

文章编号:1001-9081(2005)03-0554-03

一种 P2P 网络安全信任模型的设计与实现

史艳芬,葛燧和

(同济大学 计算机科学与工程系,上海 200092)

(shiyf2001413@126.com)

摘 要:首先分析 P2P 应用的安全需求,然后基于 JXTA 设计了一种 P2P 安全信任模型,并实现了模型中的信任机制和对等点间通信的功能,确保了 P2P 应用的安全性。

关键词:P2P;JXTA;安全模型;信任机制

中图分类号:TP393.08 **文献标识码:**A

Design and realization of security and trust model for P2P network

SHI Yan-fen, GE Sui-he

(Department of Computer Science and Engineering, Tongji University, Shanghai 200092, China)

Abstract: This paper first analysed the security demands of P2P applications, then designed a P2P security and trust model based on JXTA. Finally, it implemented the trust mechanism of the model and communication among peers to insure the security of P2P applications.

Key words: P2P; JXTA; security model; trust mechanism

0 引言

P2P 技术是一种新兴的不依赖服务器的分布式网络模型,在对等计算、信息共享、分布式搜索等领域有着广泛的应用前景。它通过系统间对等点的直接交换实现网络信息和资源的共享,在这种网络中所有的节点都是对等的,真正实现了网络间的平等沟通。

P2P 这种分布式的网络模型,在系统安全中面临着巨大的挑战,它需要在没有中心节点的情况下,提供身份验证、授权、数据信息的安全传输、数字签名、加密等机制。还要避免大量信息传输拥塞、病毒的入侵和恶意代码攻击等问题。

我们将 P2P 网络的安全问题,归结为以下三类:1) 数据安全,包括数据的完整、真实和保密;2) 网络对等节点安全问题,对等机相互信息和个人环境安全;3) 信息安全问题,网络信息内容和版权的管理。

我们分析 P2P 网络目前的安全问题后,基于 SUN 公司开发的 JXTA 设计了一种安全信任模型,实现了模型中对等点信任机制的建立和对等点间的安全通信。模型真正达到了保障信息共享完整性、真实性和保密性的目的。

1 基于 JXTA 平台的安全机制

JXTA PROJECT 是 SUN 公司为了向构建跨平台、跨操作系统和跨编程语言的 P2P 应用提供实用应用程序底层而发动的网络工程。JXTA 提供了一套支持在 P2P 环境中对等点间实现互操作性的协议。在 JXTA 中,所有的协议都被定义为 XML 消息在对等点之间传送,它将对等机的行为作了标准化,主要包括相互发现并自组织加入对等机组、广告和发现网络服务以及对等通信和监视等。

JXTA 平台为 P2P 提供的安全机制主要有两方面,一方面是 JXTA 由 J2SE 开发实现,它继承了 Java 本身的安全性;

另一方面 JXTA 本身提供了像 PureTLS 以及其他一些底层应用协议。JXTA 平台对 P2P 网络中的安全问题提供了一定的解决策略:如 JXTA 中,为了保障数据传输的安全性,传输数据的途径由 API 控制。建立对等机 ID 和对等机组,在 P2P 网络中为对等机创立对等机 ID 可以唯一标记对等机,对等机组可以限制组中成员的某些行为,也可以验证组外成员的加入,在分散结构上加强了单点控制。建立信任机制,在信任的范围内和对等机组的成员相互交互,实现共享。

2 安全模型的设计与实现

基于 JXTA 以上特性,以及 P2P 网络的安全问题,我们提出了一种新型的 P2P 网络安全模型,图 1 中描述了该模型中对等点如何加入对等组,对等点信任机制的建立以及对等点间进行安全通信的机制。

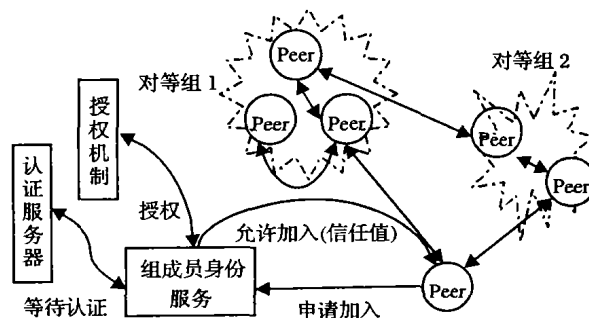


图 1 基于 JXTA 的安全模型

下面分别从模型的信任机制的建立,对等机之间安全通信等方面来讨论。

2.1 模型信任机制的建立

模型的信任机制通过综合认证、授权和加密三个标准建立:认证决定某些对等点实际上是否就是它们自称的对等点,而不是虚假或冒名的过程。授权可以授予一个认证过的对等

收稿日期:2004-08-03;修订日期:2004-10-25

作者简介:史艳芬(1979-),女,山西人,硕士研究生,主要研究方向:计算机网络、电子商务;葛燧和(1944-),男,上海人,副教授,主要研究方向:计算机网络、电子商务。

