

文章编号:1001-9081(2006)03-0571-03

基于身份的移动网动态群组密钥协商方案

邹大毕¹, 林东岱²

(1. 中国科学技术大学 计算机科学技术系, 安徽 合肥 230027;
2. 中国科学院软件研究所 信息安全国家重点实验室, 北京 100080)
(zdabi@ustc.edu)

摘 要: 群组密钥协商是群组通信中非常重要的基本工具, 如何得到一个安全有效的密钥协商协议是当前密码学研究中的一个重要问题。基于双线性对和随机预言模型, 针对移动网络提出了一个动态群组密钥协商方案。此方案就计算复杂度和通信复杂度而言都是高效的, 而且满足密钥协商所需要的安全要求。

关键词: 动态群组密钥协商; 基于身份的加密系统; 双线性对

中图分类号: TP309.2; TP393.08 **文献标识码:** A

Identity-based dynamic group key agreement protocol for mobile networks

ZOU Da-bi¹, LIN Dong-dai²

(1. Department of Computer Science and Technology, University of Science & Technology of China, Hefei Anhui 230027, China;
2. State Key Laboratory of Information Security, Institute of Software of Chinese Academy of Sciences, Beijing 100080, China)

Abstract: Group key agreement protocols are fundamental to group-oriented applications which gain more and more popularity today. Based on bilinear pairings and the random oracle model, a new identity-based dynamic group key agreement protocol for mobile networks was proposed. The analysis results show that the proposed protocol is efficient in terms of both computation and communication complexity, and satisfies most of the security requirements of key agreement.

Key words: dynamic group key agreement; identity-based cryptosystem; bilinear pairings

随着诸如视频会议等面向群组的应用的兴起, 与之紧密相关的安全性便成为了人们关注的问题。群组密钥协商协议就是其中一个很重要的基本工具。文献[1]提出了第一个仅需常数轮的静态群组密钥协商方案, 文献[2~5]也介绍了其他静态认证的群组密钥协商方案, 其中文献[5]的方案是第一个针对主动攻击可证明安全的。针对动态群组的认证密钥协商方案则可以参考文献[6], 它们都是常数轮的。

上面的密钥协商方案都是针对一般对等群组而言的, 它们不适合于新近迅速发展的无线移动网络, 因为在移动网中, 用户的处理能力千差万别, 有些节点的处理能力很低。文献[7~10]对这种应用情形作了探讨, 其中文献[9]是针对静态群组的, 其他则是针对动态群组的。文献[8]的方案最安全有效, 但是它需要假定存在一个安全高效的签名方案。在本文中, 我们也提出了一个安全有效的动态群组密钥协商方案, 而且不需要假定签名方案的存在, 我们的方案是基于一般的难题假设和随机预言模型。和文献[8]一样, 在我们的方案中, 一般的用户仅需要进行少量的计算和通信, 而大部分的计算和通信负担都转移到服务器上, 因此, 对于有着很多低处理能力用户的移动网来说, 我们方案是实际可行的。

1 双线性对及难题假设

1.1 双线性对(Bilinear pairings)

G_1 表示一加法群, G_2 表示一乘法群, 它们有着相同的阶: 素数 q 。 P 是 G_1 的一个生成元。我们假定在群 G_1 和 G_2 中,

离散对数问题(DLP)都是难解的。双线性对便是满足下列性质的一个映射 $e: G_1 \times G_1 \rightarrow G_2$ 。

双线性 $e(P_1 + P_2, Q) = e(P_1, Q)e(P_2, Q)$ 且 $e(P, Q_1 + Q_2) = e(P, Q_1)e(P, Q_2)$ 。

非退化性 该映射不会把 $G_1 \times G_1$ 中的所有对都映射到 G_2 的么元。也就是如果 P 是 G_1 的生成元, 则 $e(P, P)$ 也是 G_2 的生成元。

可计算性 对于所有 $P, Q \in G_1$, 存在一个有效的算法来计算 $e(P, Q)$ 。

1.2 难题假设

BDH 问题(Bilinear Diffie-Hellman Problem) 对于双线性映射 $e: G_1 \times G_1 \rightarrow G_2$, 给定 $P, aP, bP, cP \in G_1$, 问题是计算出 $e(P, P)^{abc}$, 其中 a, b, c 是 F_q 中的随机数。算法 A 被认为以优势 ε 解决 BDH 问题, 如果它满足:

$$\Pr[A(P, aP, bP, cP) = e(P, P)^{abc}] \geq \varepsilon$$

BDH 假设 一般认为 BDH 问题是难解的, 也就是不存在多项式时间算法以不可忽略的优势来求解出 BDH 问题。

2 群组密钥协商协议

和其他基于身份的密码协议一样, 我们假定存在一个可信的密钥生成中心(Key Generation Centre, KGC)。同时, 如前所述, 我们需要一个服务器 Server 来负责协议中的大多数计算和通信任务。

