(12) 按照专利合作条约所公布的国际申请

(19) 世界知识产权组织
国 际 局

(43) 国际公布日
2020 年 9 月 3 日 (03.09.2020)

WIPO | PCT

(10) 国际公布号
WO 2020/172898 A1

(51) 国际专利分类号:
G06F 21/60 (2013.01)

(21) 国际申请号: PCT/CN2019/076958

(22) 国际申请日: 2019 年 3 月 5 日 (05.03.2019)

(25) 申请语言: 中文

(26) 公布语言: 中文

(30) 优先权:
201910151513.5 2019年2月28日 (28.02.2019) CN

(71) 申请人: 东 北 大 学 (NORTHEASTERN UNIVERSITY) [CN/CN]; 中国辽宁省沈阳市和平区文化路三巷11号, Liaoning 110819 (CN)。

(72) 发明人: 周 福 才 (ZHOU, Fucai); 中国辽宁省沈阳市和平区文化路三巷11号, Liaoning 110819 (CN)。 郭斯栩 (GUO, Sixu); 中国辽宁省沈阳市和平区文化路三巷11号, Liaoning 110819 (CN)。 葛 悦 (GE, Yue); 中国辽宁省沈阳市和平区文化路三巷11号, Liaoning 110819 (CN)。 孙僖泽 (SUN, Xize); 中国辽宁省沈阳市和平区文化路三巷11号, Liaoning 110819 (CN)。

(74) 代理人: 沈 阳 东 大 知 识 产 权 代 理 有 限 公 司 (SHENYANG DONGDA INTELLECTUAL PROPERTY AGENCY CO., LTD); 中 国 辽 宁 省 沈 阳 市 和 平 区 文 化 路 三 巷 11 号, Liaoning 110819 (CN)。

(81) 指定国 (除另有指明, 要求每一种可提供的国家保护): AE, AG, AL, AM, AO, AT, AU, AZ, BA, BB, BG, BH, BN, BR, BW, BY, BZ, CA, CH, CL, CN, CO, CR, CU, CZ, DE, DJ, DK, DM, DO, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, GT, HN, HR, HU, ID, IL, IN, IR, IS, JO, JP, KE, KG, KH, KN, KP, KR, KW, KZ, LA, LC, LK, LR, LS, LU, LY, MA, MD, ME, MG, MK, MN, MW, MX, MY, MZ, NA, NG, NI, NO, NZ, OM, PA, PE, PG, PH, PL, PT, QA, RO, RS, RU, RW, SA, SC, SD, SE, SG, SK, SL, SM, ST, SV, SY, TH, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, ZA, ZM, ZW。

(54) **Title:** SUFFIX TREE-BASED SEARCHABLE ENCRYPTION SYSTEM AND METHOD

(54) 发明名称: 一种基于后缀树的可搜索加密系统及方法

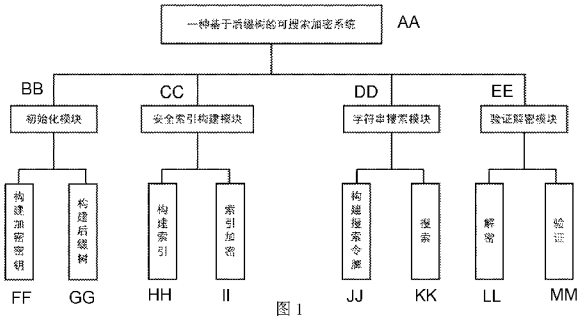

图 1

AA Suffix tree-based searchable encryption system
BB Initialization module
CC Security index construction module
DD Character string search module
EE Verification and decryption module
FF Construct an encryption key
GG Construct a suffix tree
HH Construct an index
II Index encryption
JJ Construct a search token
KK Search
LL Decrypt
MM Verify

(57) **Abstract:** The present invention relates to the technical field of the Internet, and provided therein are a suffix tree-based searchable encryption system and method. The system comprises an initialization module used for constructing an encryption key and a suffix tree, a security index construction module used for constructing an index and index encryption, a sub-character string search module used for constructing a search token and for searching, and a verification and decryption module used for decryption and verification. The method comprises: first constructing a suffix tree and encrypting an index for a given character string, and uploading the encrypted index to a server; and when a client searches for the character string, generating and transmitting a search token to the server, and the server performing the search according to the search token and transmitting a search result to the client to complete the search. The present invention implements the highly efficient search of any character string by means of the suffix tree-based searchable encryption system and method, and solves the problem of performing a sub-character string search on ciphertext data so that a user may search for the ciphertext data without using a keyword.

WO 2020/172898 A1

[见续页]

(84) 指定国(除另有指明，要求每一种可提供的地区保护)：ARIPO (BW, GH, GM, KE, LR, LS, MW, MZ, NA, RW, SD, SL, ST, SZ, TZ, UG, ZM, ZW), 欧亚 (AM, AZ, BY, KG, KZ, RU, TJ, TM), 欧洲 (AL, AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HR, HU, IE, IS, IT, LT, LU, LV, MC, MK, MT, NL, NO, PL, PT, RO, RS, SE, SI, SK, SM, TR), OAPI (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, KM, ML, MR, NE, SN, TD, TG)。

根据细则4.17的声明：

— 关于申请人有权申请并被授予专利(细则4.17(ii))

本国际公布：

— 包括国际检索报告(条约第21条(3))。

(57) 摘要：本发明提供一种基于后缀树的可搜索加密系统及方法，涉及互联网技术领域。该系统包括用于构建加密密钥和后缀树的初始化模块，用于构建索引及索引加密的安全索引构建模块，用于构建搜索令牌和搜索的子字符串搜索模块及用于解密和验证的验证解密模块；该方法首先对给定字符串构建后缀树及加密索引，并将加密索引上传至服务器；当客户端进行字符串搜索时，生成搜索令牌发送至服务器，服务器根据搜索令牌进行搜索，并将搜索结果发送到客户端，完成搜索。本发明的基于后缀树的可搜索加密系统及方法，实现对任意字符串的高效搜索，解决了密文数据的子字符串搜索问题，让用户可以不使用关键字对密文数据进行查询。

一种基于后缀树的可搜索加密系统及方法

技术领域

本发明涉及互联网技术领域，尤其涉及一种基于后缀树的可搜索加密系统及方法。

背景技术

随着互联网技术的高速发展，计算机网络技术已经惠及家家户户。2018 年全球数字报告显示，互联网用户数已经突破 40 亿大关。用户数量的不断增加导致数据信息激增，为了节省本地存储空间，并为已有数据做一个备份，云计算将虚拟化技术与网络相结合，把数据存储在远程服务器，并由服务器处理数据。但是云存储服务带来便利的同时，也使得数据存储和处理的安全问题日渐凸显。第三方服务器往往是不可信的，用户不可避免的要面对信息泄露的风险。因此，保护个人和企业数据的安全性与可用性已经成为计算机互联网技术革新的实际需求。

针对不可信云，加密再上传成为企业和个人的一个好的选择，特别是对于敏感数据，加密可以保证数据本身的安全性，即使被泄漏也不会带来严重的安全隐私问题。但加密操作影响了数据的可用性：云服务器很难对加密的数据进行操作，包括计算、搜索等。因此在保证用户数据安全性的前提下，确保密文数据的可用性，就成为了云计算、密码学等领域的重要研究内容。

可搜索加密针对用户的文件等信息，设置关键词对应文件，并对文件加密，将关键词构造成相应索引并加密，将密文和索引上传到云服务器；搜索文件时，将通过关键词计算出的搜索令牌发送给服务器，让服务器进行与索引的匹配，匹配成功服务器返回匹配的密文文件信息。

但是针对关键字进行处理的搜索加密方案，其搜索格式较为固定。文件的关键字是经过预先设定的，无法满足多种多样的用户需求。例如，电子医疗领域里的基因数据，其结构简单、相似性较大，很难定义其关键字。如何解决上述问题，是当前可搜索加密领域的研究热点。

发明内容

本发明要解决的技术问题是针对上述现有技术的不足，提供一种基于后缀树的可搜索加密系统及方法，利用后缀树结构实现任意子字符串的高效搜索。

为解决上述技术问题，本发明所采取的技术方案是：一方面，本发明提供一种基于后缀树的可搜索加密系统，采用两方实体方案，即客户端与服务器端，包括初始化模块、安全索引构建模块、子字符串搜索模块及验证解密模块；

