

Jabber e XMPP: protocolo extensível para
mensagens e presença

Thadeu Lima de Souza Cascardo

20050801

1 Introdução

Desde o surgimento do ICQ, os serviços e aplicações de mensagem instantânea e presença têm sido populares na Internet. Muitos outros protocolos surgiram, como o Yahoo! Messenger, o AIM e, mais popularmente hoje em dia, o MSN.

Entretanto, estes protocolos não são produzidos e documentados de forma aberta, como são outros protocolos na Internet, aí incluindo os protocolos IP, TCP, HTTP, SMTP, entre outros. [1] [2] [3] [4]

Em 1999, Jeremie Miller iniciou o desenvolvimento da tecnologia Jabber, com a implementação de um servidor. O objetivo era criar um protocolo aberto para mensagem instantânea e presença.

Pouco tempo depois, a IETF [5] havia produzido um modelo para mensagem instantânea e presença, mas sem um protocolo. Após mais de dois anos de trabalho da Jabber Software Foundation [6] junto à IETF para padronizar o Jabber, em outubro de 2004, foram publicados 4 RFCs especificando o XMPP: eXtensible Messaging and Presence Protocol. [7] [8] [9] [10]

2 Uma arquitetura para construção de aplicações extensíveis

Apesar de ter sido criado para a aplicação de mensagem instantânea e presença, o XMPP permite a construção de outras aplicações. Tais aplicações podem incluir publicação de conteúdo, jogos, aplicações comerciais, entre outros.

Devido à sua extensibilidade, permitida por ser basicamente um protocolo para transporte de informação estruturada em XML, qualquer um pode estender o protocolo para sua aplicação específica. A Jabber Software Foundation aceita, produz, discute, aprova e publica extensões ao padrão, denominadas JEPs, Jabber Enhancement Proposals [11]. Estas propostas são abertas e podem ser implementadas por qualquer instituição, sem pagamento de taxas [12]. Extensões proprietárias e não publicadas também podem ser produzidas, mas não são tidas como padrões. Elas podem ser padronizadas através do processo de submissão descrito no JEP-0001 [13].

3 Características

3.1 Comunicação distribuída

O XMPP permite que vários servidores sejam utilizados para a comunicação entre diversas entidades. Isto é possível pois o endereço utilizado para identificar uma entidade é da forma `name@domain/resource`, onde `domain` indica um domínio, assim como em endereços de email.

Isso garante que não há um ponto central de falha para todo o sistema nem que uma só entidade controla toda a comunicação através do protocolo, como acontece com outros serviços disponíveis por algumas instituições.

3.2 Internacionalizado

O XMPP determina que a codificação a ser utilizada para transmissão é o UTF-8 [14], codificação que permite o uso de caracteres Unicode. Os endereços também são internacionalizados através do `stringprep` [15], utilizando IDNs, Internationalized Domain Names para os domínios [16]. O uso de um atributo para a linguagem também permite localização dos textos transmitidos, seja para erros, mensagens de aplicação ou mesmo texto transmitido entre pessoas.

3.3 Segurança através de protocolos padrões

O XMPP utiliza de protocolos padronizados pela IETF para garantir a segurança. O SASL, Simple Authentication and Security Layer [17], é utilizado para autenticação, permitindo o uso dos vários métodos utilizados por outros protocolos que também utilizam do SASL, como o SMTP e o LDAP. [18] [19]

Outro protocolo utilizado é o TLS, Transport Layer Security [20], permitindo criptografia através de certificados. O TLS é o sucessor do SSL, ainda muito utilizado em protocolos como o HTTP, o LDAP, entre outros. O TLS já é utilizado com o SMTP e o LDAP também.

