RSAConference2019
San Francisco | March 4 – 8 | Moscone Center

BETTER.

SESSION ID: CRYP-F02

Secure Computation
-Context Hiding Multi-Key Homomorphic Authenticators

Lucas Schabhüser

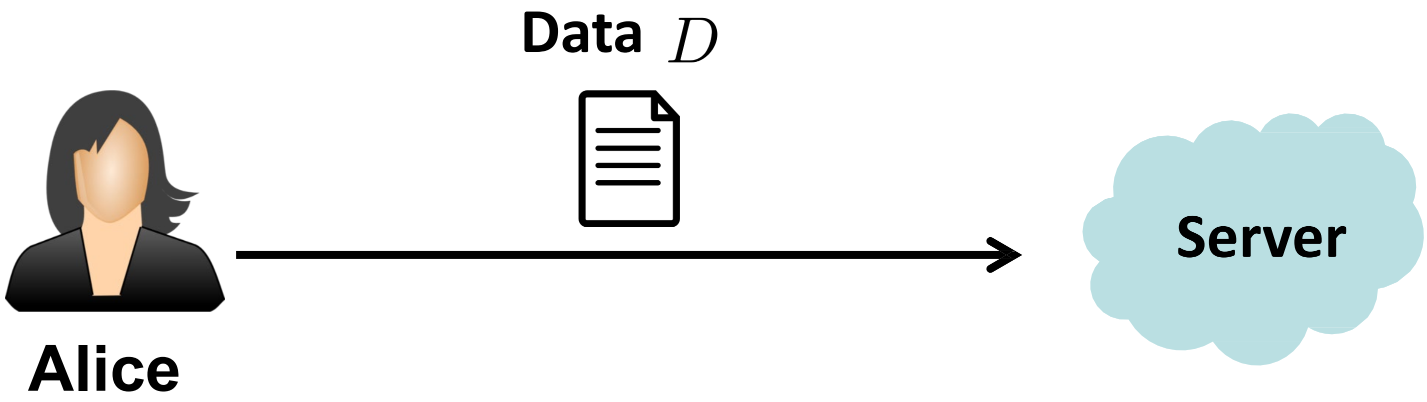
Research Assistant
TU Darmstadt

Organization

- Motivation
- Homomorphic Authenticators
- Input Privacy with respect to the Verifier
- Our Scheme
- Conclusion

