An Adaptive Secure and Practical Data Sharing System with Verifiable Outsourced Decryption

一种自适应安全和实用的可验证外包解密数据共享系统

作者：Shengmin Xu and Xingshuo Han
期刊：IEEE Transactions on Services Computing

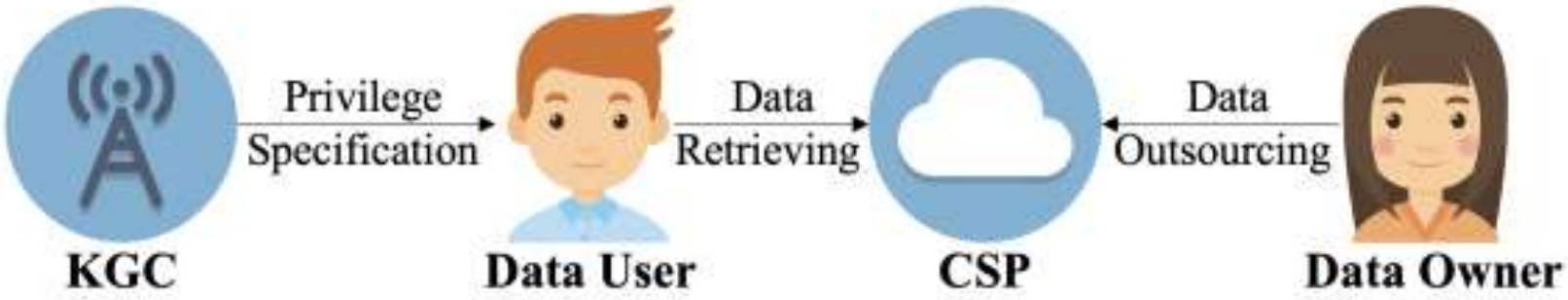
汇报人：王璇　　**指导老师：李琦**

目录

CONTENTS

01 研究背景

研究背景

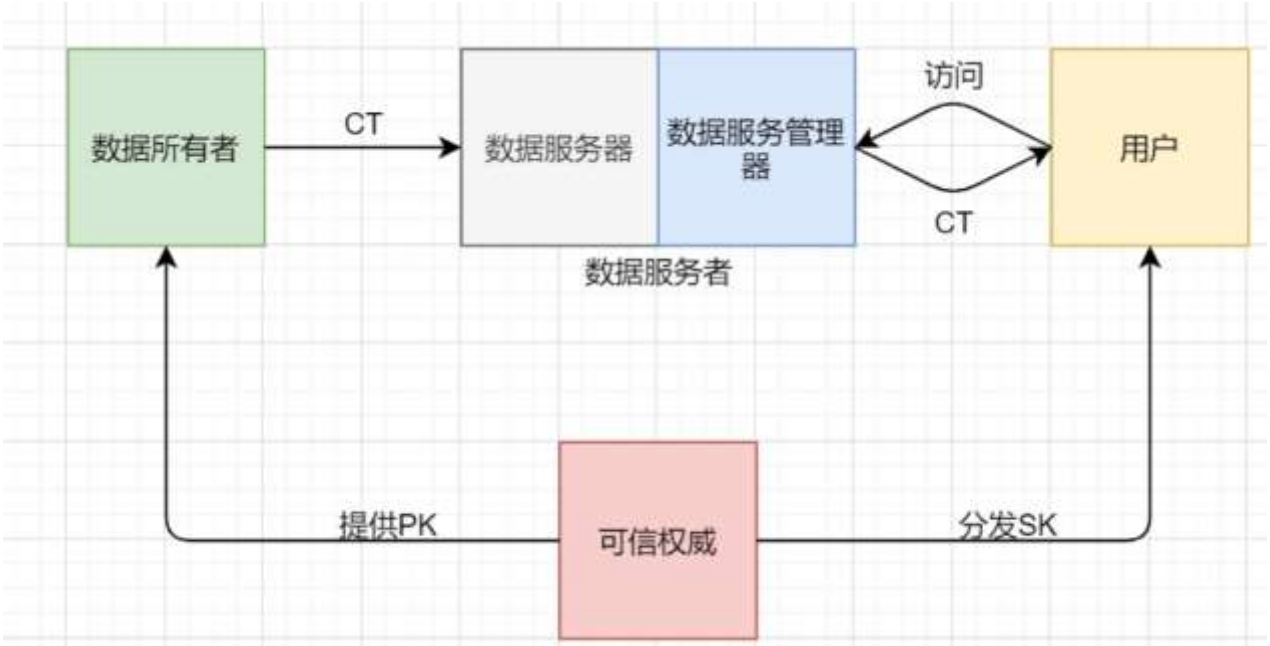


Cloud-Based Data Sharing Model

数据机密性难以保证 → 因此 → 引入KGC及其他密码学工具 → 因此 → 将计算外包给云

研究背景

ABE结构



- 密钥生成中心(权威Authority)
- 加密者(数据所有者DataOwner, DO)
- 解密者(用户User)
- 数据存储服务器

密钥生成中心负责产生公共参数PK和主密钥MK。当有用户User发出请求时，为其分配属性，生成与权限索引相关的解密钥SK。
数据所有者DO根据密文索引A和自己要共享的数据Data的安全需要生成密文C，然后发送给数据存储服务器。
