Attribute-Based Re-Encryption for Data Cloud Secure Storage and Access in End-Edge-Cloud Networks

Tingting Kou[1], Zhicheng Dong*[1], Jing Yang[2], Jing Zhao[3], Donghong Cai[2]

[1]Tibet University, [2] Jinan University, [3]Chengdu Technological University

Abstract

- Problem: Cloud storage can enhance the efficiency of the data storage and sharing, while it is challenge for the security of the data transmission. Attribute-Based Encryption (ABE) improves security of data sharing, but the main issues are key management and user-side computational efficiency.
- Algorithm: The proposed Online-Offline Re-Encryption (OORE) algorithm addresses these challenges.
- Results: The proposed OORE algorithm combines online and offline encryption/decryption operations, with most complex computations performed offline. During the online phase, it only requires a single pairing and a few divisions and multiplications, reducing the computational burden on users. Security analysis shows that the OORE algorithm can resist chosen ciphertext attacks, confirming its security. Experimental results demonstrate that the algorithm effectively lowers the computational load on users while keeping ciphertext and key storage overhead within reasonable limits.

Introduction

- With the rise of cloud storage, data security and privacy, especially in shared data access, are critical concerns.
- ABE faces the key escrow problem, where the Attribute Authority (AA) can generate users' private keys, potentially allowing it to decrypt any user's data.
- Contribution:

1. The proposed OORE algorithm addresses the key escrow problem through key computation techniques.
2. Re-encryption with distinct access control structures for each encryption.
3. We provide a formal security proof.

System Model

To address the key management issue in ABE cloud storage and enhance the security of cloud storage, we consider a end-edge-cloud network , which has four entities including Attribute Authority (AA), users, edge routers, and cloud servers, as illustrated in Fig. 1.

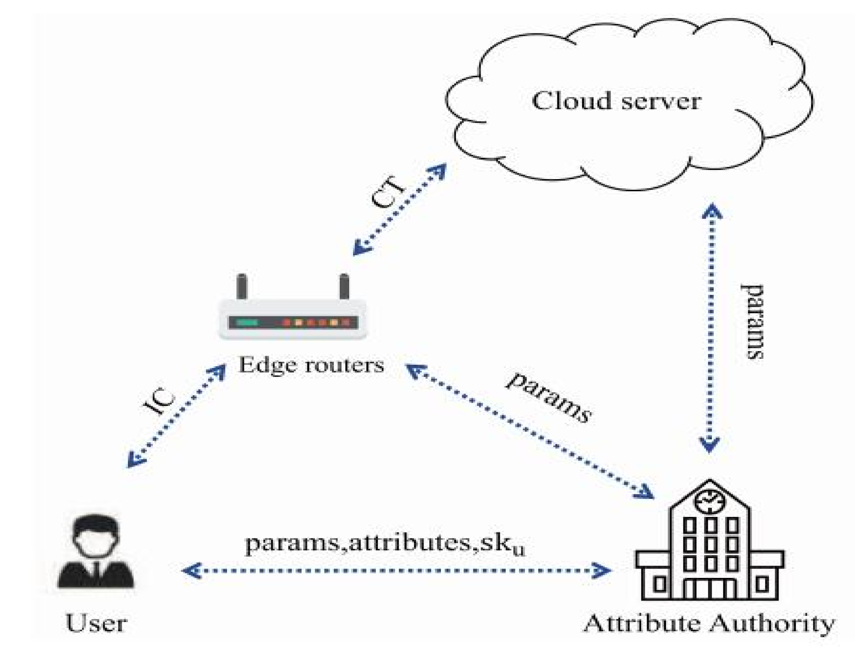

Fig. 1. System model.

Our Prosed OORE Algorithm

We propose an OORE algorithm for data cloud storage and sharing. Specifically, the key escrow issue, where AA holds the user's private key, is addressed through key computation operations. Additionally, re-encryption operations enhance the security of cloud storage. Our OORE algorithm is comprised of eight stages: Setup、Key Generation、Key Computation、Encryption、 Re-Encryption Key Generation、Re-Encrypted Ciphertext、Offline Decryption、Online Decryption.

Algorithm 1 The proposed OORE Algorithm

Input: the security parameter λ, message m, access control structure 1 $(\mathbb{M}, \rho)$, and access control structure 2 $(\bar{\mathbb{M}}, \bar{\rho})$;

Output: the message m;

PP = calculate public params (security parameters λ);
msk = generate master key ();
sk = generate private key (PP, msk, attributes);
sk_u = compute (sk, PP);
CT = encrypt (message m, PP, access control structure 1 $(\mathbb{M}, \rho)$);
rk = generate reencryption key (sk_u, PP, access control structure 2 $(\bar{\mathbb{M}}, \bar{\rho})$);
CT_{ReEnc} = reencrypt (CT, rk);
while user requests CT **do**
    CT = edge server download CT from cloud;
    if Access control structure 1 = $(\mathbb{M}, \rho)$ **then**
        $CT_{offline}$ = decrypt offline (CT, rk, CT_{ReEnc});
        return $CT_{offline}$;
    else
        return None;
    end if
end while
send $CT_{offline}$ to user;
if Access control structure 2 = $(\bar{\mathbb{M}}, \bar{\rho})$ **then**
    m = decrypt online ($CT_{offline}$, sk_u);
    return m;
else
    return None;
end if

