

基于XMPP协议文件传输的研究与实现

施济瑜, 苗放, 王华军, 李刚

(成都理工大学 信息工程学院, 四川 成都 610059)

摘要: 在 C/S 网络模式下, 利用 Java 语言及开源的开发工具 Eclipse, 结合 SWT 控件实现基于 XMPP 协议的即时通信系统, 系统主要包括单人聊天、群聊天、视频和语音聊天、文件传输、数字签名、广告播放等功能模块, 主要介绍文件传输部分——用于系统中传输文件和数字签名文件, 通过实例应用表明, 采用基于 XMPP 协议进行 XML 流的传输优于其它 IM 通用协议。

关键词: C/S 网络; XMPP 协议; 即时通信; 文件传输; 数字签名; XML 流

Design and Implementation of File Transfer Based on XMPP Protocol

Shi Jiyu, Miao Fang, Wang Huajun, Li Gang

(College of Information Engineering, Chengdu University of Technology, Chengdu 610059, China)

Abstract: In C/S network mode, the use of open-source Java language and development tools Eclipse, with SWT controls to achieve an agreement based on the XMPP instant messaging system. System include: single chat, chat groups, video and voice chat, file transfer, digital signatures, advertising and other functional modules. Mainly introduces file transfer part of the system – used to in the transmission system for documents and digital signature file. Applications show that: Based on the XMPP protocol with XML flow of transmission is superior to other IM general agreement.

Key words: C/S network; XMPP Protocol; instant messaging; file transfer; digital signature; XML stream

0 引言

随着我国计算机技术及网络技术的迅速发展, XMPP(The Extensible Messaging and Presence Protocol) 协议^[1]在计算机通信中发挥着越来越重要的作用。XMPP 的出现, 实现了整个即时通信服务协议的统一。有了这个统一的协议之后, 使用任何一个组织或者个人提供的即时通信服务, 都能够无障碍地与其他即时通信服务的用户进行交流。目前, 即时通信领域正在掀起一股新的潮流, 世界最大的搜索引擎公司 google 推出的 google talk 就是一款基于 Jabber /XMPP 协议的即时通信软件。在 JAVA 的编程中, 客户端对 XMPP 协议的使用一般可通过 Smack 类接口来处理, 使程序员从繁琐的协议定义中脱离出来, 完全专注于用户界面的开发。

1 XMPP协议标准

1.1 XMPP 协议概述及其工作原理

XMPP 是开放协议和内核代码的急促即时消息处理系统 (Jabber IM) 的基准协议技术, 它提供了一种开放式的、基于 XML (Extensible Markup Language) 的、能在分布式网络中传输即时消息和在线发现的标准, 并解决了不同 IM (Instant Messaging) 系统间互操作的问题^[2-3]。XMPP 在设计上沿袭了

Internet 上最成功的消息系统, 即 email。其路由处理的内核采用国际惯常的逻辑寻址机制, 代表格式为: 节点 @ 域 / 资源。在 XMPP 协议中, 这种模式被称为 Jabber ID (JID)。其中, 域可在 DNS 中查找, 类似于电子邮件地址中的域名部分; 节点可表示某 IM 用户、一类应用或某项服务; 资源为一类连接标识, 能让某单一用户多次重复登录连接。每个用户都有自己的本地服务器 (即自己的注册服务器), 并从该服务器上接受信息, 所有从一个客户端发给另一个客户端的消息和数据都必须通过服务端。每一个 XMPP 服务器都独立于其他 XMPP 服务器, 并且拥有其自身的用户列表, 通过 Internet, 这些服务器构成了一个类似 e-mail 系统的分布式网络。服务器知道一个用户什么时候在线, 这个能力被称为在线, 也是即时通信的核心所在。XMPP 通过两个重要特性提供这些 IM 标准功能: 首先是一个允许消息系统间协同作业的开放协议; 其次是建立在 XML 上的强大根本, 它使得非但是两个人之间的通信, 甚至是应用软件之间的通信成为了可能。XML 是 XMPP 信息传输的核心, 它最重要的作用是系统的底层可扩展性, 它能表述几乎任何一种结构化数据。XMPP 利用 XML 数据流进行客户端 - 服务器端以及服务器端 - 服务器端的通信。另一方面, 这也使得实现与第三方服务 (如: MSN Messenger、Yahoo Messenger、ICQ、AIM) 进行信息传输的“传输器”成为可能。XMPP 是由 Jabber 公司创立、用于现场消息路由处理的 XML 数据流协议^[4], 是即时消息处理系统的基准协议技术, 可以为网络间连接提供安全和易于实现的编程语言环境。XMPP 工作原理图如图 1 所示。