所述初始化模块用于构建加密密钥和后缀树；

构建加密密钥：由客户端运行，初始化客户端的公私钥及加密所需的密钥；

所述安全索引构建模块用于构建索引以及索引加密；

构建索引：由客户端运行，将待搜索数据构建成后缀树并生成后缀树的索引信息，包括字典结构，密文结构和叶子数组结构；

索引加密：运行于客户端，负责对关键字索引即字典结构、密文数组结构和叶子数组结构进行加密操作，并将加密后的安全索引，包括密文，上传至云服务器，由云服务器进行存储；

所述子字符串搜索模块用于构建搜索令牌和搜索；

构建搜索令牌：运行于客户端，负责对搜索请求计算搜索令牌；

搜索：运行于客户端和服务器端，客户端发送搜索令牌，服务器根据令牌搜索加密的安全索引，并将搜索后的结果返回给客户端；

所述验证解密模块用于解密和验证；

解密：运行于客户端，客户端收到返回的结果后，用自己的私钥对返回的结果进行解密，获取搜索结果明文；

验证：运行于客户端，对解密后的明文进行验证，如果某步验证不通过，则丢弃此明文。

另一方面，本发明还提供一种基于后缀树的可搜索加密方法，包括以下步骤：

步骤 1、根据安全参数λ，输出用户的对称密钥K；

随机选择 7 个比特串$K_D,K_C,K_L,K_1,K_2,K_3,K_4 \xleftarrow{R} \{0,1\}^{\lambda}$，其中，$K_D$为字典结构$D$的加密密钥，$K_C$为密文数组$C$的密钥，$K_L$为叶子数组结构$L$的密钥，$K_1,K_2$用于处理字典结构$D$入口值和对应后缀树节点$u$的第 i 个孩子节点$child(u,i)$的值，以及用于计算搜索令牌$Tok=T_1,\ldots,T_m$，K_3,K_4分别用来对待搜索字符串下标数$ind,\ldots,ind+m-1$以及叶子的位置$leftleaf,\ldots,leftleaf+num-1$进行伪随机置换；最后，输出$K=(K_D,K_C,K_L,K_1,K_2,K_3,K_4)$作为用户的密钥，保证每个密钥各不相同；

步骤 2、客户端对给定数据构建后缀树T_s；利用 UKK 算法将给定的字符串 s 构建成后缀树；

步骤 3、客户端根据后缀树T_s建立安全索引CT并上传至服务器，其中CT由字典结构D、密文数组C和叶子数组L组成，表示为$CT=(D,C,L)$；

所述字典结构 D 对后缀树 T_s 的每一个节点 u 建立一条包括 key 和 $value$ 的条目，表示为 $D=(key,value)$，其中，key 存储搜索的入口，键值 $value$ 由搜索结构和待返回的密文结构组成；找到搜索入口，即能返回对应的 $value$，用户进行解密操作；所述字典结构 D 采用 $\hat{p}(u)$ 进行构造，$\hat{p}(u)$ 为后缀树的根节点到节点 u 的父亲节点上所有边上的字符串的连接，再加上节点 u 的父亲节点到它本身边上的第一个字符 ω；

所述字典结构 D 的具体构建步骤如下：

（1）、用伪随机函数 F 为后缀树 T_s 的每个节点 u 计算一个 key 值，key 存储 $f_1(u)=F(K_1,\hat{p}(u))$ 作为搜索密钥；

（2）、构造 $value$ 值的搜索结构；对于节点 u 的每个孩子节点，计算 $f_{2,i}(u)=F(K_2,\hat{p}(child(u,i)))$，若节点 u 的孩子节点个数 $\deg(u)<d$，d 为叶子节点个数，则添加随机项补足：$f_{2,i}(u)\xleftarrow{R}\{0,1\}^{\lambda}$；选择随机序列 $P:\{0,1\}^{\lambda}\times[d]\to[d]$，将 $f_{2,1}(u),\ldots,f_{2,d}(u)$ 的顺序随机化，最终构造 $value$ 值的搜索结构为 $f_{2,P[1]}(u),\ldots,f_{2,P[d]}(u)$；

（3）、对于后缀树 T_s 的每个节点 u，构造待返回的密文结构，如下公式所示：

$$X_u=Enc(K_D,(ind_u,leftleaf_u,num_u,len_u,f_1(u),f_{2,P[1]}(u),\ldots,f_{2,P[d]}(u)))$$

其中，$Enc()$ 为加密函数，ind_u 为待搜索子字符串第一次出现时，在字符串 s 中的位置下标，$leftleaf_u$ 为节点 u 的最左子后裔的位置，num_u 为 $\hat{p}(u)$ 为子串重复出现的次数；

最终构造的 $value$ 值总体为 $Value_u=(f_{2,P[1]}(u),\ldots,f_{2,P[d]}(u),X_u)$；

（4）、构建字典结构 D 里的 $2n-N$ 个虚拟入口；对每个虚拟入口来说，选择随机的 $f_1(u),f_{2,1}(u),\ldots,f_{2,d}(u)\xleftarrow{R}\{0,1\}^{\lambda}$，存储搜索密钥 $key=f_1(u)$，键值 $value=(f_{2,1}(u),\ldots,f_{2,d}(u),X_u=Enc(K_D,0))$；

所述密文数组 C 的构建方法为：

对给定字符串 s 中的每个字符 $s_{\tilde{i}}$，使用伪随机置换 $P:\{0,1\}^{\lambda}\times[n]\to[n]$ 和密钥 K_3 使得 $C[P_{K_3}(\tilde{i})]=Enc(K_c,s_{\tilde{i}}\|\tilde{i})$，$\tilde{i}$ 为字符下标，$\tilde{i}=1,\cdots,m$，m 为给定字符串 s 中字符个数；

所述叶子数组 L 的构建方法为：$P:\{0,1\}^{\lambda}\times[n]\to[n]$；

对后缀树 T_s 的每个叶子节点，使用伪随机置换和密钥 K_4 使得 $L[P_{K_4}(j)]=Enc(K_L,ind_{leaf_j}\|j)$，$j$ 为叶子结点下标，$j=1,\cdots,d$，ind_{leaf_j} 是根节点到第 j 个叶子节点上的字符串在给定字符串 s 中第一次出现时的位置下标；

客户端与服务器之间的交互搜索如下：

步骤 4、在用户搜索时，选取待搜索字符串 $p=p_1\ldots p_m\in\Sigma^{s.len}$，其中，$s.len$ 为待搜索字符串长度，并计算搜索令牌上传至服务器；

步骤 4-1、获取加密数据结构所需的密钥 $K_{enc}=\{K_1,K_2\}$；

步骤 4-2、计算搜索令牌 $Tok=T_1,\ldots,T_m$；对于 $i'=1,\ldots,m$，计算每一个伪随机函数 $f(i')$ 的值：

$$f_1(i')=F(K_1,p[1\ldots i'])$$

$$f_2(i')=F(K_2,p[1\ldots i'])$$

$$T_{i'}=Enc(F(K_1,p[1\ldots i']),F(K_2,p[1\ldots i']))$$

并将搜索令牌 $T_{i'}$ 上传至服务器；

步骤 5、服务器接收到搜索令牌 Tok 后，先找到字典结构的初始入口，即根节点 u_0 的字典项 $D(F(K_1,\in))$，其中，$\in$ 表示空字符串；

步骤 6、对于每个搜索令牌 $T_{i'}$，使用 $f_{2,1}(u),\ldots,f_{2,d}(u)$ 对搜索令牌按顺序进行解密，具体方法为：

步骤 6-1、设定字母 a,b，并令 $a=1$，$b=1$；

步骤 6-2、令 $i''=a,j''=b$，且当 $1\le i''\le m,1\le j''\le d$ 时，执行如下操作：

$Y\leftarrow Dec(f_{2,j''},T_{i''})$，将 Y 与字典结构 D 的入口 key 值对比：若 $Y=f_1(u')$，则更新数据到字典项 $D(F(K_1,f_1(u')))$，$value$ 值为：$f_{2,P[1]}(u'),\mathrm{K},f_{2,P[d]}(u'),X_{u'}=Enc(K_D,X_{u'})$，并令 $a=a+1$，跳转到步骤 3-2；若 $Y\neq f_1(u')$，则令 $b=b+1$，跳转到步骤 6-3；