User想获取某个被共享的数据时，向服务器发起请求，服务器为User发送请求访问的密文数据C。
当用户User满足A的要求时，可以利用Authority分配的解密钥SK和收到的密文C，得到授权并获得Data。

研究背景

ABE的优缺点

优势

- 支持细粒度访问控制
- 保护数据隐私

劣势

- 高计算和存储成本（与资源受限的设备不兼容）
- 安全性和可信任问题

研究背景

现有解决方案（经过改进的ABE）的优缺点

优势

- 实现无安全通道的密钥发布
- 无需昂贵的数据解密
- 实现可验证外包解密

劣势

- 仅提供选择性安全性
- CSP可能不诚实地处理外包数据解密
- 底层ABE必须是ElGamal类型的ABE

研究背景

"Can we design a practical VO-ABE with adaptive security under the **standard assumption** to realize a **practical** and **secure** data-sharing system?"

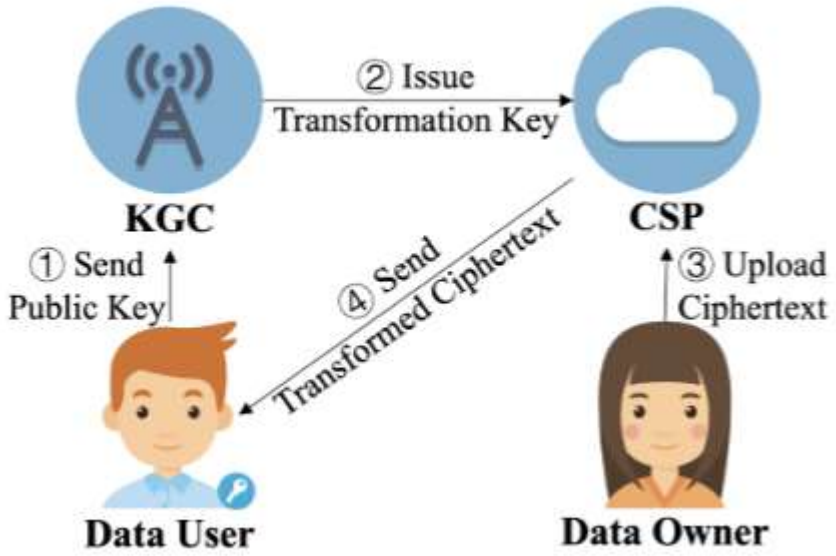
本文介绍了第一个自适应安全的标准假设下的实用的VO-ABE。与之前的相比，我们的解决方案继承了所有的实际属性（例如，细粒度的访问控制、可扩展的密钥发布、可验证的外包解密）并且享有更高级别的安全性。

