# Provably Secure Identity based Provable Data Possession

Yong Yu, Yafang Zhang
University of Electronic Science and Technology of China

Yi Mu, Willy Susilo
University of Wollongong, Australia

# Outline

- Cloud data integrity
- Basic idea of cloud data auditing
- Flaws of an ID-based auditing protocol
- Generic construction of ID-based auditing protocol
- A new construction of ID-based auditing protocol with zero-knowledge privacy
- Conclusion

# 1 Cloud data integrity

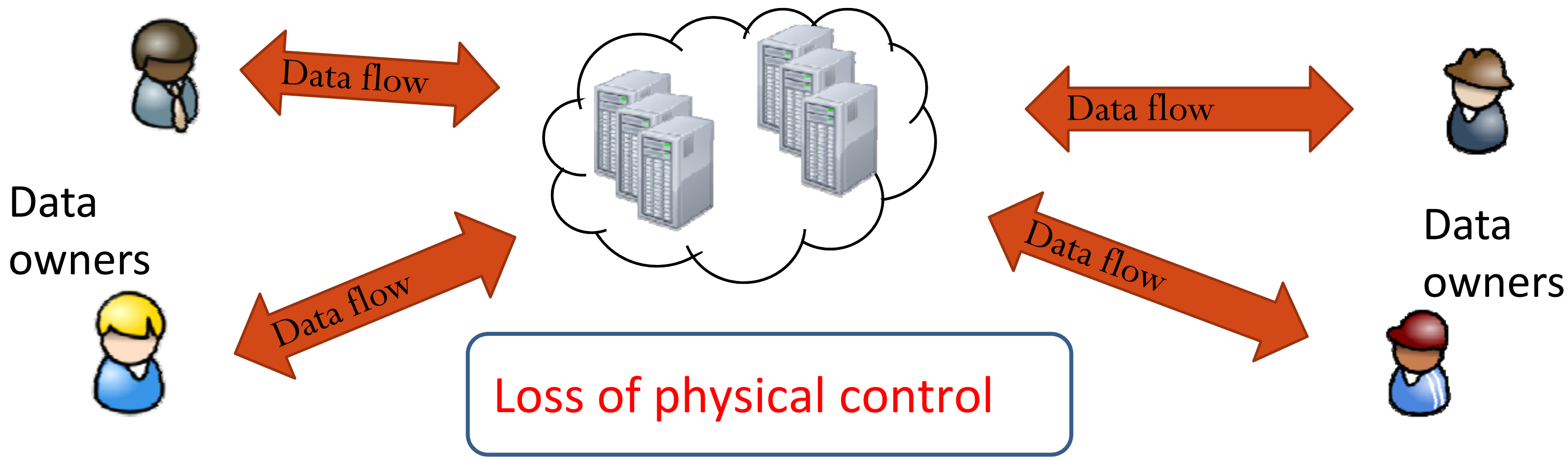
## Cloud Computing: Advantages

- Cloud computing enjoys a "pay-per-use model for enabling available, convenient and on-demand network access to a shared pool of configurable computing resources (e.g., networks, servers, storage, applications and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction." – NIST

# Cloud Storage vs. Data Integrity


- Cloud storage service allows owners to outsource their data to cloud servers for storage and maintenance.
  - Low capital costs on hardware and software, low management and maintenance overheads, universal on-demand data access, etc
  - E.g., Amazon S3.

# Cloud Storage vs. Data Integrity




- However, data outsourcing also eliminates owners' ultimate control over their data.
- The cloud server is not fully trusted.
  - Try to hide data loss incidents in order to maintain their reputation.
  - Might discard the data that have not been or are rarely accessed for monetary reasons.

