第33卷 第23期 计 算 机 工 程 2007年12月
Vol.33 No.23 Computer Engineering December 2007
·安全技术· 文章编号：1000—3428(2007)23—0176—03 文献标识码：A 中图分类号：TP393.08

基于PIM-DM实时密钥的更新仿真

张 伟，许 勇，赵克淳
(安徽师范大学数学计算机学院，芜湖 241000)

摘 要：解决多播安全的主要措施是多播密钥管理，包括实时密钥管理和批量密钥管理。该文讨论了实时密钥更新管理中的密钥分发问题，分析了密集模式协议实现方法，并针对密集模式协议，添加了实时密钥更新管理模型，分别对星型结构和树型结构的密钥管理，利用NS仿真软件中的多播协议进行了仿真。结果增强了该协议的安全性。

关键词：多播安全；密钥管理；网络仿真

Simulation of Real-time Group Rekeying Based on PIM-DM

ZHANG Wei, XU Yong, ZHAO Ke-chun
(College of Mathematics and Computer Science, Anhui Normal University, Wuhu 241000)

【Abstract】 The main solution to multicast security is group key management, which includes real time key management and batch key management. This paper mainly discusses the group key distribution in real time group rekeying management, analyzes emphatically the realization of the multicast protocol, especially the PIM-DM protocol, directs at the PIM-DM protocol, and develops a model of real time group rekeying management. The author makes use of multicast protocol in software NS to conduct simulation study on key management of centralized mechanism and decentralized mechanism. As a result, the security of this protocol is enhanced.

【Key words】 multicast security; key management; network simulation

随着网络技术的不断发展，传统的数据业务(如FTP、HTTP、SMTP等)已经难以满足人们对信息业务的需求，多播作为20世纪80年代后期出现的新技术，以其自身的特点和优势在互联网中的应用越来越广泛，如视频点播、远程教学、新闻发布、视频会议等。它能够在网络上提供单点到多点以及多点到多点的通信，既可以避免单播所带来的效率低、负载重等弊端，又可以解决广播的盲目性、引发广播风暴等问题。然而，由于多播自身的特殊性和内在的复杂性，其可靠性和安全性变得比单播更为复杂[1]。在多播安全问题中，组密钥的安全管理已成为目前研究的焦点，而组密钥更新问题是组密钥安全管理的核心问题[2]。目前，对密钥更新的研究，人们主要提出两种更新策略[3]：实时密钥更新和批量密钥更新。实时密钥更新是在每次收到加入或离开请求后立即进行密钥更新，而批量密钥更新是在收到请求后先等待一段时间(rekey interval)，然后把这段时间收到的所有请求一起处理。对这两种策略的研究大多数还是停留在理论或者网络仿真阶段，利用网络仿真可以使得在虚拟网络中对已有的理论进行验证。本文主要针对实时密钥更新进行网络仿真研究。

网络仿真作为一种行之有效的、对实际网络进行模拟与分析的方法，在多播技术研究中同样有着重要的意义。但是目前大部分仿真软件，包括目前在网络研究领域使用较广泛的NS-2网络仿真器，都不提供对密钥更新的应用仿真。本文利用NS-2中的多播模拟器，对其中的多播协议PIM-DM进行分析并修改，添加对实时密钥更新的管理，并对星型结构和树型结构的组成员进行实时密钥更新仿真，对其仿真结果进行分析。

1 PIM-DM协议的分析

PIM-DM是密集模式的协议无关多播(PIM)路由协议，它可以利用现成的单播路由表来实现RPF校验功能，因此，与其他多播协议相比，它的开销降低了许多。与PIM-SM相比，它不需要周期性地发送加入消息，也没有汇合点*RP*，避免了单点失效问题[4]。PIM-DM的工作机制如下：采用“扩散-剪枝”的方式在最短路径树(SPT)中周期性(其值可由PruneTimeout来设置，默认为0.5s)广播多播信息，中间节点对多播信息进行反向路径转发，同时为组*G*和源*S*创建相应的多播路由项(*S*, *G*)，该路由项包括多播源地址、多播组地址、流入接口(iif)、流出接口列表(oiflist)和定时器等。当非组成员接收到该信息时，向上游发送剪枝信息，上游节点将二者之间的转发接口剪枝。如果该节点的所有转发接口剪枝，继续向上游转发剪枝信息，并将自身转为剪枝状态，剪枝状态对应着超时定时器，当定时器超时又重新变为转发状态。

