(19)中华人民共和国国家知识产权局

(12)发明专利

(10)授权公告号 CN 112039660 B
(45)授权公告日 2021.06.08

(21)申请号 202010811970.5
(22)申请日 2020.08.13
(65)同一申请的已公布的文献号
申请公布号 CN 112039660 A
(43)申请公布日 2020.12.04
(73)专利权人 南京航空航天大学
地址 211016 江苏省南京市江宁区将军大道29号
(72)发明人 常相茂 王杜毅
(74)专利代理机构 南京钟山专利代理有限公司 32252
代理人 陈月菊
(51)Int.Cl.
H04L 9/08(2006.01)
H04L 9/32(2006.01)
H04L 29/06(2006.01)
(56)对比文件
CN 111031519 A,2020.04.17
审查员 万林青

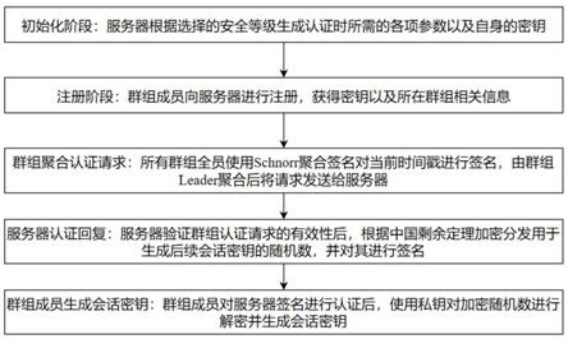
权利要求书2页 说明书5页 附图2页

(54)发明名称

一种物联网节点群组身份安全认证方法

(57)摘要

本发明公开了一种物联网节点群组身份安全认证方法，包括：服务器根据选择的安全等级生成认证时所需的各项参数以及自身的密钥；群组成员向服务器进行注册，获得密钥以及所在群组相关信息；所有群组全员使用Schnorr聚合签名对当前时间戳进行签名，由群组Leader聚合后将请求发送给服务器；服务器验证群组认证请求的有效性后，根据中国剩余定理加密分发用于生成后续会话密钥的随机数，并对其进行签名；群组成员对服务器签名进行认证后，使用私钥对加密随机数进行解密并生成会话密钥。本发明能够降低大规模群组申请身份认证时的数据通信量，群组聚合认证申请和服务器回复信息的大小均为恒定值，不随着群组成员数量的变化而变化。



CN 112039660 B

1.一种物联网节点群组身份安全认证方法,其特征在于,所述认证方法包括以下步骤:

S1,初始化阶段:服务器根据选择的安全等级生成认证时所需的各项参数以及自身的密钥;

S2,注册阶段:群组成员向服务器进行注册,获得密钥以及所在群组相关信息;

S3,群组聚合认证请求:所有群组全员使用Schnorr聚合签名对当前时间戳进行签名,由群组Leader聚合后将请求发送给服务器;

S4,服务器认证回复:服务器验证群组认证请求的有效性后,根据中国剩余定理加密分发用于生成后续会话密钥的随机数,并对其进行签名;

S5,群组成员生成会话密钥:群组成员对服务器签名进行认证后,使用私钥对加密随机数进行解密并生成会话密钥;

步骤S1中所述服务器根据选择的安全等级生成认证时所需的各项参数以及自身的密钥的过程包括以下步骤:

S11,服务器选择系统安全等级参数k,选择素数$q>2^k$;

S12,选择q阶循环群$E(F_q)$,其生成元为G;

S13,选择哈希函数$hash():\{0,1\}^* \rightarrow Z_q^*$,其中$Z_q$为模q的加法循环群;

S14,服务器选择随机数$x_c \in Z_q$作为私钥,计算公钥$P_c=x_c \cdot G$;

S15,服务器公布$\{q,E(F_q),G,hash(),P_c\}$;

步骤S2中,所述群组成员向服务器进行注册,获得密钥以及所在群组相关信息的过程包括以下步骤:

