北京交通大學
BEIJING JIAOTONG UNIVERSITY

# Attribute Based Proxy Re-encryption with Keyword Search

Yanfeng Shi, Jiqiang Liu, Zhen Han, Qingji Zheng,
Rui Zhang, Shuo Qiu

# Cloud Data

SinaAppEngine
YouTube
Microsoft
facebook
Baidu 百度
新浪微博 weibo.com
Google
EQUIFAX
verizon
YAHOO!
acxiom

# Cloud Privacy

**Micosoft SkyDrive：**
企业或个人的数据、
文档和重要资料等

**FaceBook：**
全球最大照片分享
每天850万张照片上传量

**Twitter：**
用户最新动态
用户新奇想法
照片分享

**YouTube：**
全球最大视频分享
电影剪辑、电视短片、
音乐录像

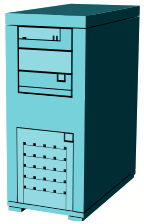
**腾讯QQ：**
用户敏感聊天记录
QQ好友信息

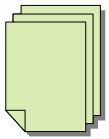
**淘宝网：**
用户与商家交易信息
用户住址
手机信息

**新浪微博：**
好友信息
用户的位置信息
个人相片

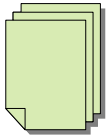
**人人网：**
校友关系信息
用户学校专业等信息

# Encrypt and Outsource





File 1
Owner: John

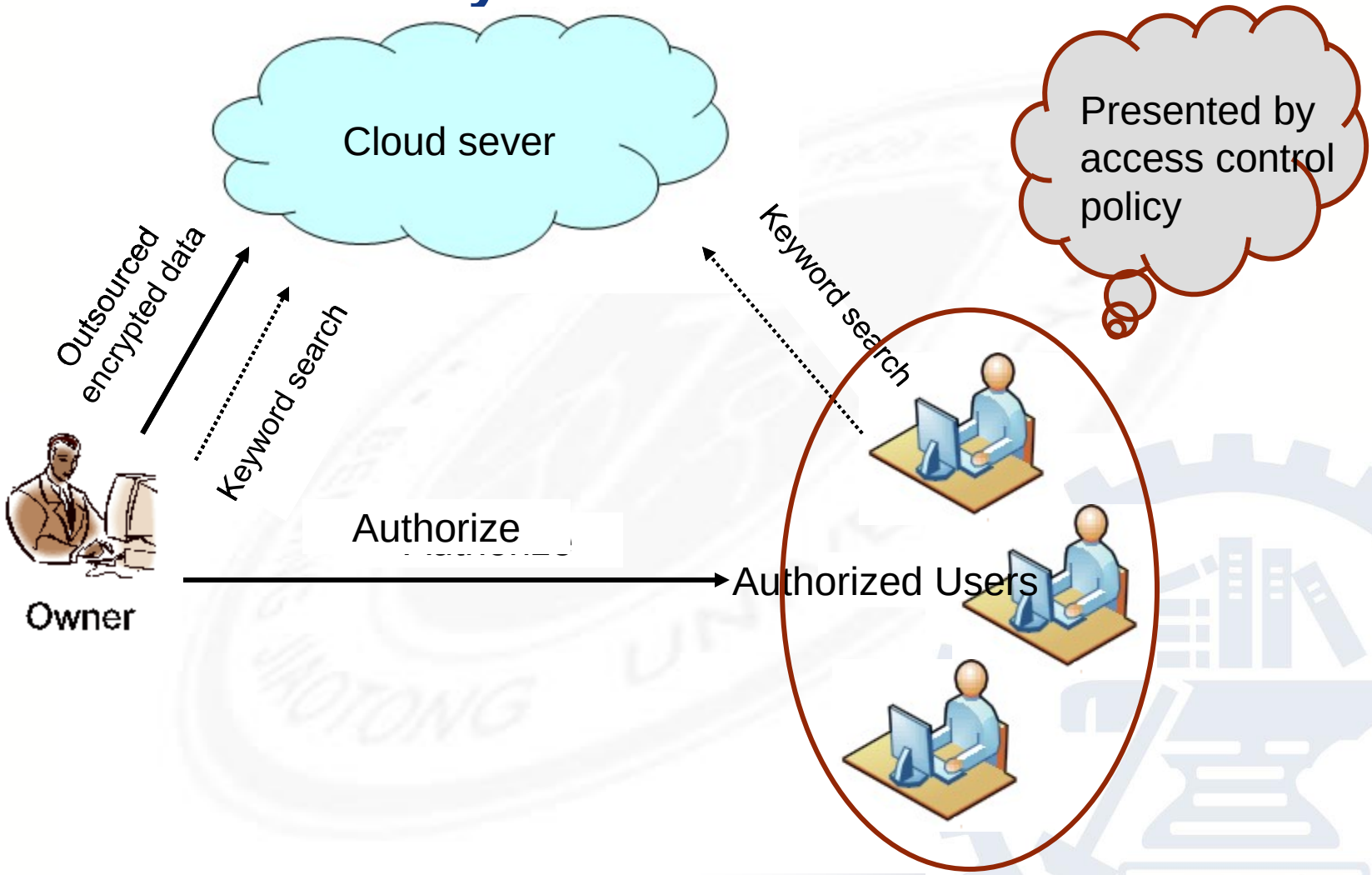


File 2
Owner: Tim

- Encrypted Files stored on Untrusted Server
- Every user can decrypt its own files