Esta segurança é oferecida entre pares que estabelecem uma conexão direta. Como as entidades se comunicam através de mensagens e presença enviadas através de um servidor ou mais, não há privacidade na comunicação entre essas entidades, a não ser que utilizem o protocolo documentado no RFC 3923 [10], que descreve como enviar "stanzas" criptografadas e assinadas, garantindo, assim, a privacidade e autenticidade da informação transmitida. Entretanto, este padrão não está difundido e outros também já foram de-

scritos na forma de JEPs. Portanto, ainda há de se definir um padrão de fato para a comunicação fim-a-fim de forma privada e autêntica.

3.4 Privacidade

Apesar de não permitir a transmissão de mensagens (e outras "stanzas") fim-a-fim de forma privada, o XMPP define listas para bloqueio de qualquer tipo de comunicação [8].

Esta mesma lista é utilizada para a implementação do estado invisível, que é justamente uma das diversas formas de privacidade desejada pelo usuário.

3.5 Organização de contatos

A lista de contatos, denominada "roster", é armazenada no servidor e permite a organização de itens em diversos grupos [8]. Esta lista é integrada com o sistema de subscrição, permitindo diferentes estados, utilizados pelo servidor como regra para entrega de presença para outras entidades.

3.6 Um mecanismo de migração

Mesmo antes de sua padronização pela IETF, o Jabber já oferecia um mecanismo que permitia a troca de mensagens e presença entre entidades Jabber e entidades de sistemas legados, através de componentes denominados "transports" ou "gateways".

Assim, um usuário que tivesse um registro em um outro sistema, poderia utilizá-lo para configurar o "gateway" e ter acesso a sua lista de contatos, à presença deles e poder enviar mensagens a usuários desse outro sistema.

4 Extensões

4.1 Descoberta de Informação

Uma das extensões [21] publicadas e padronizadas permite a descoberta de informações sobre uma entidade. Estas informações envolvem o tipo e identidade da entidade, bem como o suporte oferecido para diversas extensões.

Itens associados a uma entidade também podem ser descobertos através desta mesma extensão. É possível utilizar de uma hierarquia de itens, o que permitiria classificação destes. Uma aplicação comercial poderia utilizar

esta hierarquia para um catálogo de produtos, como é exemplificado no documento que descreve a extensão.

4.2 Formulários genéricos

Formulários podem ser utilizados por diversas extensões, permitindo a utilização de campos além dos exigidos ou propostos [22]. Tais formulários podem até mesmo ser preenchidos automaticamente, seja devido a uma configuração prévia pelo usuário ou a alguma preferência da aplicação ou mesmo a consulta a dados existentes em outra base, como um diretório LDAP.

4.3 "Chat" Multi-Usuário

O "chat" ou conversa entre mais de um usuários simultaneamente também é possível através de uma extensão [23]. Salas podem ser criadas e administradas. Entidades podem assumir diferentes papéis, seja como moderador, participante, visitante. Também há afiliações, como dono, administrador, membro e expulso. Cada um desses papéis e afiliações garantem privilégios distintos.

Esta extensão utiliza as extensões de descoberta e de formulário para identificação de tipos da sala (entre eles, temporária, persistente, pública, privada, moderada, protegida, entre outros) e para configurações, que incluem até "log" da sala.

O documento que descreve a extensão chega a apresentar um mapeamento de comandos do IRC para o protocolo MUC, como é denominado.

4.4 Formatação de Mensagens

Para permitir formatação de mensagens ou marcação de texto, uma adaptação do XHTML é utilizada conforme descrita em uma extensão [24].

Essa adaptação foi definida de forma a ser leve e atender a requisitos próprios de sistemas de mensagem instantânea.

4.5 Transferência de Arquivos

Diversos mecanismos podem ser utilizados para transferência de arquivos, incluindo transmissão de URLs, utilização de proxy SOCKS5 ou mesmo utilizando o roteamento padrão de mensagens do XMPP [25].

4.6 Publicação e Subscrição de conteúdo

Uma extensão particularmente útil é a denominada PubSub [26], que permite que conteúdo seja publicado e que entidades se inscrevam a este conteúdo, recebendo notificações de sua atualização.