S21,服务器选择$x_i \in Z_q$作为UE_i私钥,群组内成员私钥两两互质,计算公钥$P_i=x_i \cdot G$以及$M=\sum x_i$,$M_i=M/x_i$,$var_i=M_i \cdot y_i$,其中$M_i \cdot y_i \equiv 1 (\mathrm{mod}\ x_i)$;

S22,服务器为群组生成群组身份标识符GID、群组密钥g_k、群组公共参数$L=hash(g_k,P_1,P_2,\cdots,P_n)$和群组公钥$P=\sum[hash(L,P_i) \cdot P_i]$;

S23,服务器将$\{x_i,P_i,GID,g_k,L,P\}$发送给群组成员UE_i;

步骤S3中,所述所有群组全员使用Schnorr聚合签名对当前时间戳进行签名,由群组Leader聚合后将请求发送给服务器的过程包括以下步骤:

S31,群组成员UE_i选择随机数$r_i \in Z_q$,计算$R_i=r_i \cdot G$并发送给群组Leader;

S32,群组Leader收集所有群组成员UE_i的R_i,计算$R=\sum R_i$;

S33,群组Leader将当前时间戳T_u和R发送给群组成员UE_i;

S34,群组成员UE_i验证时间戳T_u有效性后,使用私钥x_i对时间戳进行签名$s_i=r_i+hash(P,R,T_u) \cdot hash(L,P_i) \cdot x_i$,并发送至群组Leader;

S35,群组Leader收集所有群组成员UE_i的s_i,计算聚合签名$s=\sum s_i$,并将$\{GID,s,R,T_u\}$发送给服务器。

2.根据权利要求1所述的物联网节点群组身份安全认证方法,其特征在于,步骤S4中所述服务器验证群组认证请求的有效性后,根据中国剩余定理加密分发用于生成后续会话密钥的随机数,并对其进行签名的过程包括以下步骤:

S41,服务器验证时间戳T_u有效性,通过计算等式$s \cdot G=R+hash(P,R,T_u) \cdot P$是否成立来验证群组身份的合法性;

S42,为群组成员UE_i生成随机数$k_i \in Z_q$,生成$K=\sum(k_i \cdot var_i) \pmod M$;

S43,服务器选择随机数$r_c \in Z_q$,计算$R_c=r_c \cdot G$,对K和当前时间戳T_c进行签名$s_c=r_c+hash(P,R_c,T_c,K) \cdot x_c$;

S44,服务器将$\{K,T_c,s_c,R_c\}$发送给群组Leader。

3.根据权利要求2所述的物联网节点群组身份安全认证方法,其特征在于,步骤S5中,所述群组成员对服务器签名进行认证后,使用私钥对加密随机数进行解密并生成会话密钥的过程包括以下步骤:

S51,群组Leader将$\{K,T_c,s_c,R_c\}$转发给群组成员UE_i;

S52,群组成员UE_i验证时间戳T_c的有效性,通过计算等式$s_c \cdot G=R_c+hash(P,R_c,T_c,K) \cdot P_c$是否成立来验证服务器身份的合法性;

S53,群组成员UE_i使用私钥x_i解密得到$k_i=K \pmod{x_i}$;

S54,群组成员UE_i根据k_i生成会话密钥SK。

一种物联网节点群组身份安全认证方法

技术领域

[0001] 本发明涉及安全认证技术领域，具体而言涉及一种物联网节点群组身份安全认证方法，主要用于解决大规模物联网设备安全认证时效率低下和占用大量带宽资源的问题。

背景技术

[0002] 伴随着物联网技术的迅猛发展，物联网设备已经深入人们的生成生活中，被广泛应用于各种领域，物联网设备的数量也呈爆发式增长。与此同时，物联网设备的安全问题也愈加突出，尤其是那些用于收集和传输用户敏感数据的设备。物联网设备通常计算能力有限，不能采用高复杂度的安全技术，极易遭受攻击。一旦这些设备遭受攻击，导致用户敏感数据遭到泄露，将造成不可挽回的损失。因此，在访问认证过程和数据传输过程中，对用户身份隐私和传输数据安全的保证至关重要。