# Data Integrity Accidents

cloud security alliance℠

Top Threats
to
Cloud Computing V1.0

Prepared by the
Cloud Security Alliance
March 2010

- **Insure Interfaces &APIs**
- **Data Loss & Leakage**
- **Hardware Failure**

**64%!**

BUSINESS INSIDER AUSTRALIA

Tech | Money & Markets | Briefing | Ideas | Executive Life | Video

TECH

**Amazon's Cloud Crash Disaster Permanently Destroyed Many Customers' Data**

HENRY BLODGET | APR 28 2011, 9:10 PM | 2

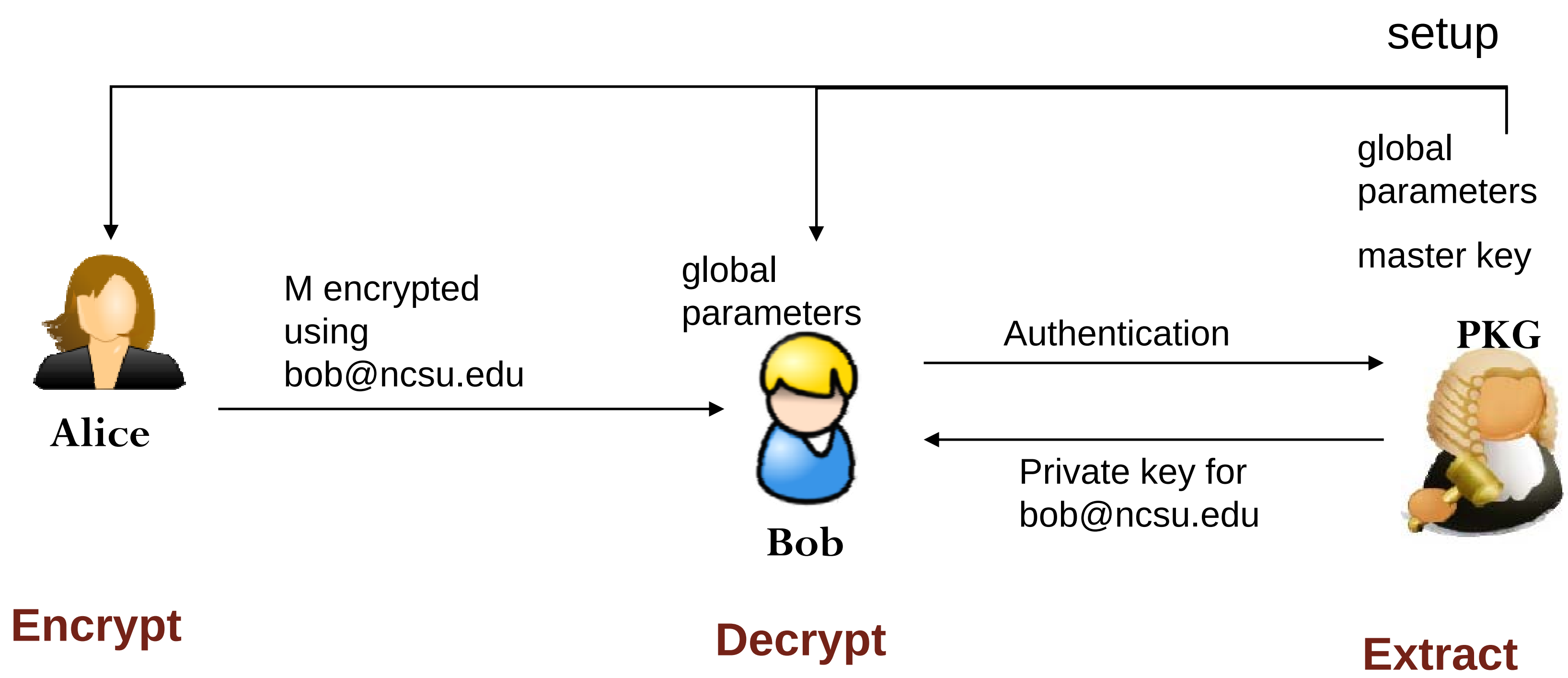
- **Amazon's Huge EC2 Cloud Service Crash**

# Remote Data Integrity Checking

- Trivial Schemes
  - Check data upon retrieval
  - Ask the storage server to MAC the entire file
  - Ask the cloud server to send a subset of randomly-picked file blocks along with their MACs

