP server software
- Secure communication
- SIMPLE
- Matrix

References

1. Johansson, Leif (April 18, 2005). "XMPP as MOM - Greater NOrdic MiddlewAre Symposium (GNOMIS)" (<http://web.archive.org/web/20110510012207/http://www.gnomis.org/presentasjoner/oslo2005/xmpp.pdf>) (PDF). Oslo: University of Stockholm. Archived from the original (<http://www.gnomis.org/presentasjoner/oslo2005/xmpp.pdf>) (PDF) on May 10, 2011.
2. Saint-Andre, P. (March 2011). *Extensible Messaging and Presence Protocol (XMPP): Core* (<https://tools.ietf.org/html/rfc6120>). IETF. RFC 6120. <https://tools.ietf.org/html/rfc6120>. Retrieved May 4, 2014.
3. "Jabber Inc" (<http://www.cisco.com/web/about/ac49/ac0/ac1/ac258/JabberInc.html>). Cisco.com. Retrieved 2012-11-24.
4. "Jabber Instant Messaging User Base Surpasses ICQ" (<https://web.archive.org/web/20071103080257/http://www.xmpp.org/xsf/press/2003-09-22.shtml>) (Press release). XMPP Standards Foundation. September 22, 2003. Archived from the original (<http://xmpp.org/xsf/press/2003-09-22.shtml>) on November 3, 2007. Retrieved November 30, 2007.
5. "Open Real Time Messaging System" (<http://tech.slashdot.org/article.pl?sid=99/01/04/1621211>). Tech.slashdot.org. 1999-01-04. Retrieved 2012-11-24.
6. Chatting Up the Chef (<http://www.linuxjournal.com/article/6489>) *Linux Journal* March 1, 2003 by Marcel