步骤 6-3、服务器将搜索结果 $X=Enc(K_D,(ind,leftleaf,num,len,f_1,f_{2,P[1]},\ldots,f_{2,P[d]}))$ 发送给客户端；

步骤 7、用户运行解密操作，令 $W=Dec(K_D,X_u)$，若 $W=\perp$，则解密失败，输出 $\perp$ 并中

止搜索协议，若 $W \neq \perp$，将 W 展开为 $(ind, leftleaf, num, len, f_1, f_{2,P[1]}, \ldots, f_{2,P[d]})$，检验 $f_1 = F(K_1, p[1 \ldots i''])$ 是否成立，如果等式不成立，则返回值出错，输出 $\perp$ 并中止搜索协议；如果等式成立，验证在 $a = i' + 1, \ldots, m$ 且 $b = 1, \ldots, d$ 时，$Dec(f_{2,b}, T_a) = \perp$ 是否成立，若成立，则表明 $p[1 \ldots i'']$ 是能在字典 D 中搜索到最长的匹配前缀；解出 ind，选取一个随机序列 $\pi_1 : [m] \to [m]$；将 $ind, \ldots, ind + m - 1$ 置换，使 $x_{\pi_1(i'')} = P(K_3, ind + i'' - 1)$，用户发送随机序列 $(x_1, \ldots x_m)$ 给服务器；

步骤 8、服务器根据 $(x_1, \ldots x_m)$ 搜索密文数组 C，令 $C_{i''} = C[x_{i''}]$，$i'' = 1, \ldots, m$，并发送 $(C_1, \ldots C_m)$ 给用户；

步骤 9、客户端执行 $Y = Dec_{K_C}(C_{\pi_1}(i''))$，循环执行步骤 7 m 次，i'' 初始值为 1，每执行一次，i'' 增加 1，当 $i'' > m$ 时，终止协议；

若 $Y = \perp$，输出 $\perp$ 并中止查询协议，否则，使搜索结果为 (p_i', j'')；

若 $j'' \neq ind + i'' - 1$，输出 $\perp$；

当 $j'' = ind + i'' - 1$ 时，如果 $p_1', \cdots, p_m' \neq p$，用户输出 Φ 作为答案然后结束协议；如果 $p_1', \ldots, p_m' = p$，$i'' = 1, \ldots, num$，用户选择随机序列 $\pi_2 : [num] \to [num]$，使 $y_{\pi_2}(i'') = P_{K_4}(leftleaf + i'' - 1)$，用户发送随机序列 $(y_1, \ldots y_{num})$ 给服务器；

步骤 10、如果 $i'' = m$，跳转到步骤 8，否则，将 i'' 加 1，跳转到步骤 7；

步骤 11、服务器端设置 $L_{i''} = L[y_{i''}]$，$i'' = 1, \ldots, num$，并将 $(L_1, \ldots L_{num})$ 发送给客户端；

步骤 12、客户端执行 $A = Dec_{K_L}(L_{\pi_2}(i''))$，$i''$ 初始值为 1，循环执行步骤 12 num 次，当 $i'' > num$ 时终止搜索；

如果搜索结果为 $\perp$，客户端输出 $\perp$ 作为答案；

如果搜索结果为 $(a_{i''}, j'')$，且 $j'' \neq leftleaf + i'' - 1$，则客户端输出 $\perp$ 作为答案，如果 $j'' = leftleaf + i'' - 1$，则客户端输出 $A = \{a_1, \ldots a_{num}\}$ 作为答案。

采用上述技术方案所产生的有益效果在于：本发明提供的一种基于后缀树的可搜索加密系统及方法，利用后缀树结构中每个非叶子节点至少有 2 个孩子节点，及每条边的起始字符

不相同等特性，构建具有后缀树结构、支持子字符串搜索的安全索引。并结合密码学原语，对构建的安全索引进行保护，保证数据在加密上传和搜索过程中的安全。通过保留后缀树节点和边的特点，模拟遍历后缀树边来有效的在加密索引中执行字符串匹配，同时找到待搜索子字符串在原始数据中的所有位置，实现对任意字符串的高效搜索。解决了密文数据的子字符串搜索问题，让用户可以不使用关键字对密文数据进行查询。还可以保证用户的隐私信息安全存储在云服务器上，并支持高效的子字符串搜索功能。

附图说明

图 1 为本发明实施例提供的一种基于后缀树的可搜索加密系统的结构框图；

图 2 为本发明实施例提供的一种基于后缀树的可搜索加密方法的流程图；

图 3 为本发明实施例提供的构建安全索引的流程图；

图 4 为本发明实施例提供的客户端与服务器之间进行交互搜索的示意图；

图 5 为本发明实施例提供的客户端与服务器之间进行交互搜索的流程图。

具体实施方式

下面结合附图和实施例，对本发明的具体实施方式作进一步详细描述。以下实施例用于说明本发明，但不用来限制本发明的范围。

一种基于后缀树的可搜索加密系统，采用两方实体方案，即客户端与服务器端，如图 1 所示，包括初始化模块、安全索引构建模块、子字符串搜索模块及验证解密模块；

所述初始化模块用于构建加密密钥和后缀树；

构建加密密钥：由客户端运行，初始化客户端的公私钥及加密所需的密钥；

所述安全索引构建模块用于构建索引以及索引加密；

构建索引：由客户端运行，将待搜索数据构建成后缀树并生成后缀树的索引信息，包括字典结构，密文结构和叶子数组结构；

索引加密：运行于客户端，负责对关键字索引即字典结构、密文数组结构和叶子数组结构进行加密操作，并将加密后的安全索引，包括密文，上传至云服务器，由云服务器进行存储；

所述子字符串搜索模块用于构建搜索令牌和搜索；

构建搜索令牌：运行于客户端，负责对搜索请求计算搜索令牌；

搜索：运行于客户端和服务器端，客户端发送搜索令牌，服务器根据令牌搜索加密的安全索引，并将搜索后的结果返回给客户端；

所述验证解密模块用于解密和验证；

解密：运行于客户端，客户端收到返回的结果后，用自己的私钥对返回的结果进行解密，

获取搜索结果明文；

验证：运行于客户端，对解密后的明文进行验证，如果某步验证不通过，则丢弃此明文。

一种基于后缀树的可搜索加密方法，如图 2 所示，包括以下步骤：

步骤 1、根据安全参数λ，输出用户的对称密钥K；

随机选择 7 个比特串$K_D,K_C,K_L,K_1,K_2,K_3,K_4 \xleftarrow{R} \{0,1\}^{\lambda}$，其中，$K_D$为字典结构$D$的加密密钥，$K_C$为密文数组$C$的密钥，$K_L$为叶子数组结构$L$的密钥，$K_1,K_2$用于处理字典结构$D$入口值和对应后缀树节点$u$的第 i 个孩子节点$child(u,i)$的值，以及用于计算搜索令牌$Tok=T_1,\ldots,T_m$，K_3,K_4分别用来对待搜索字符串下标数$ind,\ldots,ind+m-1$以及叶子的位置$leftleaf,\ldots,leftleaf+num-1$进行伪随机置换；最后，输出$K=(K_D,K_C,K_L,K_1,K_2,K_3,K_4)$作为用户的密钥，保证每个密钥各不相同；

步骤 2、客户端对给定数据构建后缀树T_s；利用 UKK 算法将给定字符串 s 构建成后缀树；

步骤 3、客户端根据后缀树T_s建立安全索引CT并上传至服务器，具体方法如图 3 所示，其中CT由字典结构D、密文数组C和叶子数组L组成，表示为$CT=(D,C,L)$；

所述字典结构D对后缀树T_s的每一个节点u建立一条包括 key 和 $value$ 的条目，表示为$D=(key,value)$，其中，key存储搜索的入口，键值$value$由搜索结构和待返回的密文结构组成；找到搜索入口，即能返回对应的$value$，用户进行解密操作；所述字典结构D采用$\hat{p}(u)$进行构造，$\hat{p}(u)$为后缀树的根节点到节点u的父亲节点上所有边上的字符串的连接，再加上节点u的父亲节点到它本身边上的第一个字符ω；