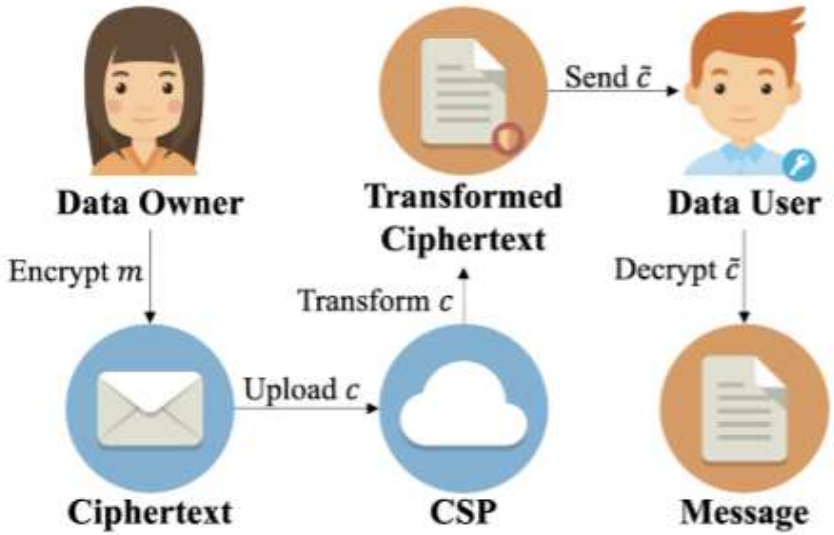
02

研　究　内　容

研究内容

系统概览





· KGC负责初始化系统并为每个数据用户发布转换密钥。

· 数据所有者是希望通过CSP与指定的数据用户共享敏感数据的客户端。

· 数据用户是生成密钥对并享受可验证外包数据共享的客户端，其中数据用户可以通过资源受限的终端设备接收共享的数据。

· 作为远程服务器的CSP占用大量存储空间，并提供按需数据服务。

研究内容

系统概览 VO-ABE的形式定义

四个标识符空间

标识符空间I

属性域U

消息空间M

标签空间T

六个算法

$\mathcal{ABE}_{VO}.\mathsf{Setup}(1^\lambda) \rightarrow (pp, mpk, msk)$

$\mathcal{ABE}_{VO}.\mathsf{KeyGen}(id) \rightarrow (pk_{id}, sk_{id})$

$\mathcal{ABE}_{VO}.\mathsf{TKGen}(msk, pk_{id}, \mathcal{S}) \rightarrow tk_{id}$

$\mathcal{ABE}_{VO}.\mathsf{Enc}(\mathbb{A}, m) \rightarrow (c, vk)$

$\mathcal{ABE}_{VO}.\mathsf{Transform}(tk_{id}, c) \rightarrow \tilde{c}_{id}/\perp$

$\mathcal{ABE}_{VO}.\mathsf{Dec}(sk_{id}, \tilde{c}_{id}, vk) \rightarrow m/\perp$

研究内容

系统概览

- 无需任何安全通道的可扩展密钥发布协议

在传统模型中,由KGC生成data user的(sk,pk),但是在作者的算法中sk由数据用户生成，用户将pk传递给KGC。KGC通过公共通道返回基于属性的转换密钥。没有秘密密钥，相应的基于属性的转换密钥是无用的。

研究内容

系统概览

- 生成公私钥对的算法

$\mathcal{ABE}_{\mathcal{VO}}$.KeyGen$(id) \rightarrow (pk_{id}, sk_{id})$: *On input an identifier* $id \in \mathcal{I}$, *and output a public key* pk_{id} *and a secret key* sk_{id}. *In our system, each data user runs the key generation algorithm to generate* (pk_{id}, sk_{id}), *where* pk_{id} *is outsourced to the CSP and* sk_{id} *is kept in secret.*

- 生成转换密钥pk的算法

$\mathcal{ABE}_{\mathcal{VO}}$.TKGen$(msk, pk_{id}, \mathcal{S}) \rightarrow tk_{id}$: *On input a master secret key* msk, *a public key* pk_{id} *and a set of attributes* $\mathcal{S} \subseteq \mathbb{U}$, *and output a transformation key* tk_{id}. *In our system, the KGC runs the transformation key generation algorithm to derive* tk_{id}, *where* tk_{id} *is outsourced to the CSP.*

$\mathcal{ABE}_{\mathcal{VO}}$.KeyGen$(id) \rightarrow (pk_{id}, sk_{id})$: Pick $\alpha \in \mathbb{Z}_p^*$. Return $sk_{id} = \alpha$ and $pk_{id} = g^{\alpha}$.

$\mathcal{ABE}_{\mathcal{VO}}$.TKGen$(msk, pk_{id}, \mathcal{S}) \rightarrow tk_{id}$: Parse $msk = (a_1, a_2, b_1, b_2, g^{d_1}, g^{d_2}, g^{d_3}, d_3)$.