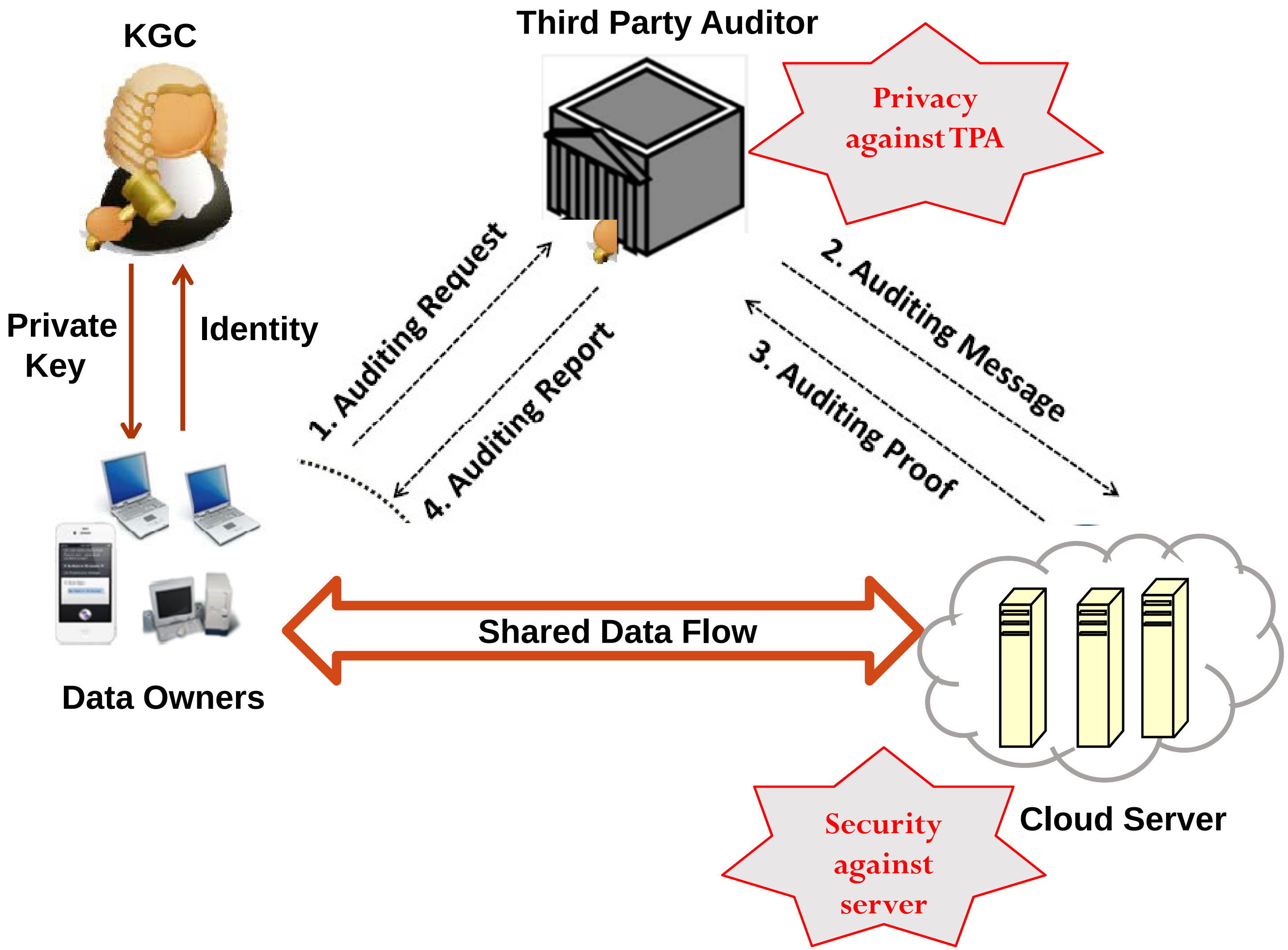
# 3 ID-based Cloud Auditing

## ID-based Cryptography




**Simplify Key Management**

# ID-based PDP



KGC
Third Party Auditor
Privacy against TPA
Private Key
Identity
1. Auditing Request
2. Auditing Message
3. Auditing Proof
4. Auditing Report
Shared Data Flow
Data Owners
Cloud Server
Security against server

# Wang et al.'s ID-based PDP

Setup: PKG's secret key $x \in Z_q^*$, public key $y = g^x$.

PPs: (G1,G2,q,g,H,h,h1,f,\pi,y);

Extract: $sk_{ID} = (R, \tau)$ $R = g^r, \tau = r + xH(ID, R) \bmod q.$

$$g^{\tau} = Ry^{H(ID,R)}.$$

TagGen: Compute $F_{ij} = h_1(\widehat{F}_{ij})$, $\sigma_i = (h(Ni, CSi, i)\prod_{j=1}^{s} u_j^{F_{ij}})^{\tau}.$

Challenge: (c,k1,k2)

ProofGen: $\sigma = \prod \sigma_i^{a_i}$, $F_j = \sum a_i F_{ij} (1 \le j \le s)$

Verify: $e(\sigma, g) \stackrel{?}{=} e(\prod_{i=1}^{c} h_i^{a_i} \prod_{j=1}^{s} u_j^{F'_j}, Ry^{H(ID,R)}).$

Huaqun Wang: Identity-Based Distributed Provable Data Possession in Multicloud Storage. IEEE T. Services Computing 8(2): 328-340 (2015)