所述字典结构D的具体构建步骤如下：

（1）、用伪随机函数 F 为后缀树T_s的每个节点 u 计算一个key值，key存储$f_1(u)=F(K_1,\hat{p}(u))$作为搜索密钥；

（2）、构造$value$值的搜索结构；对于节点u的每个孩子节点，计算$f_{2,i}(u)=F(K_2,\hat{p}(child(u,i)))$，若节点$u$的孩子节点个数$\deg(u)<d$，$d$为叶子节点个数，则添加随机项补足：$f_{2,i}(u) \xleftarrow{R} \{0,1\}^{\lambda}$；选择随机序列$P:\{0,1\}^{\lambda}\times[d]\to[d]$，将$f_{2,1}(u),\ldots,f_{2,d}(u)$的顺序随机化，最终构造$value$值的搜索结构为$f_{2,P[1]}(u),\ldots,f_{2,P[d]}(u)$；

（3）、对于后缀树 T_s 的每个节点 u，构造待返回的密文结构，如下公式所示：

$$X_u = Enc(K_D, (ind_u, leftleaf_u, num_u, len_u, f_1(u), f_{2,P[1]}(u), \ldots, f_{2,P[d]}(u)))$$

其中，$Enc()$ 为加密函数，ind_u 为待搜索子字符串第一次出现时，在字符串 s 中的位置下标，$leftleaf_u$ 为节点 u 的最左子后裔的位置，num_u 为 $\hat{p}(u)$ 为子串重复出现的次数；

最终构造的 $value$ 值总体为 $Value_u = (f_{2,P[1]}(u), \ldots, f_{2,P[d]}(u), X_u)$；

（4）、构建字典结构 D 里的 $2n-N$ 个虚拟入口；对每个虚拟入口来说，选择随机的 $f_1(u), f_{2,1}(u), \ldots, f_{2,d}(u) \xleftarrow{R} \{0,1\}^{\lambda}$，存储搜索密钥 $key = f_1(u)$，键值 $value = (f_{2,1}(u), \ldots, f_{2,d}(u)$，$X_u = Enc(K_D, 0))$；

所述密文数组 C 的构建方法为：

对给定字符串 s 中的每个字符 $s_{\tilde{i}}$，使用伪随机置换 $P: \{0,1\}^{\lambda} \times [n] \to [n]$ 和密钥 K_3 使得 $C[P_{K_3}(\tilde{i})] = Enc(K_c, s_{\tilde{i}} \| \tilde{i})$，$\tilde{i}$ 为字符下标，$\tilde{i} = 1, \cdots, m$，m 为待搜索字符串中字符个数；

所述叶子数组 L 的构建方法为：$P: \{0,1\}^{\lambda} \times [n] \to [n]$；

对后缀树 T_s 的每个叶子节点，使用伪随机置换和密钥 K_4 使得 $L[P_{K_4}(j)] = Enc(K_L, ind_{leaf_j} \| j)$，$j$ 为叶子结点下标，$j = 1, \cdots, d$，ind_{leaf_j} 是根节点到第 j 个叶子节点上的字符串在给定字符串 s 中第一次出现时的位置下标；

客户端与服务器之间的交互搜索如图 4 和图 5 所示，具体为：

步骤 4、在用户搜索时，选取待搜索字符串 $p = p_1 \ldots p_m \in \Sigma^{s.len}$，其中，$s.len$ 为待搜索字符串长度，并计算搜索令牌上传至服务器；

步骤 4-1、获取加密数据结构所需的密钥 $K_{enc} = \{K_1, K_2\}$；

步骤 4-2、计算搜索令牌 $Tok = T_1, \ldots, T_m$；对于 $i' = 1, \ldots, m$，计算每一个伪随机函数 $f(i')$ 的值：

$$f_1(i') = F(K_1, p[1 \ldots i'])$$

$$f_2(i') = F(K_2, p[1 \ldots i'])$$

$$T_{i'} = Enc(F(K_1, p[1 \ldots i']), F(K_2, p[1 \ldots i']))$$

并将搜索令牌 $T_{i'}$ 上传至服务器；

步骤 5、服务器接收到搜索令牌 *Tok* 后，先找到字典结构的初始入口，即根节点u_0的字典项$D(F(K_1,\in))$，其中，$\in$表示空字符串；

步骤 6、对于每个搜索令牌$T_{i'}$，使用$f_{2,1}(u),\ldots,f_{2,d}(u)$对搜索令牌按顺序进行解密，具体方法为：

步骤 6-1、设定字母a,b，并令$a=1$，$b=1$；

步骤 6-2、令$i''=a, j''=b$，且当$1\le i''\le m,1\le j''\le d$时，执行如下操作：

$Y\leftarrow Dec(f_{2,j'},T_{i'})$，将$Y$与字典结构$D$的入口 *key* 值对比：若$Y=f_1(u')$，则更新数据到字典项$D(F(K_1,f_1(u')))$，*value* 值为：$f_{2,P[1]}(u'),\ldots,f_{2,P[d]}(u'),X_{u'}=Enc(K_D,X_{u'})$，并令$a=a+1$，跳转到步骤 3-2；若$Y\neq f_1(u')$，则令$b=b+1$，跳转到步骤 6-3；

步骤 6-3、服务器将搜索结果$X=Enc(K_D,(ind,leftleaf,num,len,f_1,f_{2,P[1]},\ldots,f_{2,P[d]}))$发送给客户端；

步骤 7、用户运行解密操作，令$W=Dec(K_D,X_u)$，若$W=\perp$，则解密失败，输出$\perp$并中止搜索协议，若$W\neq\perp$，将W展开为$(ind,leftleaf,num,len,f_1,f_{2,P[1]},\ldots,f_{2,P[d]})$，检验$f_1=F(K_1,p[1\ldots i''])$是否成立，如果等式不成立，则返回值出错，输出$\perp$并中止搜索协议；如果等式成立，验证在$a=i'+1,\ldots,m$且$b=1,\ldots,d$时，$Dec(f_{2,b},T_a)=\perp$是否成立，若成立，则表明$p[1\ldots i'']$是能在字典$D$中搜索到最长的匹配前缀，解出$ind$，选取一个随机序列$\pi_1:[m]\rightarrow[m]$；将$ind,\ldots,ind+m-1$置换，使$x_{\pi_1(i'')}=P(K_3,ind+i''-1)$，用户发送$(x_1,\ldots x_m)$给服务器；

步骤 8、服务器根据$(x_1,\ldots x_m)$搜索密文数组C，令$C_{i''}=C[x_{i''}]$，$i''=1,\ldots,m$，并发送$(C_1,\ldots C_m)$给用户；

步骤 9、客户端执行$Y=Dec_{K_C}(C_{\pi_1}(i''))$，循环执行步骤 7$m$次，$i''$初始值为 1，每执行一次，$i''$增加 1，当$i''>m$时，终止协议；

若$Y=\perp$，输出$\perp$并中止查询协议，否则，使搜索结果为(p_i',j'')，若$j''\neq ind+i''-1$,输出$\perp$；若$j''=ind+i''-1$，如果$p_1',\cdots,p_m'\neq p$，用户输出 Φ 作为答案然后结束协议；如果

$p_1^{'},\ldots,p_m^{'}=p$，$i''=1,\ldots,mum$，用户选择随机序列$\pi_2:[mum]\to[num]$，使$y_{\pi_2}(i'')=P_{K_4}(leftleaf+i''-1)$，用户发送$(y_1,\ldots y_{num})$给服务器；

步骤 10、如果$i''=m$，跳转到步骤 8，否则，将i''加 1，跳转到步骤 7；

步骤 11、服务器端设置$L_{i'}=L[y_{i'}]$，$i''=1,\ldots,mum$，并将$(L_1,\ldots L_{num})$发送给客户端；

步骤 12、客户端执行$A=Dec_{K_L}(L_{\pi_2}(i''))$，$i=1,\ldots,mum$，$i''$初始值为 1，循环执行步骤 12$num$次，当$i''>mum$时终止搜索；