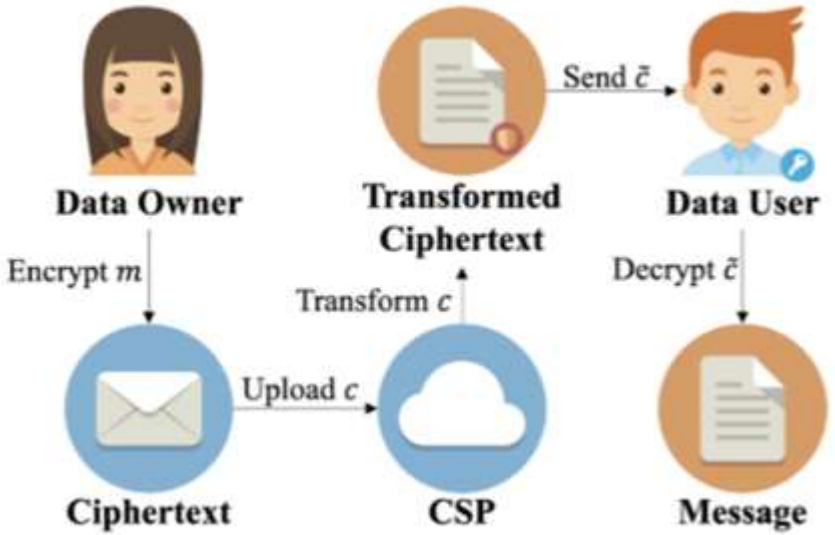
- Pick $r_1, r_2, \{\sigma_y\}_{y \in \mathcal{S}}, \sigma' \in \mathbb{Z}_p$. Set $tk_0 = (h^{b_1 r_1}, h^{b_2 r_2}, h^{r_1 + r_2})$.
- Compute $tk_{y,t} = \mathcal{H}(y1t)^{\frac{b_1 r_1}{a_t}} \cdot \mathcal{H}(y2t)^{\frac{b_2 r_2}{a_t}} \cdot \mathcal{H}(y3t)^{\frac{r_1 + r_2}{a_t}} \cdot g^{\frac{\sigma_y}{a_t}}$ for all $y \in \mathcal{S}$ and $t = 1, 2$. Set $tk_y = (tk_{y,1}, tk_{y,2}, g^{-\sigma_y})$.
- Compute $tk'_t = g^{d_t} \cdot \mathcal{H}(011t)^{\frac{b_1 r_1}{a_t}} \cdot \mathcal{H}(012t)^{\frac{b_2 r_2}{a_t}} \cdot \mathcal{H}(013t)^{\frac{r_1 + r_2}{a_t}} \cdot g^{\frac{\sigma'}{a_t}}$ for $t = 1, 2$. Set $tk' = (tk'_1, tk'_2, pk_{id}^{d_3} \cdot pk_{id}^{-\sigma'})$.
- Return $tk_{id} = (tk_0, \{tk_y\}_{y \in \mathcal{S}}, tk')$.

研究内容

系统概览

- 一个可验证的外包解密机制，用于资源受限的终端设备对抗恶意云服务提供商

外包：数据所有者通过指定访问策略对消息m进行封装，并将封装后的消息c外包给CSP。CSP使用tk生成转换后的密文c'以替代访问策略对数据用户的秘密密钥的限制。最后，数据用户使用sk来解密c'以揭示消息。因此，昂贵的工作负载(如匹配属性和访问策略)分配给了CSP，数据用户从中受益，享受轻量级数据解密。



研究内容

系统概览

- 一个可验证的外包解密机制，用于资源受限的终端设备对抗恶意云服务提供商

可验证：我们利用了私有可验证性的概念，其中验证密钥（vk）拥有者，即其转换密钥在密文转换期间被处理的数据用户，可以验证转换后的密文的格式。

$\mathcal{ABE}_{VO}.\mathsf{Enc}(\mathbb{A}, m) \rightarrow (c, vk)$: *On input an access policy* $\mathbb{A} \in 2^{\mathbb{U}}$ *and a message* $m \in \mathcal{M}$, *and output a ciphertext* c *and a verification key* $vk \in \mathcal{T}$. *In our system, the data owner runs the encryption algorithm to derive* (c, vk), *where* c *and* vk *are outsourced to the CSP.*

$\mathcal{ABE}_{VO}.\mathsf{Dec}(sk_{id}, \tilde{c}_{id}, vk) \rightarrow m/\perp$: *On input a secret key* sk_{id}, *a transformed ciphertext* $\tilde{c}_{id}$ *and a verification key* $vk \in \mathcal{T}$, *and output a message* $m \in \mathcal{M}$ *if* $\tilde{c}_{id}$ *is well-formed; otherwise output a failure symbol* $\perp$. *In our system, the data user runs the decryption algorithm to reveal* m.