Key Computation:

$$sk_u = (S, sk_{u1} = g^{\delta} g^{ac} g^{t}, sk_{u2} = g^{c}, sk_{u4} = g^{t}, \forall x \in S, sk_{ux} = d_x^{c})$$

Encryption:

$$C = m \cdot e(g,g)^{\delta s}, C_1 = g^{s}, C_2 = b^{s}, \sigma_m = H(m), C_{3,i} = g^{a\lambda_i} d_{\rho(i)}^{-h_i}, C_{4,i} = g^{h_i}, \forall i \in [1, l]$$

Re-Encryption Key Generation:

$$rk_1 = sk_{u1}^{\chi} \cdot b^{\delta}, rk_2 = g^{\delta}, rk_3 = sk_{u2}^{\chi}. rk_{4,x} = sk_{ux}^{\chi}, rk_5 = \chi \cdot e(g,g)^{\bar{s}}, rk_{6,i} = e(g,g)^{\bar{\lambda}_i}$$

Re-Encrypted Ciphertext:

$$CT_{ReEnc} = C \cdot rk_5$$

Results

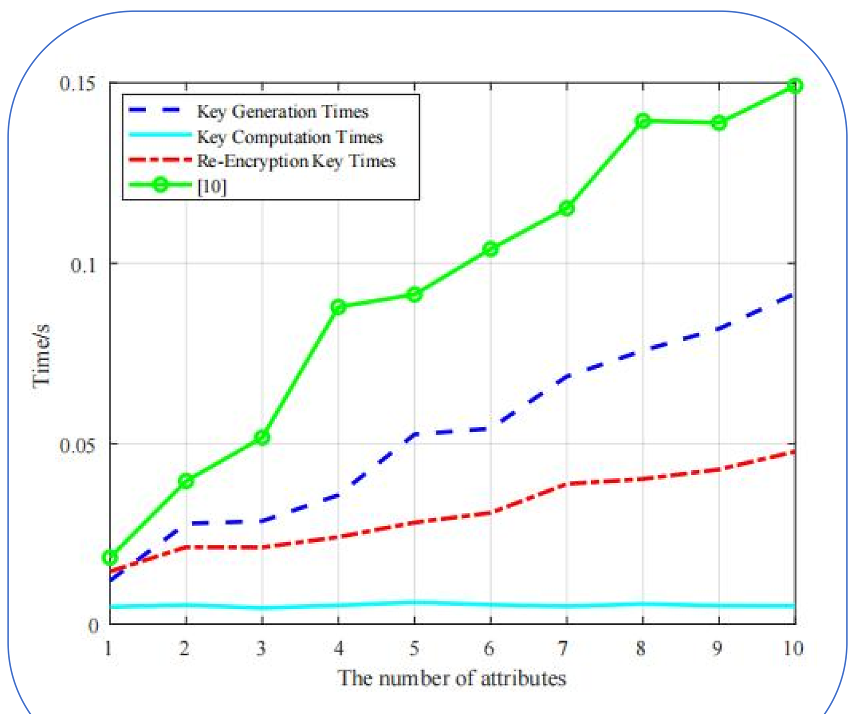

Fig. 2. KeyGen Time.

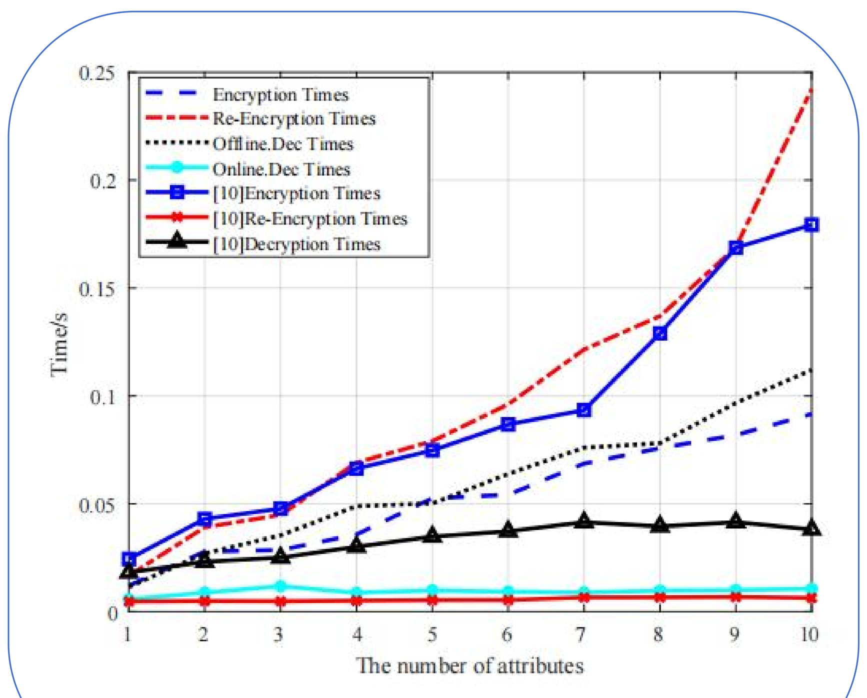

Fig. 3. Encryption and Decryption Time.