如果搜索结果为⊥，则客户端输出⊥作为答案；

如果搜索结果为$(a_{i''},j'')$，且$j''\neq leftleaf+i''-1$，则客户端输出⊥作为答案，如果$j''=leftleaf+i''-1$，则客户端输出$A=\{a_1,\ldots a_{num}\}$作为答案。

最后应说明的是：以上实施例仅用以说明本发明的技术方案，而非对其限制；尽管参照前述实施例对本发明进行了详细的说明，本领域的普通技术人员应当理解：其依然可以对前述实施例所记载的技术方案进行修改，或者对其中部分或者全部技术特征进行等同替换；而这些修改或者替换，并不使相应技术方案的本质脱离本发明权利要求所限定的范围。

权 利 要 求 书

1、一种基于后缀树的可搜索加密系统，采用两方实体方案，即客户端与服务器端，其特征在于：包括初始化模块、安全索引构建模块、子字符串搜索模块及验证解密模块；

所述初始化模块用于构建加密密钥和后缀树；

构建加密密钥：由客户端运行，初始化客户端的公私钥及加密所需的密钥；

所述安全索引构建模块用于构建索引以及索引加密；

构建索引：由客户端运行，将待搜索数据构建成后缀树并生成后缀树的索引信息，包括字典结构，密文结构和叶子数组结构；

索引加密：运行于客户端，负责对关键字索引即字典结构、密文数组结构和叶子数组结构进行加密操作，并将加密后的安全索引，包括密文，上传至云服务器，由云服务器进行存储；

所述子字符串搜索模块用于构建搜索令牌和搜索；

构建搜索令牌：运行于客户端，负责对搜索请求计算搜索令牌；

搜索：运行于客户端和服务器端，客户端发送搜索令牌，服务器根据令牌搜索加密的安全索引，并将搜索后的结果返回给客户端；

所述验证解密模块用于解密和验证；

解密：运行于客户端，客户端收到返回的结果后，用自己的私钥对返回的结果进行解密，获取搜索结果明文；

验证：运行于客户端，对解密后的明文进行验证，如果某步验证不通过，则丢弃此明文。

2、一种基于后缀树的可搜索加密方法，其特征在于：包括以下步骤：

步骤 1、根据安全参数λ，输出用户的对称密钥K；

随机选择 7 个比特串$K_D,K_C,K_L,K_1,K_2,K_3,K_4 \xleftarrow{R} \{0,1\}^{\lambda}$，其中，$K_D$为字典结构$D$的加密密钥，$K_C$为密文数组$C$的密钥，$K_L$为叶子数组结构$L$的密钥，$K_1,K_2$用于处理字典结构$D$入口值和对应后缀树节点$u$的第 i 个孩子节点$child(u,i)$的值，以及用于计算搜索令牌$Tok=T_1$、…、T_m，K_3,K_4分别用来对待搜索字符串下标数$ind,\ldots,ind+m-1$以及叶子的位置$leftleaf,\ldots,leftleaf+num-1$进行伪随机置换；最后，输出$K=(K_D,K_C,K_L,K_1,K_2,K_3,K_4)$作为用户的密钥，保证每个密钥各不相同；

步骤 2、客户端对给定数据构建后缀树T_s；利用 UKK 算法将给定的字符串 s 构建成后缀树；

步骤 3、客户端根据后缀树 T_s 建立安全索引 CT 并上传至服务器，其中 CT 由字典结构 D、密文数组 C 和叶子数组 L 组成，表示为 $CT=(D,C,L)$；

所述字典结构 D 对后缀树 T_s 的每一个节点 u 建立一条包括 *key* 和 *value* 的条目，表示为 $D=(key,value)$，其中，*key* 存储搜索的入口，键值 *value* 由搜索结构和待返回的密文结构组成；找到搜索入口，即能返回对应的 *value*，用户进行解密操作；所述字典结构 D 采用 $\hat{p}(u)$ 进行构造，$\hat{p}(u)$ 为后缀树的根节点到节点 u 的父亲节点上所有边上的字符串的连接，再加上节点 u 的父亲节点到它本身边上的第一个字符 ω；

客户端与服务器之间的交互搜索如下：

步骤 4、在用户搜索时，选取待搜索字符串 $p=p_1\dots p_m\in\Sigma^{s.len}$，其中，*s.len* 为待搜索字符串长度，并计算搜索令牌上传至服务器；

步骤 5、服务器接收到搜索令牌 *Tok* 后，先找到字典结构的初始入口，即根节点 u_0 的字典项 $D(F(K_1,\in))$，其中，$\in$ 表示空字符串；

步骤 6、对于每个搜索令牌 $T_{i'}$，使用伪随机函数 $f(i')$ 按搜索令牌顺序进行解密；

步骤 7、用户运行解密操作，若解密失败，输出 $\perp$ 并中止搜索协议，否则验证解密操作是否成立，若不成立，则输出 $\perp$ 并中止搜索协议；否则，用户发送随机序列 $(x_1,\dots x_m)$ 给服务器；

步骤 8、服务器根据 $(x_1,\dots x_m)$ 搜索密文数组 C，令 $C_{i''}=C[x_{i''}]$，$i''=1,\dots,m$，并发送 $(C_1,\dots C_m)$ 给用户；

步骤 9、客户端执行 $Y=Dec_{K_C}(C_{\pi_1}(i''))$，循环执行步骤 7 m 次，i'' 初始值为 1，每执行一次，i'' 增加 1，当 $i''>m$ 时，终止协议；

若 $Y=\perp$，输出 $\perp$ 并中止查询协议，否则，使搜索结果为 (p_i',j'')，若 $j''\neq ind+i''-1$，输出 $\perp$；当 $j''=ind+i''-1$ 时，如果 $p_1',\cdots,p_m'\neq p$，用户输出 Φ 作为答案然后结束协议；如果 $p_1',\dots,p_m'=p$，$i''=1,\dots,num$，用户选择随机序列 $\pi_2:[num]\to[num]$，使 $y_{\pi_2}(i'')=P_{K_4}(leftleaf+i''-1)$，用户发送随机序列 $(y_1,\dots y_{num})$ 给服务器；

步骤 10、如果$i''=m$，跳转到步骤 8，否则，将i''加 1，跳转到步骤 7；

步骤 11、服务器端设置$L_{i''}=L[y_{i''}]$，$i''=1,\ldots,num$，并将$(L_1,\ldots L_{num})$发送给客户端；

步骤 12、客户端执行$A=Dec_{K_L}(L_{\pi_2}(i''))$，$i=1,\ldots,num$，$i''$初始值为 1，循环执行步骤 12$num$次，当$i''>num$时终止搜索；

如果搜索结果为⊥，客户端输出⊥作为答案；

如果搜索结果为$(a_{i''},j'')$，且$j''\neq leftleaf+i''-1$，则客户端输出⊥作为答案，如果$j''=leftleaf+i''-1$，则客户端输出$A=\{a_1,\ldots a_{num}\}$作为答案。

3、根据权利要求 2 所述的一种基于后缀树的可搜索加密方法，其特征在于：步骤 3 所述字典结构D的具体构建方法为：

（1）、用伪随机函数 F 为后缀树T_s的每个节点u计算一个key值，key存储$f_1(u)=F(K_1,\hat{p}(u))$作为搜索密钥；

（2）、构造$value$值的搜索结构；对于节点u的每个孩子节点，计算$f_{2,i}(u)=F(K_2,\hat{p}(child(u,i)))$，若节点$u$的孩子节点个数$\deg(u)<d$，$d$为叶子节点个数，则添加随机项补足：$f_{2,i}(u)\xleftarrow{R}\{0,1\}^{\lambda}$；选择随机序列$P:\{0,1\}^{\lambda}\times[d]\to[d]$，将$f_{2,1}(u),\ldots,f_{2,d}(u)$的顺序随机化，最终构造$value$值的搜索结构为$f_{2,P[1]}(u),\ldots,f_{2,P[d]}(u)$；

（3）、对于后缀树T_s的每个节点u，构造待返回的密文结构，如下公式所示：

$$X_u=Enc(K_D,(ind_u,leftleaf_u,num_u,len_u,f_1(u),f_{2,P[1]}(u),\ldots,f_{2,P[d]}(u)))$$