系统概览

$\mathcal{ABE}_{\mathcal{VO}}.\mathsf{Enc}(\mathbb{A}, m) \to (c, vk)$: Parse $\mathbb{A} = (\mathbb{M}, \pi)$, where $\mathbb{M}$ has n_1 rows and n_2 columns, and $\pi : [n_1] \to \mathbb{G}$ is a mapping function.

- Pick $k \in \mathbb{G}_T$, $s_1, s_2 \in \mathbb{Z}_p^*$. Set $c_0 = (H_1^{s_1}, H_2^{s_2}, h^{s_1+s_2})$.
- Compute $c_{i,\ell} = \mathcal{H}(\pi(i)\ell 1)^{s_1} \cdot \mathcal{H}(\pi(i)\ell 2)^{s_2} \cdot \prod_{j=1}^{n_2} [\mathcal{H}(0j\ell 1)^{s_1} \cdot \mathcal{H}(0j\ell 2)^{s_2}]^{\mathbb{M}_{i,j}}$ for $i = 1, \ldots, n_1$ and $\ell = 1, 2, 3$, where $\mathbb{M}_{i,j}$ denotes the (i, j)th element of $\mathbb{M}$. Set $c_i = (c_{i,1}, c_{i,2}, c_{i,3})$.
- Compute $c' = k \cdot T_1^{s_1} \cdot T_2^{s_2}$ and $\tilde{k} = \tilde{\mathcal{H}}(k)$. Run $\mathcal{SE}.\mathsf{Enc}(\tilde{k}, m)$ to generate $c_{\mathcal{SE}}$.
- Return $c = (c_{\mathcal{SE}}, c_0, c_1, \ldots, c_{n_1}, c')$, $vk = \ddot{\mathcal{H}}(\dot{\mathcal{H}}(k), c_{\mathcal{SE}})$.

$\mathcal{ABE}_{\mathcal{VO}}.\mathsf{Dec}(sk_{id}, \tilde{c}_{id}, vk) \to m/\perp$: Parse $sk_{id} = \alpha$.

- Compute $k = \tilde{c}_1 / \tilde{c}_2^{\frac{1}{\alpha}}$. Return $\perp$ if $vk \neq \ddot{\mathcal{H}}(\dot{\mathcal{H}}(k), c_{\mathcal{SE}})$.
- Compute $\tilde{\mathcal{H}}(k) = \tilde{k}$. Return $m \leftarrow \mathcal{SE}.\mathsf{Dec}(\tilde{k}, c_{\mathcal{SE}})$.

可验证外包解密可以实现自适应安全性，即使攻击者可以在加密和解密过程中进行动态调整和选择，系统也能够保持数据的安全性和正确性。这得益于验证密钥的存在，以及对解密过程的严格验证，从而防止了潜在的攻击。