当某成员加入多播组时，主动向上游发送嫁接报文(graft)。上游节点如果处于转发状态，直接将多播信息报文复制一份转发给刚加入成员；如果处于剪枝状态且不是多播源，继续向上游转发嫁接信息，并取消定时器，这时该节点变为转发状态。

当组成员离开多播组时，不是主动发送剪枝信息，而是先将该成员从组成员列表中删除。这样，当多播信息再次到达该节点时，才触发剪枝操作，其具体操作同“扩散-剪枝”方式中对非组成员的剪枝操作。

从对PIM-DM协议的分析来看，当成员加入或离开多播组时，并没有对组信息的安全性进行管理，即缺少组密钥管

基金项目：安徽省高校自然科学基金资助重点项目(2005KJ009ZD)
作者简介：张 伟(1984－)，男，硕士研究生，主研方向：计算机网络，网络安全；许 勇，博士、教授；赵克淳，硕士研究生
收稿日期：2006-12-31 **E-mail：**hzw53374616@163.com

理协议，因而不能保证“前向安全性”和“后向安全性”。为了解决这些问题，就必须对该协议进行修改，在已有协议的基础上添加密钥管理协议，既要保证不影响已有协议的运行，又要对组信息安全进行管理。

2 PIM-DM 协议的修改

实时密钥管理协议要求成员加入或离开多播组时，要立即对组密钥进行更新。因此，对 PIM-DM 协议的修改主要是针对其成员加入或离开时，中间节点接收到嫁接报文或剪枝信息的操作进行修改。当某成员加入多播组，中间节点收到该节点发送的嫁接报文时，首先要通知多播源进行组密钥更新，更新之后再对其转发多播信息。同样，当组成员离开多播组时，中间节点也要通知多播源进行组密钥更新。对已有的接受嫁接函数(*recv-graft*)和接受剪枝函数(*recv-prune*)进行修改，添加实时密钥更新管理。

2.1 主要算法描述

对每个中间节点引入一个成员数组 *M*，以记录当前节点的下一跳中属于多播组的节点，这样，当节点发送或转发密钥报文时可根据其成员数组 *M* 来实现报文的多播。由于在成员离开和周期性“扩散-剪枝”时都产生剪枝信息，为了区分还需设置一个标志位 *Flag*，缺省为 false，当有成员离开时置为 true。

接收嫁接函数：当中间节点收到嫁接信息时，先将发送或转发嫁接信息的节点添加到其数组 *M* 中，然后判断当前节点的活动状态，如果处于剪枝状态，则转发信息，否则单播密钥更新信息到多播源，多播源产生新的密钥报文对多播成员进行密钥更新。

接收剪枝函数：当节点收到成员离开的剪枝信息时，先将其数组 *M* 中删除发送或转发信息的节点，然后判断当前节点的活动状态，如果处于剪枝状态，则转发该信息，否则单播更新信息给多播源要求进行实时密钥更新。

2.2 算法实现框架

改进的 recv-graft 算法：

```
将发送或转发信息的节点加入当前节点的成员数组 M 中
If  当前节点不是多播源
    If  当前节点处于转发状态
        单播更新信息给多播源
    Else
        同 recv-graft ()函数操作
Else
    产生密钥报文并根据其成员数组 M 来更新组密钥
```

改进的 recv-prune 算法：

```
If  Flag 为 true
{
将当前节点的成员数组 M 中删除发送或转发信息的节点
If  当前节点不是多播源
If  当前节点处于剪枝状态
        转发剪枝信息
    Else
        单播更新信息给多播源
 Else
    产生密钥报文并根据其成员数组 M 来更新组密钥
置 Flag 为 false
}
设置定时器等同 recv-prune 操作
```

