基于树结构和门限思想的组密钥协商协议*

王志伟[1+], 谷大武[2]

[1](华东理工大学 计算机科学与工程系,上海 200237)
[2](上海交通大学 计算机科学与工程系,上海 200030)

A Group Key Agreement Protocol Based on Tree and Threshold Idea

WANG Zhi-Wei[1+], GU Da-Wu[2]

[1](Department of Computer Science and Engineering, East China University of Science & Technology, Shanghai 200237, China)
[2](Department of Computer Science and Engineering, Shanghai Jiaotong University, Shanghai 200030, China)
+ Corresponding author: Phn: +86-21-64247115, E-mail: wangzhiwei110@tom.com
Received 2003-07-04; Accepted 2003-10-15

Wang ZW, Gu DW. A group key agreement protocol based on tree and threshold idea. *Journal of Software*, 2004,15(6):924~927.
http://www.jos.org.cn/1000-9825/15/924.htm

Abstract: Dynamic peer communication is the most complicated of all the group communication mode. The paper concisely analysed five typical group key agreement protocols put forward in recent years, they are: centralized key distribution (CKD), burmester-desmedt (BD), steer et al. (STR), group diffie-hellman (GDH) and tree-based group diffie-hellman (TGDH). A new group key agreement protocol (TTS) is proposed and compared with other protocols. This protocol has the advantage in computation, and fit to common network condition.
Key words: group communication; group key agreement; key tree; threshold scheme

摘 要: 动态对等通信(dynamic peer communication)是目前最复杂的群组通信方式之一.简要分析了近几年提出的适合这种通信方式的5种组密钥协商协议,即CKD(centralized key distribution)协议、BD(burmester-desmedt)协议、STR(steer, et al.)协议、GDH(group diffie-hellman)协议和TGDH(tree-based group diffie-hellman)协议,进而提出了一种基于树结构和门限思想的组密钥协商协议TTS(tree and threshold scheme).与现有的协议比较,TTS协议在计算量方面具有较大优势,适用于现有的网络环境.
关键词: 群组通信;组密钥协商;密钥树;门限方案
中图法分类号: TP309 **文献标识码**: A

随着基于群组通信的新业务的大量涌现,如网络游戏、视频会议等,其安全性要求也在日益提高.为了防止组通信被非授权用户访问,所有组内成员必须共享一个组密钥,所有的组通信都是通过这个组密钥加密的.为了确保组通信的安全,组通信的基本要求是:(1) 前向安全性,即一个新加入的成员不能访问以前的通信;(2) 后向

* Supported by the National Natural Science Foundation of China under Grant No.60203012 (国家自然科学基金)
作者简介: 王志伟(1977－),男,江苏扬州人,硕士生,主要研究领域为信息安全与密码学;谷大武(1970－),男,博士,教授,主要研究领域为计算机通信网络安全,密钥交换与管理技术,密码体制.

安全性,即一个离开的成员不能访问目前的通信.也就是说,当群组成员发生变动时,组通信密钥必须改变.

目前组通信的方式大致分为 3 类:(1) 一对多:一个发送者和多个接收者,例如有线电视、有线广播等;(2) 少对多:少数发送者和多个接收者,例如电视辩论、GPS 等;(3) 多对多:所有成员都是对等体,可以动态地成为发送者或接收者.这种方式又被称为动态对等通信(dynamic peer communication),例如视频会议、网络游戏等.

1 动态对等通信的组密钥协商协议

动态对等通信无疑是最复杂的一种群组通信方式,目前适合这种方式的组密钥协商协议不多.其中,比较有代表性的有 5 个:(1) CKD,动态选取组成员充当中心密钥管理器,然后利用 Diffie-Hellman 协议建立中心密钥管理器和各成员之间的两两通信密钥,再利用这个通信密钥分发组密钥;(2) GDH,将两方的 Diffie-Hellman 协议扩展到多方,各成员利用这个协议协商建立组密钥;(3) TGDH,将二叉树结构和 Diffie-Hellman 协议结合起来,各成员协商建立组密钥;(4) STR,这是 TGDH 的一种特例,它采用一种非平衡的二叉树结构;(5) BD,这是 GDH 的一种变体.