03 实 验 结 果

效率分析

TABLE 2: Computational Complexity Comparison

	Setup	KeyGen	TKGen	Enc	Transform	Dec
Wat11 [42]	$\mathcal{O}(\mathbb{U})$	$\mathcal{O}(\mathcal{S})$	N/A	$\mathcal{O}(\mathbb{A})$	N/A	$\mathcal{O}(\mathcal{S})$
GHW11 [21]	$\mathcal{O}(1)$	$\mathcal{O}(\mathcal{S})$	$\mathcal{O}(\mathcal{S})$	$\mathcal{O}(\mathbb{A})$	$\mathcal{O}(\mathcal{S})$	$\mathcal{O}(1)$
RW13 [35]	$\mathcal{O}(1)$	$\mathcal{O}(\mathcal{S})$	N/A	$\mathcal{O}(\mathbb{A})$	N/A	$\mathcal{O}(\mathcal{S})$
LDGW13 [23]	$\mathcal{O}(\mathbb{U})$	$\mathcal{O}(\mathcal{S})$	$\mathcal{O}(\mathcal{S})$	$\mathcal{O}(\mathbb{A})$	$\mathcal{O}(\mathcal{S})$	$\mathcal{O}(1)$
QDLL15 [33]	$\mathcal{O}(\mathcal{I})$	$\mathcal{O}(1)$	$\mathcal{O}(\log \mathcal{N})$	$\mathcal{O}(1)$	$\mathcal{O}(1)$	$\mathcal{O}(1)$
QDLM15 [34]	$\mathcal{O}(\mathbb{U})$	$\mathcal{O}(1)$	$\mathcal{O}(\mathcal{S})$	$\mathcal{O}(\mathbb{A})$	$\mathcal{O}(\mathcal{S})$	$\mathcal{O}(1)$
CDLQ16 [14]	$\mathcal{O}(1)$	$\mathcal{O}(1)$	$\mathcal{O}(\mathcal{S} \cdot \log \mathcal{N})$	$\mathcal{O}(\mathbb{A})$	$\mathcal{O}(\mathcal{S})$	$\mathcal{O}(1)$
AC17 [2]	$\mathcal{O}(1)$	$\mathcal{O}(\mathcal{S})$	N/A	$\mathcal{O}(\mathbb{A})$	N/A	$\mathcal{O}(\mathcal{S})$
NCD+18 [29]	$\mathcal{O}(1)$	$\mathcal{O}(1)$	$\mathcal{O}(\mathcal{S})$	$\mathcal{O}(\mathbb{A})$	$\mathcal{O}(\mathcal{S})$	$\mathcal{O}(1)$
LWZH20 [26]	$\mathcal{O}(\mathbb{U})$	$\mathcal{O}(\mathcal{S})$	$\mathcal{O}(\mathcal{S})$	$\mathcal{O}(\mathbb{A})$	$\mathcal{O}(\mathcal{S})$	$\mathcal{O}(1)$
XNH+21 [44]	$\mathcal{O}(\mathcal{T})$	$\mathcal{O}(1)$	$\mathcal{O}(\mathcal{S}_{\text{rec}} \cdot \log \mathcal{N})$	$\mathcal{O}(\mathcal{S}_{\text{snd}} + \mathbb{A}_{\text{rec}})$	$\mathcal{O}(\mathcal{S}_{\text{rec}})$	$\mathcal{O}(1)$
XNM+21 [45]	$\mathcal{O}(1)$	$\mathcal{O}(1)$	$\mathcal{O}(\mathcal{S}_{\text{rec}})$	$\mathcal{O}(\mathcal{S}_{\text{snd}} + \mathbb{A}_{\text{rec}})$	$\mathcal{O}(\mathcal{S}_{\text{rec}})$	$\mathcal{O}(1)$
XLD+22 [43]	$\mathcal{O}(1)$	$\mathcal{O}(\mathcal{S}_{\text{rec}})$	N/A	$\mathcal{O}(\mathcal{S}_{\text{snd}} + \mathbb{A}_{\text{rec}})$	N/A	$\mathcal{O}(\mathcal{S}_{\text{rec}})$
Our SA-ABE	$\mathcal{O}(1)$	$\mathcal{O}(1)$	$\mathcal{O}(\mathcal{S})$	$\mathcal{O}(\mathbb{A})$	$\mathcal{O}(\mathcal{S})$	$\mathcal{O}(1)$
Our VO-ABE	$\mathcal{O}(1)$	$\mathcal{O}(1)$	$\mathcal{O}(\mathcal{S})$	$\mathcal{O}(\mathbb{A})$	$\mathcal{O}(\mathcal{S})$	$\mathcal{O}(1)$

$\mathbb{U}$: the capacity of the attribute universe;
$\mathcal{S}$: the capacity of the attribute set;
$\mathcal{N}$: the maximum number of system users;
$\mathcal{S}_{\text{rec}}$: $\mathcal{S}$ of the data user;
$\mathbb{A}_{\text{rec}}$: $\mathbb{A}$ of the data user;
$\mathbb{A}$: the length of the access policy;
$\mathcal{I}$: the space of the identifier;
$\mathcal{T}$: the system bounded lifetime;
$\mathcal{S}_{\text{snd}}$: $\mathcal{S}$ of the data owner;