# Search

# System Model



# Related Work

Authorized

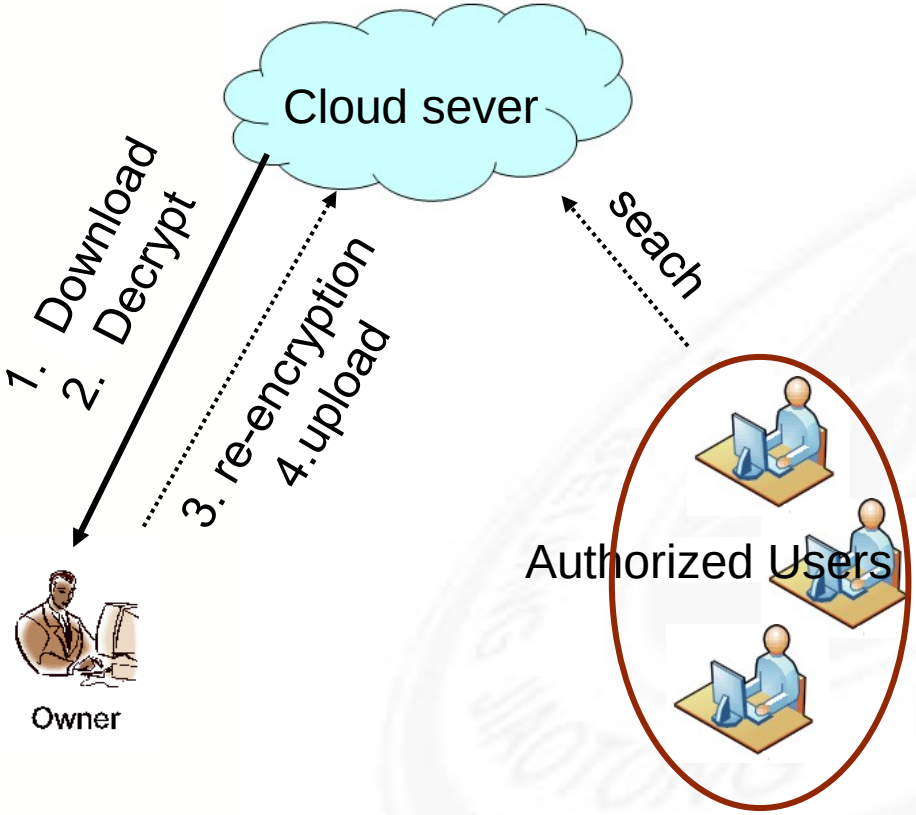
seachable

Fine-grained

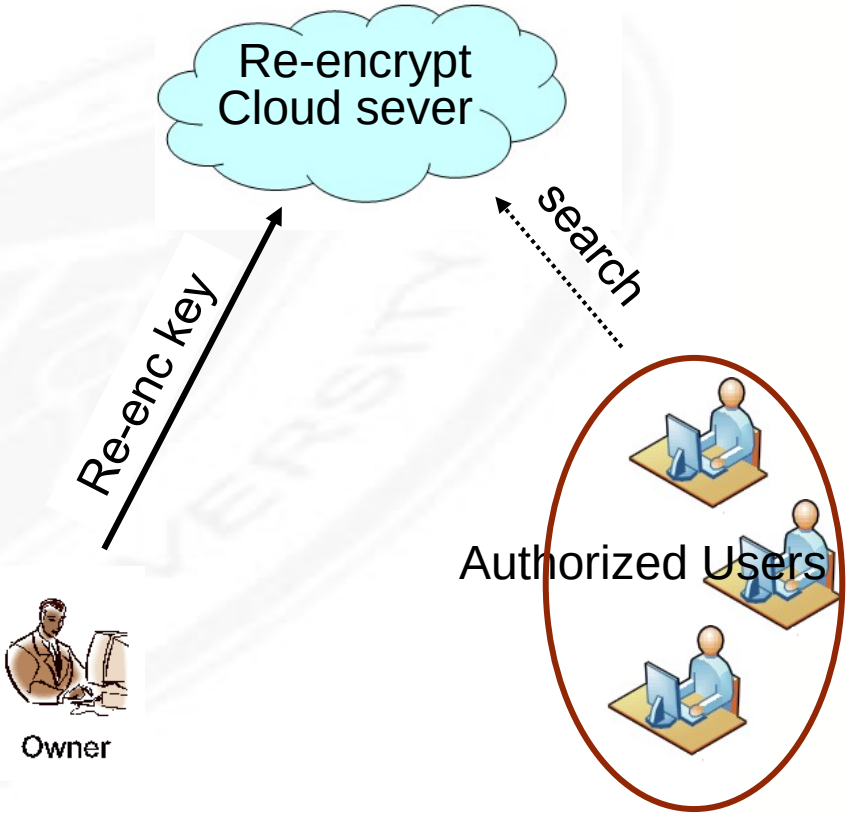
| Scheme | Proxy Re-encryption | Keyword Search | Access Control |
|---|---|---|---|
| PRES [6–11] | √ | √ | × |
| ABE [12–18] | × | × | √ |
| ABKS [19, 20] | × | √ | √ |
| ABPRE [21–26] | √ | × | √ |
| ABRKS(Our solution) | √ | √ | √ |