工作原理:

- (1) 节点连接到服务器。
- (2) 服务器利用本地目录系统中的证书对其认证。

收稿日期: 2008-07-09; 修回日期: 2008-08-27。

基金项目: 国家发改委项目 (发改办高技 [2006] 2535 号)。

作者简介: 施济瑜 (1984-), 男, 安徽人, 硕士研究生, 主要从事计算机软件与理论方向的研究。

苗放 (1958-), 男, 北京市人, 博士, 教授、博士生导师, 主要从事计算机网络与 3S 技术方向的研究。

王华军 (1964-), 男, 四川眉山人, 博士后, 教授、博士生导师, 主要从事通信及计算机方向的研究。

- (3) 节点指定目标地址, 让服务器告知目标状态。
- (4) 服务器查找、连接并进行相互认证。
- (5) 节点之间进行交互。

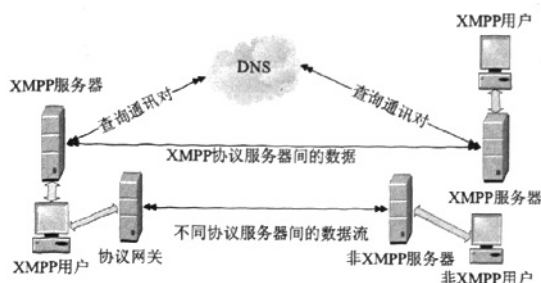


图1 XMPP工作原理图

1.2 即时消息协议的比较

目前, IM 通用协议包括 XMPP、SIMPLE、IMPP 共 3 个代表。

(1) XMPP 协议: 由 IETF 的 XMPP 工作组依据 RFC2779^[9] 种定义的需求, 结合了实际应用中 Jabber 协议制定。采用 C/S 系统模式, 有网关与其他即时消息服务器相连, 从而实现多种不同即时消息的互联互通。在应用领域, 主要以 Google Talk, Jive Messenger 为代表。XMPP 在 XML 结构化数据中扮演通用“传输层”的角色, 能使数据高效路由到最合适的请求源。除了实现 IM 的互操作性, XMPP 另一个重大用途是通过各类应用实现现场实时信息处理。比如, CRM、协同软件等都可作为 XMPP 的客户端, 所有实体对于 XMPP 服务器都是透明的, 实体中的任何用户的状态对与其他用户都是可见的。

(2) SIMPLE 协议: 由 IETF 的 SIMPLE 工作组织制定, 通过对于 SIP 协议的扩展, 使其支持 IMPS 服务。体系结构与 SIP 基本相同, SIMPLE 的目标是要用一种统一的形式来管理多种媒体形式。利用了 SIP 的请求重定向功能来发送 Presence 信息, 与 XMPP 所要实现的提供实时、可靠的异步数据传输服务的目标不同。

(3) IMPP 协议: 由 IETF 的 IMPP 工作组制定。包括了 IM 服务模型定义、通用功能接口的定义、统一数据信息的定义。用来构建一个具有空间接收、发布能力的即时信息系统。但在实际领域没有完整的应用。

2 基于XMPP协议的即时通信系统

2.1 XMPP 服务器

XMPP 的服务器由若干个组件组成, 他们分别完成逻辑上独立的功能。项目主要是在 Openfire 基础上进行扩展。服务器的内核是一个 XMPP 路由器。完成基本组件之间的数据包路由和交换。XMPP 服务器内核处理以下的公共任务组件:

- (1) 会话管理器: 负责客户端会话认证、在线状态、用户联系表等。
- (2) 数据存储 (XDB): 连接数据库系统、保存用户信息、通信日志等。XDB 包括 XDB_file 和 XDB_ldap 子组件, 在对用户认证时, 明文认证包发送给 XDB_file 子组件, 数字

认证包通过检查 LDAP (Light Directory Access Protocol) 用于 XDB_ldap 子组件。

(3) 连接管理器: 管理和客户端之间的连接。

(4) 服务器连接器: 管理 XMPP 服务器之间的连接。

(5) 传输器: 建立 XMPP 服务器与非 XMPP 服务器通信。

2.2 XMPP 客户端

项目的 XMPP 客户端界面设计主要采用 SWT, 业务处理继承 Smack 库。XMPP 客户端的形态也是多种多样的, 有基于消息模式, 支持端到端的, 比如 JiveMessage 的 Spark 客户端; 有通过 Web 浏览器接入 XMPP 客户端的, 像 Google Talk, Jwchat 等。