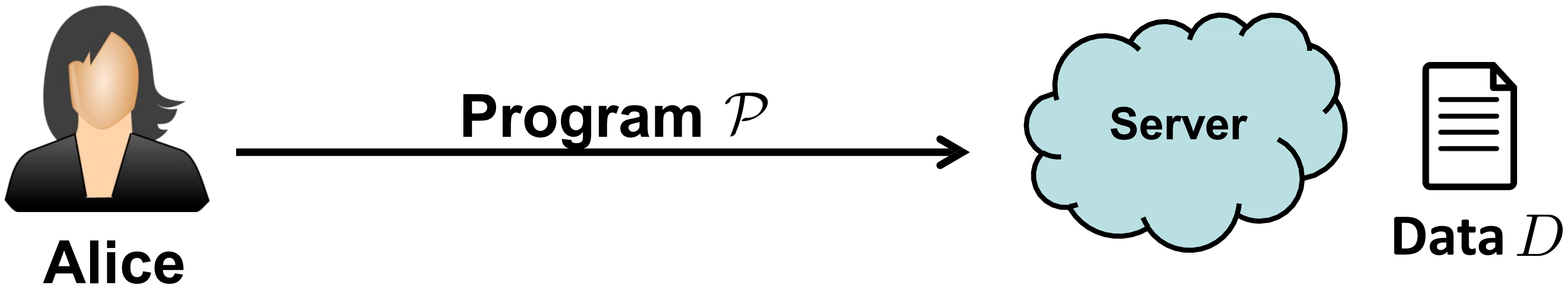
Motivation

Motivation

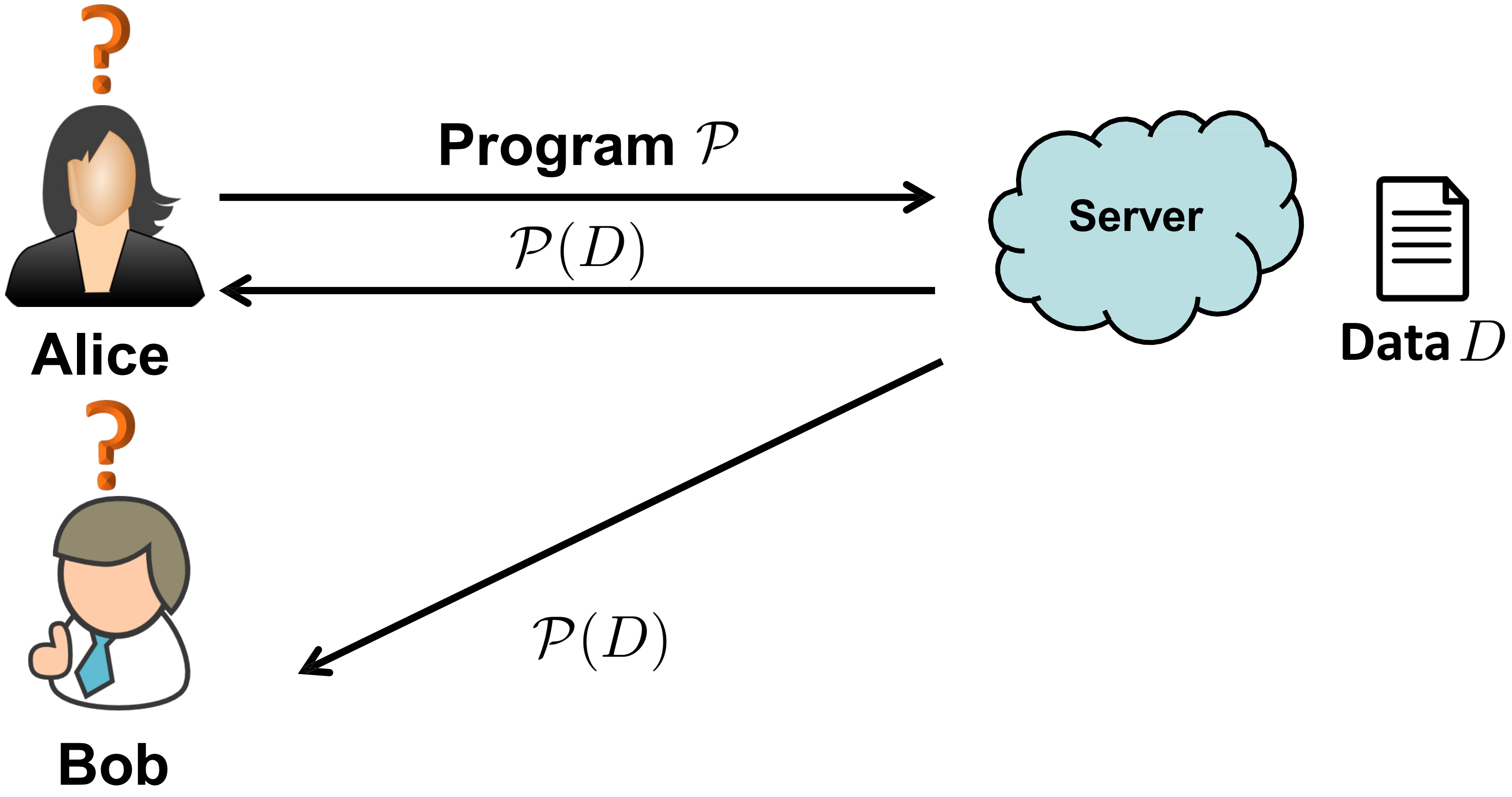


Motivation

Motivation

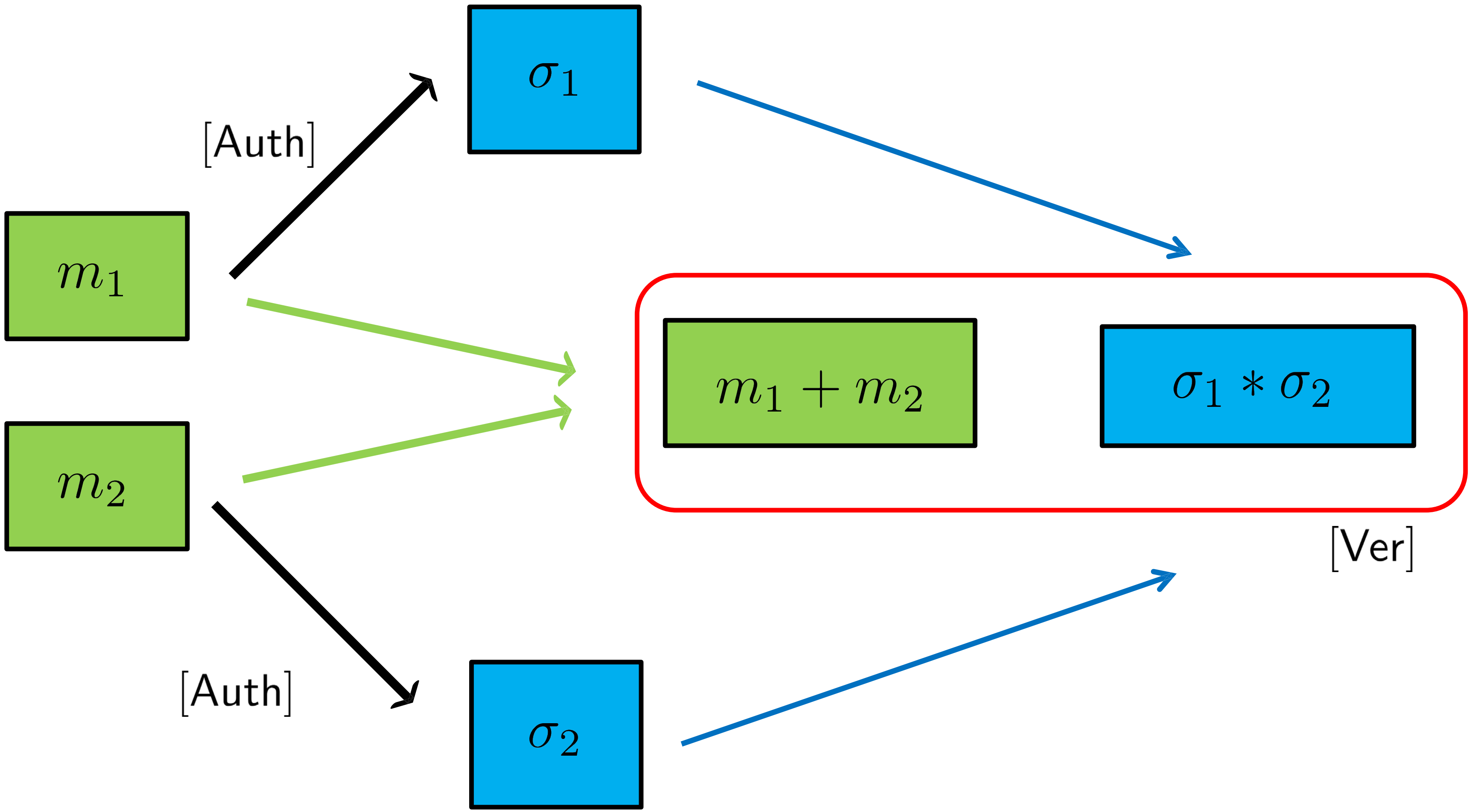


Motivation

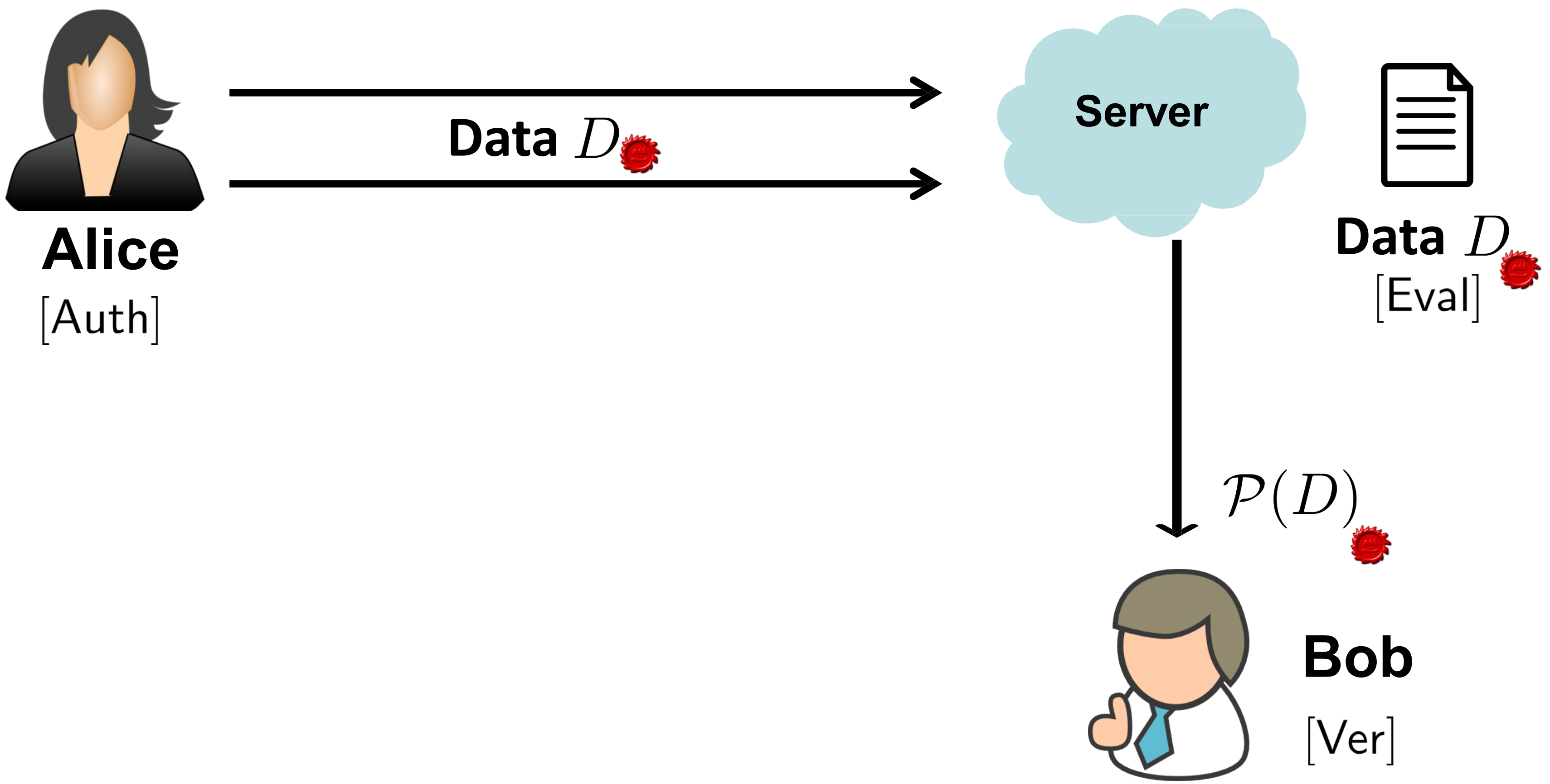


Program $\mathcal{P}$
$\mathcal{P}(D)$
Server