doi:10.1371/journal.pone.0116325.t001

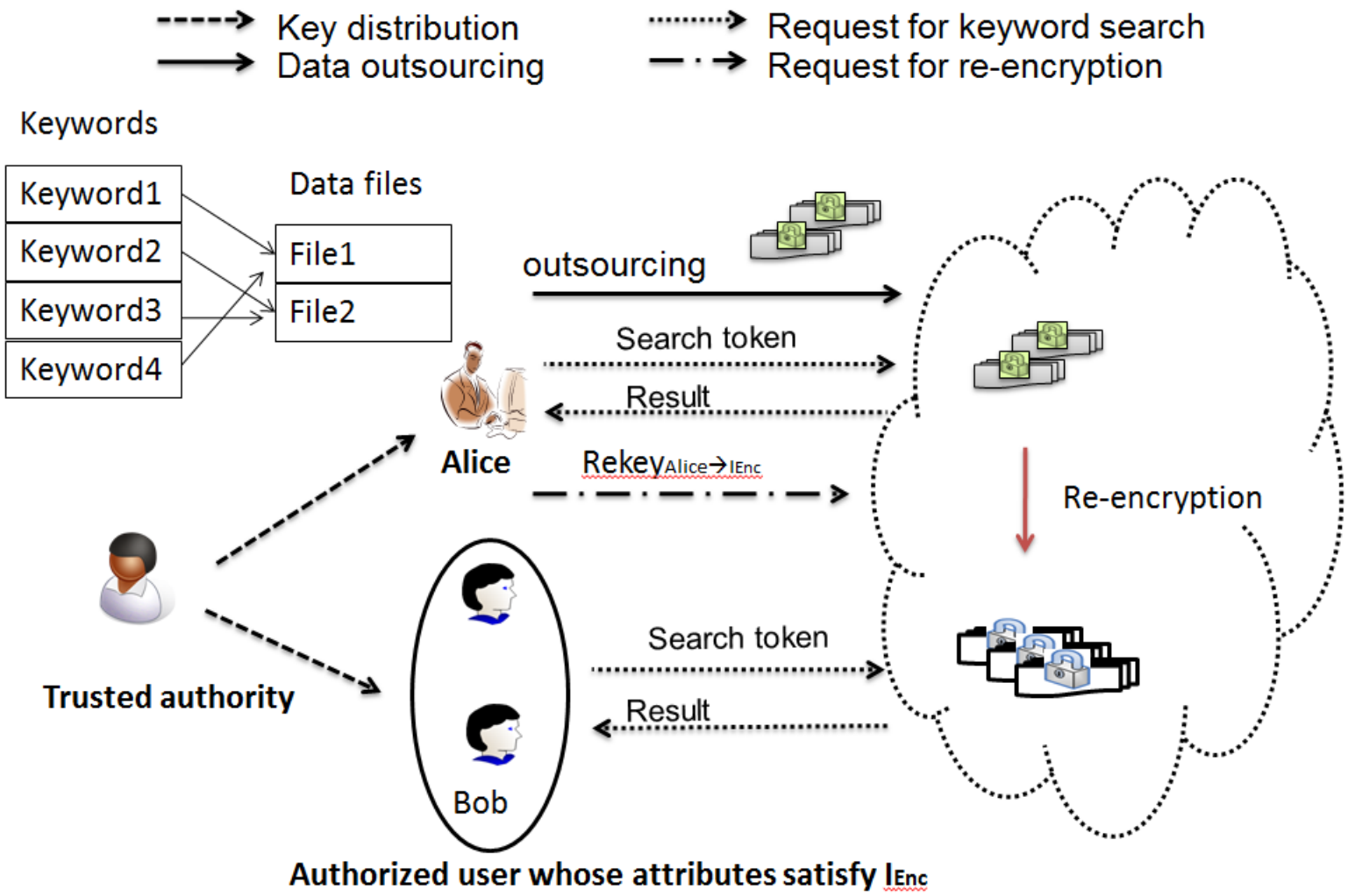

## Traditional Work




## Our work



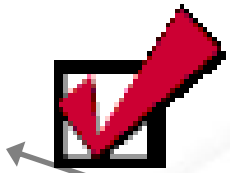
# System Model



# Key-Policy Attribute-Based Proxy Re-encryption