在算法实现的过程中涉及到更新信息和密钥报文的产生，因此要对 NS 中相关文件进行修改。首先要对新协议进行注册，其方法是在 packet.h 中添加更新信息和密钥协议的条目等，代码如下：

```
enum packet_t
{…
  PT_SECRET, PT_UPDATE,
};
class p_info
{
  public: p_info()
  {
    …
    name_[ PT_SECRET]= "secret";
    name_[ PT_UPDATE]= "update";
    }
    …
    }
```

然后对 mcast_ctrl.cc 文件中 command 函数进行修改，代码如下：

```
int command(int argc, const char*const* argv)
{
    if (argc == 4)
{
        if (strcmp(argv[1], "send") == 0)
{
            switch (*argv[2])
{
                …
                CASE('s', "secret", PT_ SECRET);
                CASE('u', "update", PT_ UPDATE);
            }
…}
```

目的是产生相关报文并发送，以及其他相关文件的修改，最后对 NS 重新编译。

3 实时密钥更新仿真

由于本文提出的改进方法在 NS 上实现，并进行了验证，下面先对 NS 机制中多播协议进行简单分析。

3.1 NS 中多播机制的分析

NS仿真软件中对多播协议的配置有 4 种模式[5]：核心模式(CtrMcast)，密集模式(DM)，共享树模式(ST)和双向共享树模式(BST)。核心模式、共享树模式和双向共享树模式属于多播路由协议中的稀疏模式，特点是采用共享树，使用显式加入模型，通过周期性向共享树发送加入消息来维护分支信息。密集模式(DM)类似于多播路由协议中的密集模式，特点是采用周期性“扩散-剪枝”方式和反向路径检查技术。它的运行模式有两种：pimdm模式(类似PIM-DM协议)和dvmrp模式(类似DVMRP协议)，可根据DM类中变量*CacheMissMode*的值来决定，默认值为pimdm。

3.2 NS 模拟实现及验证

下面利用 NS 软件对改进的 DM 协议进行仿真，由于协议的改进没有对测试脚本产生影响，脚本代码可参照文献[5]中多播路由一章。

星型和树型的拓扑结构如图 1、图 2 所示。

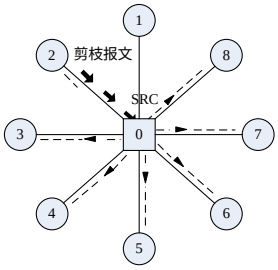

图 1　星型结构仿真示意图

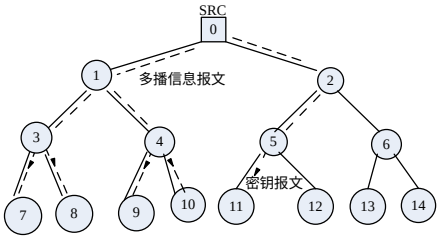

图 2　树型结构仿真示意图

在图 1 和图 2 中，虚线报文表示多播信息报文，三角形报文表示密钥报文，箭头报文表示剪枝报文，节点 0 为多播源(SRC)。图 1 显示的是：当节点 2 离开组时，多播源对其他组成员(节点 3~节点 8)进行密钥报文发送。图 2 显示的是：当节点 11 加入组中，多播源对组成员(节点 7~节点 11)进行密钥更新。由于密钥更新是实时更新，因此一旦有成员加入或离开组就可以触发密钥更新操作。

在模拟的过程中，对密钥报文进行跟踪，周期性计算链路中密钥报文的数量，其代码如下[5]：

```
set mcastmonitor [new McastMonitor]
set chan [open secret.tr w]     ;# 打开跟踪文件 secret.tr
$mcastmonitor attach $chan    ;# 把跟踪文件连接到监控对象上
$mcastmonitor set period_ 0.005   ;# 监控的时间间隔为 0.005s
$mcastmonitor trace-topo              ;# 跟踪所有链路
$mcastmonitor filter Common ptype_ $ptype(secret)
;# 对密钥报文(secret)进行过滤
$mcastmonitor print-trace             ;# 开始提取数据
```