Data D
Alice
Bob
$\mathcal{P}(D)$

Homomorphic Authenticators - Intuition



Homomorphic Authenticators



Labeled programs

- Messages are stored in datasets identified by an identifier Δ
- Typically, the dataset size is fixed by a value n
- Functions can only be evaluated over messages in the same dataset

$$
\begin{array}{cccc}
 & \Delta_1 & \Delta_2 & \Delta_3 \\
l_1 & m_1 & m_1^* & m_1' \\
l_2 & m_2 & m_2^* & m_2' \\
l_3 & m_3 & m_3^* & m_3' \\
\dots & \dots & \dots & \dots \\
\dots & \dots & \dots & \dots \\
\dots & \dots & \dots & \dots \\
l_n & m_n & m_n^* & m_n'
\end{array}
$$

Homomorphic Authenticators

Setup : $\quad$ security parameter $\mapsto$ public parameters

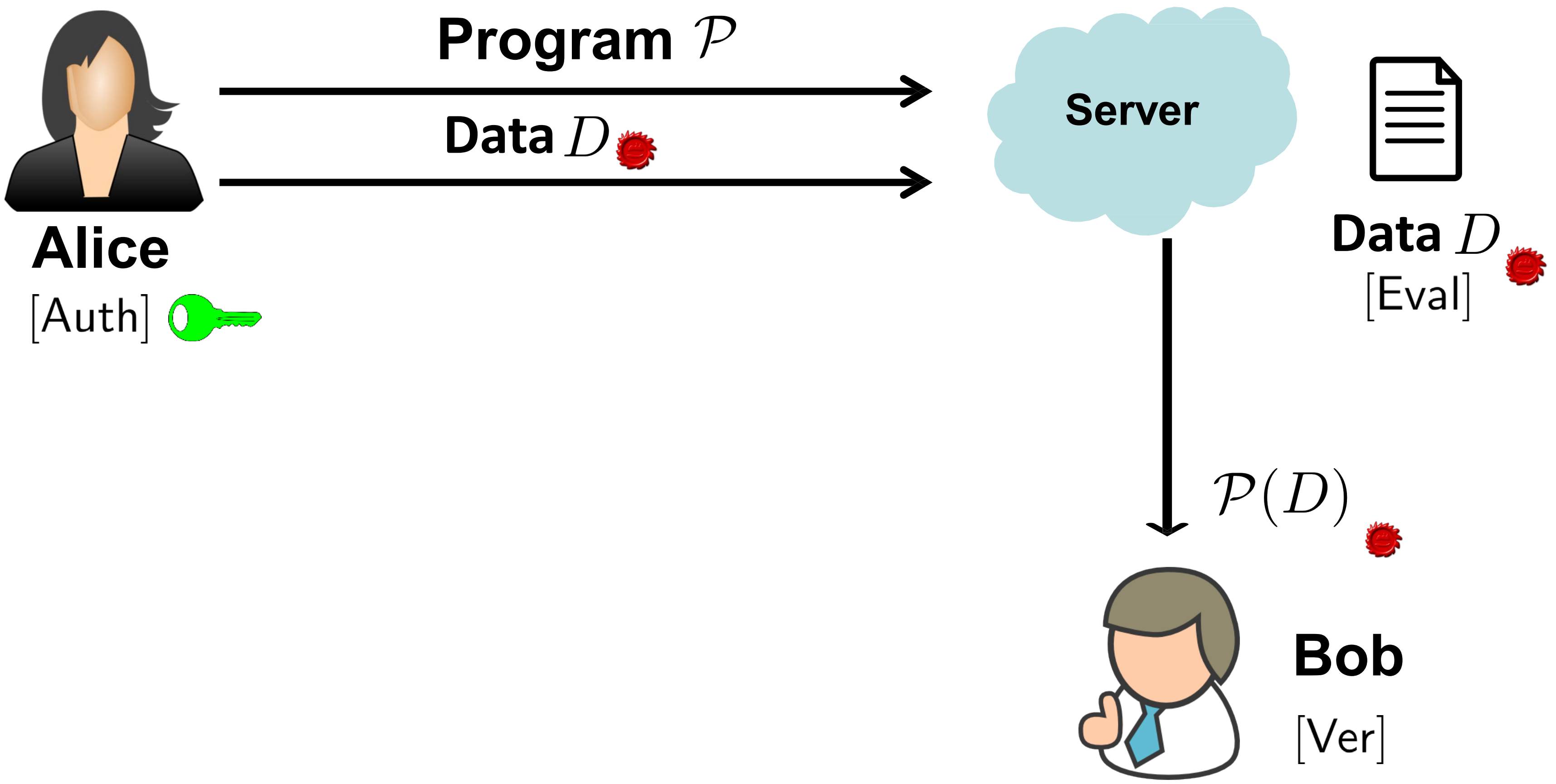
KeyGen : $\quad$ public parameters $\mapsto$ key triple $(\mathsf{sk}, \mathsf{ek}, \mathsf{vk})$