3 基于XMPP协议的文件传输设计与实现

基于 XMPP 协议的文件传输主要采用两套方案: (1) 对于同一局域网内用户的文件传输主要通过 Sockect (套接字) 进行点对点文件的互发; (2) 在 Sockect (套接字) 建立不了连接的情况下使用服务代理来处理文件传输。

3.1 XMPP 协议的应用

XMPP 协议主要定义了三个顶级 XML 元素 <message>, <presence>, <iq> (info/query)^[6], 每一个元素通过属性和名字空间包含大量的数据, 这些属性和名字空间都是 XMPP 协议的组成部分。在本文的文件传输中主要用到了 XMPP 协议的 <message> 顶级元素, 将文件的二进制流信息通过设置属性和对应的名字空间在 Internet 上从一客户机传输到另一客户机。

3.2 使用 Socket 技术

Socket 是建立在传输层协议 (主要是 TCP 和 UDP) 上的一种套接字规范。Socket 屏蔽了底层通信软件 and 具体操作系统的差异, 使得任何 2 台安装了 TCP 协议软件和实现了 Socket 规范的计算机之间的通信成为可能。Socket 接口是 TCP/IP 网络最为通用的 API, 它也是在 Internet 上进行应用开发最常用的 API。开发人员可以直接调用 Java 中的 API 来实现基于 Socket 的文件传输。

3.3 文件传输的流程图

文件传输的基本流程如图 2 所示。

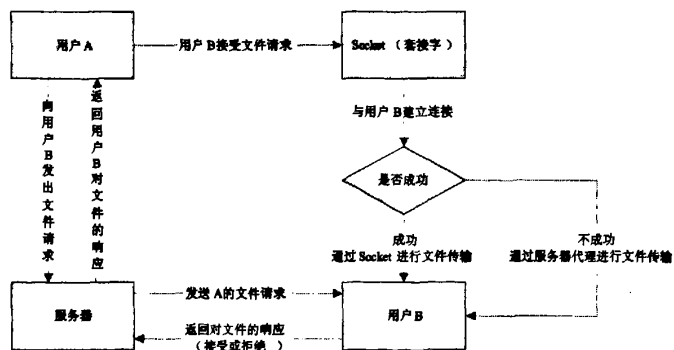


图2 文件传输的基本流程图

3.4 软件实现

发送请求、请求响应部分主要用到了 XMPP 协议的 3 个顶级元素中 <message> 顶级元素, 设置节点将接收方、发送方、文件流等信息通过 XML 流的形式进行传递。

(下转第 741 页)

信技术必将在民机、军机的机载设备中得到越来越广泛的应用。

参考文献:

[1] 陈云洽, 等. CPLD 应用技术与数字系统设计 [M]. 北京: 电子工业出版社, 2003.
[2] Device Engineering Inc. DEI 1016ARINC 429 Transceiver [Z].2000.

[3] Device Engineering Inc. BD429/RS-429 Line Driver Integrated Circuit [Z].2000.
[4] 董长飞. C8051F 系列单片机开发与 C 语言编程 [M]. 北京: 北京航空航天大学出版社, 2005.
[5] 徐维. 数字电子技术与逻辑设计 [M]. 北京: 中国电力出版社, 2006.

(上接第 733 页)

3.4.1 文件发送与实现

(1) 建立 ServerSocket 监听的主要代码
启动一个线程建立 ServerSocket 监听,
public void run(){
// 启动 7777 端口监听
ServerSocket listeningSocket = new ServerSocket
(7777);
Socket conn = listeningSocket.accept();
}

并向发送方发出两组 IP 信息: 一组本地 IP 和服务器 IP 信息

一组带 proxy.IP 和服务器 proxy.IP 信息

(2) 在建立的 Socket 连接中传输数据的主要代码
InputStream inputStream = null;
OutputStream outputStream = null;
try {
// 待发送文件的输入流 file 为文件对象
inputStream = new FileInputStream(file);
// 通过建立 Socket 套接字的输出流
outputStream = new BufferedOutputStream
(conn.getOutputStream());
// 读取输入流中的数据写入输出流中
writeToStream(inputStream,outputStream);
}

3.4.2 文件接收与实现

(1) 建立与发送方连接的主要代码

接收方收到发送方的两组 IP 信息进行解析, 首先直接连接发送方

// 与发送方建立连接, address 为发送方 IP
Socket socket = new Socket(address,7777);