1) 系统建立

收稿日期:2005-09-08 修订日期:2005-11-14

基金项目:国家自然科学基金资助项目(90204016);国家 863 计划项目(2003AA144030);国家 973 规划项目(2004CB318004)

作者简介:邹大毕(1981-),男,广东湛江人,硕士研究生,主要研究方向:密码学以及协议; 林东岱(1964-),男,山东聊城人,教授,博士生导师,主要研究方向:密码理论、安全协议、符号计算、软件设计。

通过 KGC, 获得以下系统参数: $q, G_1, G_2, e; G_1 \times G_1 \rightarrow G_2, P$, 以及 4 个 Hash 函数: $H_1: \{0,1\}^* \rightarrow G_1, H_2: G_1 \times G_1 \rightarrow G_1, H_3: F_q \times G_1 \rightarrow \{0,1\}^l (l \text{ 是会话密钥的长度}), H_4: F_q \times \{0,1\}^l \times G_1 \rightarrow G_1$ 。KGC 的私钥 k , 公钥 $Q_K = kP$ 。Server 的私钥 s , 公钥 $Q_S = sP$ 。所以系统公开参数是 $\{F_q, P, G_1, G_2, e, H_1, H_2, H_3, H_4, Q_K, Q_S\}$, 而 k 和 s 则分别由 KGC 和 Server 各自秘密保存。

2) 用户密钥获取(Extract)

用户 U_i 的标识是 ID_i 。在 U_i 通过验证后, KGC 为它生成私钥 $SK_i = kH_1(ID_i)$, 并经由安全信道发送给它。而对应的公钥, 任何人都可以计算 $PK_i = H_1(ID_i)$ 。

2.1 初始密钥协商 $P_{initial}$

$U = \{U_1, \dots, U_n\}$ 表示在协议 $P_{initial}$ 中协商会话密钥的 n 个用户集合。下面是具体协议描述:

步骤 1 每个用户 $U_i (i = 1, \dots, n)$ 选择随机整数 $a_i, k_i \in F_q$, 预计算 $Q_i = a_i P, W_i = a_i Q_S = a_i sP, R_i = k_i PK_i, S_i = (H_2(Q_i, R_i) + k_i) SK_i$, 然后将 $\{W_i, R_i, S_i\}$ 发送给 Server。

Server 选择随机整数 $b \in F_q$, 并计算 $Q_b = bP$ 。

步骤 2 收到用户 U_i 发来的 $\{W_i, R_i, S_i\}$ 后, Server 计算 $Q_i = s^{-1} W_i = a_i P, e_i = e(S_i, P), e_i' = e(Q_K, R_i + H_2(Q_i, R_i) PK_i)$, 并通过测试 $e_i = e_i'$ 是否成立来验证用户。若等式成立, 则计算 $Q_{Si} = b Q_i = a_i b P$ 。

在所有 n 个用户都通过验证之后, Server 选择随机整数 $r \in F_q$, 并计算:

$$K = \bigoplus_{i=1}^n H_3(r \| Q_{Si})$$

对每个用户 U_i , Server 计算 $K_i = K \oplus H_3(r \| Q_{Si}), T_i = s^{-1} H_4(r \| K_i \| Q_b)$, 并将 $\{r, K_i, Q_b, T_i\}$ 发送给 U_i 。

步骤 3 U_i 在收到 Server 发来 $\{r, K_i, Q_b, T_i\}$ 后, 计算 $t_i = H_4(r \| K_i \| Q_b) \in G_1$, 并测试 $e(t_i, P) = e(T_i, Q_S)$ 是否成立。若等式成立, 则可以计算出会话密钥 $K = K_i \oplus H_3(r \| Q_{Si})$, 其中 Q_{Si} 通过计算 $Q_{Si} = a_i Q_b$ 得到。

2.2 成员离开 P_{leave}

当有用户离开通信群组时, 应该协商一个新的会话密钥来代替之前的密钥。假定 m 个用户离开, 用 $U_L = \{U_{n-m+1}, \dots, U_n\}$ 表示这 m 个用户的集合。则此时的协商方案如下:

步骤 1 Server 选择一个新的随机数 $r^* \in F_q$, 并计算:

$$K^* = \bigoplus_{i=1}^{n-m} H_3(r^* \| Q_{Si})$$

对每个用户 $U_i (i = 1, \dots, n-m)$, Server 计算 $K_i^* = K^* \oplus H_3(r^* \| Q_{Si}), T_i^* = s^{-1} H_4(r^* \| K_i^* \| Q_b)$, 并将 $\{r^*, K_i^*, Q_b, T_i^*\}$ 发送给 U_i 。

步骤 2 U_i 在收到 Server 发来 $\{r^*, K_i^*, Q_b, T_i^*\}$ 后, 计算 $t_i^* = H_4(r^* \| K_i^* \| Q_b) \in G_1$, 并测试 $e(t_i^*, P) = e(T_i^*, Q_S)$ 是否成立。若等式成立, 则可以计算出会话密钥 $K^* = K_i^* \oplus H_3(r^* \| Q_{Si})$ 。

2.3 成员加入 P_{join}

同样, 如果有新的用户加入到通信群组中来, 也需要更新会话密钥。假设 $U_J = \{U_{n+1}, \dots, U_{n+m}\}$ 是要加入的 m 个用户的集合, 则新群组为 $U = U + U_J = \{U_1, \dots, U_n, U_{n+1}, \dots, U_{n+m}\}$ 。

步骤 1 U_J 中每个用户 $U_i (i = n+1, \dots, n+m)$ 选择随机整数 $a_i, k_i \in F_q$, 预计算 $Q_i = a_i P, W_i = a_i Q_S = a_i sP, R_i = k_i PK_i, S_i = (H_2(Q_i, R_i) + k_i) SK_i$ 。然后将 $\{W_i, R_i, S_i\}$ 发送给 Server。

步骤 2 收到用户 $U_i (i = n+1, \dots, n+m)$ 发来的 $\{W_i,$

$R_i, S_i\}$ 后, Server 计算 $Q_i = s^{-1} W_i = a_i P, e_i = e(S_i, P), e_i' = e(Q_K, R_i + H_2(Q_i, R_i) PK_i)$, 并通过测试 $e_i = e_i'$ 是否成立来验证用户。若等式成立, 则计算 $Q_{Si} = b Q_i = a_i b P$ 。

在所有 m 个用户都通过验证之后, Server 选择随机整数 $r' \in F_q$, 并计算:

$$K' = \bigoplus_{i=1}^{n+m} H_3(r' \| Q_{Si})$$

对每个 $U_i (i = 1, \dots, n+m)$, Server 计算 $K_i' = K' \oplus H_3(r' \| Q_{Si}), T_i' = s^{-1} H_4(r' \| K_i' \| Q_b)$, 并将 $\{r', K_i', Q_b, T_i'\}$ 发送给 U_i 。

步骤 3 $U_i (i = 1, \dots, n+m)$ 在收到 Server 发来 $\{r', K_i', Q_b, T_i'\}$ 后, 计算 $t_i' = H_4(r' \| K_i' \| Q_b) \in G_1$, 并测试 $e(t_i', P) = e(T_i', Q_S)$ 是否成立。若等式成立, 则可以计算出会话密钥 $K' = K_i' \oplus H_3(r' \| Q_{Si})$, 其中 U_J 中用户需要计算 $Q_{Si} = a_i Q_b$ 。

3 协议分析

3.1 复杂度分析

通信复杂性 在协议 $P_{initial}$ 中, 有一个来回的通信: 步骤 1 中每个用户发送一个消息, 步骤 2 中 Server 发送 n 个消息, 所以通信量是 $2n$ 。类似地可得到, P_{leave} 的通信量是 $n-m$, P_{join} 的是 $n+2m$ 。

计算复杂性 文中协议的计算量如表 1 所示。根据该表可以得到, 在协议 $P_{initial}$ 中由于步骤 1 中的计算可以预先进行, 因此一般用户的在线计算复杂度是 $1SC + 2PC$, 而 Server 的在线计算复杂度则是 $4nSC + 2nPC$ (求逆 IN 也可以预计算)。

类似可得, 在 P_{leave} 中, 一般用户的计算量是 $2PC$, Server 的则是 $(n-m)SC$ 。在 P_{join} 中, 原用户的复杂度是 $2PC$, 新加入用户的是 $1SC + 2PC$, Server 的则是 $(n+4m)SC + 2mPC$ 。

表 1 协议各步骤复杂度

协议	步骤序号	复杂度
$P_{initial}$	1	$4SC / 1SC$
	2	$1IN + 4nSC + 2nPC$
	3	$1SC + 2PC$
P_{leave}	1	$(n-m)SC$
	2	$2PC$
P_{join}	1	$4SC / 0$
	2	$(4m+n)SC + 2mPC$
	3	$2PC / 2PC + 1SC$