with Keyword search

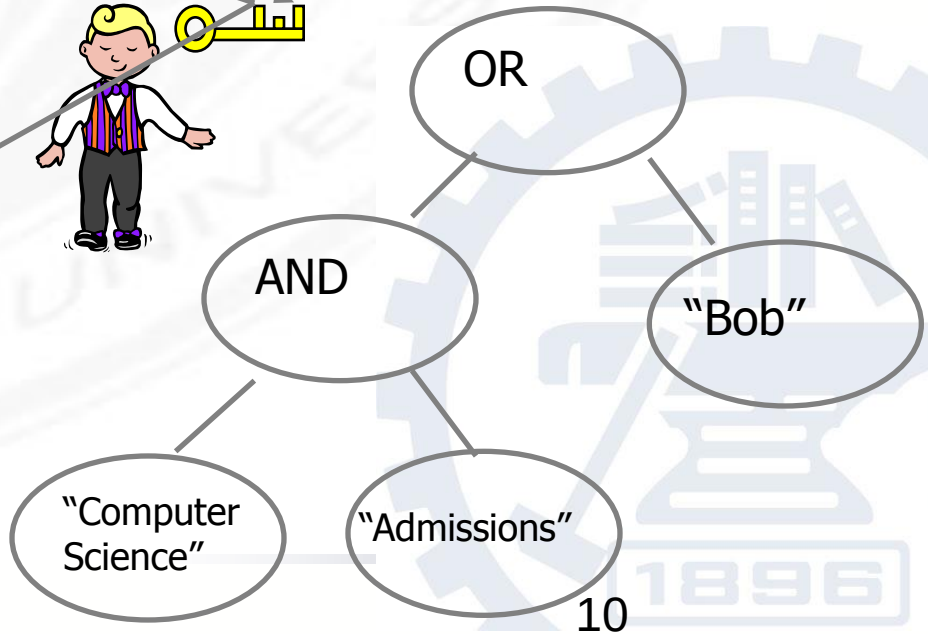
File 1
- "Creator: John"
- "Computer Science"
- "Admissions"
- "Date: 04-11-06"

File 2
- "Creator: Tim"
- "History"
- "Admissions"
- "Date: 03-20-05"