点实施某些行为或访问某些资源。加密可以保护对等点间传输的信息,与每个对等点的安全认证结合在一起,确保了交换数据不会在通信中被窃听。

模型对等机组的管理模式中,任何想要加入该组的对等机都必须申请成员关系,只有取得了该组的信任后,才能被授予一定权限,在这个权限内和组中的其他对等机交互信息。模型在保证数据传输安全,主要是加密的基础上,可以通过验证和授权来建立对等机之间的信任机制。下面描述的 MembershipService 提供了具体的成员服务接口。

```
public abstract interface Service{
    Advertisement getImplAdvertisement();
    Service getInterface();
}
//引用 net. jxta. service 包,提供服务接口
public abstract interface MembershipService extends Service {
    Authenticator apply( AuthenticationCredential application);
    Enumeration getAuthCredentials();
    Enumeration getCurrentCredentials();
    Service getInterface();
    Service getName();
    Credential join( Authenticator authenticator);    //申请加入组
    Credential makeCredential( Element element);    //授予信任
    Credential getDefaultCredential();
    void addPropertyChangeListener
        ( PropertyChangeListener propertyChangeListener);
    void removePropertyChangeListener
        ( PropertyChangeListener propertyChangeListener);
    void resign();
}
public final class AuthenticationCredential implements Credential {
    private String authenticationMethod;
}
```

一个对等机组当收到一个陌生的加入请求时, AuthenticationCredential 构造函数将会接收验证方法,然后调用目标对等机组的 getMembershipService 方法从中获得成员服务;并使用验证证书调用成员服务的 apply 方法,创建验证器(authenticator);最后测试 authenticator,若为真,则允许加入该组,若为假,则将抛出异常。在执行加入操作时,申请对象将是最终证书以保证其可以合法地使用服务,这其中也包括了授予的权限。通过这样的信任机制,P2P 网络的安全性加强,有利于大型对等网的运行和维护。

可见,认证、授权和加密的使用在 P2P 应用中可以建立起信任机制。

2.2 模型安全通信的实现

我们应用一个实例来描述 JXTA 平台下这个安全模型中任意两个对等点是如何交互通信的。

本实例中,我们构造了一个对等机组 PeerGroup,以及其中的成员 PeerA、PeerB、PeerC 和 PeerD,在它们之间建立通信。构造的每个对等点都包括有对等机 ID (PeerID)、信任值 (TrustID) 和传输地址 (TransportAddress) 等信息。如图 3 的程序流程图所示,将两个对等点通信的过程简单描述为:首先创建一个管道,然后创建一条消息,最后通过管道把消息发送到另一个对等机。

实例中,我们假设 PeerA 和 PeerB 要完成通信的过程。我们首先构造这两个对等实体,PeerA 用于提供服务,相当于 Service 实体,它建立一条输入管道,生成一个管道广告并且发布广告,通过监听管道,Service 等待从客户方来的消息。另外,我们建立的 PeerB 实体用于发现服务并且提供消息,相当于 User 实体,它在网络中发现 Service 实体发布的广告,生

成一条输出管道,创建一个消息并通过管道发送消息。

在对等点 PeerA 实体中完成这样的步骤:

- 1) 进行身份验证,确认得到信任,可以通信;
- 2) 创建一条管道广告 (myPipeAdv);
- 3) 生成一条基于该广告的输出管道 (myInpPipe);
- 4) 在对等组中发布一条管道广告 (myDiscovery publish);
- 5) 等待输入管道信息;
- 6) 用私钥解密接收的消息,确认成功。

在对等点 PeerB 实体中完成这样的步骤:

- 1) 查找发布的管道广告 (search res);
- 2) 生成一条基于该广告的输出管道 (dwoutpipe);
- 3) 创建一条消息 (dwMsg);
- 4) 用公钥加密消息,并进行数字签名;
- 5) 通过管道发送消息 (send dwMsg dwoutpipe)。

我们设计实现模型中的一些功能,引用一些包,其功能描述如表 1。

表 1 引用包及其功能描述

包	功能
net. jxta. pipe	管道中使用 InputPipe 和 Pipe 接口
net. jxta. message	消息中使用 Message 接口
net. jxta. Advertisement	广告中使用 PipeAdvertisement 接口
met. jxta. protocol	协议使用的类和接口