其中，$Enc()$为加密函数，ind_u为待搜索子字符串第一次出现时，在字符串s中的位置下标，$leftleaf_u$为节点u的最左子后裔的位置，num_u为$\hat{p}(u)$为子串重复出现的次数；

最终构造的$value$值总体为$Value_u=(f_{2,P[1]}(u),\ldots,f_{2,P[d]}(u),X_u)$；

（4）、构建字典结构D里的$2n-N$个虚拟入口；对每个虚拟入口来说，选择随机的$f_1(u),f_{2,1}(u),\ldots,f_{2,d}(u)\xleftarrow{R}\{0,1\}^{\lambda}$，存储搜索密钥$key=f_1(u)$，键值$value=(f_{2,1}(u),\ldots,f_{2,d}(u),X_u=Enc(K_D,0))$。

4、根据权利要求 3 述的一种基于后缀树的可搜索加密方法，其特征在于：步骤 3 所述密文数组 C 的构建方法为：

对给定字符串 s 中的每个字符 $s_{\tilde{i}}$，使用伪随机置换 $P:\{0,1\}^{\lambda}\times[n]\rightarrow[n]$ 和密钥 K_3 使得 $C[P_{K_3}(\tilde{i})]=Enc(K_c,s_{\tilde{i}}\|\tilde{i})$，$\tilde{i}$ 为字符下标，$\tilde{i}=1,\cdots,m$，m 为给定字符串 s 中字符个数；

所述叶子数组 L 的构建方法为：$P:\{0,1\}^{\lambda}\times[n]\rightarrow[n]$

对后缀树 T_s 的每个叶子节点，使用伪随机置换和密钥 K_4 使得 $L[P_{K_4}(j)]=Enc(K_L,ind_{leaf_j}\|j)$，$j$ 为叶子结点下标，$j=1,\cdots,d$，ind_{leaf_j} 是根节点到第 j 个叶子节点上的字符串在给定字符串 s 中第一次出现时的位置下标。

5、根据权利要求 4 述的一种基于后缀树的可搜索加密方法，其特征在于：所述步骤 4 的具体方法为：

步骤 4-1、获取加密数据结构所需的密钥 $K_{enc}=\{K_1,K_2\}$；

步骤 4-2、计算搜索令牌 $Tok=T_1,\ldots,T_m$；对于 $i'=1,\ldots,m$，计算每一个伪随机函数 $f(i')$ 的值：

$$f_1(i')=F(K_1,p[1\ldots i'])$$

$$f_2(i')=F(K_2,p[1\ldots i'])$$

$$T_{i'}=Enc(F(K_1,p[1\ldots i']),F(K_2,p[1\ldots i']))$$

并将搜索令牌 $T_{i'}$ 上传至服务器。

6、根据权利要求 5 述的一种基于后缀树的可搜索加密方法，其特征在于：所述步骤 6 的具体方法为：

步骤 6-1、设定字母 a,b，并令 a=1，b=1；

步骤 6-2、令 $i''=a,j''=b$，且当 $1\le i''\le m,1\le j''\le d$ 时，执行如下操作：

$Y\leftarrow Dec(f_{2,j''},T_{i''})$，将 Y 与字典结构 D 的入口 key 值对比：若 $Y=f_1(u')$，则更新数据到字典项 $D(F(K_1,f_1(u')))$，$value$ 值为：$f_{2,P[1]}(u'),\mathrm{K},f_{2,P[d]}(u'),X_{u'}=Enc(K_D,X_{u'})$，并令 $a=a+1$，跳转到步骤 3-2；若 $Y\neq f_1(u')$，则令 $b=b+1$，跳转到步骤 6-3；

步骤 6-3、服务器将搜索结果 $X=Enc(K_D,(ind,leftleaf,num,len,f_1,f_{2,P[1]},\ldots,f_{2,P[d]}))$ 发送给客户端。

7、根据权利要求 6 述的一种基于后缀树的可搜索加密方法，其特征在于：所述步骤 7 的

具体方法为：

用户运行解密操作，令$W=Dec(K_D,X_u)$，若$W=\perp$，则解密失败，输出$\perp$并中止搜索协议，若$W\neq\perp$，将W展开为$(ind,leftleaf,num,len,f_1,f_{2,P[1]},\ldots,f_{2,P[d]})$，检验$f_1=F(K_1,p[1\ldots i''])$是否成立，如果等式不成立，则返回值出错，输出$\perp$并中止搜索协议；如果等式成立，验证在$a=i'+1,\ldots,m$且$b=1,\ldots,d$时，$Dec(f_{2,b},T_a)=\perp$是否成立，若成立，则表明$p[1\ldots i'']$是能在字典$D$中搜索到最长的匹配前缀；解出$ind$，选取一个随机序列$\pi_1:[m]\rightarrow[m]$；将$ind,\ldots,ind+m-1$置换，使$x_{\pi_1(i'')}=P(K_3,ind+i''-1)$，用户发送$(x_1,\ldots x_m)$给服务器。

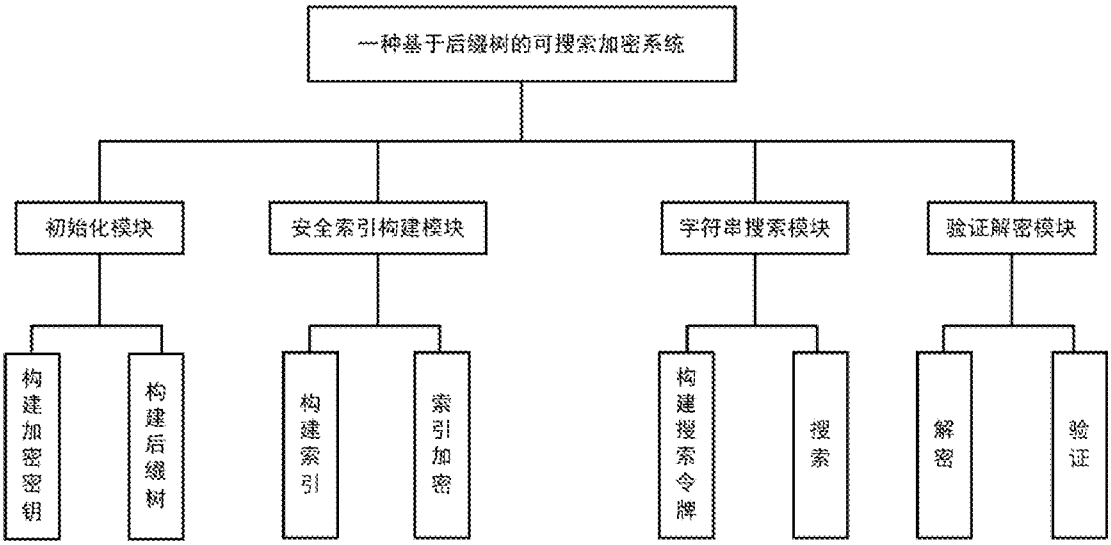
一种基于后缀树的可搜索加密系统
初始化模块
安全索引构建模块
字符串搜索模块
验证解密模块
构建加密密钥
构建后缀树
构建索引
索引加密
构建搜索令牌
搜索
解密
验证

图 1

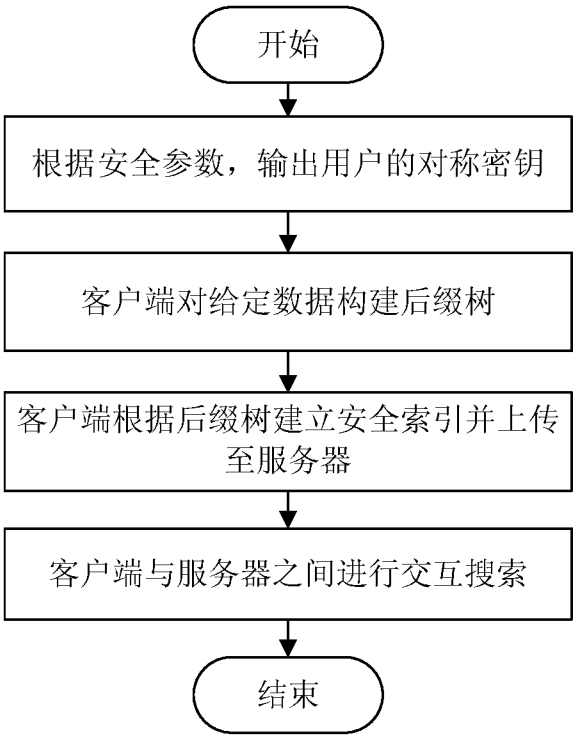
开始
根据安全参数，输出用户的对称密钥
客户端对给定数据构建后缀树
客户端根据后缀树建立安全索引并上传至服务器
客户端与服务器之间进行交互搜索
结束

