大型动态多播群组的密钥管理和访问控制*

刘 璟, 周明天

(电子科技大学 计算机科学与工程学院,四川 成都 610054)
E-mail: jingmliu@263.net; jingliu_uestc@yahoo.com.cn; mtzhou@uestc.edu.cn
http://www.uestc.edu.cn

摘要: 随着因特网用户的急剧增加和因特网不断的商业化,多播技术呈现出极为广阔的应用领域.在国际上,多播是一个崭新的学术研究领域,主要的研究成果集中在多播的路由算法、流量控制、拥塞控制和可靠传输上,多播安全领域的研究成果相对较少(尤其是在组通信密钥管理方面).研究了多播安全机制中的组通信密钥管理和访问控制问题.提出了一种基于子组安全控制器的组通信密钥管理和访问控制方案,该安全方案改进并解决了 IOLUS 系统和 WGL 方案中存在的若干问题,简化了访问控制策略,达到了预期的设计目标和要求.

关 键 词: 多播安全;组通信;组密钥管理;访问控制;密钥树
中图法分类号: TP309 文献标识码: A

1 多播安全机制的密钥管理问题

为了防止组通信被非授权用户访问,所有的组内成员必须共享一个组通信密钥,所有的组通信都通过这个组通信密钥加密.为了确保组通信的安全,组通信的一个基本要求是:每当有用户离开或加入多播组时,这个组通信密钥必须更新,以使离开后的成员无法访问目前的通信(称为后向安全性),而一个新加入的成员无法访问以前的通信(前向安全性).对一个大型动态多播群组,每当有成员变动时都要更新一次组通信密钥,系统要付出的开销是可想而知的.

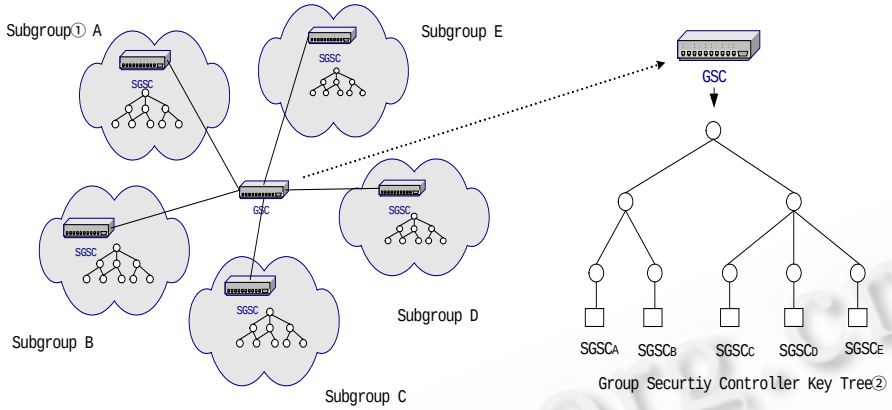
针对组通信的密钥管理问题,国际学术界提出一些解决方案,如:GKMP(group key management protocol)协议[3]、基于核心基本树路由算法 CBT(core-base trees)的可伸缩多播秘钥分配方案 SMKD(scalable multicast key distribution)[4]、MKMP(multicast key management protocol)秘钥管理协议[6]、HCD[5]、文献[8]、IOLUS[2]系统和 WGL 方案[1]等.其中 IOLUS 系统和 WGL 方案[1]是具有开创性的两种方案,因为前述的很多方案都或多或少地借鉴了 IOLUS 系统或 WGL 方案的思想.这些方案也就或多或少地存在和 IOLUS 系统或 WGL 方案相同的一些问题.

2 系统核心框架

我们的方案在系统设计上从始至终都贯彻了以下目标和要求:可伸缩性;路由协议的独立性;安全技术的独立性;系统的健壮性、灵活性和高效性.下面让我们分别介绍子组密钥树、子组安全控制器和组安全控制器密钥树的概念.密钥树的概念详见文献[1].

子组密钥树由子组安全控制器 SGSC(subgroup security controller)维护和管理.通过多播安全控制器的星型结构将相对简单的子组密钥树级联为一个统一的庞大而复杂的全局密钥树.如图 1 所示.