注: SC 表示标量乘法, PC 表示线性对计算, IN 表示整数求逆运算。

3.2 安全性分析

由于 P_{leave}, P_{join} 和 $P_{initial}$ 的相似性, 在下面的分析中仅考虑协议 $P_{initial}$ 。

定理 我们的密钥协商协议是一个双向认证的协议。

引理 1 协议 $P_{initial}$ 中, Server 正确认证了 U_i 。

证明: Server 在步骤 2 中计算 $e_i = e(S_i, P) = e(P, (H_2(Q_i, R_i) + k_i) k H_1(ID_i))$, $e_i' = e(Q_K, R_i + H_2(Q_i, R_i) PK_i) = e(kP, k H_1(ID_i) + H_2(Q_i, R_i) H_1(ID_i))$, 并测试 $e_i = e_i'$ 是否成立。很显然, 如果是正确的用户, 那么 U_i 就可以从 KGC 那里获得它自己的私钥 SK_i , 从而可以正确计算 $S_i = (H_2(Q_i, R_i) + k_i) SK_i$, 这样它就可以通过 Server 的验证。否则, 根据离散对数难题假设, 它是很难成功找到一个 S_i' 通过上述测试而欺骗到 Server 的。另外, 根据文献[11] 知道, 在敌手不知道私钥的情况下成功伪造合法的 R_i', S_i' 对 (即可以通过 Server 的验证) 的概

率可以忽略不计,故引理得证。

引理2 协议 $P_{initial}$ 中, U_i 正确认证了 Server。

证明: U_i 在步骤3收到 Server 发来的信息中的 T_i 包含了 Server 的标记 s^{-1} , 这样在测试 $e(t_i, P) = e(T_i, Q_s)$ 时等式就成立, 因为 $e(T_i, Q_s) = e(s^{-1}t_i, sP) = e(t_i, P)$ 。可以看到, 上面的验证中, 用户用到了 Server 的公钥 Q_s , 所以等式的成立就说明了发送信息的正是 Server, 而不是别人。证毕。

由于引理1和引理2, 很自然可以得出定理是成立的, 即我们的协议是一个双向认证的协议。

进一步, 我们的协议方案满足现在普遍认可的密钥协商安全要求:

1) 隐式密钥认证。除了协议参与者, 其他人是不能够得知所协商出来的会话密钥的。因为虽然其他人可以窃听到消息 $\{r, K_i, Q_b, T_i\}$, 但密钥的计算需要知道 Q_{si} , 而 Q_{si} 只有 U_i 和 Server 才能计算。

2) 已知会话密钥安全性。某次会话密钥的泄漏不会影响到别的轮次协商的会话密钥的保密性。在协议中, 每次会话密钥的计算都包含了一个新的随机数, 而 Hash 函数是抗碰撞的, 所以对于两个轮次的随机数 $r, r', H_2(r \parallel Q_{si})$ 和 $H_2(r' \parallel Q_{si})$ 应是相互独立的, 其中一个被破解不会影响另一个的安全性。

3) 向前安全性。协议参与者的长期私钥的泄漏不会影响到在这之前所协商的密钥的安全性。因为敌手虽然获得了 SK_i (或者 s), 但它不能得知用户的内部状态, 即 a_i (或者 b), 因此它就不能计算 Q_{si} , 因此也就无法得到之前协商的密钥。

4) 密钥泄漏的伪装攻击安全。敌手破解用户的长期私钥后不能够伪装成别人来欺骗该被破解的用户, 因为我们的协议是双向认证的。

5) 未知密钥协商攻击安全。用户不能被欺骗去和未知的第三者协商密钥, 因为我们的协议是双向认证的, 所以不会存在这种欺骗。

参考文献:

- [1] BURMESTER M, DESMETS Y. A Secure and Efficient Conference Key Distribution System[A]. Eurocrypt'94[C], 1994. 270 - 290.
- [2] BRESSON E, CATALANO D. Constant Round Authenticated Group Key Agreement via Distributed Computation[A]. PKC04, LNCS 2947[C]. Springer-Verlag, 2004. 115 - 129.
- [3] BOYD C, NIETIO J. Round-Optimal Contributory Conference Key Agreement[A]. PKC'03, LNCS 2567[C]. Springer-Verlag, 2003. 161 - 174.
- [4] CHOI K, HWANG J, LEE D. Efficient ID-based Group Key Agreement with Bilinear Maps[A]. PKC 2004, LNCS 2947[C]. Springer-Verlag, 2004. 130 - 144.
- [5] KATZ J, YUNG M. Scalable Protocols for Authenticated Group Key Exchange[A]. Crypto'03, LNCS 2729[C]. Springer-Verlag, 2003. 110 - 125.
- [6] KIME H, LEE S, LEE D. Constant-Round Authenticated Group Key Exchange for Dynamic Groups[A]. Asiacrypt'04, LNCS 3329[C]. Springer-Verlag, 2004. 245 - 259.
- [7] BRESSON E, CHEVASSUT O, ESSIARI A, et al. Mutual Authentication and Group Key Agreement for Low-Power Mobile Devices[A]. The Fifth IEEE ICMWCN[C], 2003.
- [8] CHO S, NAM J, KIM S, et al. An Efficient Dynamic Group Key Agreement for Low-Power Mobile Devices[A]. ICCSA 2005, LNCS 3480[C]. Springer-Verlag, 2005. 498 - 507.
- [9] HERRANZ J, VILLAR J. An Unbalanced Protocol for Group Key Exchange[A]. TrustBus 2004, LNCS 3184[C]. Springer-Verlag, 2004. 172 - 180.
- [10] NAM J, KIM S, WON D. Report 2004/251, Attacks on Bresson-Chevassut-Essari-Pointcheval's Group Key Agreement Scheme for Low-Power Mobile Devices[R]. Cryptography ePrint Archive, 2004.
- [11] CHA JC, CHEON JH. Report 2002/018, An Identity-Based Signature from Gap Diffie-Hellman Groups[R]. Cryptography ePrint Archive, 2002.

(上接第570页)

一,如果策略表现为通过邮件警告方式,收到邮件的用户不会一直转发该邮件;第二,如果策略表现为补丁,通过自动的机制传送,停止传播是为了防止补丁传播过度使网络饱和。这样是为了排除策略传播带来的负面影响。为了讨论策略的传播速度,我们定义了策略传播概率 $\rho_c = (\lambda/\delta)$ 。

表1列出了模型中参数的符号及含义,我们的目的是通过对从实际病毒传播问题抽象建模,产生有效的策略。

3 反病毒策略仿真

反病毒策略仿真可以检测四种不同的反病毒策略在多种不同的网络拓扑上的作用效果,如图3所示。在仿真中,稳定状态就是没有节点处于“1”状态或者所有的节点都被感染了。

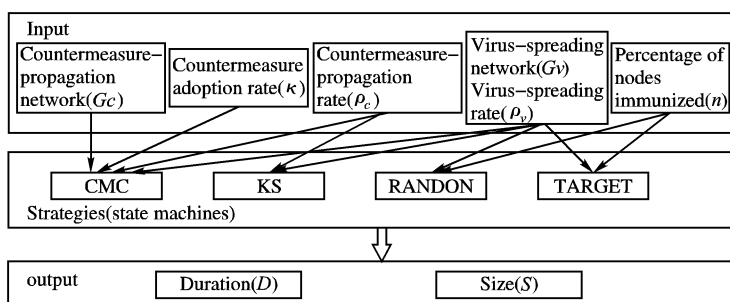


图3 反病毒策略仿真

输出包括病毒传播的时间 D 和传播的范围 S 。 D 指的是系统会聚时间, S 指的是网络中被感染的节点总数。使用仿真模型,可以管理几个虚拟实验组,每个虚拟实验在由动态系统会聚成稳定状态时停止。实验中所有的数据都来源于 The Wile List (TWL) 的病毒报告记录。在实验中,为了更好地抑制病毒传播,还提出了两个限制条件:第一,策略传播要比病毒传播快 $(\rho_v/\rho_c > 1)$;第二,每个节点采取策略的可能性必须大于 $0.1 (\kappa > 0.1)$ 。

具体实验数据表明,当策略传播网络具有较高的连通性时,CMC 较其他三种更有效。通过对不同病毒和在不同网络拓扑上实验,还发现 G_v 的拓扑跟具体的病毒有关。例如,通过邮件传播病毒的 G_v 与通过网页浏览传播病毒的 G_v 是不同的;相同的, G_c 的拓扑跟具体的反病毒策略有关。因此,如果 G_v 的拓扑可以确定,我们就可以设计出策略传播比病毒传播快的 G_c 。

参考文献:

- [1] CHEN L-C, CARLEY K-M. The Impact of Countermeasure Propagation on the Prevalence of Computer Viruses[J]. IEEE Transactions on Systems, 2004, 34(2).
- [2] 陆丽华, 罗鹏飞, 李星, 等. 网络恶意移动代码扩散模型综述[J]. 计算机应用, 2003, 23(6).
- [3] 刘俭, 唐朝京, 张森强. 一种计算机病毒的检测方法[J]. 计算机工程, 2004, 30(6).