[0003] 随着低功耗广域网技术的发展成熟，给大规模物联网应用的部署提供了一种有效的传输解决方案。但是，当大规模的物联网设备同时向服务器进行身份认证请求时，产生的信令对物理信道和服务器会造成巨大的通信负担和计算压力，可能会引起信道堵塞，降低系统的运行效率。

[0004] CN110149214A的发明提出了一种无证书聚合签名的LTE-R网络群组认证密钥协商方法，其主要操作步骤是：A、系统建立及参与者注册：参与认证的实体在密钥生成中心处完成注册，获取公私钥信息；B、初始接入认证：用户设备在列车发车前统一接入LTE-R网络时，执行无证书签名算法生成各自签名后发送至中继服务器，由中继服务器执行无证书聚合签名算法，实现用户设备、中继服务器及路旁基站三者间快速的相互认证并协商共享密钥；C、切换认证：列车运行过程中，用户设备始终与中继服务器保持稳定连接，中继服务器将和路旁基站通过执行无证书签名算法实现快速频繁的切换认证；D、终止会话。该方法认证效率高，信令开销小，安全性好。然而，在该发明中，完成认证后建立的是用户到中继服务器再到路旁基站的安全通信，并不能直接建立组内用户成员与路旁基站的直接安全通信，针对的是需要中继设备的大规模用户的认证，并不适用于使用低功耗广域网技术能够直接与基站服务器通信的物联网设备。在该发明中，中继服务器仅聚合用户认证请求消息的签名，仍然会转发组内所有成员包括ID在内的认证请求消息，路旁基站接受的数据量与组内成员数量正相关，同时验证时所需要的计算量也会随着成员数量的增加而增加。因此，前述发明只能从一定程度上缓解信令开销的问题，但事实上，当涉及到的设备过多时，群组聚合认证申请和服务器回复信息的运算量仍十分巨大，数据通信量仍然对系统的运行效率提出了较高的要求。

发明内容

[0005] 本发明针对现有技术中的不足，提供一种物联网节点群组身份安全认证方法，能够降低大规模群组申请身份认证时的数据通信量，群组聚合认证申请和服务器回复信息的大小均为恒定值，不随着群组成员数量的变化而变化，尤其适用于NB-IoT等大规模物联网

设备部署的场景。

[0006] 为实现上述目的，本发明采用以下技术方案：

[0007] 一种物联网节点群组身份安全认证方法，所述认证方法包括以下步骤：

[0008] S1，初始化阶段：服务器根据选择的安全等级生成认证时所需的各项参数以及自身的密钥；

[0009] S2，注册阶段：群组成员向服务器进行注册，获得密钥以及所在群组相关信息；

[0010] S3，群组聚合认证请求：所有群组全员使用Schnorr聚合签名对当前时间戳进行签名，由群组Leader聚合后将请求发送给服务器；

[0011] S4，服务器认证回复：服务器验证群组认证请求的有效性后，根据中国剩余定理加密分发用于生成后续会话密钥的随机数，并对其进行签名；

[0012] S5，群组成员生成会话密钥：群组成员对服务器签名进行认证后，使用私钥对加密随机数进行解密并生成会话密钥。

[0013] 为优化上述技术方案，采取的具体措施还包括：

[0014] 进一步地，步骤S1中所述服务器根据选择的安全等级生成认证时所需的各项参数以及自身的密钥的过程包括以下步骤：

[0015] S11，服务器选择系统安全等级参数k，选择素数$q>2^k$；

[0016] S12，选择q阶循环群$E(F_q)$，其生成元为G；

[0017] S13，选择哈希函数hash()：$\{0,1\}^* \rightarrow Z_q^*$，其中$Z_q$为模q的加法循环群；

[0018] S14，服务器选择随机数$x_c \in Z_q$作为私钥，计算公钥$P_c=x_c \cdot G$；

[0019] S15，服务器公布$\{q,E(F_q),G,hash(),P_c\}$。

[0020] 进一步地，步骤S2中，所述群组成员向服务器进行注册，获得密钥以及所在群组相关信息的过程包括以下步骤：

[0021] S21，服务器选择$x_i \in Z_q$作为UE_i私钥，群组内成员私钥两两互质，计算公钥$P_i=x_i \cdot G$以及$M=\sum x_i$，$M_i=M/x_i$，$var_i=M_i \cdot y_i$，其中$M_i \cdot y_i \equiv 1 (mod x_i)$；