Gagne

7. "Jabber.org – XMPP Server Migration" (<https://web.archive.org/web/20091213180443/http://www.jabber.org/2009/08/xmpp-server-migration/>). August 12, 2009. Archived from the original (<http://www.jabber.org/2009/08/xmpp-server-migration/>) on December 13, 2009. Retrieved December 14, 2009.
8. Burd, Gary (January 17, 2006). "XMPP Federation" (<http://googletalk.blogspot.com/2006/01/xmpp-federation.html>). Retrieved November 30, 2007.
9. "Google's chat client drops Jabber compatibility" (<http://www.h-online.com/open/news/item/Google-s-chat-client-drops-Jabber-compatibility-1866129.html>). Heise Online. May 20, 2013. Retrieved May 27, 2013.
10. Jensen, Florian (2008-01-17). "AOL adopting XMPP aka Jabber" (<http://florianjensen.com/2008/01/17/aol-adopting-xmpp-aka-jabber/>). Archived (<https://web.archive.org/web/20080120143857/http://florianjensen.com/2008/01/17/aol-adopting-xmpp-aka-jabber/>) from the original on January 20, 2008. Retrieved 2008-01-17.
11. "AOL XMPP Gateway" (<http://www.aim.com/xmpp>). 2011-05-14. Archived (<https://web.archive.org/web/20110522092331/http://www.aim.com/xmpp>) from the original on May 22, 2011. Retrieved 2011-05-14.
12. "Cisco Announces Definitive Agreement to Acquire Jabber" (https://web.archive.org/web/20091223033256/http://newsroom.cisco.com/dlls/2008/corp_091908.html). Archived from the original (http://newsroom.cisco.com/dlls/2008/corp_091908.html) on December 23, 2009. Retrieved January 2, 2010.
13. "Facebook Chat Now Available Everywhere" (<http://blog.facebook.com/blog.php?post=297991732130>). Retrieved February 11, 2010.
14. "Chat API (deprecated)" (https://developers.facebook.com/docs/chat/?_fb_noscript=1). Retrieved July 5, 2015.
15. Obasanjo, Dare (2011-12-14). "Anyone can build a Messenger client—with open standards access via XMPP" (http://windowsteamblog.com/windows_live/b/windowslive/archive/2011/12/14/anyone-can-build-a-windows-live-messenger-client-with-open-standards-access-via-xmpp.aspx). Windowsteamblog.com. Retrieved 2012-11-24.
16. Roettgers, Janko (2011-06-28). "Skype adds XMPP support, IM interoperability next? — Tech News and Analysis" (<http://gigaom.com/2011/06/28/skype-xmpp-support/>). Gigaom.com. Retrieved 2012-11-24.
17. RFC 6122
18. "Lotus Sametime 7.5 Interoperates with AIM, Google Talk", eWeek, December 6, 2006 Eweek.com (<http://www.eweek.com/c/a/Messaging-and-Collaboration/Lotus-Sametime-75-Interoperates-with-AIM-Google-Talk/>)
19. "Lotus ships gateway to integrate IM with AOL, Yahoo, Google", Network World, December 6, 2006 Networkworld.com (<http://www.networkworld.com/news/2006/120606-sametime-links-up-with-aim.html>) Archived (<https://web.archive.org/web/20081101072959/http://www.networkworld.com/news/2006/120606-sametime-links-up-with-aim.html>) November 1, 2008, at the Wayback Machine.
20. "Unified Communications: Uniting Communication Across Different Networks", Microsoft Press Release, October 1, 2009 Microsoft.com (<https://www.microsoft.com/Presspass/Features/2009/oct09/10-01UCInterop.msp>) Archived (<https://web.archive.org/web/20100105062129/https://www.microsoft.com/Presspass/Features/2009/oct09/10-01UCInterop.msp>) January 5, 2010, at the Wayback Machine.
21. <https://xmpp.org/extensions/xep-0030.html>
22. "Clients" (<http://xmpp.org/xmpp-software/clients/>). *xmpp.org*.
23. Joe Hildebrand; Craig Kaes; David Waite (2009-06-03). "XEP-0025: Jabber HTTP Polling" (<http://xmpp.org/extensions/xep-0025.html>). Xmpp.org. Retrieved 2012-11-24.
24. Ian Paterson; Dave Smith; Peter Saint-Andre; Jack Moffitt (2010-07-02). "XEP-0124: Bidirectional-streams Over Synchronous HTTP ([BOSH])" (<http://xmpp.org/extensions/xep-0124.html>). Xmpp.org. Retrieved 2012-11-24.
25. "Question FAQ #270-What is LJ Talk?" (<http://www.livejournal.com/support/faqbrowse.bml?faqid=270>).