衡量一个组密钥协商协议的性能主要有两个因素,即计算量和通信量.GDH 协议和 CKD 协议的计算量都很高,但是通信量较小.BD 协议则是通信量很高,计算量较小.TGDH 协议的计算量和通信量都较小,它是 5 种协议中综合性能最好的.STR 协议与 TGDH 协议相比通信量略有降低,但计算量有所上升.一般来说,通信量大而计算量小的协议适用于高速局域网,因为在这种网络中,通信开销相对于计算开销可以忽略不计.相反,计算量大而通信量小的协议适用于高延迟的广域网,因为在这种网络中通信量才是最重要的因素.

2 基于树结构和门限思想的组密钥协商协议

2.1 树结构

一般的网络环境是由一台总服务器管理几台服务器,每台服务器再管理几台主机.这是一个二层树结构,如图 1 所示.

2.2 TTS协议

如果位于叶子节点的主机构成一个组,当它们相互通信时,就需要产生一个共同的组密钥.我们假设总服务器有 1 个,服务器有 m 个(依次编号为 1,…,m,分别记为 $S_1,S_2,\ldots,S_m$),i 号服务器所管理的主机有 n_i 个(依次编号为 1,…,n_i,分别记为 $C_1^{(i)},C_2^{(i)},\ldots,C_{n_i}^{(i)}$)($i\in\{1,2,\ldots,m\}$).

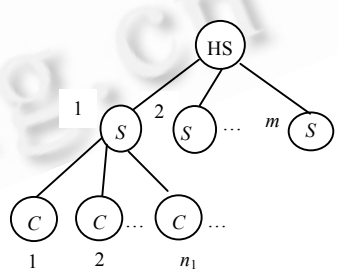

Fig.1 Two layers tree structure

图 1 二层树结构

TTS 协议分为 3 个阶段,初始化阶段、更新阶段和组密钥生成阶段.

(1) 初始化阶段(此阶段只执行 1 次)

Step 1. 总服务器选定如下参数:p,q 为大素数,满足 $q|p-1$;g 为有限域 Z_p 中 q 阶的生成元.

Step 2. 总服务器选取一个初始的群组通信密钥,记为 S,再选取有限域 Z_q 上的 $k-1$ 次方程:

$$f(x)=\sum_{f=0}^{k-1}a_f x^f \bmod q,\text{且}a_0=S\ (k>\max\{m,n_1,n_2,\ldots,n_m\}).$$

Step 3. 总服务器对每一个 $i\in\{1,2,\ldots,m\}$ 计算出 $S_i=f(i)$(为简便起见,我们将第 i 号服务器与经计算得到的密钥均用 S_i 表示).再任选 $k-1$ 个数 $t_1,t_2,\ldots,t_{k-1}$,对每一个 $t_i\in\{t_1,t_2,\ldots,t_{k-1}\}$ 计算出 $S_{t_i}=f(t_i)$.

Step 4. 总服务器将 $\{(t_i,S_{t_i})|(1\le i\le k-1)\}$ 留下.将 (i,S_i) 秘密发送给第 i 号服务器 S_i.

Step 5. 1 号服务器选取有限域 Z_q 上的 $k-1$($k>n_1$)次方程

$$f^{(1)}(x)=\sum_{f=0}^{k-1}a_f^{(1)}x^f \bmod q,$$

其中 $a_0^{(1)}=S_1$.

Step 6. 1 号服务器进行如下计算:先对每一个 $i \in \{1,2,...,n_1\}$ 计算出 $S_{1i} = f^{(1)}(i)$.再对每一个 $t_i \in \{t_1,t_2,...,t_{k-1}\}$ 计算出 $S_{t_i}^{(1)} = f^{(1)}(t_i)$.

Step 7. 1 号服务器将 $\{(t_i, S_{t_i}^{(1)}) | (1 \le i \le k-1)\}$ 留下,再分别将 $(i, S_{1i})(i \in \{1,2,...,n_1\})$ 秘密发送给其所管理的各主机 $C_1^{(1)}, C_2^{(1)}, ..., C_{n_1}^{(1)}$.

Step 8. 其余服务器均仿照 1 号服务器,重复 Step 5~Step 7.直至每个主机和服务器都收到一份消息.