* 收稿日期: 2000-04-05; 修改日期: 2000-07-14
基金项目: 电子对抗国防科技重点实验室基金项目(NEWL9703)
作者简介: 刘璟(1972－),男,四川绵阳人,博士生,主要研究领域为计算机网络安全,分布对象技术;周明天(1939－),男,广西容县人,教授,博士生导师,主要研究领域为计算机网络,分布对象技术,并行分布处理,系统集成,虚拟现实.



① 子组 ② 组安全控制器密钥树

Fig. 1 Sketch Map of System Core Architechture

图1 系统核心框架示意图

整个系统框架的核心思想是:GSC(group security controller)通过 GSC 密钥树(如图 1 右部)向各 SGSC 安全发送更新后的组通信密钥.各 SGSC 一方面在子组成员变动时,通过子组密钥树更新子组组密钥;另一方面向各用户成员安全发送更新后的组通信密钥(该组通信密钥通过当前的子组组密钥加密).

3 组通信密钥管理策略和协议

子组密钥树更新算法和协议采用 WGL 方案中的两个协议(详见文献[1]中的图 6 和图 8).我们称为:多播组有成员加入时的密钥树更新协议(协议 1),多播组有成员离开时的密钥树更新协议(协议 2).算法和协议的具体说明以及特殊记号和表示法详见文献[1].

3.1 访问控制简化策略和组通信密钥的更新算法及协议

3.1.1 多播安全组的访问控制简化策略

本系统简化了多播安全组的访问控制策略.如图 2 所示,多播组 B 的一个子组 1 离开并加入到多播组 A.所有的认证过程只发生一次,即新加入的子组安全控制器与组安全控制器的相互认证过程.不再需要对该子组的每一个成员单独进行繁琐的认证过程(目前的所有方案都利用这种模式).这样,不仅解决了子组安全控制器不可信问题,而且组安全控制器的访问控制过程得到大幅度的简化,系统性能得到大幅度的提升.在以下算法和协议中,将用到以下两个概念:树的深度 h(指树中最长有向路径的长度:以边的数目计),树中接点的度数 d(指在树中与该点关联的边的数目),假设所有接点的度数相同.

3.1.2 系统核心算法和协议:组通信密钥更新算法和协议

组通信密钥更新算法和协议是整个系统的核心算法和协议.它解决了组通信密钥的安全更新、访问控制和子组安全控制器的不可信等问题.其中子组密钥树的更新算法和协议(由 SGSC 执行)基本上与 WGL 的子组密钥树更新算法和协议相同(协议 1 和 2).下面分 3 种情况简述我们的组通信密钥更新算法和协议:

1) 当另一个多播组的子组加入时

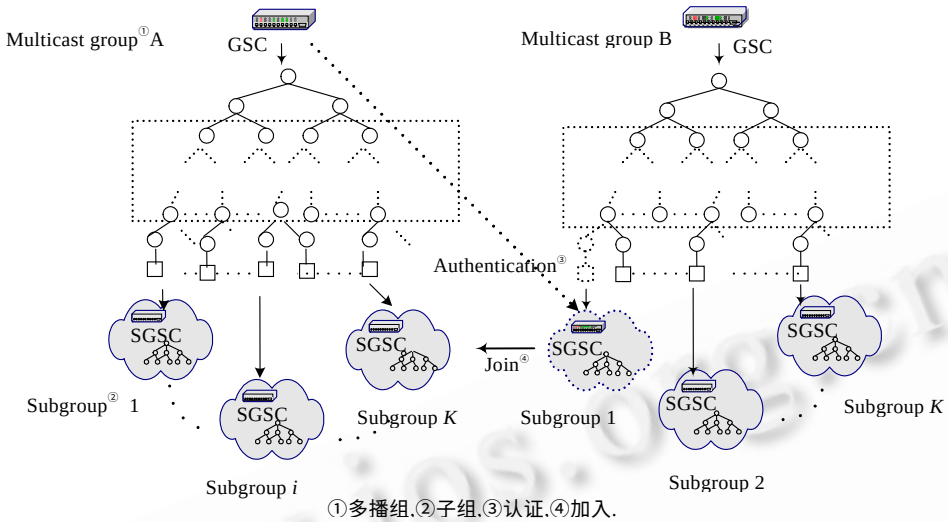
此时,组安全控制器密钥树的更新算法和协议与协议 1 基本相同.以图 2 为例(为简单起见,本文均以完全二叉树为例),组通信密钥更新协议(协议 3)如下:

(1) $SGSC_{B1}$→GSC_A:加入请求;

(2) GSC_A<=>$SGSC_B$:相互认证并发送会话密钥 $k_{SGSC_{B1}}$ (该密钥同时又为该子组的子组组密钥);

(3) GSC_A:生成新的组通信密钥 K',同时执行协议 1 更新组安全控制器密钥树;

(4) A 中各 $SGSC_i$(包括 $SGSC_{B1}$)→{子组 i 的所有成员}: $\{K'\}_{K_{SGSC_i}}$ 以多播方式发送.



①多播组,②子组,③认证,④加入.

Fig. 2 Sketch map of the core algorithm

图 2 核心算法示意图

注意,多播组初始化时,同样执行该算法和协议.以图 2 为例,多播组 A 和 B 均有 K 个子组且每个子组有 L 个成员.GSC_A 需要付出的加密计算开销是 $2(h-1)=2\log_2 K$,需要发送的 rekey 信息包数是为 $h=1+\log_2 K$.各 $SGSC$ 需要付出的加密计算开销是 1,rekey 信息包数是 1(注,$SGSC$ 的产生算法相对简单,篇幅所限故略去).

2) 当一个子组离开时

此时,子组安全控制器树的更新算法和协议与协议 2 基本相同.以图 2 为例,组通信密钥更新协议(协议 4)如下:

(1) $SGSC_k \rightarrow GSC_B$: $\{$离开 – 请求$\}_{k_{SGSC_i}}$;

(2) $GSC_B \rightarrow SGSC_i$: $\{$离开 –同意$\}_{k_{SGSC_i}}$;

(3) GSC_B:生成新的组通信密钥 K',执行协议 2 更新组安全控制器密钥树;

(4) B 中各 $SGSC_i(i \neq k) \rightarrow$ {子组 i 的所有成员}: $\{K'\}_{K_{SGSC_i}}$ 以多播方式发送.

GSC_B 需要付出的加密计算开销是 $d(h\text{-}1)=2\log_2 K$,需要发送的 rekey 信息包数是$(d\text{-}1)(h\text{-}1)=\log_2 K$.各 $SGSC$ 需要付出的加密计算开销是 1,rekey 信息包数是 1.

3) 当子组内有成员变动(加入或离开)时

此时组安全控制器密钥树的更新算法有所不同.对组安全控制器密钥树而言,此时并没有成员的离开和加入情况.即不能删除 u 节点.组通信密钥更新算法(协议 5)如下:

(1) $SGSC_k$:得知有成员变动(离开或加入)后分别执行协议 2 或协议 1,更新子组密钥树.生成新的子组组密钥 K'_{SGSC_k}.

(2) $SGSC_k \rightarrow GSC$: $\{K'_{SGSC_k}\}_{K_{SGSC_k}}$;

(3) $SGSC_k \rightarrow GSC$: $\{$密钥更新 – 请求$\}_{K'_{SGSC_k}}$;

(4) $GSC \rightarrow \mathcal{S}GSC_k$: $\{$密钥更新 –同意$\}_{K'_{SGSC_k}}$,同时产生新的组通信密钥 K';

(5) $GSC \rightarrow$各$\{SGSC_i | SGSC_i \in userset\{son_j(root\ key)\}\}$集合: $\{K'\}_{son_j(root\ key)}$ 分别以多播方式发送(注: $son_j(root\ key)$ 表示 $root\ key$ 节点的第 j 个子节点所代表的密钥.$j=1,2,\ldots,m$(m 表示 $root\ key$ 有 m 个子节点).

(6) 各 $\mathcal{S}GSC_i$(包括 $\mathcal{S}GSC_k$)$\rightarrow$ {子组 i 的所有成员}: $\{K'\}_{K_{SGSC_i}}$,以多播方式发送.