TABLE 3: Space Complexity Comparison

	System Parameter	Secret Key	Transformation Key	Ciphertext	Transformed Ciphertext
Wat11 [42]	$\mathcal{O}(\mathbb{U})$	$\mathcal{O}(\mathcal{S})$	N/A	$\mathcal{O}(\mathbb{A})$	N/A
GHW11 [21]	$\mathcal{O}(1)$	$\mathcal{O}(\mathcal{S})$	$\mathcal{O}(\mathcal{S})$	$\mathcal{O}(\mathbb{A})$	$\mathcal{O}(1)$
RW13 [35]	$\mathcal{O}(1)$	$\mathcal{O}(\mathcal{S})$	N/A	$\mathcal{O}(\mathbb{A})$	N/A
LDGW13 [23]	$\mathcal{O}(\mathbb{U})$	$\mathcal{O}(\mathcal{S})$	$\mathcal{O}(\mathcal{S})$	$\mathcal{O}(\mathbb{A})$	$\mathcal{O}(1)$
QDLL15 [33]	$\mathcal{O}(\mathcal{I})$	$\mathcal{O}(1)$	$\mathcal{O}(\log \mathcal{N})$	$\mathcal{O}(1)$	$\mathcal{O}(1)$
QDLM15 [34]	$\mathcal{O}(\mathbb{U})$	$\mathcal{O}(1)$	$\mathcal{O}(\mathcal{S})$	$\mathcal{O}(\mathbb{A})$	$\mathcal{O}(1)$
CDLQ16 [14]	$\mathcal{O}(1)$	$\mathcal{O}(1)$	$\mathcal{O}(\mathcal{S} \cdot \log \mathcal{N})$	$\mathcal{O}(\mathbb{A})$	$\mathcal{O}(1)$
AC17 [2]	$\mathcal{O}(1)$	$\mathcal{O}(\mathcal{S})$	N/A	$\mathcal{O}(\mathbb{A})$	N/A
NCD+18 [29]	$\mathcal{O}(1)$	$\mathcal{O}(1)$	$\mathcal{O}(\mathcal{S})$	$\mathcal{O}(\mathbb{A})$	$\mathcal{O}(1)$
LWZH20 [26]	$\mathcal{O}(\mathbb{U})$	$\mathcal{O}(\mathcal{S})$	$\mathcal{O}(\mathcal{S})$	$\mathcal{O}(\mathbb{A})$	$\mathcal{O}(1)$
XNH+21 [44]	$\mathcal{O}(\mathcal{T})$	$\mathcal{O}(1)$	$\mathcal{O}(\mathcal{S}_{\text{rec}} \cdot \log \mathcal{N})$	$\mathcal{O}(\mathcal{S}_{\text{snd}} + \mathbb{A}_{\text{rec}})$	$\mathcal{O}(1)$
XNM+21 [45]	$\mathcal{O}(1)$	$\mathcal{O}(1)$	$\mathcal{O}(\mathcal{S}_{\text{rec}})$	$\mathcal{O}(\mathcal{S}_{\text{snd}} + \mathbb{A}_{\text{rec}})$	$\mathcal{O}(1)$
XLD+22 [43]	$\mathcal{O}(1)$	$\mathcal{O}(\mathcal{S}_{\text{rec}})$	N/A	$\mathcal{O}(\mathcal{S}_{\text{snd}} + \mathbb{A}_{\text{rec}})$	N/A
Our SA-ABE	$\mathcal{O}(1)$	$\mathcal{O}(1)$	$\mathcal{O}(\mathcal{S})$	$\mathcal{O}(\mathbb{A})$	$\mathcal{O}(1)$
Our VO-ABE	$\mathcal{O}(1)$	$\mathcal{O}(1)$	$\mathcal{O}(\mathcal{S})$	$\mathcal{O}(\mathbb{A})$	$\mathcal{O}(1)$

对于计算复杂度，我们的VO-ABE方案享有与最先进的SA-ABE相当的结果，并且与最先进的VO-ABE相比具有上级性能。

空间复杂度的结果与计算复杂度的性能相似，其中我们的解决方案具有与最先进的SA-ABE相当的性能，并且具有优于最先进的VO-ABE的性能。

- 我们的解决方案可以与所有现有的SA-ABE和VO-ABE相媲美。然而，所有现有的SA-ABE和VO-ABE解决方案都是选择性安全的，并且选择性安全的解决方案通常比自适应安全的解决方案更快
- 为了展示可验证外包解密的优势，并公平地评估我们与现有最先进解决方案相比的代价，我们在下一节中给出了自适应安全ABE和我们之间的实验分析。