[0022] S22，服务器为群组生成群组身份标识符GID、群组密钥g_k、群组公共参数$L=hash(g_k,P_1,P_2,\cdots,P_n)$和群组公钥$P=\sum[hash(L,P_i) \cdot P_i]$；

[0023] S23，服务器将$\{x_i,P_i,GID,g_k,L,P\}$发送给群组成员UE_i。

[0024] 进一步地，步骤S3中，所述所有群组全员使用Schnorr聚合签名对当前时间戳进行签名，由群组Leader聚合后将请求发送给服务器的过程包括以下步骤：

[0025] S31，群组成员UE_i选择随机数$r_i \in Z_q$，计算$R_i=r_i \cdot G$并发送给群组Leader；

[0026] S32，群组Leader收集所有群组成员UE_i的R_i，计算$R=\sum R_i$；

[0027] S33，群组Leader将当前时间戳T_u和R发送给群组成员UE_i；

[0028] S34，群组成员UE_i验证时间戳T_u有效性后，使用私钥x_i对时间戳进行签名$s_i=r_i+hash(P,R,T_u) \cdot hash(L,P_i) \cdot x_i$，并发送至群组Leader；

[0029] S35，群组Leader收集所有群组成员UE_i的s_i，计算聚合签名$s=\sum s_i$，并将$\{GID,s,R,T_u\}$发送给服务器。

[0030] 进一步地，步骤S4中所述服务器验证群组认证请求的有效性后，根据中国剩余定理加密分发用于生成后续会话密钥的随机数，并对其进行签名的过程包括以下步骤：

[0031] S41，服务器验证时间戳T_u有效性，通过计算等式$s \cdot G=R+hash(P,R,T_u) \cdot P$是否成

立来验证群组身份的合法性；

[0032] S42，为群组成员UE_i生成随机数$k_i \in Z_q$，生成$K=\Sigma(k_i \cdot var_i)(modM)$；

[0033] S43，服务器选择随机数$r_c \in Z_q$，计算$R_c=r_c \cdot G$，对K和当前时间戳T_c进行签名$s_c=r_c+hash(P,R_c,T_c,K) \cdot x_c$；

[0034] S44，服务器将$\{K,T_c,s_c,R_c\}$发送给群组Leader。

[0035] 进一步地，步骤S5中，所述群组成员对服务器签名进行认证后，使用私钥对加密随机数进行解密并生成会话密钥的过程包括以下步骤：

[0036] S51，群组Leader将$\{K,T_c,s_c,R_c\}$转发给群组成员UE_i；

[0037] S52，群组成员UE_i验证时间戳T_c的有效性，通过计算等式$s_c \cdot G=R_c+hash(P,R_c,T_c,K) \cdot P_c$是否成立来验证服务器身份的合法性；

[0038] S53，群组成员UE_i使用私钥x_i解密得到$k_i=K(modx_i)$；

[0039] S54，群组成员UE_i根据k_i生成会话密钥SK。

[0040] 本发明的有益效果是：

[0041] 本发明能够降低大规模群组申请身份认证时的数据通信量，群组聚合认证申请和服务器回复信息的大小均为恒定值，不随着群组成员数量的变化而变化，尤其适用于NB-IoT等大规模物联网设备部署的场景；本发明能够保证群组身份认证的安全性，能够抵御重放攻击、中间人攻击等常见攻击；在保证身份认证安全性的同时，还能够完成群组成员同服务器后续通信的会话密钥协商。

附图说明

[0042] 图1是本发明的物联网节点群组身份安全认证方法的流程图。

[0043] 图2是本发明的其中一个具体实施例的流程图。

具体实施方式

[0044] 现在结合附图对本发明作进一步详细的说明。

[0045] 需要注意的是，发明中所引用的如“上”、“下”、“左”、“右”、“前”、“后”等的用语，亦仅为便于叙述的明了，而非用以限定本发明可实施的范围，其相对关系的改变或调整，在无实质变更技术内容下，当亦视为本发明可实施的范畴。