对密钥报文的跟踪结果如图 3、图 4 所示。图中，Tree 曲线表示树型结构链路中密钥报文的变化曲线，Star 曲线表示星型结构链路中密钥报文的变化曲线。图 3 表示 8 个节点依次加入组时链路中密钥报文数量变化示意图，图 4 表示 8 个节点依次离开组时链路中密钥报文数量变化示意图。在模拟脚本中，其 8 个节点的加入时间是 0.1s~0.45s，离开时间是 0.6s~0.95s，相邻两个节点之间的间隔时间是 0.05s。从对两种拓扑结构的仿真结果来看，由于成员变化信息传递给多播源需要时间，在星型结构中密钥更新操作是在节点加入或离开后 0.01s 进行，树型结构在变化后 0.03s 进行，基本上实现密钥的实时更新，从而有效地实现组信息的“前向安全性”和“后向安全性”。从变化曲线来看，树型结构对密钥报文数量的缓冲比星型结构要好些，可以减缓链路中密钥报文传输的突发性对链路带宽产生的不利影响。从对已有协议的影响来看，密钥管理协议并没有对 PIM-DM 协议进行本质的改变，在不影响其执行性能上，增加对该协议的安全性。

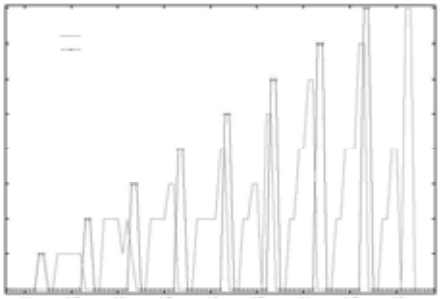

图 3　成员加入时密钥数量变化示意图

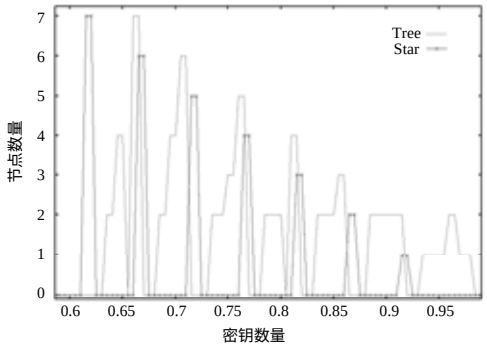

图 4　成员离开时密钥数量变化示意图

4　结束语

本文对PIM-DM协议进行了具体分析并修改，添加对实时密钥更新管理，并且对星型结构和树型结构进行实时密钥更新仿真，增强PIM-DM协议的安全性。由于文中讨论的是实时密钥管理，当成员加入或离开组时，组密钥会立即更新。组成员频繁变动时，会出现单个成员实时密钥更新算法存在低效和失序(out-of-sync)问题[6]，如何解决频繁变动的多播系统，对密钥进行批量更新管理，是下一步需要继续深入解决的问题。

参考文献

1 许　勇, 凌　龙, 顾冠群. 可靠可缩放安全多播密钥更新实现研究[J]. 计算机研究与发展, 2004, 41(6): 934-939.

2 Wallner D M, Harder E, Agee R C. Key Management for Multicast: Issues and Architectures[S]. RFC 2627, 1999-06.

3 马义忠, 付东亚, 易纪海, 等. TTR: 用于安全组密钥管理的改进批次更新方案[J]. 计算机工程与应用, 2005, 41(10): 149-152.

4 Adams A, Nicholas J, Siadak W. Protocol Independent Multicast——Dense Mode (PIM-DM): Protocol Specification (Revised)[S]. RFC 3973, 2005-01.

5 Fall K, Varadhan K. The NS Manual (Formerly NS Notes and Documentation)[EB/OL]. (2003-03). http://www.isi.edu/nsnam/ns/ns-documentation.html.

6 许　勇. 批量密钥更新中密钥组织方法的研究与实现[J]. 东南大学学报(自然科学版), 2006, 36(3): 488-492.