如果连接建立不上将该连接 IP 地址放入不能直接建立连接的 List 列表中。并进行带有 proxy.IP 的连接, 该 proxy.IP 是通过服务器代理的连接。黑名单 IP 列表主要是方便客户端在发送第二次文件时如果该 IP 已在 List 中则直接采用服务器代理连接。

(2) 在建立的 Socket 连接中传输数据的主要代码

InputStream inputStream = null;
OutputStream outputStream = null;
try {
// 待保存文件的输出流 file 为保存文件对象
outputStream = new FileOutputStream (file);
// 通过任务的形式获取 Socket 套接字中的输入流
FutureTask<InputStream> streamNegotiatorTask

= new FutureTask<InputStream>
(new Callable<InputStream>()
{

public InputStream call() throws Exception {
// 返回输入流
return streamNegotiator
.createIncomingStream(recieveRequest
.getStreamInitiation());
}
});
// 任务启动
streamNegotiatorTask.run();
启动任务从建立的 Socket 套接字中获取连接输入流。

4 系统测试

在系统测试中, 对于同一个局域网的两客户端在传输率为 100Mbps 的以太网网线传输速度可以达到 5~9MB/s, 对于处于不同局域网的两客户端可以达到 20~200kB/s。以及常用传输的对比如表 1 所示。

表 1 系统测试结果列表

比较	XMPP 协议文件传输	QQ 文件传输	飞鸽传输
局域网	5-9MB/S	1M左右	1-10MB/S
非局域网	20-200KB/S	10-100KB/S左右	不能传输

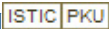
5 结束语

通过开源开发工具 Eclipse 结合 SWT 控件开发了基于 XMPP 协议的即时通信系统, 文件传输只是作为系统的最重要的组成部分, 用于传输普通的文件及在系统中用到的数字签名文件。基于 XMPP 协议的文件传输也是基于 XML 流的传输方式传输数据, 为以后不同风格代码和外来模块的加入打下了良好的基础, 提高了扩展能力, 可集成于基于 XML 数据处理的应用系统中。由于该模块是用 Java 编写可在不同的操作平台上运行, 有很好的移植性。通过系统的测试, 基于 XMPP 协议的文件传输也具有一定可靠性和安全性, 并且相对其他软件的文件传输具有一定的优势。

参考文献:

[1] RFC 3920 ,Extensible Messaging and Presence Protocol (XMPP): Core/ Introduction[S].
[2] RFC 3920- 2000, Instant messaging/presence protocol requirements [S].
[3] PeterSA.Draft-ietf-xmpp-core-224XMPP:core [EB/OL] http://www.xmpp.org/specs/draft-ietf-xmpp-core-24.htm1, 2004205201.
[4] XMPP RFCS. (Request For Comments, Internet 标准草案) [S]. 2004.
[5] RFC 2779- 2000, Instant messaging/presence protocol requirements [S].
[6] Peter S A. XMPP Instant messaging and presence[EB]. RFC 3921, 2004.

基于XMPP协议文件传输的研究与实现

作者: [施济瑜](#), [苗放](#), [王华军](#), [李刚](#), [Shi Jiyu](#), [Miao Fang](#), [Wang HuaJun](#), [Li Gang](#)
作者单位: [成都理工大学, 信息工程学院, 四川, 成都, 610059](#)
刊名: [计算机测量与控制](#) 
英文刊名: [COMPUTER MEASUREMENT & CONTROL](#)
年, 卷(期): 2009, 17 (4)
被引用次数: 1次

参考文献(6条)

1. [Extensible Messaging and Presence Protocol \(XMPP\):Core/Introduction](#)
2. [RFC 3920-2000. Instant messaging/presence protocol requirements](#)
3. [PeterSA Draft-ietf-xmpp-core224XMPP:core](#) 2004
4. [XMPP RFCS. \(Request For Comments, Internet标准草案\)](#) 2004
5. [Instant messaging/presence protocol requirements](#) 2000
6. [Peter S A XMPP Instant messaging and presence](#) 2004

引证文献(1条)

1. [周文琼](#), [王乐球](#), [周桐](#), [周春光](#) [基于XMPP的企业即时通信系统研究与应用\[期刊论文\]-吉林大学学报\(信息科学版\)](#) 2010 (1)

本文链接: http://d.wanfangdata.com.cn/Periodical_jsjzdclykz200904038.aspx

下载时间: 2011年1月28日