[0046] 结合图1，本发明提出一种物联网节点群组身份安全认证方法，所述认证方法包括以下步骤：

[0047] S1，初始化阶段：服务器根据选择的安全等级生成认证时所需的各项参数以及自身的密钥。

[0048] S2，注册阶段：群组成员向服务器进行注册，获得密钥以及所在群组相关信息。

[0049] S3，群组聚合认证请求：所有群组全员使用Schnorr聚合签名对当前时间戳进行签名，由群组Leader聚合后将请求发送给服务器。

[0050] S4，服务器认证回复：服务器验证群组认证请求的有效性后，根据中国剩余定理加密分发用于生成后续会话密钥的随机数，并对其进行签名。

[0051] S5，群组成员生成会话密钥：群组成员对服务器签名进行认证后，使用私钥对加密随机数进行解密并生成会话密钥。

[0052] 下面结合图2,通过其中一个具体实施例对本发明的认证方法进行阐述。

[0053] 一、初始化阶段

[0054] 初始化:服务器选择系统安全等级参数k,生成并公布协议认证时所使用的各项安全参数和服务器自身的公钥{q,E(F_q),G,hash(),P_c}。

[0055] 步骤S1中所述服务器根据选择的安全等级生成认证时所需的各项参数以及自身的密钥的过程包括以下步骤:

[0056] S11,服务器选择系统安全等级参数k,选择素数$q>2^k$。

[0057] S12,选择q阶循环群E(F_q),其生成元为G。

[0058] S13,选择哈希函数hash():$\{0,1\}^* \rightarrow Z_q^*$,其中$Z_q$为模q的加法循环群。

[0059] S14,服务器选择随机数$x_c \in Z_q$作为私钥,计算公钥$P_c=x_c \cdot G$。

[0060] S15,服务器公布{q,E(F_q),G,hash(),P_c}。

[0061] 二、注册阶段

[0062] 注册:群组成员设备UE_i向服务器进行注册,UE_i得到私钥、群组身份标识符、群组密钥、群组公共参数和群组公钥{x_i,GID,g_k,L,P}。

[0063] 步骤S2中,所述群组成员向服务器进行注册,获得密钥以及所在群组相关信息的过程包括以下步骤:

[0064] S21,服务器选择$x_i \in Z_q$作为UE_i私钥,群组内成员私钥两两互质,计算公钥$P_i=x_i \cdot G$以及$M=\sum x_i$,$M_i=M/x_i$,$var_i=M_i \cdot y_i$,其中$M_i \cdot y_i \equiv 1 (mod x_i)$。

[0065] S22,服务器为群组生成群组身份标识符GID、群组密钥g_k、群组公共参数$L=hash(g_k,P_1,P_2,\cdots,P_n)$和群组公钥$P=\sum[hash(L,P_i) \cdot P_i]$。

[0066] S23,服务器将{x_i,P_i,GID,g_k,L,P}发送给群组成员UE_i。

[0067] 三、群组聚合认证请求

[0068] 群组聚合认证请求:群组Leader收集所有UE_i的R_i,计算并分发$R=\sum R_i$和当前时间戳T_u;UE_i对时间戳T_u进行签名,并将签名s_i发送给Leader;Leader收集所有UE_i的签名s_i,生成聚合签名s并发送给服务器。

[0069] 步骤S3中,所述所有群组全员使用Schnorr聚合签名对当前时间戳进行签名,由群组Leader聚合后将请求发送给服务器的过程包括以下步骤:

[0070] S31,群组成员UE_i选择随机数$r_i \in Z_q$,计算$R_i=r_i \cdot G$并发送给群组Leader。

[0071] S32,群组Leader收集所有群组成员UE_i的R_i,计算$R=\sum R_i$。

[0072] S33,群组Leader将当前时间戳T_u和R发送给群组成员UE_i。

[0073] S34,群组成员UE_i验证时间戳T_u有效性后,使用私钥x_i对时间戳进行签名$s_i=r_i+hash(P,R,T_u) \cdot hash(L,P_i) \cdot x_i$,并发送至群组Leader。