图 2

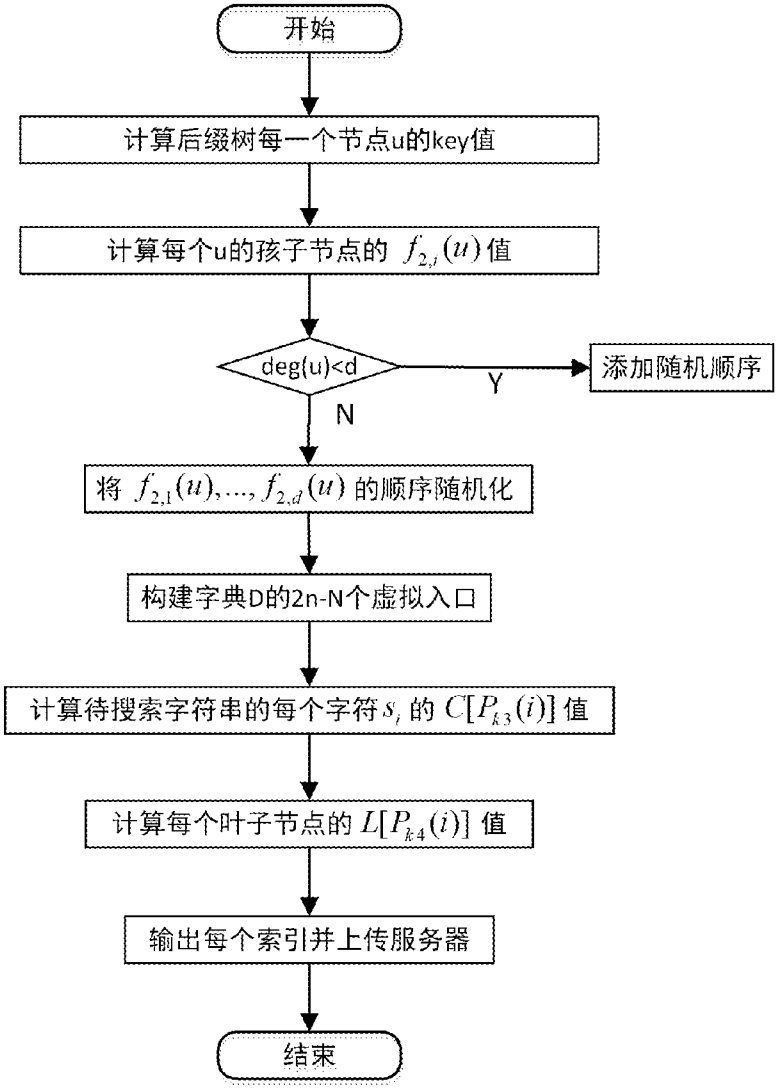
开始
计算后缀树每一个节点u的key值
计算每个u的孩子节点的 $f_{2,l}(u)$ 值
deg(u)<d
Y
添加随机顺序
N
将 $f_{2,1}(u),...,f_{2,d}(u)$ 的顺序随机化
构建字典D的2n-N个虚拟入口
计算待搜索字符串的每个字符 s_i 的 $C[P_{k3}(i)]$ 值
计算每个叶子节点的 $L[P_{k4}(i)]$ 值
输出每个索引并上传服务器
结束

图 3

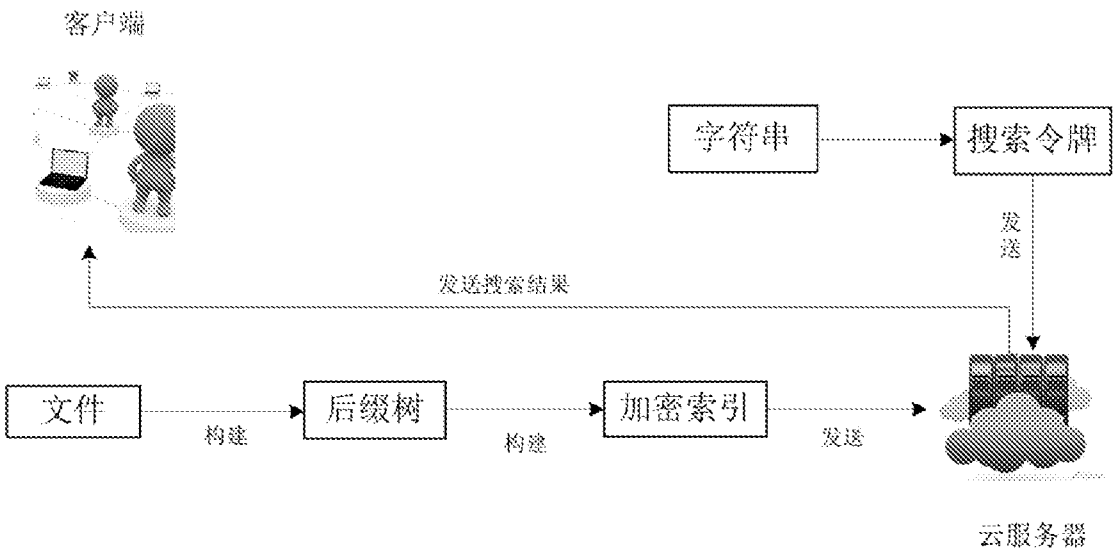
客户端
字符串
搜索令牌
发送
发送搜索结果
文件
构建
后缀树
构建
加密索引
发送
云服务器