Auth : $\quad \begin{pmatrix} \text{secret key } \mathsf{sk}, \text{message } m, \\ \text{metadata } l = (\tau, \mathsf{ID}), \Delta \end{pmatrix} \mapsto \text{authenticator } \sigma$

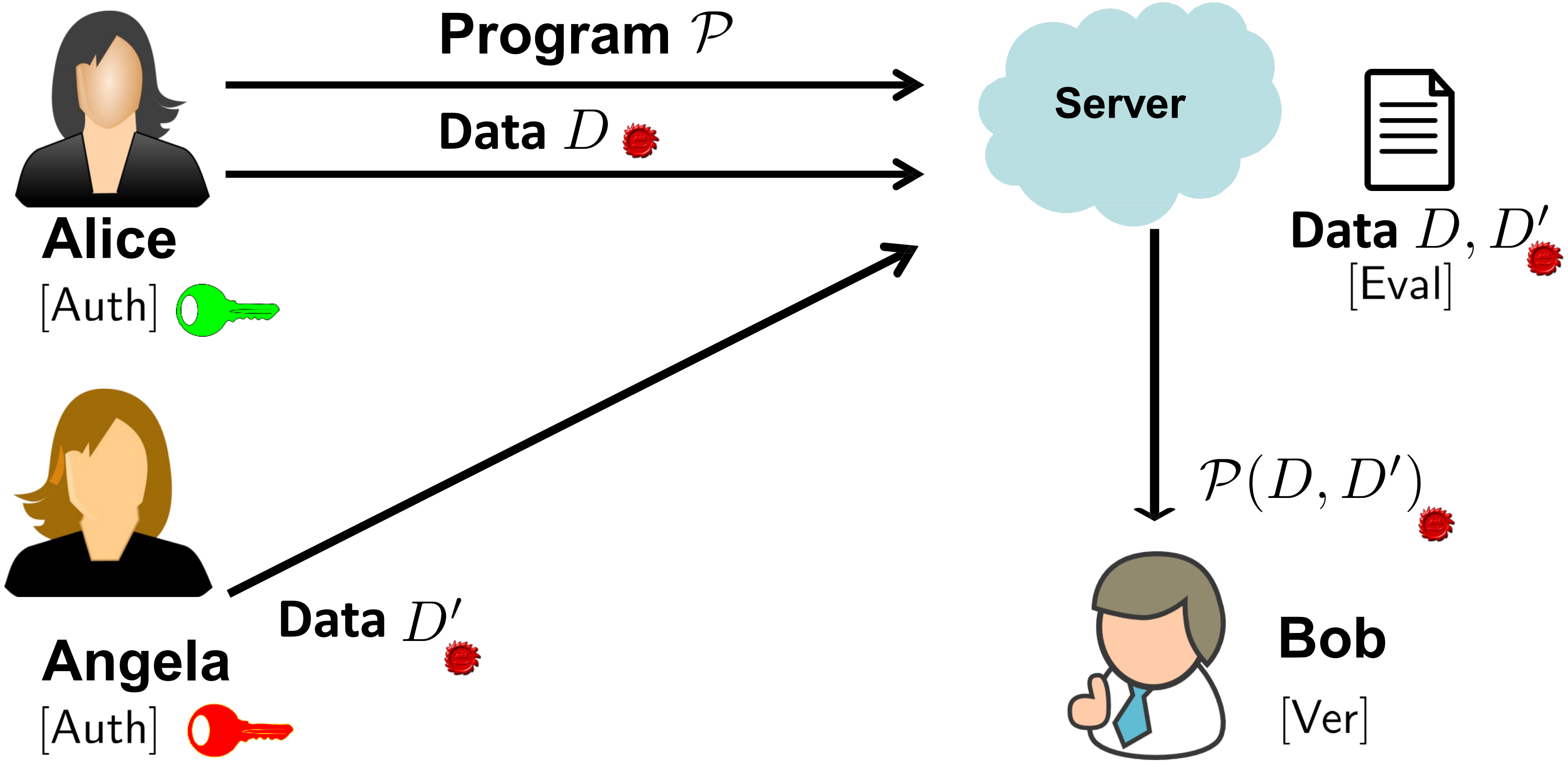
Eval : $\quad \begin{pmatrix} \text{public evaluation key } \mathsf{ek}, \text{program } \mathcal{P}_\Delta, \\ \text{authenticators } \sigma_1, \ldots, \sigma_n \end{pmatrix} \mapsto \text{authenticator } \sigma^*$