(2) 更新阶段

当组成员发生变动(加入或离开)时,就需要改变组密钥,以保证通信的安全.如果 j 号($1 \le j \le m$)服务器管理的子组中有成员变动,则 j 号服务器重新选择有限域 Zq 上的 $k-1(k>n_j)$次方程 $f'^{(j)}(x) = \sum_{f=0}^{k-1} a_f^{(j)^f} \bmod q$ (其中 $a_0^{(j)}$ 保持不变还是等于 S_j),再对每一个 $i \in \{1,2,...,n_j\}$ 计算出 $S_{ji} = f'^{(j)}(i)$,对每一个 $t_i \in \{t_1,t_2,...,t_{k-1}\}$ 计算出 $S_{t_i}^{(j)} = f'^{(j)}(t_i)$.计算完毕后,$j$ 号服务器将 $\{(t_i, S_{t_i}^{(j)}) | (t_1 \le t_i \le t_{k-1})\}$ 留下,再分别将 $(i, S_{ji})(i \in \{1,2,...,n_j\})$ 秘密发送给其所管理的各主机.

凡是有成员变动的子组,各服务器均按上述步骤处理.在没有变动的子组中,各主机先前收到的那份消息依然有效.

(3) 组密钥生成阶段

在子密钥分发完毕之后,群组中的任何一个成员(主机)均可成为此次更新的发起者(sponsor).发起者任选 $r \in Z_q$,计算出 $X = g^r \bmod p$,并广播 X.然后,总服务器对每一个 $t_i \in \{t_1,t_2,...,t_{k-1}\}$ 计算出 $M_{t_i} = X^{S_{t_i}} \bmod p$,再将 $\{(t_i, M_{t_i}) | t_1 \le t_i \le t_{k-1}\}$ 广播.其余服务器仿照总服务器,例如,j 号服务器对每一个 $t_i \in \{t_1,t_2,...,t_{k-1}\}$ 计算出 $M_{t_i}^{(j)} = X^{S_{t_i}^{(j)}} \bmod p$,再将 $\{(t_i, M_{t_i}^{(j)}) | t_1 \le t_i \le t_{k-1}\}$ 广播.

各主机可根据本子组的服务器和总服务器广播的消息计算出组密钥.例如,j 号($j \in \{1,2,...,m\}$)服务器管理的第 i 号($i \in \{1,2,...,n_j\}$)成员,先计算出 $\tilde{U} = X^{S_{ji} \times L(\Lambda \cup \{i\}, i)} \times \prod_{t_i \in \Lambda} \left[M_{t_i}^{(j)}\right]^{L(\Lambda \cup \{i\}, t_i)} \bmod p$,其中 Λ 为集合 $\{t_1,t_2,...,t_{k-1}\}$,$L(\psi, \{\omega\}) = \prod_{t \in \psi / \{\omega\}} \frac{t}{t-\omega} \bmod q$ (ψ是任意的集合,ω是任意的整数).因为 $M_{t_i}^{(j)} = X^{S_{t_i}^{(j)}} \bmod p$,所以

$$\tilde{U} = g^{r \times S_{ji} \times L(\Lambda \cup \{i\}, i)} \times \prod_{t_i \in \Lambda} g^{r \times S_{t_i}^{(j)} \times L(\Lambda \cup \{i\}, t_i)} (\bmod p)$$

$$= g^{r \times [S_{ji} \times L(\Lambda \cup (i), i) + \sum_{t_i \in \Lambda} S_{t_i}^{(j)} \times L(\Lambda \cup \{i\}, t_i)]} \bmod p$$

$$= g^{r \times S_j} \bmod p$$

然后,计算出组通信密钥 $U = \tilde{U}^{L(\Lambda \cup \{j\}, j)} \times \prod_{t_i \in \Lambda} M_{t_i}^{L(\Lambda \cup \{j\}, t_i)} \bmod p$,因为 $M_{t_i} = X^{S_{t_i}} \bmod p$,所以

$$U = g^{r \times S_{ji} \times L(\Lambda \cup \{j\}, j)} \times \prod_{t_i \in \Lambda} g^{r \times S_{t_i} \times L(\Lambda \cup \{j\}, t_i)} (\bmod p)$$

$$= g^{r \times [S_j \times L(\Lambda \cup (j), j) + \sum_{t_i \in \Lambda} S_{t_i} \times L(\Lambda \cup \{j\}, t_i)]} \bmod p$$

$$= g^{r \times S} \bmod p$$

其他成员可仿照上述运算得出同样的 U,这就是组密钥.