# Comments on the Protocol

1 Soundness: $F_{ij} = h_1(\widehat{F}_{ij}),$

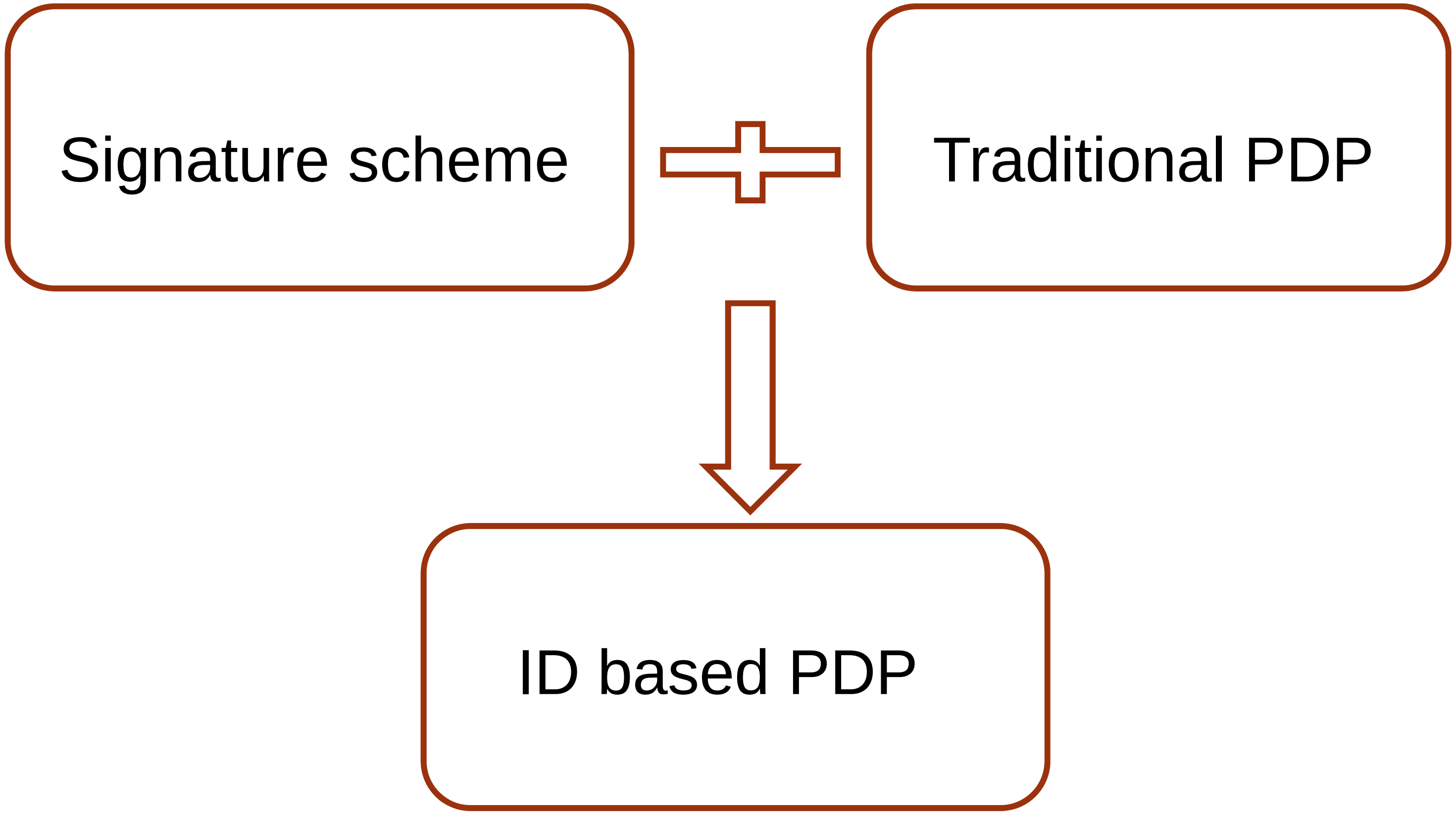
2 ID-based: $R$

3 Security model: Unforgeability

3. *Challenge: $\mathcal{C}$ generates a challenge chal which defines a ordered collection $\{ID^*, i_1, i_2, \ldots, i_c\}$, where $ID^* \notin S_1$, $\{i_1, i_2, \ldots, i_c\} \not\subseteq \mathbb{I}_1$, and $c$ is a positive integer. The adversary is required to provide the data possession proof for the blocks $F_{i_1}, \ldots, F_{i_c}$.*
4. *Second-Phase Queries: Similar to the First-Phase Queries. Let the* Extract *query identity set be $S_2$ and the* TagGen *query index set be $\mathbb{I}_2$. The restriction is that $\{i_1, i_2, \ldots, i_c\} \not\subseteq (\mathbb{I}_1 \cup \mathbb{I}_2)$ and $ID^* \notin (S_1 \cup S_2)$.*
5. *Forge: The adversary $\mathcal{A}$ responses $\theta$ for the challenge chal.*

Huaqun Wang: Identity-Based Distributed Provable Data Possession in Multicloud Storage. IEEE T. Services Computing 8(2): 328-340 (2015)

# Generic Construction of ID-based PDP




M. Bellare, C. Namprempre, G. Neven. Security proofs for identity-based identification and signature schemes, Eurocrypt 2004, LNCS 3027, 268-286, 2004.