Esta extensão é utilizada por outras que permitem publicação de diversas informações, como localização geográfica e física, música sendo ouvida pelo usuário, humor e atividade atual do usuário.

Também existe uma extensão para a publicação de "avatars", ou imagens associadas ao usuário através desta extensão.

4.7 Áudio e Vídeo

Para transmissão de áudio e vídeo, ou sessões multimídia em geral, existe a extensão TINS [27] que permite a negociação de uma sessão através do SDP, protocolo padrão. Este mecanismo é similar ao SIP, protocolo também padrão, atualmente muito utilizado para comunicação de voz sobre IP.

5 Implementações

Há diversas implementações do protocolo XMPP/Jabber, além das extensões aqui descritas. Muitas dessas implementações são multi-plataformas e há implementações para plataformas específicas. Uma grande parte dessas implementações está disponível sob uma licença de uso, distribuição e modificação consideradas livres pela Free Software Foundation, como GPL, LGPL, MPL, entre outras. Diversas linguagens são utilizadas, como C, Python, C++, Java, XUL. [28]

5.1 Servidores

Entre os servidores, podemos destacar o jabberd, o servidor originalmente desenvolvido por Jeremie Miller e diversos contribuidores. Atualmente, está sendo reformulado como jabberd2.

Outro que merece destaque é o ejabberd, escrito na linguagem Erlang, e razoavelmente popular.

5.2 Componentes

Há componentes disponíveis, como gateways para sistemas legados como ICQ, MSN, AIM e Yahoo! Messenger.

Há também componentes que implementam algumas das extensões descritas acima, como o MUC e o PubSub.

5.3 Clientes

Clientes para os mais variados gostos, desde clientes para console, GTK+, QT, Windows, Mozilla, Java, PDAs, etc.

6 Conclusão

O XMPP é um protocolo padronizado pela IETF para mensagem instantânea e presença. É um protocolo extensível e permite o desenvolvimento de diversas outras aplicações. A JSF tem cumprido muito bem seu papel em padronizar as extensões e há um grande potencial para a implementação de diversas das extensões já publicadas.

7 Bibliografia

- [1] <http://www.ietf.org/rfc/rfc791.txt>
- [2] <http://www.ietf.org/rfc/rfc793.txt>
- [3] <http://www.ietf.org/rfc/rfc2616.txt>
- [4] <http://www.ietf.org/rfc/rfc2821.txt>
- [5] <http://www.ietf.org/>
- [6] <http://www.jabber.org/>
- [7] <http://www.ietf.org/rfc/rfc3920.txt>
- [8] <http://www.ietf.org/rfc/rfc3921.txt>
- [9] <http://www.ietf.org/rfc/rfc3922.txt>
- [10] <http://www.ietf.org/rfc/rfc3923.txt>
- [11] <http://www.jabber.org/jeps/>
- [12] <http://www.jabber.org/jsf/ipr-policy.shtml>
- [13] <http://www.jabber.org/jeps/jep-0001.html>
- [14] <http://www.ietf.org/rfc/rfc3629.txt>
- [15] <http://www.ietf.org/rfc/rfc3454.txt>
- [16] <http://www.ietf.org/rfc/rfc3490.txt>
- [17] <http://www.ietf.org/rfc/rfc2222.txt>

- [18] <http://www.ietf.org/rfc/rfc2554.txt>
- [19] <http://www.ietf.org/rfc/rfc2829.txt>
- [20] <http://www.ietf.org/rfc/rfc2246.txt>
- [21] <http://www.jabber.org/jeps/jep-0030.html>
- [22] <http://www.jabber.org/jeps/jep-0004.html>
- [23] <http://www.jabber.org/jeps/jep-0045.html>
- [24] <http://www.jabber.org/jeps/jep-0071.html>
- [25] <http://www.jabber.org/jeps/jep-0096.html>
- [26] <http://www.jabber.org/jeps/jep-0060.html>
- [27] <http://www.jabber.org/jeps/jep-0111.html>
- [28] <http://jabberstudio.org/>