- Livejournal.com. 2010-09-27. Retrieved 2012-11-24.
26. Leigh, Dylan. "Introduction to Jabber/XMPP" (http://www.dylanleigh.net/notes/jabber-intro.html#Frequently_Asked_Questions). *www.dylanleigh.net*. Retrieved 2017-10-16.
 27. Rémond, Mickaël (2014-02-23). "Whatsapp, Facebook, Erlang and realtime messaging: It all started with ejabberd" (<https://blog.process-one.net/whatsapp-facebook-erlang-and-realtime-messaging-it-all-started-with-ejabberd/>). *ProcessOne — Blog*. Retrieved 2017-10-16.
 28. "Origin game platform sends login and messages in plain-text" (<https://www.slightfuture.com/security/ea-origin-chat-unencrypted>). Slight Future. 2015-06-30. Retrieved 2016-02-22.
 29. "libsteam.c" (<https://github.com/EionRobb/pidgin-opensteamworks/blob/master/steam-mobile/libsteam.c>). Github. Retrieved 2016-02-22.
 30. "XEP-0004: Data Forms" (<http://xmpp.org/extensions/xep-0004.html>). *xmpp.org*.
 31. "XEP-0030: Service Discovery" (<http://xmpp.org/extensions/xep-0030.html>). *xmpp.org*.
 32. "XEP-0045: Multi-User Chat" (<http://xmpp.org/extensions/xep-0045.html>). *xmpp.org*.
 33. "XEP-0060: Publish-Subscribe" (<http://xmpp.org/extensions/xep-0060.html>). *xmpp.org*.
 34. "XEP-0163: Personal Eventing Protocol" (<http://xmpp.org/extensions/xep-0163.html>). *xmpp.org*.
 35. "XEP-0071: XHTML-IM" (<http://xmpp.org/extensions/xep-0071.html>). *xmpp.org*.
 36. "XEP-0096: SI File Transfer" (<http://xmpp.org/extensions/xep-0096.html>). *xmpp.org*.
 37. "XEP-0115: Entity Capabilities" (<http://xmpp.org/extensions/xep-0115.html>). *xmpp.org*.
 38. "XEP-0322: Efficient XML Interchange (EXI) Format" (<http://xmpp.org/extensions/xep-0322.html>). *xmpp.org*.
 39. "XEP-0323: Internet of Things - Sensor Data" (<http://xmpp.org/extensions/xep-0323.html>). *xmpp.org*.
 40. "XEP-0324: Internet of Things - Provisioning" (<http://xmpp.org/extensions/xep-0324.html>). *xmpp.org*.
 41. "XEP-0325: Internet of Things - Control" (<http://xmpp.org/extensions/xep-0325.html>). *xmpp.org*.
 42. "XEP-0326: Internet of Things - Concentrators" (<http://xmpp.org/extensions/xep-0326.html>). *xmpp.org*.
 43. "XEP-0347: Internet of Things - Discovery" (<http://xmpp.org/extensions/xep-0347.html>). *xmpp.org*.
 44. "Tech pages/IoT systems" (http://wiki.xmpp.org/web/Tech_pages/IoT_systems). *xmpp.org*.
 45. "IOT Info Page" (<http://mail.jabber.org/mailman/listinfo/iot>). *jabber.org*.
 46. "XMPP rises to face SIMPLE standard", Infoworld magazine, April 17, 2003 XMPP rises to face SIMPLE standard (<http://www.infoworld.com/d/developer-world/xmpp-rises-face-simple-standard-018>)
 47. "XMPP vs SIMPLE: The race for messaging standards", Infoworld magazine, May 23, 2003 Infoworld.com (<http://www.infoworld.com/t/platforms/xmpp-vs-simple-race-messaging-standards-295>)
 48. "Specifications" (<http://xmpp.org/extensions/index.html>). *xmpp.org*. XSF. p. 1. Retrieved January 6, 2017.
 49. "Google Wave Federation Protocol" (<http://www.waveprotocol.org/draft-protocol-specs/draft-protocol-spec#intro-overview>). Google.

External links

- Official website (<http://xmpp.org/>)
- xmpp-iot.org - the XMPP-IoT (Internet of Things) initiative (<http://xmpp-iot.org>)
- Real-Time Communications Quick Start Guide (<http://rtcquickstart.org/>)
- Jabber User Guide (<http://archive.jabber.org/userguide/>)
- "IETF Publishes XMPP RFCs: Core Jabber Protocols Recognized As Internet-Grade Technologies" (<https://web.archive.org/web/20091024103254/http://xmpp.org/xsf/press/2004-10-04.shtml>). Oct 4, 2004. Archived from the original (<http://xmpp.org/xsf/press/2004-10-04.shtml>) on October 24, 2009.
- "Peter Saint-Andre on Jabber/XMPP" (<https://twit.tv/shows/floss-weekly/episodes/49>), *FLOSS Weekly*, Twit TV, Dec 7, 2008, interviewed by Randal Schwartz and Leo Laporte

Retrieved from "<https://en.wikipedia.org/w/index.php?title=XMPP&oldid=806229134>"

This page was last edited on 20 October 2017, at 16:08.

Text is available under the Creative Commons Attribution-ShareAlike License; additional terms may apply. By using this site, you agree to the Terms of Use and Privacy Policy. Wikipedia® is a registered trademark of the Wikimedia Foundation, Inc., a non-profit organization.

- [Contact Wikipedia](#)
- [Developers](#)
- [Cookie statement](#)