ID-PDP.Setup $(1^k)$: $\mathrm{DS.Setup}(1^k) \rightarrow (sk, pk) \Rightarrow (msk, mpk)$

ID-PDP.Extract(ID,mpk,msk):

$\mathrm{PDP.KeyGen}(1^k) \rightarrow (\mathrm{pk,sk})$

$\mathrm{DS.Sign}(msk, id \,||\, pk) \rightarrow k_{ID}$

$(k_{ID}, pk, sk)$

ID-PDP.Store$(F, ID, mpk, k_{ID})$:

$(k_{ID}, pk, sk)$ $\mathrm{PDP.Store}(\mathrm{F,sk,pk}) \rightarrow \mathrm{F}^*$

ID-PDP.Proof(mpk,ID):

$\mathrm{DS.Verify}(mpk, id \,||\, pk, k_{ID}) = 1$

$\mathrm{PDP.Verify}(pk, id \,||\, pk, k_{ID}) = 1$

## ID-PDP.Proof(mpk,ID):

**Verifier** — **Cloud Server**

Cloud Server → Verifier: $(k_{ID}, pk)$

Verifier: $\mathrm{DS.Verify}(mpk, id \,\|\, pk, k_{ID})$

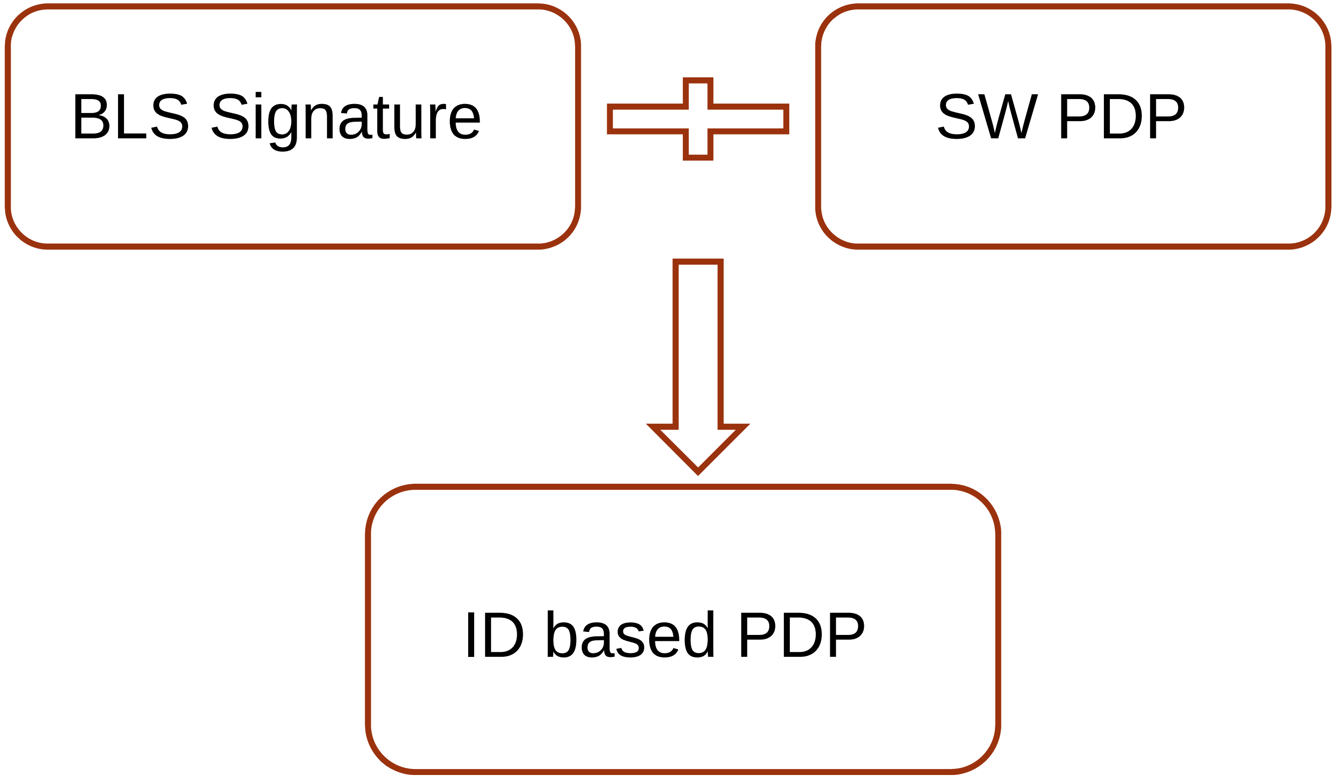
Verifier → Cloud Server: PDP.Challenge(pk)

Cloud Server: proof=PDP.Proof(pk,$F^*$,chal)