Ver : $\quad \begin{pmatrix} \text{verification key } \mathsf{vk}, \text{program } \mathcal{P}_\Delta, \\ \text{message } m, \text{authenticator } \sigma \end{pmatrix} \mapsto \text{accept/reject}$

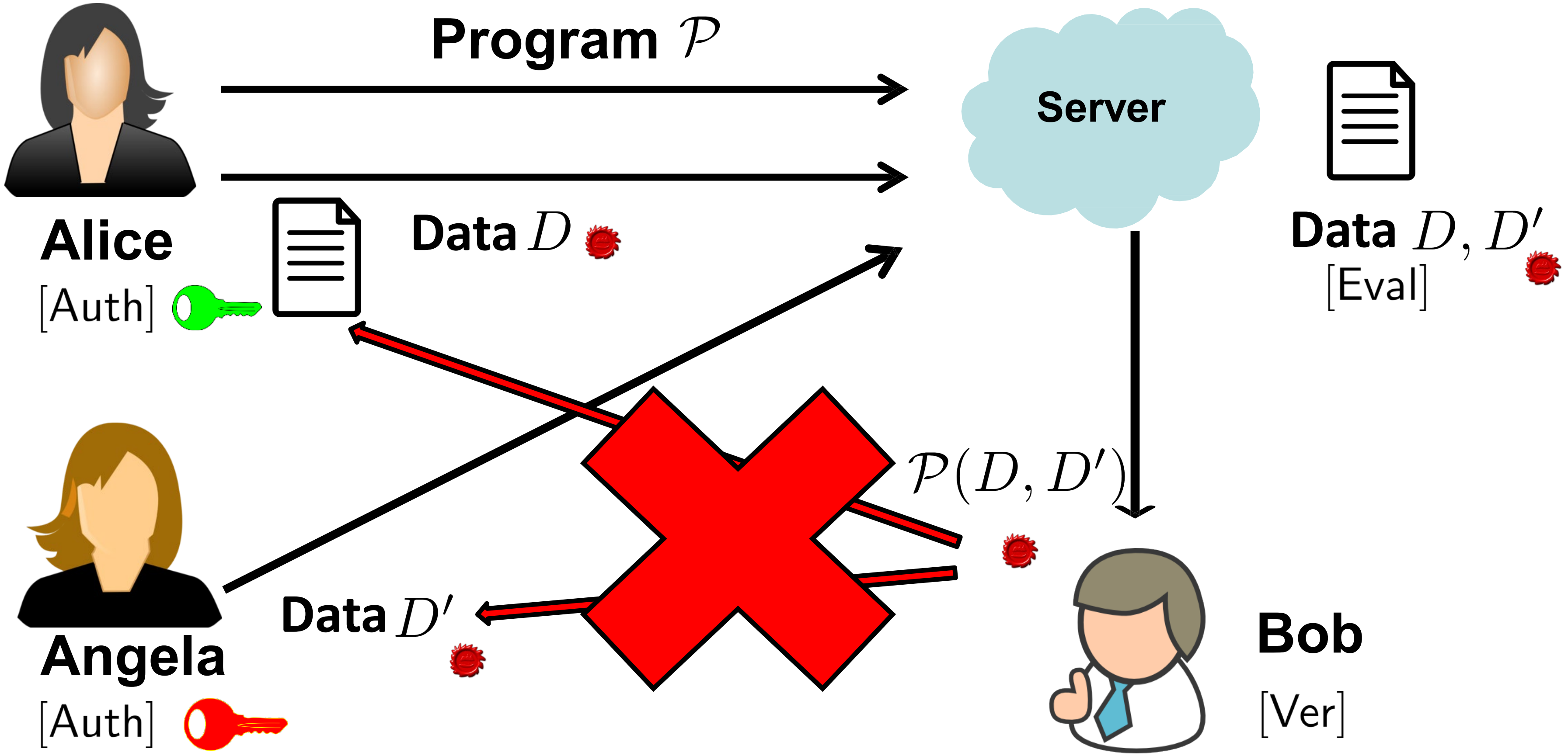
Homomorphic Authenticators



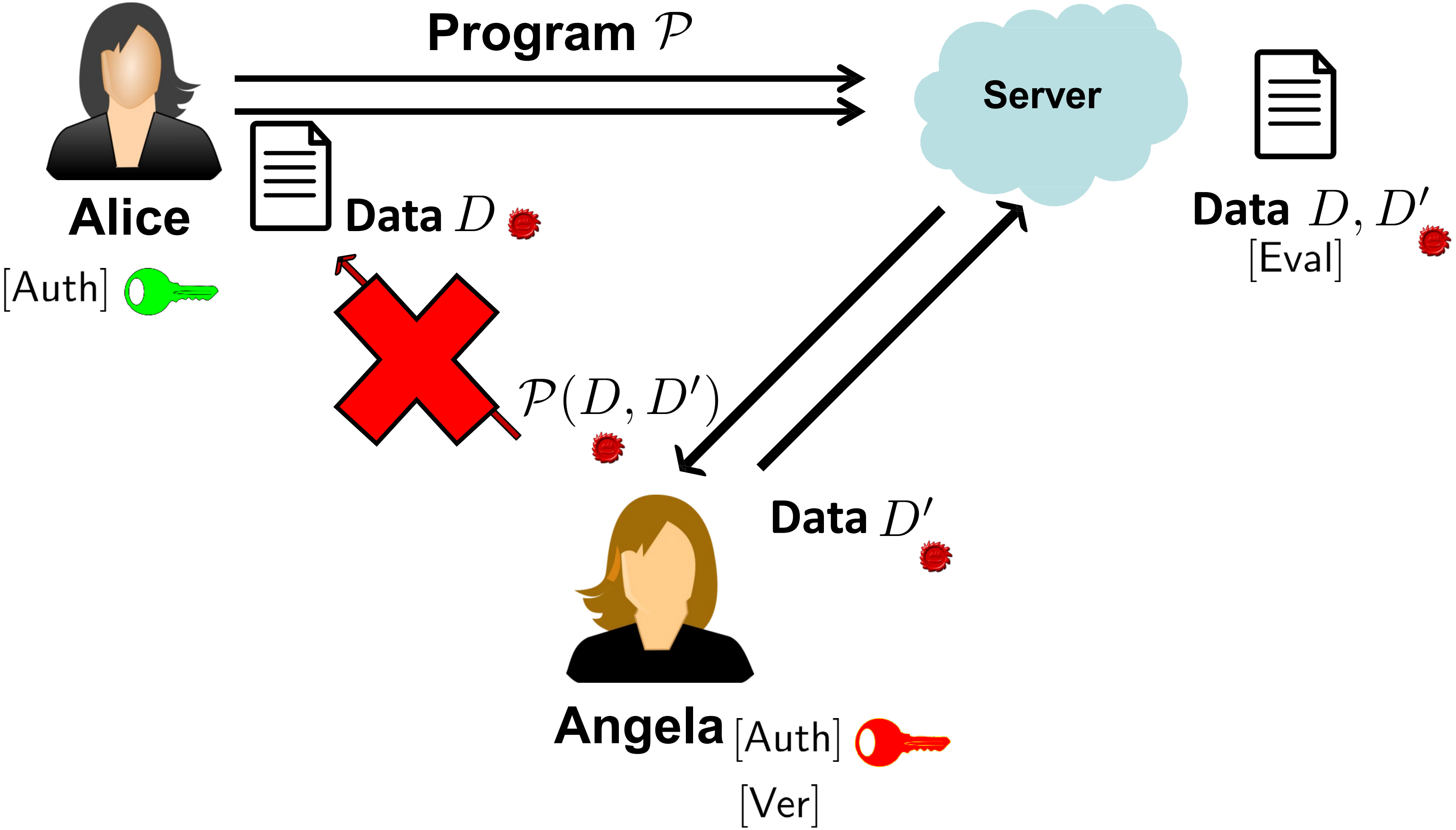
Multi- Key Homomorphic Authenticators



Input Privacy with Respect to the Verifier (external)