我们设计定义了类 GetPipeText,它定义要使用的 JXTA 对象类型 (广告、管道、消息等) 的 private 实例。类 LoginContext 完成对等点间身份认证的过程,定义了完成认证、授权、加密等功能的方法。类 DSA 完成对等点间信息的数字签名的过程,定义了加密、数字签名和解密的方法。

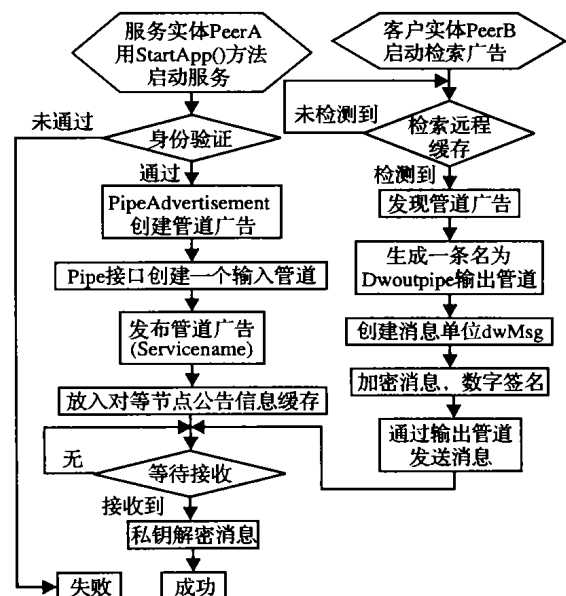


图 2 对等组中 PeerA 和 PeerB 的通信程序流程图

```
public class GetPipeText extends ShellApp {
    private PipeAdvertisement myPipeAdv = null;    //管道广告
    private InputPipe myInpPipe = null;          //输入管道
    private Message myMsg = null;                //消息
    private Discovery myDiscovery = null;         //发现广告
    private PeerGroup myGroup;                  //对等组实体
}

Public class LoginContext {
    Public LoginContext( string name, string subject);
```



```

Public void Login() {};
Public void Sign() {};
Public string getSubject() {};
}
//身份认证
public class DSA {
    public void Encryption() {};
    public void Decryption() {};
    public void Signature() {};
}
//数字签名

```

1) 服务实体 PeerA 在启动服务方法 StartApp() 后完成以下步骤:

a) 首先实体 PeerA 和 PeerB 进行相互的身份认证,我们用 LoginContext 类中的方法完成身份认证,确认得到信任后,用 AdvertisementFactory 静态类来创建管道广告,通过创建一个新的 PipeID 实例来创建 ID,每个 PipeID 都包含它从中被创建的对等组的 ID。

```

myPipeAdv = ( PipeAdvertisement ) AdvertisementFactory.
    newAdvertisement
    ( PipeAdvertisement. getAdvertisementType() );
myPipeAdv.setPipeID( new PipeID( group. getID() ) );
if ( myPipeAdv == null ) {
    println( "waitptext: Cannot create a Pipe Advertisement" );
    return ShellApp. appMiscError;
}
println( "created a pipe advertisement..." );

```

b) 在创建了管道广告之后,我们必须创建一条基于该广告的输出管道,通过使用 jxta. pipe. Pipe 接口完成这项任务:

```

myInpPipe = pipes. createInputPipe ( myPipeAdv )
c) 用原始的 JXTA 发现协议来发布这广告:
myDiscovery = group. getDiscovery();
myDiscovery. publish( myPipeAdv, Discovery. ADV );
println( "published the pipe advertisement to the group..." );
d) 继续等待着接收输入消息,假定可以等到为止:

```

```

println( "waiting at the input pipe for a message..." );
myMsg = myInpPipe. poll(0);

```

如果等待到了消息,用 DSA. Decryption() 方法解密消息,验证是否成功。

2) 客户实体 PeerB 通过以下步骤发送消息

a) 通过 getRemoteAdvertisements 方法查找远程服务 Servicename 的服务广告,将它放在本地缓存中:

```

myDiscovery. getRemoteAdvertisements
    ( null, DiscoveryService. ADV, "name", Servicename, 1 );

```

b) 一段时间后,对等点开始从本地机器检索服务广告:

```

res = myDiscovery. getLocalAdvertisements
    ( DiscoveryService. ADV, "name" Servicename );

```

c) 创建一条新消息:

```

Message dwMsg = pipes. createMessage();