Univ. Key Authority





# Multilinear Maps

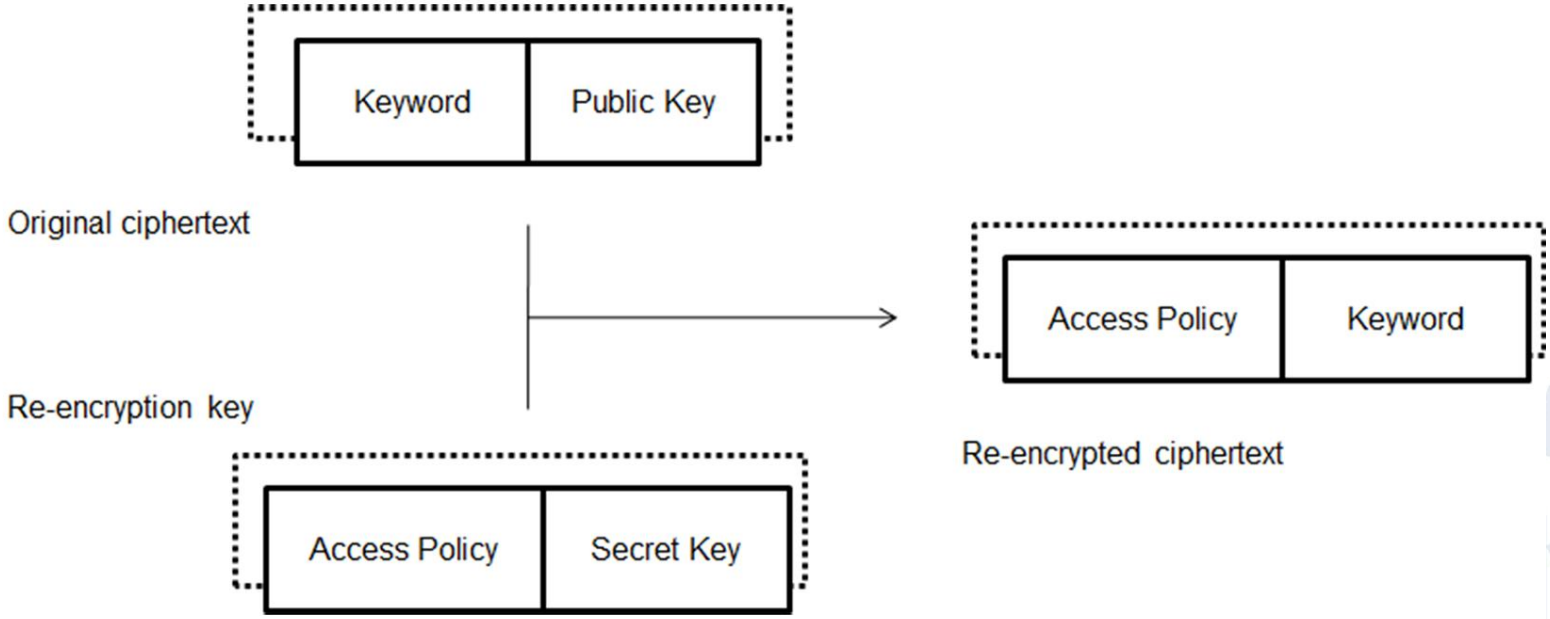
4: $(G_0, G_1, \ldots, G_3)$ order $p$

3 mappings $e_i: G_0 \times G_i \rightarrow G_{i+1}, i = 0, \ldots, 3$

Properties:

- ✓ Given generator $g_0 \in G_0$, then $g_{i+1} = e_i(g_0, g_i)$ is the generator of $G_{i+1}$
- ✓ $e_i\left({g_0}^{\alpha}, {g_i}^{\beta}\right) = e_i(g_0, g_i)^{\alpha\beta}$
- ✓ $e_i$ can be efficiently computed

# Main idea


Keyword
Public Key
Original ciphertext
Access Policy
Keyword
Re-encryption key
Re-encrypted ciphertext
Access Policy
Secret Key

# System Setup

$a, b \xleftarrow{R} Z_p$ , $H_1: \{0,1\}^* \rightarrow Z_p$, $H_0: \{0,1\}^* \rightarrow G_0$,

$h_j \xleftarrow{R} G_0, 0 \leq j \leq max, max$ is the maximum number of attributes. Define $Q(y) = \prod_{j=0}^{max} \left( {h_j}^{y^j} \right), y \in Z_p$

pm=$(e_0, e_1, e_2, G_0, \ldots, G_3, g_0, \ldots, g_2, {g_0}^a, {g_0}^b, H_1, H_2, h_0, \ldots, h_{max})$

mk=$(a, b)$

# Key Generation

mk=$(a, b)$

(1) $v_2, v_3, \ldots, v_k \overset{R}{\leftarrow} Z_p$.