Input Privacy with Respect to the Verifier (internal)



Input Privacy with Respect to the Verifier (internal) - Intuition

Not always possible:

Example :

$$\mathrm{ID}_1 : m$$
$$\mathrm{ID}_2 : m'$$
$$m^* = m + m'$$

Example:

$$\mathrm{ID}_1 : m_1, \ldots, m_{365}$$
$$\mathrm{ID}_2 : m'_1, \ldots, m'_{365}$$
$$\mathrm{ID}_3 : \hat{m}_1, \ldots, \hat{m}_{365}$$
$$m^* = m_1 + \ldots + m_{365} + m'_1 + \ldots + m'_{365} + \hat{m}_1 + \ldots + \hat{m}_{365}$$

ID_2 can learn $m_1 + \ldots + m_{365} + \hat{m}_1 + \ldots + \hat{m}_{365}$

but not about individual $m_i, \hat{m}_i$

Our Solution

- A new multi-key linearly homomorphic signature scheme:
  - Supports linear functions
  - Unforgeable under DL, DDH and FDHI [CFN15] assumption
  - First multi-key homomorphic authenticator scheme to provide input privacy w.r.t. the verifier
    - both external and internal
    - even information theoretic input privacy

Our Solution – Comparison with State of the Art

	Functions	Privacy	Signature Size	Verification	Security
[ABBF10]	Linear	×	O(#ID)	O(#Inputs)	Pairings
[FMNP16]	Boolean Circuits	×	O(#ID)	O(#ID)	Lattices
[LTWC18]	Depends	Depends	Depends ≥O(#ID)	Depends ≥O(#Inputs)	SNARKs
This scheme	Linear	✓	O(#ID)	O(#ID)	Pairings

Our Solution - Efficiency

- Succinctness:
  - Authenticators size independent of number of inputs
  - Authenticators of size O(#ID)
- Efficient Verification
  - After a function-dependent preprocessing
    - Verification time independent of number of inputs
    - Verification in time O(#ID)

Our Solution – Signature Size

$e : \mathbb{G}_1 \times \mathbb{G}_2 \to \mathbb{G}_t$

$\mathsf{Auth}(\mathsf{sk}, \Delta, l_i, m_i) \mapsto \sigma_i$ (regular signature: 1, $\mathbb{G}_2$: 2, $\mathbb{G}_1$: 3)

$\mathsf{Eval}(\mathsf{ek}, \mathcal{P}_\Delta, \sigma_1, \ldots, \sigma_n) \mapsto \sigma^*$ (regular signature: #ID, $\mathbb{G}_2$: #ID+1, $\mathbb{G}_1$: 2#ID+1)

Our Solution – High Level Intuition

$\mathcal{P}$ = Aggregation: $m^* = m_1 + \ldots + m_{365} + m'_1 + \ldots + m'_{365} + \hat{m}_1 + \ldots + \hat{m}_{365}$

σ^* : 3 regular signatures and 3 $\mathbb{G}_2$ elements

2 $\mathbb{G}_1$ elements associated to ID_1	2 $\mathbb{G}_1$ elements associated to ID_2	2 $\mathbb{G}_1$ elements associated to ID_3
$m_1 + \ldots + m_{365}$ metadata $r_{\mathsf{ID}_1}, s_{\mathsf{ID}_1}$	$m'_1 + \ldots + m'_{365}$ metadata $r_{\mathsf{ID}_2}, s_{\mathsf{ID}_2}$	$\hat{m}_1 + \ldots + \hat{m}_{365}$ metadata $r_{\mathsf{ID}_3}, s_{\mathsf{ID}_3}$

Global $\mathbb{G}_1$ element	Global $\mathbb{G}_2$ element
$r_{\mathsf{ID}_1} + r_{\mathsf{ID}_2} + r_{\mathsf{ID}_3}$	$s_{\mathsf{ID}_1} + s_{\mathsf{ID}_2} + s_{\mathsf{ID}_3}$

Summary

- Introduced new notion of input privacy in the multi-key setting

- New scheme
  - First multi-key homomorphic authenticator scheme to achieve any type of input privacy w.r.t. the verifier
  - Input privacy even in an information theoretic sense
  - Amortized efficiency + succinctness

Open Problems

- Achieving a stronger form of succinctness
  - Size independent of **both** number of inputs and number of identities
- Achieving a stronger form of efficient verification
  - Verification time independent of **both** number of inputs and number of identities
- Multi-key homomorphic authenticators with input privacy beyond the linear case

"Secure Outsourcing in Practice"

- Be aware!
  - Threat of maliciously computed results
- Homomorphic Authenticators can help
  - (https://github.com/imdea-software/homomorphic-authentication-library)

Conclusion

Thank you for your attention!

Questions?