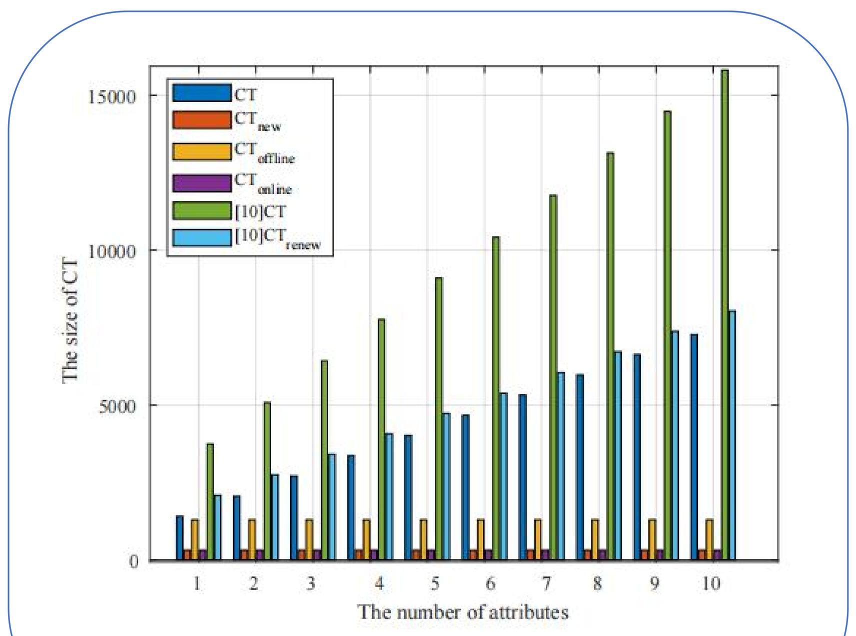

Fig. 4. Ciphertext Sizes.

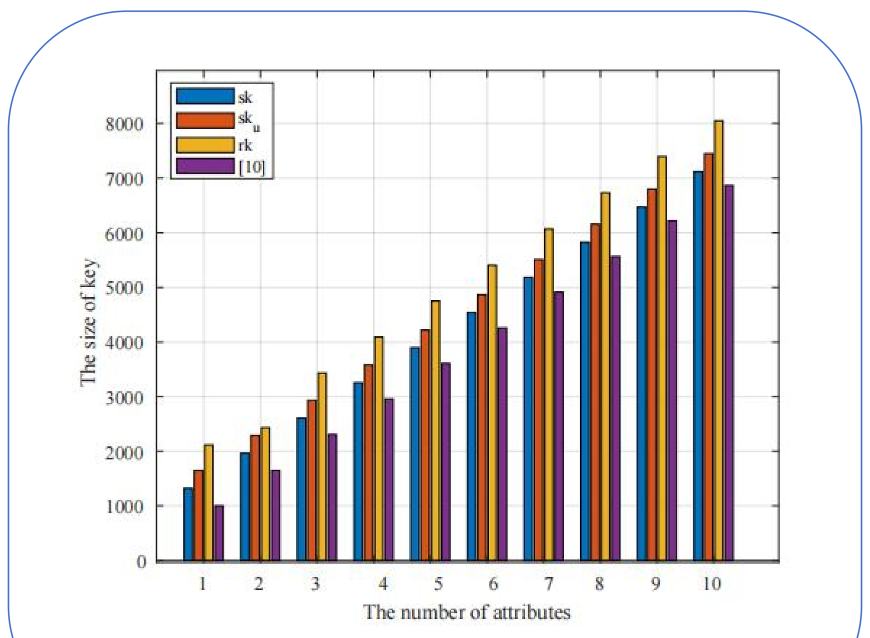

Fig. 5. Key Sizes in the System.

This work was supported by the Science and Technology Major Project of Tibet an Autonomous Region of China under Grant No.XZ202201ZD0006G. The Lhasa Science and Technology Plan Project under Grant No.LSKJ202405. The Xizang Autonomous Region Central Guided Local Science and Tech_x0002_nology Development Fund Project under Grant No.XZ202401YD0015. The National Key Research and Development Program of China under Grant No.2023YFC3321705. The Basic and Applied Basic Research Foundation of Guangdong Province under Grant No.2024A1515012398.