有成员变动的子组的 $SGSC_k$ 需要付出的开销是:当有成员加入时的加密计算开销为 $2(h-1)+2=2+2\log_2 L$,同时需要发送的 rekey 信息包数是为 $h+2=3+\log_2 L$;当有成员离开时的加密计算开销为 $d(h-1)+2=2+2\log_2 L$,需要

发送的 rekey 信息包数是$(d-1)(h-1)+2=2+\log_2 L$.*GSC* 需要付出的加密计算开销是 $d=2$,需要发送的 rekey 信息包数是 $d=2$.各 $SGSC_i$ $(i\neq k)$需要付出的加密计算开销是 1,需要发送的 rekey 信息包数是 1.

4 结 论

4.1 与IOLUS方案的对比

IOLUS 方案[3]通过引入“安全发送树”来处理可伸缩性问题.IOLUS 方案存在的问题是:

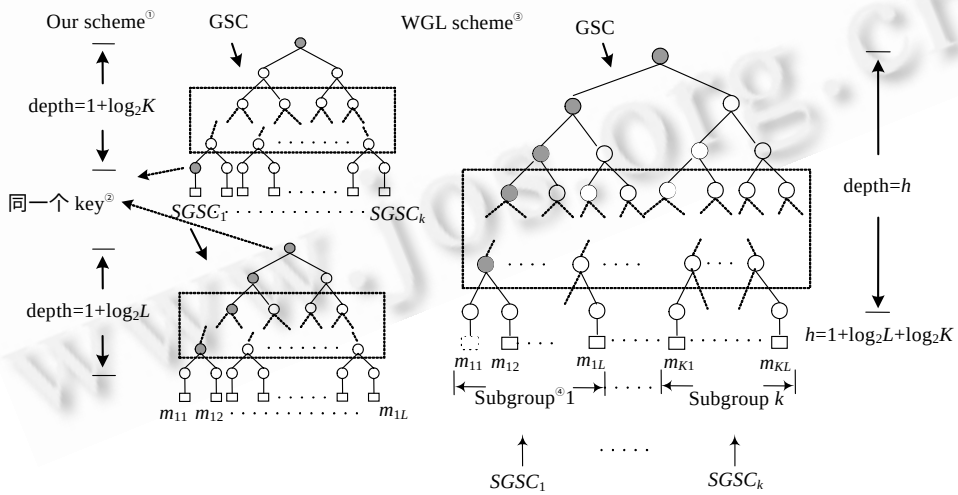
- ✧ 由于各个 GSIs(group security intermediaries)拥有不同的子组通信密钥,由 GSIs 解密和重新加密每一个数据包所引入的延迟开销是影响系统性能的重要因素.这是 IOLUS 系统存在的最大问题.
- ✧ 各个子组通信密钥由各 GSI 自行产生,因此 GSI 的信任也是一个复杂的问题.当 GSI 变为不可信时,系统如何更新也是一个困难的问题.这一问题[2~4]等方案都没有解决.

我们的方案中 GSC(group security controller)通过 GSC 密钥树来管理各个 SGSC.整个多播组的所有成员共享一个唯一的组通信密钥,所有的组通信都通过该密钥加密.不存在 IOLUS 方案存在的第 1 个问题.通过前述的简化的访问控制策略(协议 3),我们的方案解决了 IOLUS 方案存在的第 2 个问题(GSI 不可信的问题).

4.2 与WGL方案[1]的对比(详细对比参见表1)

依据表 1 中的详细对比,我们可以得出如下结论:

- ✧ 当组内的成员增加时(例如上万甚至上百万用户时),文献[1]的方案中的组服务器必将管理和维护一个庞大的密钥树.组服务器管理和维护的密钥数目增加,而且多播组成员拥有的密钥数目增加,即 keyset(u)集合中的密钥数目增加.如何保证所有密钥的安全和保密对计算能力相对有限的组成员端系统来说是一个挑战.
- ✧ 记忆和维护庞大的密钥树,要组服务制器付出巨大的计算开销.组服务器很容易成为整个安全系统的性能瓶颈.系统的可伸缩性成为问题.
- ✧ 在我们系统中,访问控制策略在一定程度上的到了简化.如前面图 2 所示.同时密钥树的更新算法也得到了简化.