(2) $\mathbf{v} = (ab, v_2, v_3, \ldots, v_k)$,

for $i = 1, \ldots, l$, compute $\lambda_{\pi(i)} = \mathrm{M}_i \cdot \mathbf{v}$, M is an $l \times k$ matrix;

$A_i = g_0^{\lambda_{\pi(i)}} Q(H_1(\pi(i)))^{r_i}, B_i = g_0^{r_i},\ r_i \overset{R}{\leftarrow} Z_p$;

Access control policy

sk=(uid, (M,$\pi$),$(A_i, B_i)_{i\in[1,l]}$)

(3)$sk_{uid} \overset{R}{\leftarrow} Z_p$, compute $sk_{uid} = x_{uid}$, $pk_{uid} = g_0^{x_{uid}}$

# Encryption

$$pk_{uid} = g_0^{x_{uid}}$$

Encrypt the keyword kw $\in \{0,1\}^*$ under the public key $pk_{uid}$

$$C_1 = g_0^r, \quad C_2 = e_1\left(H(kw)^r, e_0(pk_{\mathrm{uid}}, {g_0}^b\right), \; r \xleftarrow{R} Z_p;$$

cph=$(C_1, C_2)$

# ReKey Generation

pm=$(e_0, e_1, e_2, G_0, \dots, G_3, g_0, \dots, g_2, {g_0}^a, {g_0}^b, H_1, H_2, h_0, \dots, h_{max})$

Generate the re-encryption key under attribute set S

(1)$R_1 = \mathrm{d}/x_{\mathrm{uid}}, R_2 = g_0^d, d \overset{R}{\leftarrow} Z_p;$

(2) $R_{\mathrm{at}} = Q(H_1(\mathrm{at}))^{\,d}, \mathrm{at} \in \mathrm{S};$

The Attribute set

$rk_{uid \to S}$=$(R_1, R_2, \{R_{\mathrm{at}}\}_{\mathrm{at} \in \mathrm{S}})$

# Re-Encryption

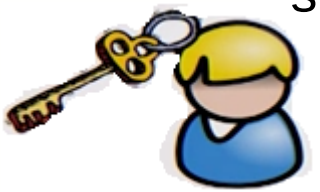
$rk_{uid \to S}$=($R_1$, $R_2$, $\{R_{at}\}_{at \in S}$), cph=($C_1$, $C_2$)

Re-encrypt the ciphertext cph under attribute set S,

$$C_2' = C_2^{R_1}$$

cph$^R$=($C_1$, $C_2'$, $C^{(2)}$, $R_2$, $\{R_{at}\}_{at \in S}$ )

# Token Generation



$\mathrm{sk} = (\mathrm{uid}, (\mathrm{M},\pi), (A_i, B_i)_{i\in[1,l]}), sk_{uid} = x_{uid},$

For each "satisfied" node (S satisfies $(\mathrm{M},\pi)$) perform a computation

(1) With uid, the token for the keyword $kw$ over original encrypted keywords as $token = H(kw)^{x_{uid}}$

(2) With the secret key associated with access control policy, the token for the keyword $kw$over re-encrypted keywords as

$A'_i = e_0(H(kw), A_i),\quad B'_i = e_0(H(kw), B_i), token^R = \left((\mathrm{M},\pi), (A'_i, B'_i)_{i\in[1,l]}\right)$

# Search

(1)With the token over original encrypted keywords , the search algorithm outputs 1 if $C_2 = e_2(\text{token}, e_1(C_1,\ e_0(g_0^a, {g_0}^b))$