Cloud Server → Verifier: proof

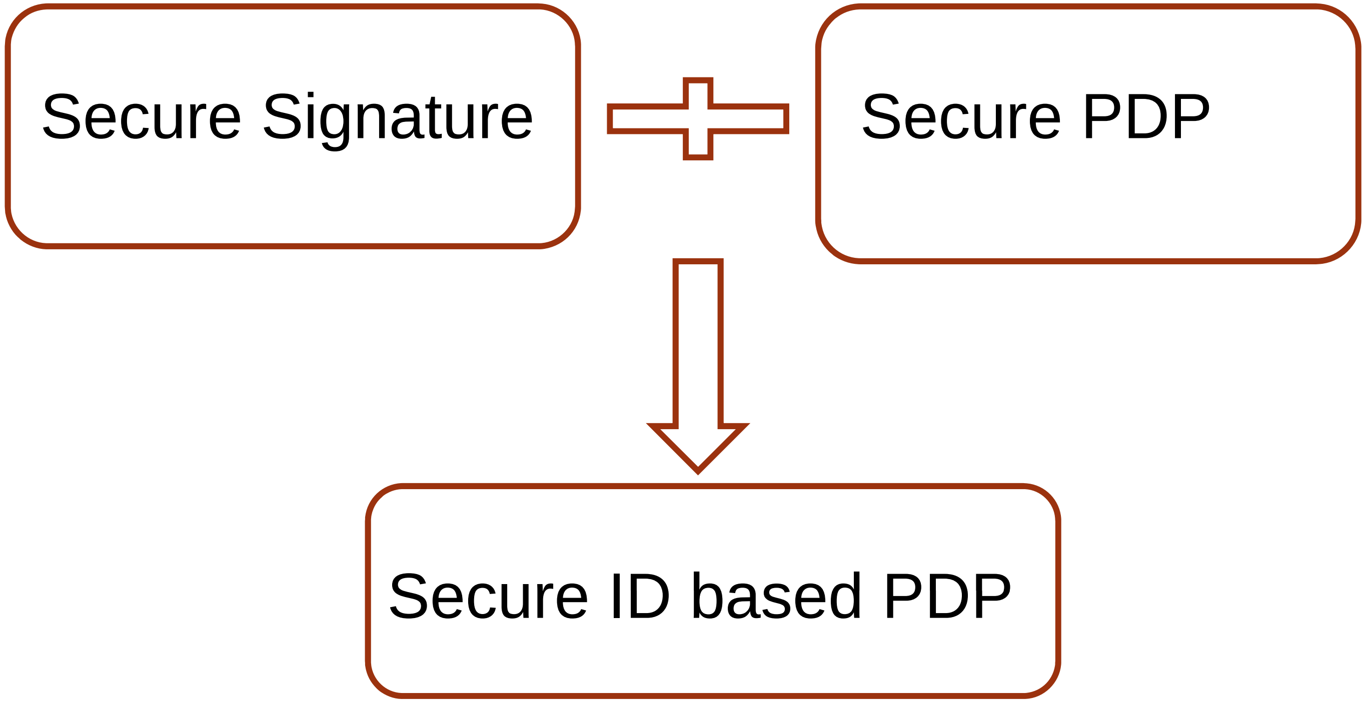
Verifier: PDP.Verify(pk,proof,chal)