```

d) 利用 DSA. Encryption() 和 DSA. Signature() 方法对消息进行加密和数字签名,最后生成一个输出管道,通过输出管道发送消息:

```

dwoutpipe = pipes. createOutputPipe( myPipeAdv, )
dwoutpipe. sendMessage( dwMsg );

```

两个对等实体的通信结束,停止服务。

参考文献:

- [1] DAVIDSON A. Peer-to-Peer File Sharing Privacy and Security [DB/OL]. Center for Democracy and technology, 2003.
- [2] YEAGER B. Enterprise strength security on a JXTA P2P network [A]. Peer-to-Peer Computing 2003 (P2P 2003) Proceedings [C], 2003. 7-8.
- [3] 周功业,黎书生. 新一代网络计算模型——P2P 及其 JXTA 体系结构的设计与实现[J]. 计算机应用研究, 2003, (9): 139-142.
- [4] 陈宇,唐旭章. 基于 P2P 系统的 JXTA 技术探析[J]. 计算机工程, 2002, 28(10): 18-19.
- [5] 邵丽炯,贺亮. 利用 JXTA 平台保障 P2P 安全的研究[J]. 微型电脑应用, 2004, 20(1): 19-22.

(上接第 545 页)

一般采用两个参数对这种变化进行描述,即像素变化率 (NPCR) 和平均变化强度 (UACI)^[7]。对于一幅原始图像,其加密图像为 C_1 ,若对其修改一个像素点的灰度值进行加密的结果为 C_2 ,比较灰度值矩阵 C_1 和 C_2 所有点的值。如果 $C_1(i, j) = C_2(i, j)$, 则 $D(i, j) = 1$, 否则为 0。

则 NPCR 和 UACI 可分别定义为:

$$NPCR = \frac{\sum_{i,j} D(i, j)}{M \times N} \times 100\% \quad (26)$$

$$UACI = \frac{1}{M \times N} \sum_{i,j} \frac{|C_1(i, j) - C_2(i, j)|}{255} \times 100\% \quad (27)$$

其中, M 和 N 分别代表图像的长与宽。

以 Lena 图像的加密结果为 C_1 , 以经过修改后 (改变第一个像素的灰度值) 的 Lena 图像的加密结果为 C_2 , 通过计算得到其 $NPCR = 0.0051$, $UACI = 0.2224$ 。这一结果表明对原始图像进行微小的改变会导致加密图像很大的变化,故算法具有较强的抵抗差分攻击能力。

正如文献[7]所指出,一个好的加密系统应该具有如下特征:对明文敏感、对密钥敏感以及将明文映射为随机暗文。这一点和混沌系统所具有的遍历性、对初值敏感性、对参数敏感性是很相似的。以上的讨论与分析表明,本文所讨论的算法具有较强的安全性和抵抗统计与差分攻击的能力。

2.2.4 算法效率

从算法上来看,本文所采用的加密/解密规则都具有迭代

结构,适合于用计算机快速计算。从明文角度看,对于图像矩阵,也不需要先进行拉直预处理,省掉了预处理时间。从密钥的产生来看,本文算法采用 128 位自定义密钥通过映射产生混沌序列的初始条件,具有对选择密钥敏感的特性。又由于算法中所采用的混沌系统具有强耦合特性,因此,算法的迭代轮数不需要选择太多,这些都有效地提高了算法的执行效率。

参考文献:

- [1] HOWARD C, LI XB. Partial encryption of compressed images and videos [J]. IEEE Transactions on Signal Processing, 2000, 48(8): 2439-2551.
- [2] DANG PP, CHAN PM. Image encryption for secure Internet multimedia applications [J]. IEEE Transactions on Consumer Electronics, 2000, 46 (3): 395-403.
- [3] CHENG YJ, GUO JN. A new chaotic key based design for image encryption and decryption [A]. ISCAS 2000 [C]. Geneva, Switzerland, 2000.
- [4] JESSA M. Data encryption algorithms using one - dimensional chaotic maps [A]. ISCAS2000 [C], 2000. 28-31.
- [5] 吕金虎,陆君安,陈士华. 混沌时间序列分析及其应用 [M]. 武汉:武汉大学出版社, 2002. 224226.
- [6] SHANNON CE. Communication theory of secrecy system [J]. Bell System Technical Journal, 1949, 28: 656-715.
- [7] CHEN GY, MAO YB, CHUI CK. A symmetric image encryption scheme based on 3D chaotic cat maps [J]. Chaos Solitons & Fractals, 2004, 21: 749-761.