效率分析

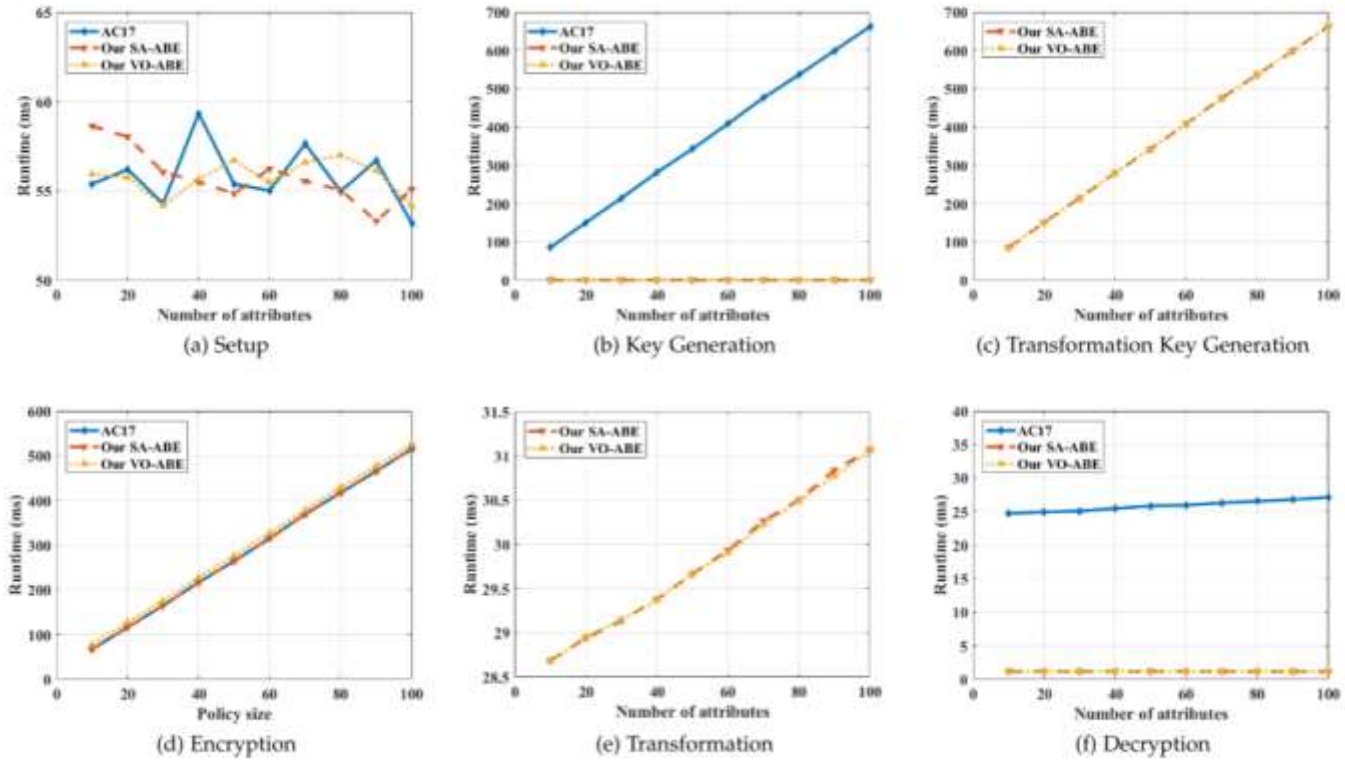


Fig. 6: Experimental Performance of AC17 and Ours

为了模拟资源受限的设备

- 实验模拟（数据解密除外）在PC端实现
- 数据解密在Android设备上进行（在现实世界的场景中，很多情况下数据的加密和解密需要在移动设备上进行，比如在移动应用程序中对加密数据进行解密）
- 每种情况下，已经测试了100次，我们计算的平均值作为实验数据的结果。

我们的解决方案在设置和加密方面可以与最先进的ABE(AC17)相媲美，并且解密速度更快。因此，我们的解决方案比最先进的解决方案更实用，适用于资源受限的终端设备的数据共享系统。

04

总　　结

总结

成果：这篇论文探讨了数据共享系统的实际应用和安全问题，并提出了一种适应资源受限终端设备的自适应安全数据共享系统。

- 我们开发了一种可扩展的密钥分发模型，无需安全通道即可完成密钥分发，简化了密钥分发过程，减少了不同实体之间建立安全通信渠道所需的开销，同时做到了自适应安全，精细化的数据共享。
- 我们提出的解决方案适用于各种基于云的应用程序，提供了更好的数据访问控制、轻量级解密和适应性安全保障。

思考：现阶段只实现私有可验证性，是否能够实现公开可验证性（有助于提高系统的安全性和信任度）

可以考虑采用数字签名（使用私钥对数据进行签名后，任何人都可以使用公钥来验证数据的真实性）零知识证明（允许证明者向验证者证明某个陈述的真实性，同时不透露除了陈述为真之外的任何其他信息）

恳请各位老师批评指正！