# An instance



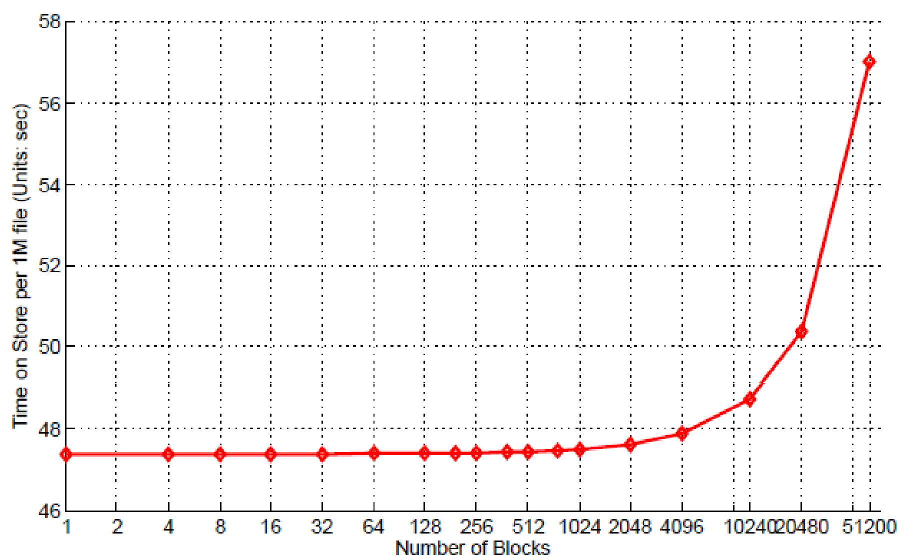

H. Shacham and B. Waters, Compact Proofs of Retrievability, Asiacrypt 2008, LNCS 5350, pp. 90-107, 2008.

# Security

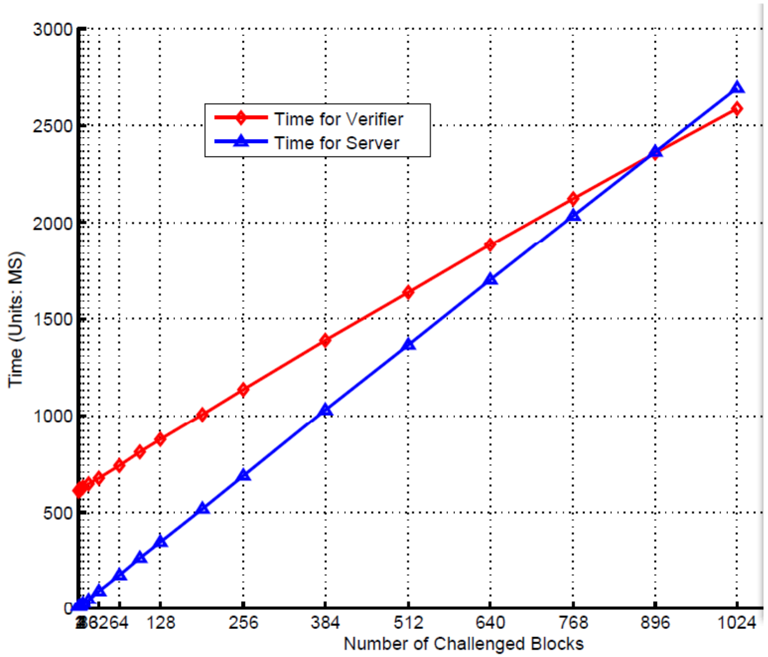



H. Shacham and B. Waters, Compact Proofs of Retrievability, Asiacrypt 2008, LNCS 5350, pp. 90-107, 2008.

# Evaluation




**Block size: 1k-4k**




**Time cost of prove protocol**

# A Novel Construction

# Basic Idea

**Key-Aggregate Cryptosystem**

**Asymmetric Group Key Agreement**

Key-Aggregate Cryptosystem for Scalable Data Sharing in Cloud Storage, Cheng-Kang Chu, S. M. Chow, Jianying Zhou, R. H. Deng et al. IEEE Trans. on Parallel and Distributed Systems, 25(2), 2014.

Qianhong Wu, Yi Mu, Willy Susilo, Bo Qin, Josep Domingo-Ferrer: Asymmetric Group Key Agreement. EUROCRYPT 2009: 153-170

Lei Zhang, Qianhong Wu, Bo Qin: Authenticated Asymmetric Group Key Agreement Protocol and Its Application. ICC 2010: 1-5

# Basic Tools

## Bilinear Pairing

$$e : G_1 \times G_1 \rightarrow G_2$$

Bilinearity | Non-Degeneracy | Efficient Computation

## Equality of Discrete Logarithm

$$POK\{(x) : Y_1 = g_1^x \wedge Y_2 = g_2^x\}$$

| Prover | | Verifier |
|---|---|---|
| $\rho \in Z_q, T_1 = g_1^{\rho}, T_2 = g_2^{\rho}$ | $(T_1, T_2)$ → | |
| | ← $c$ | $c \in \{0,1\}^{\lambda}$ |
| $z = \rho - cx \bmod q$ | $z$ → | |
| | | $T_1 = g_1^z Y_1^c \wedge T_2 = g_2^z Y_2^c$ |

# Our Construction

**Setup**

$$\alpha \in Z_q^*, P_{pub} = g^{\alpha}. \quad H_1, H_2 : \{0,1\}^* \to G_1, H_3 : G_2 \to \{0,1\}^l$$

**System Parameter**: $(G_1, G_2, e, g, P_{pub}, H_1, H_2, H_3, l)$

**Extract** $s = H_1(ID)^{\alpha}$

**TagGen** $M = m_1 m_2 \cdots m_n$

$$(1) \eta \in Z_q^*, r = g^{\eta}.$$

$$(2) \sigma_i = s^{m_i} H_2(fname \| i)^{\eta}.$$

**Upload**: $(M, r, \{\sigma_i\}, IDS(r \| fname))$

# Challenge-GenProof-CheckProof

| The Verifier | | Cloud Server |
|---|---|---|
| 1. Choose a challenge set $Q = \{(i, v_i)\}$;<br>2. Compute $c_1 = g^\rho, Z = e(H_1(ID), P_{pub}), c_2 = Z^\rho$;<br>3. Generate a knowledge proof $pf$:<br>$POK\{(\rho) : c_1 = g^\rho \ \wedge \ c_2 = Z^\rho\}$;<br>4. Generate a challenge<br>$chal = (c_1, c_2, Q, pf)$ | $\xrightarrow{chal}$ | 5. Verify $pf$;<br>6. Compute<br>$\mu = \sum_{i \in I} v_i m_i$,<br>$\sigma = \prod_{i \in I} \sigma_i^{v_i}$, |
| 7. Verify $\mathsf{IDS}(r\|\|fname)$;<br>8. Verify $m' \stackrel{?}{=} H_3(\prod_{i \in I} e(H_2(fname\|\|i)^{v_i}, r^\rho))$. | $\xleftarrow{m', r, \ \mathsf{IDS}(r\|\|fname)}$ | $m' = H_3(e(\sigma, c_1) \cdot c_2^{-\mu})$. |