[0074] S35,群组Leader收集所有群组成员UE_i的s_i,计算聚合签名$s=\sum s_i$,并将{GID,s,R,T_u}发送给服务器。

[0075] 四、服务器认证回复

[0076] 服务器认证回复:服务器验证时间戳T_u和聚合签名s的有效性,为UE_i生成随机数k_i,并使用中国剩余定理加密得到K;服务器对K和当前时间戳T_c进行签名s_c,将{K,T_c,s_c}发送给群组Leader。

[0077] 步骤S4中所述服务器验证群组认证请求的有效性后,根据中国剩余定理加密分发

用于生成后续会话密钥的随机数，并对其进行签名的过程包括以下步骤：

[0078] S41，服务器验证时间戳T_u有效性，通过计算等式$s \cdot G = R + hash(P,R,T_u) \cdot P$是否成立来验证群组身份的合法性。原理如下：

[0079]
$$\begin{aligned} s \bullet G &= \sum s_i \bullet G \\ &= \sum \left(r_i + hash(P,R,T_u) \bullet hash(L,P_i) \bullet x_i \right) \bullet G \\ &= \sum r_i \bullet G + hash(P,R,T_u) \bullet \sum [hash(L,P_i) \bullet P_i] \\ &= R + hash(P,R,T_u) \bullet P \end{aligned}$$

。

[0080] S42，为群组成员UE_i生成随机数$k_i \in Z_q$，生成$K = \Sigma (k_i \cdot var_i) (mod M)$。

[0081] S43，服务器选择随机数$r_c \in Z_q$，计算$R_c = r_c \cdot G$，对K和当前时间戳T_c进行签名$s_c = r_c + hash(P,R_c,T_c,K) \cdot x_c$。

[0082] S44，服务器将$\{K,T_c,s_c,R_c\}$发送给群组Leader。

[0083] 五、群组成员生成会话密钥

[0084] 群组成员生成会话密钥：群组Leader将$\{K,T_c,s_c\}$转发给UE_i，UE_i验证时间戳T_c和签名s_c的有效性，解密得到k_i，并以此生成会话密钥SK。

[0085] 步骤S5中，所述群组成员对服务器签名进行认证后，使用私钥对加密随机数进行解密并生成会话密钥的过程包括以下步骤：

[0086] S51，群组Leader将$\{K,T_c,s_c,R_c\}$转发给群组成员UE_i。

[0087] S52，群组成员UE_i验证时间戳T_c的有效性，通过计算等式$s_c \cdot G = R_c + hash(P,R_c,T_c,K) \cdot P_c$是否成立来验证服务器身份的合法性。原理如下：

[0088]
$$\begin{aligned} s_c \bullet \mathrm{G} &= \left(r_c + hash(P,R_c,T_c,K) \bullet x_c \right) \\ &= r_c \bullet G + hash(P,R_c,T_c,K) \bullet \left(x_c \bullet G \right) \\ &= R_c + hash(P,R_c,T_c,K) \bullet P_c \end{aligned}$$

。

[0089] S53，群组成员UE_i使用私钥x_i解密得到$k_i = K (mod x_i)$。

[0090] S54，群组成员UE_i根据k_i生成会话密钥SK。

[0091] 以上仅是本发明的优选实施方式，本发明的保护范围并不仅局限于上述实施例，凡属于本发明思路下的技术方案均属于本发明的保护范围。应当指出，对于本技术领域的普通技术人员来说，在不脱离本发明原理前提下的若干改进和润饰，应视为本发明的保护范围。