①本文的方案,②同一个 key,③WGL 方案,④子组.

Fig.3 Contrast of the two rekeying scheme

图 3 两种密钥更新策略的对比

以图 3 为例,有一多播组拥有 $K\cdot L$ 个成员.按照 WGL 方案,GSC 管理一张密钥树(如图 3 右部所示).按照本文的方案,将 $K\cdot L$ 个成员分为 K 个子组,由 K 个 *SGSC* 分别管理,每一个子组有 L 个成员,同时 *GSC* 通过 *GSC* 密钥树管理这 K 个 *SGSC*.如图 3 左部所示.考虑图 2 的情形,对本文的方案,多播组 *A* 和 *B* 都按图 3 左部的方式

组织.对 *WGL* 方案,它们按图 3 右部所示的方式组织.这样限制不失一般性.下面是这种情形下,两种方案的详细对比:

Table 1 Comparison of the two schemes

表 **1** 两种方案的对比

The items to be contrasted(2) \ Scheme(1)			WGL scheme(3)	Our scheme(4)
The size of the key tree maintained by the GSC(5)			Big(6)	Small(7)
Subgroup security controller(8)			Has no(9)	Has(10)
The number of keys maintained by the GSC(11)			$2LK-1$	$2K-1$
The number of keys maintained by the $SGSC_i$(12)			-	$2L-1$
The number of keys hold by group member sush as m_1(13)			$1+\log_2 L+\log_2 K$	$1+\log_2 L$
When subgroups member, for example m_1 of subgroup1, leaves from multicast group B(14)	Algorithm and protocol(15) The number of keys updated(18)		Protocol 2(16) $\log_2 L+\log_2 K$	Protocol 5(17) $1+\log_2 L$
	GSC	The encryption cost(19) The number of rekey messages(20)	2 $(\log_2 L+\log_2 K)$ $1+\log_2 L+\log_2 K$	2 2
	$SGSC_i$	The encryption cost The number of rekey messages	- -	$2+2\log_2 L(i=1),\ 1(i\neq 1)$ $2+\log_2 L(i=1),\ 1(i\neq 1)$
When subgroup member, for example m_1 of subgroup 1, joins in multicast group B(21)	Algorithm and protocol The number of keys updated		Protocol 1(22) $1+\log_2 L+\log_2 K$	Protocol 5 $2+\log_2 L$
	GSC	The encryption cost The number of rekey messages	2 $(\log_2 L+\log_2 K)$ $1+\log_2 L+\log_2 K$	2 2
	$SGSC_i$	The encryption cost The number of rekey messages	- -	$2+2\log_2 L(i=1),\ 1(i\neq 1)$ $3+\log_2 L(i=1),\ 1(i\neq 1)$
When a whole subgroup, for example all members of subgroup 1, leave from multicast group B(23)	Algorithm and protocol The execution rounds of group key updating algorithm(25) The number of keys updated(28)		Protocol 2 L(26) $L(\log_2 L+\log_2 K)$	Protocol 4(24) 1(27) $\log_2 K$
	GSC	The encryption cost The number of rekey messages	2 $(\log_2 L+\log_2 K)$ $L(\log_2 L+\log_2 K)$	$2\log_2 K$ $\log_2 K$
	$SGSC_i$ $(i\neq 1)$	The encryption cost The number of rekey messages	- -	1 1
When a whole subgroup, for example all members of subgroup 1, join in multicast group B(29)	Algorithm and protocol The execution rounds of group key updating algorithm The number of keys updated		Protocol 1 L $L(1+\log_2 L+\log_2 K)$	Protocol 3(30) 1 $1+\log_2 K$
	GSC	The encryption cost The number of rekey messages	$2L(\log_2 L+\log_2 K)$ $L(1+\log_2 L+\log_2 K)$	$2\log_2 K$ $1+\log_2 K$
	$SGSC_i$ $(i\neq 1)$	The encryption cost The number of rekey messages	- -	1 1
The scalability and robustness of system(31)			Poor(32)	Fine(33)