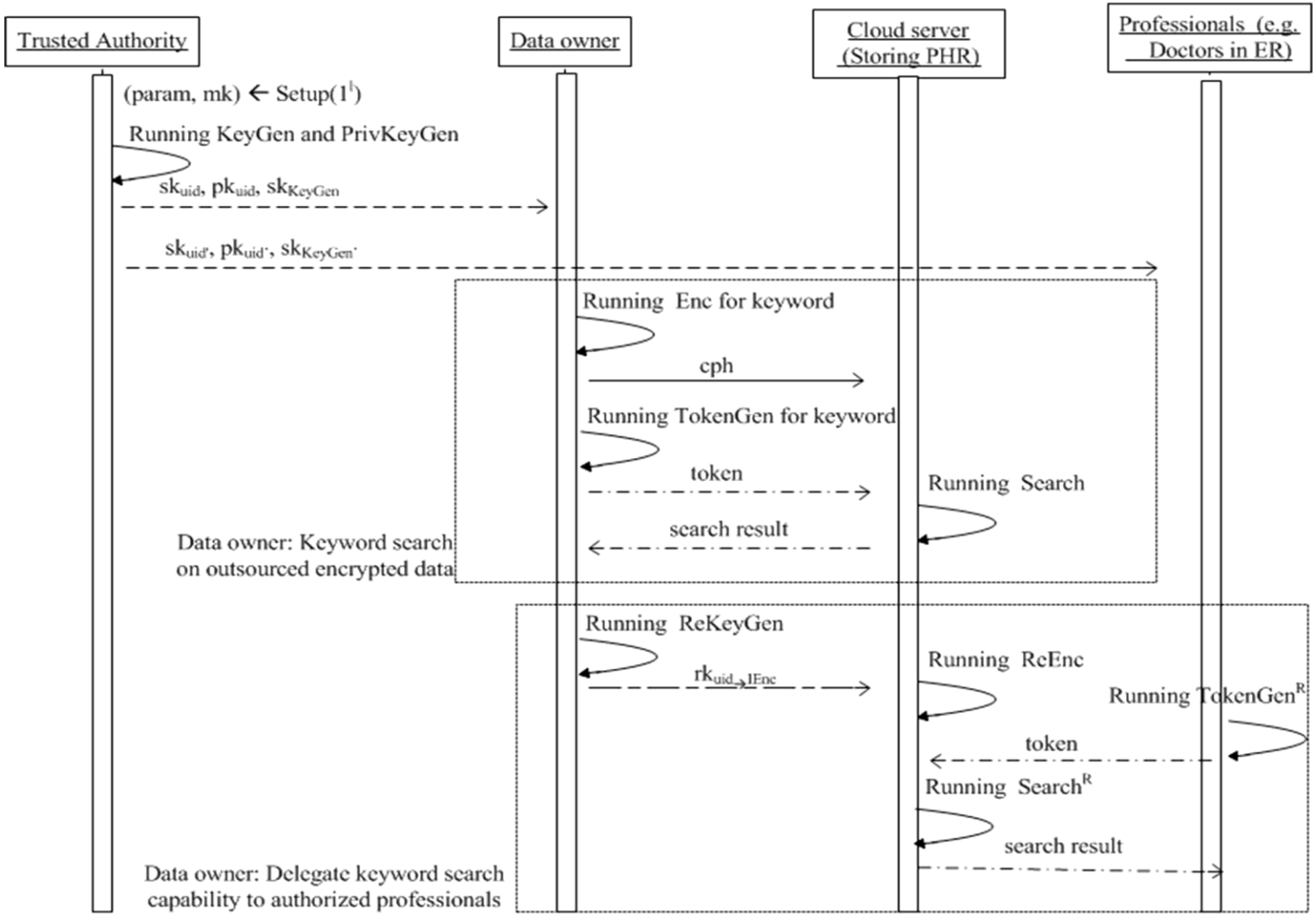
(2) With the token over re-encrypted keywords, the search algorithm can be done as follows:

If S satisfies (M,$\pi$), exist $c_i,\ s.t.\ \sum_{\pi(i)\in S} c_i M_i$=(1,0,...,0), then

$$K = \prod_{\pi(i)\in S} \left(\frac{e_1(R_2, A_i')}{e_1(R_{\pi(i)}, B_i')}\right)^{c_i}.$$

If $e_2$(K, $C_1$ )=$C'_2$, output 1 and 0 otherwise.

# Work Process



北京交通大學
BEIJING JIAOTONG UNIVERSITY

# Thanks!