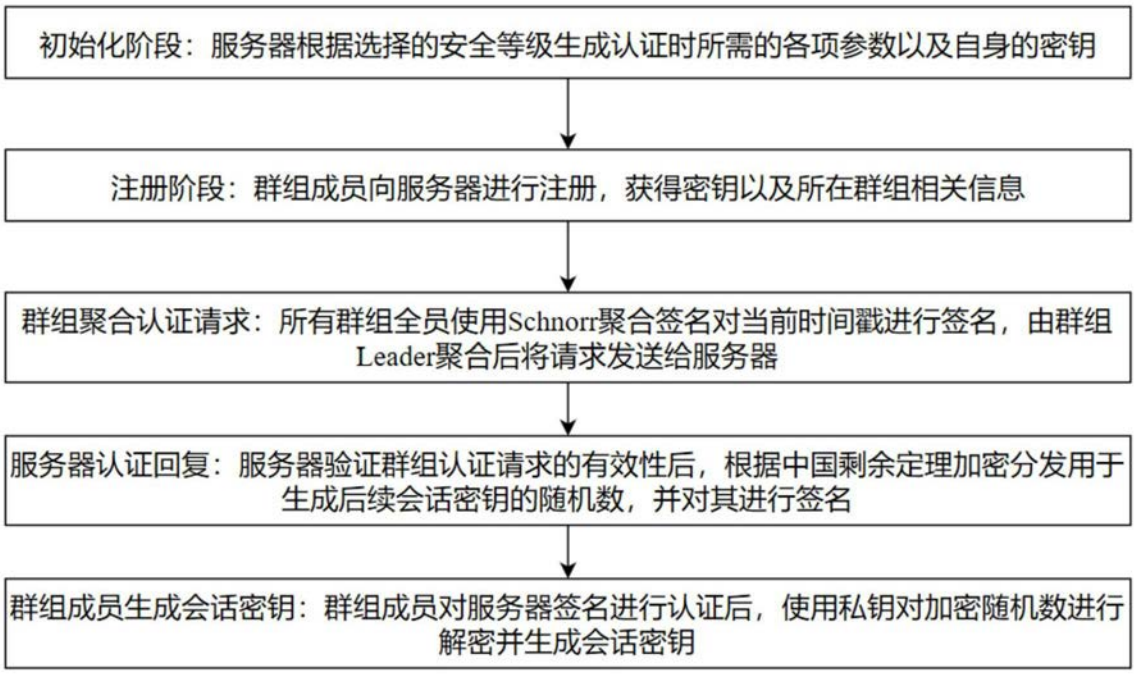

初始化阶段：服务器根据选择的安全等级生成认证时所需的各项参数以及自身的密钥
注册阶段：群组成员向服务器进行注册，获得密钥以及所在群组相关信息
群组聚合认证请求：所有群组全员使用Schnorr聚合签名对当前时间戳进行签名，由群组Leader聚合后将请求发送给服务器
服务器认证回复：服务器验证群组认证请求的有效性后，根据中国剩余定理加密分发用于生成后续会话密钥的随机数，并对其进行签名
群组成员生成会话密钥：群组成员对服务器签名进行认证后，使用私钥对加密随机数进行解密并生成会话密钥