(1)方案,(2)比较细目,(3)WGL 方案,(4)本文方案,(5)GSC 维护的密钥树大小,(6)大,(7)较小,(8)有无子组安全控制器,(9)没有,(10)有,(11)GSC 维护的密钥数,(12)$SGSC_i$ 维护的密钥数,(13)组内成员例如 m_1 需要保管的密钥数,(14)子组成员例如子组 1 中 m_1 离开 B 时,对系统 B,(15)算法及协议,(16)协议 2,(17)协议 5,(18)需要更新的密钥数,(19)加密计算量,(20)发送的 rekey 信息包数,(21)子组成员例如子组 1 中 m_1 加入 B 时,对系统 B,(22)协议 1,(23)当一个子组例如 B 中子组 1 内所有成员同时离开时,对系统 B,(24)协议 4,(25)组通信密钥更新算法的执行次数,(26)L 次,(27)1 次,(28)更新的密钥数,(29)当一个子组例如 B 中子组 1 加入 A 时,对系统 A,(30)协议 3,(31)系统的可伸缩性和健壮性,(32)较差,(33)好.

References:

[1] Wong, C. K., Gouda, M., Lam, S. S. Secure group communication using key graphs. Technical Report, TR 97-23, Department of Computer Science, University of Texas at Austin.

[2] Mittra, S. IOLUS: a framework for scaleable secure multicast. ACM Computer Communication, 1997,27(3):277~288.

[3] Harney, H., Muckenhirn,C. Group key management protocol (GKMP) architecture. RFC 2094, 1997.

[4] Ballardie, A. Scalable multicast key distribution. RFC 1949, 1996.

[5] Hardjono, T., Cain, B., Doraswamy, N. A framework for group key management for multicast security. Internet draft draft-ietf-ipsec-gkmframework-00.txt, 1998. http://www.ipmulticast.com/techcent.htm.

[6] Harkins, D., Doraswamy, N. A secure, scalable multicast key management protocol (MKMP). IETF Internet draft (work in progress), 1998. http://www.ipmulticast.com/techcent.htm.

[7] Quinn, B. IP Multicast Applications: Challenges and Solutions. draft-quinn-multicast-apps-00.txt, Nov 1998. http://www. Ipmulticast.com/techcent.htm.

[8] Balenson, D. McGrew, A. Sherman. Key Management for Large Dynamic Groups: One-Way Function Trees andAmortized Initialization. Internet Draft. http://www.ipmulticast.com/techcent.htm.

Key Management and Access Control for Large Dynamic Multicast Groups*

LIU Jing, ZHOU Ming-tian

(*College of Computer Science and Engineering, University of Electronic Science and Technology of China, Chengdu* 610054, *China*)

E-mail: jingmliu@263.net; jingliu_uestc@yahoo.com.cn

http://www.uestc.edu.cn

Abstract: As the process of proliferation of the Internet progresses and multicast applications are deployed for mainstream use, the need to IP multicasting technology will become critical. Now, the research of multicast technology has become a fresh new thriving academic field. Many main results converge to such fields as the multicast routing, the flow control, the congestion control and reliable multicast. But the results referred to multicast security are very few (especially in the field of group key management). In this paper, the problems of group key management and access control for large dynamic multicast groups have been researched. A scheme based on subgroup secure controllers are proposed which solves many problems of the Iolus System and the solution of WGL, and at the same time, which simplifies the scheme of access control and meets our design aims and requirements.

Key words: multicast security; group communication; group key management; access control; secret key tree

* Received April 5, 2000; accepted July 14, 2000

Supported by the 'Computer Information Steal and Counter-Steal' Project of National Defense Technological Key Laboratory of Electronic Countermeasure under Grant No.NEWL9703

Subgroup① A Subgroup B Subgroup C Subgroup D Subgroup E

SGSC SGSC SGSC SGSC SGSC GSC GSC

$SGSC_A$ $SGSC_B$ $SGSC_C$ $SGSC_D$ $SGSC_E$

Group securtiy controller key tree②

①子组,②组安全控制器密钥树.

Fig.1 Sketch map of system core architechture

图 1 系统核心框架示意图