3 安全性分析

TTS 协议的安全性是基于计算离散对数的困难性和门限方案的安全性.

TTS 协议不会泄露组密钥 U 或其他秘密参数.随机数 r 是包含在 $X = g^r \bmod p$ 的指数部分,获取 r 相当于解决离散对数问题.S 只有系统管理者知道,它的安全由门限方案保证.因此,攻击者获取 $U = g^{r \times S} \bmod p$ 是不可能的.

在 TTS 协议中,即使攻击者伪造出 X 并广播,它也无法获得组密钥 U,因为它不知道 S.

一个已离开的成员,用它原有的消息和服务器广播的消息是无法计算出 U 的,因为在它离开之后,系统管理者重新拆分了它所在子组的 $S_j(1 \le j \le m)$,所以它所持有的那份消息也已经作废.

几个离去的成员共谋也无法获得 S.因为我们在初始化阶段和更新阶段选取的方程次数 k–1 满足 $k > n_j(j \in \{1,2,...,m\})$,所以即使该子组成员全部离去,并共谋,也无法构成门限值 k,从而计算出 $S_j(j \in \{1,2,...,m\})$,更不用说获得 S.同理,它们也无法共谋得到 U.

4 TTS 协议的性能分析

TTS 协议是一个较为实用的协议,是根据一般的网络环境设计的.其优点是,在协商组密钥时,每个用户的计算量很小,仅需 2 次模指数运算.GDH 协议、STR 协议和 CKD 协议均需要 $O(n)$次模指数运算,BD 协议在步骤 3 有 n–1 次小指数的模指数运算,运算量也相当大,5 种协议中计算量最小的 TGDH 协议也需要 $\frac{3}{2}h$ (h 为二叉树树高)次模指数运算[6].与这些协议相比,显然 TTS 协议计算量要低很多.它存在的缺点是,系统管理者需要为每个成员和服务器分发密钥.这带来了比较大的通信开销,尤其是单播次数很多.

5 结 论

本文提出了一个基于树结构和门限思想的组密钥协商协议.该协议采用的是二层树结构和门限思想的有机集成,所以与其他协议相比,在计算量方面具有很大的优势.同时,它是根据一般的网络环境设计的,所以很实用.它的缺点是通信开销比较大.如何找到一种通信量和计算量都很小的协议,是值得进一步研究的课题.

最后需要说明的是,TTS 协议是基于门限思想的,具体方案可以选择.本文之所以采用 Shamir 的 lagrange 公式方案,是因为它有两个优点:① 简单明了;② 安全性可达 Shannon 在信息论中所定义的“完全安全”的特性.若需要防止对服务器的主动攻击,可以选用可证实的共享协议(verifiable secret sharing).

致谢 感谢上海交通大学密码学与网络安全联合实验室的侯科鑫、陆海宁、王奕、曾宝珠、何勇等同学在本文的写作过程中所给予的帮助!

References:

[1] Burmestrer M, Desmedt Y. A security and efficient conference-key agreement key distribution system. Advances in Cryptology – EUROCRYPT'94. Berlin: Springer-verlag, 1994. 275~287.

[2] Kim Y, Perrig A, Tsudik G. Communication-Efficent group key agreement. In: Dupuy M, Paradinas P, eds. Proc. of the IFIP SEC 2001. 2001. 229~244.

[3] Stein M, Tsudik G, Waidner M. Key agreement in dynamic peer groups. IEEE Trans. on Parallel and Distributed Systems, 2000,11(8):769~780.

[4] Kim Y, Perrig A, Tsudik G. Simple and fault-tolerant key agreement for dynamic collaborative groups. In: Sander T, ed. Proc. of the 7th ACM Conf. on Computer and Comunications Security. New York: ACM Press, 2000. 235~244.

[5] Lai XS, Han L, Zhang ZC ,Zhang YQ, Xiao GZ. Computer Cryptography and Application. Beijing: National Defence Industry Publishing House, 2001(in Chinese).

[6] Amir Y, Kim Y, Nita-Rotaru C, Tsudik G. On the performance of group key agreement protocols, 2001. http://www.cnds.jhu.edu/pub/papers/cnds-2001-5.ps.gz

附中文参考文献:

[5] 赖溪松,韩亮,张真诚,张玉清,肖国镇.计算机密码学及其应用.北京:国防工业出版社,2001.