图1

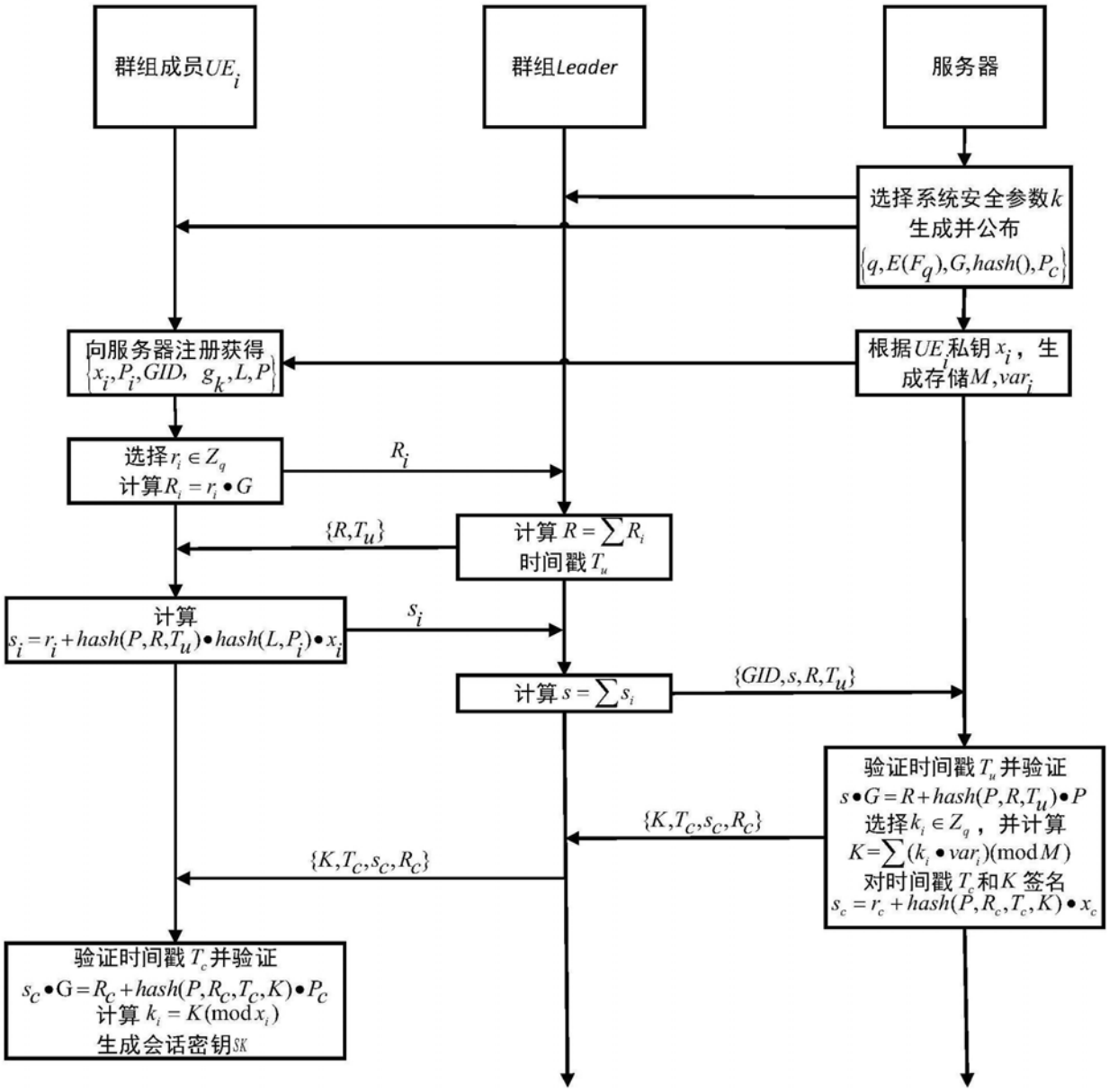

群组成员UE_i
群组Leader
服务器
选择系统安全参数k
生成并公布
$\{q,E(F_q),G,hash(),P_C\}$
向服务器注册获得
$\{x_i,P_i,GID,\ g_k,L,P\}$
根据UE_i私钥x_i，生成存储M,var_i
选择$r_i\in Z_q$
计算$R_i=r_i\bullet G$
R_i
$\{R,T_u\}$
计算$R=\sum R_i$
时间戳T_u
计算
$s_i=r_i+hash(P,R,T_u)\bullet hash(L,P_i)\bullet x_i$
s_i
计算$s=\sum s_i$
$\{GID,s,R,T_u\}$
验证时间戳T_u并验证
$s\bullet G=R+hash(P,R,T_u)\bullet P$
选择$k_i\in Z_q$，并计算
$K=\sum(k_i\bullet var_i)(\mathrm{mod}\,M)$
对时间戳T_c和K签名
$s_c=r_c+hash(P,R_c,T_c,K)\bullet x_c$
$\{K,T_C,s_C,R_C\}$
$\{K,T_C,s_C,R_C\}$
验证时间戳T_c并验证
$s_C\bullet G=R_C+hash(P,R_C,T_C,K)\bullet P_C$
计算$k_i=K(\mathrm{mod}\,x_i)$
生成会话密钥SK

图2