# Security Proof Challenge

## Soundness

$$\mu = \sum_{i \in I} v_i m_i, \qquad \sigma = \prod_{i \in I} \sigma_i^{v_i}$$

**Knowledge of Exponent Assumption**:

Foy any adversary **A** that takes input $(N, g, g^s)$ and returns group elements(C,Y) such that $Y = C^s$ , there exists an "**extractor**" **B** which, given the same inputs as A, returns x such that $C = g^x$.

# Security Proof Challenge

## Challenge:

**There is no $\mu$ in our response, but**

$$(m', r, IDS(r \parallel fname))$$

## Solution:

**Generic Group Model**

Lower bounds for discrete logarithms and related problems, Eurocrypt '97, 256-266, 1997

# Zero-knowledge privacy

- Public parameters and the response are independent of the file stored except the name of the file.

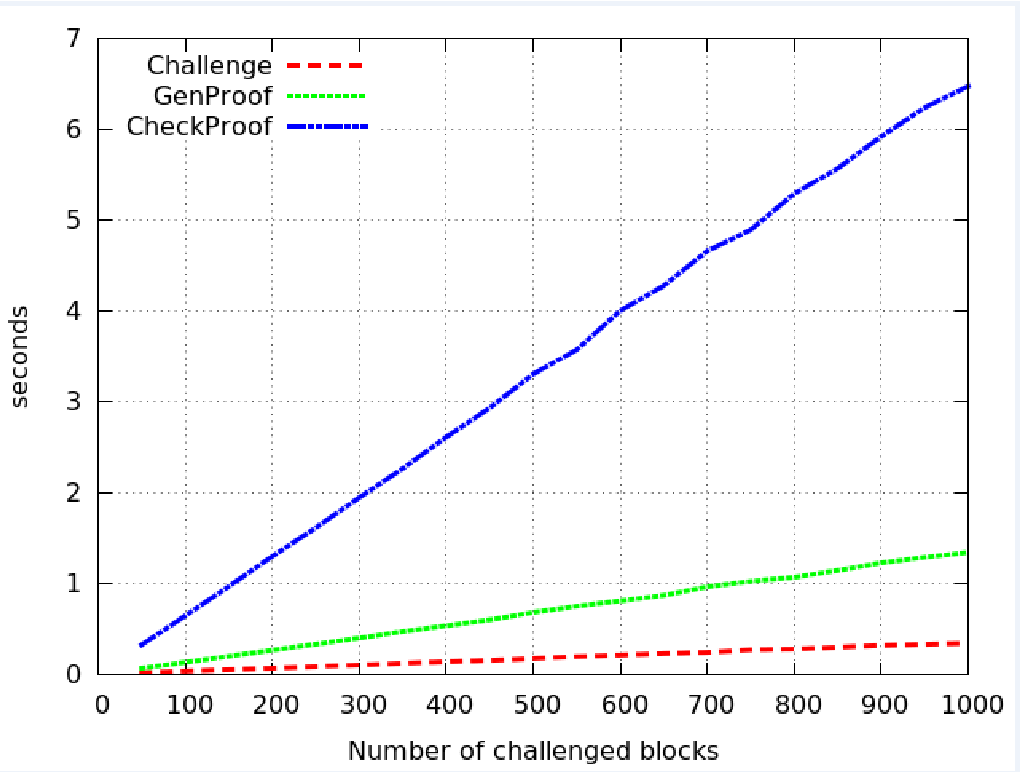
- (r,fname,IDS(r||fname),m') are not related to the content of the file.

# Implementation

| Setup | Extract | TagGen: off-line | TagGen: on-line | Challenge | GenProof | CheckProof |
|---|---|---|---|---|---|---|
| 4.8 ms | N/A | 241.9 second | 20.3 second | 351 ns per challenge | 1.3 ms per challenge | 6.6 ms per challenge |

TABLE I

SUMMERISE OF THE TIME COST FOR A 1 MB FILE




Increasing number of challenges for fixed size of file

# Conclusion

- Cloud data integrity Checking
- Flaws of an ID-based auditing protocol
- Generic construction of ID-based auditing protocol
- A new construction of ID-based auditing protocol with zero-knowledge privacy
- Soundness and zero-knowledge privacy models for ID-based auditing

# Thank YOU!

yuyong@uestc.edu.cn