图 4

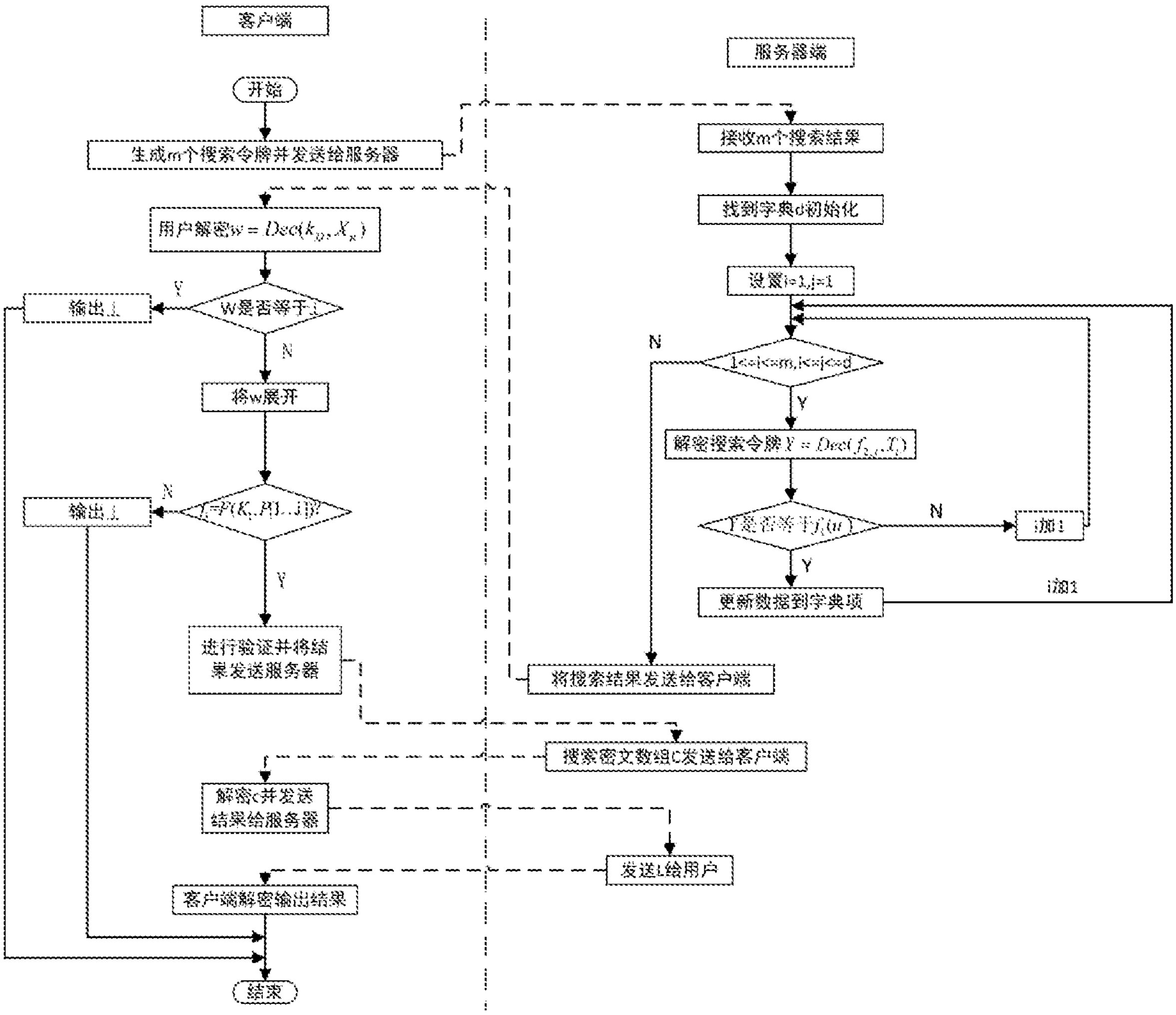
客户端
服务器端
开始
生成m个搜索令牌并发送给服务器
接收m个搜索结果
找到字典d初始化
用户解密$w = Dec(k_D, X_x)$
设置i=1,j=1
Y
输出⊥
W是否等于⊥
N
N
1<=i<=m,k<=j<=d
将w展开
Y
解密搜索令牌$Y = Dec(f_{2,i}, T_i)$
N
输出⊥
N
j加1
Y
更新数据到字典项
i加1
进行验证并将结果发送服务器
将搜索结果发送给客户端
搜索密文数组C发送给客户端
解密c并发送结果给服务器
发送L给用户
客户端解密输出结果
结束

图 5

INTERNATIONAL SEARCH REPORT

International application No.
PCT/CN2019/076958

A. CLASSIFICATION OF SUBJECT MATTER

G06F 21/60(2013.01)i

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)

G06F

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)

CNABS; CNTXT; VEN; USTXT; EPTXT; WOTXT; CNKI: 搜索, 查询, 可搜索加密, 后缀树, 树, 子字符串, 加密, 密钥, 公钥, 私钥, 索引, 云, search, query, searchable encryption, suffix tree, tree, substring, encryption, secret key, public key, private key, index, cloud

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	CHASE, Melissa et al. "Substring-Searchable Symmetric Encryption" *Proceedings on Privacy Enhancing Technologies,* 31 December 2015 (2015-12-31), ISSN: 2299-0984, pp. 263-272	1-7
A	CN 103607405 A (NORTHEASTERN UNIVERSITY) 26 February 2014 (2014-02-26) entire document	1-7
A	CN 108388807 A (SOUTH CHINA UNIVERSITY OF TECHNOLOGY) 10 August 2018 (2018-08-10) entire document	1-7
A	US 2017091475 A1 (ROBERT BOSCH GMBH) 30 March 2017 (2017-03-30) entire document	1-7

☐ Further documents are listed in the continuation of Box C. ☑ See patent family annex.

* Special categories of cited documents:
"A" document defining the general state of the art which is not considered to be of particular relevance
"E" earlier application or patent but published on or after the international filing date
"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)
"O" document referring to an oral disclosure, use, exhibition or other means
"P" document published prior to the international filing date but later than the priority date claimed
"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention
"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone
"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art
"&" document member of the same patent family

Date of the actual completion of the international search	Date of mailing of the international search report
20 September 2019	**29 October 2019**
Name and mailing address of the ISA/CN **China National Intellectual Property Administration (ISA/CN)** **No. 6, Xitucheng Road, Jimenqiao Haidian District, Beijing 100088** **China** Facsimile No. **(86-10)62019451**	Authorized officer Telephone No.

INTERNATIONAL SEARCH REPORT
Information on patent family members

International application No.
PCT/CN2019/076958

Patent document cited in search report	Publication date (day/month/year)	Patent family member(s)	Publication date (day/month/year)
CN 103607405 A	26 February 2014	CN 103607405 B	23 November 2016
CN 108388807 A	10 August 2018	None	
US 2017091475 A1	30 March 2017	US 9971904 B2	15 May 2018
		WO 2017055256 A1	06 April 2017
		EP 3356954 A1	08 August 2018

国际检索报告

国际申请号
PCT/CN2019/076958

A. 主题的分类

G06F 21/60(2013.01)i

按照国际专利分类(IPC)或者同时按照国家分类和IPC两种分类

B. 检索领域

检索的最低限度文献(标明分类系统和分类号)

G06F

包含在检索领域中的除最低限度文献以外的检索文献

在国际检索时查阅的电子数据库(数据库的名称，和使用的检索词（如使用））

CNABS;CNTXT;VEN;USTXT;EPTXT;WOTXT;CNKI：搜索，查询，可搜索加密，后缀树，树，子字符串，加密，密钥，公钥，私钥，索引，云，search，query，searchable encryption，suffix tree，tree，substring，encryption，secret key，public key，private key，index，cloud

C. 相关文件

类 型*	引用文件，必要时，指明相关段落	相关的权利要求
X	CHASE, Melissa 等. "Substring-Searchable Symmetric Encryption" Proceedings on Privacy Enhancing Technologies, 2015年 12月 31日 (2015-12-31), ISSN：2299-0984, 第263-272页	1-7
A	CN 103607405 A (东北大学) 2014年 2月 26日 (2014-02-26) 全文	1-7
A	CN 108388807 A (华南理工大学) 2018年 8月 10日 (2018-08-10) 全文	1-7
A	US 2017091475 A1 (ROBERT BOSCH GMBH) 2017年 3月 30日 (2017-03-30) 全文	1-7

☐ 其余文件在C栏的续页中列出。

☑ 见同族专利附件。

* 引用文件的具体类型：

“A” 认为不特别相关的表示了现有技术一般状态的文件

“E” 在国际申请日的当天或之后公布的在先申请或专利

“L” 可能对优先权要求构成怀疑的文件，或为确定另一篇引用文件的公布日而引用的或者因其他特殊理由而引用的文件（如具体说明的）

“O” 涉及口头公开、使用、展览或其他方式公开的文件

“P” 公布日先于国际申请日但迟于所要求的优先权日的文件

“T” 在申请日或优先权日之后公布，与申请不相抵触，但为了理解发明之理论或原理的在后文件

“X” 特别相关的文件，单独考虑该文件，认定要求保护的发明不是新颖的或不具有创造性

“Y” 特别相关的文件，当该文件与另一篇或者多篇该类文件结合并且这种结合对于本领域技术人员为显而易见时，要求保护的发明不具有创造性

“&” 同族专利的文件

国际检索实际完成的日期	国际检索报告邮寄日期
2019年 9月 20日	2019年 10月 29日
ISA/CN的名称和邮寄地址 中国国家知识产权局(ISA/CN) 中国北京市海淀区蓟门桥西土城路6号 100088 传真号 (86-10)62019451	受权官员 崔岩 电话号码 86-(20)-28950393

表 PCT/ISA/210 (第2页) (2015年1月)

国际检索报告
关于同族专利的信息

国际申请号
PCT/CN2019/076958

检索报告引用的专利文件	公布日 (年/月/日)	同族专利	公布日 (年/月/日)
CN 103607405 A	2014年 2月 26日	CN 103607405 B	2016年 11月 23日
CN 108388807 A	2018年 8月 10日	无	
US 2017091475 A1	2017年 3月 30日	US 9971904 B2	2018年 5月 15日
		WO 2017055256 A1	2017年 4月 6日
		EP 3356954 A1	2018年 8月 8日