Computations on Encrypted Data for the Cloud

David Pointcheval
CNRS - ENS - INRIA

Secure Cloud Services and Storage Workshop
Oslo, Norway - September 10th, 2017

The Cloud

Dropbox

iCloud Drive

amazon cloud drive

Google Drive

OneDrive

Anything from Anywhere

One can store

- Documents to share
- Pictures to edit
- Databases to query

and access from everywhere

Security Requirements

As from a local hard drive/server, one expects

- **Storage** guarantees
- **Privacy** guarantees
  - **confidentiality** of the data
  - **anonymity** of the users
  - **obliviousness** of the queries/processing

How to proceed?

Confidentiality vs Sharing & Computations

Classical Encryption allows to protect data

- the provider stores them without knowing them
- nobody can access them either, except the owner

How to share the data?

How to compute on the data?

Broadcast Encryption

[Fiat-Naor - Crypto '94]

No computations!

The sender chooses a target set

Users get **all-or-nothing** about the data

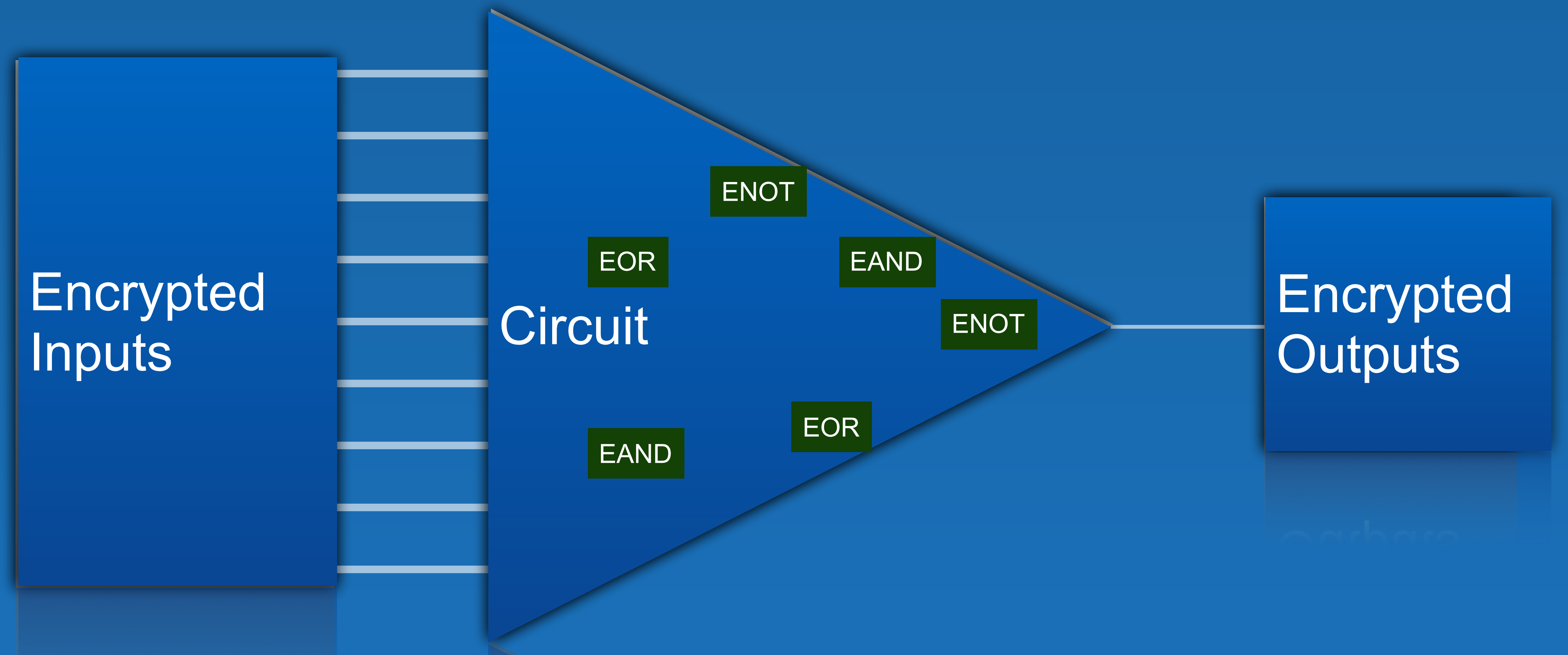
Fully Homomorphic Encryption

[Rivest-Adleman-Dertouzos - FOCS ’78]
[Gentry - STOC ’09]

FHE allows any computations on encrypted data

But the result is **encrypted** as the inputs!

No sharing!



Functional Encryption

[Boneh-Sahai-Waters - TCC '11]

Result in clear

The authority generates functional decryption keys DK_f according to functions f

- From $C = \textbf{Encrypt}(x)$, $\textbf{Decrypt}(DK_f, C)$ outputs $f(x)$
- This allows **controlled sharing of data**

Functional Encryption is Powerful

Functional Encryption allows access control:

- with $f_{\mathsf{id}}(x \,||\, y) = (\text{if } y = \text{id, then } x\text{, else } \perp)$: **identity-based** encryption
- with $f_{\mathsf{G}}(x \,||\, y) = (\text{if } y \in \text{G, then } x\text{, else } \perp)$: **broadcast** encryption

Functional Encryption allows computations:

- any function f: in theory, with iO (Indistinguishable Obfuscation)
- concrete functions: inner product

FE: Concrete Case

Student Name	English		CS		Math	
	Written	Spoken	Theory	Practice	Algebra	Analysis
Year 1						
Year 2						
Year 3						

Class	English	CS	Math
Year 1			
Year 2			
Year 3			

Class	English		CS		Math	
	Written	Spoken	Theory	Practice	Algebra	Analysis
Total						

Class	Total
Year 1	
Year 2	
Year 3	

Class	Total
3Years	

- For each student: transcript with all the grades
- Access to partial information for each student
- And even global grades for the class

FE: Inner Product

[Abdalla-Bourse-De Caro-P. - PKC ’15 - EPrint 2015/017]

Cells of derived tables are linear combinations of the grades from the main table:

$$c_i = \sum_j a_{i,j} b_j = \overrightarrow{a_i} \cdot \overrightarrow{b}$$

- $\overrightarrow{b}$: vector of the private grades, encrypted in the main table
- $\overrightarrow{a_i}$: vector of the public coefficients for the cell c_i, defines f_i
- With ElGamal encryption:
  - computations modulo p
  - if grades, coefficients, and classes small enough: DLog computation

FE: Limitations

Initial result: selective security [Abdalla-Bourse-De Caro-P. - PKC ’15 - EPrint 2015/017]

But improved to adaptive security [Agrawal-Libert-Stehlé - Crypto ’16 - EPrint 2015/608]

Anyway:

- one key limits to one function on any vector 🙂
- a malicious player could ask many functional keys
  - too many keys reveal the plaintexts… 😞
- a unique sender can encrypt a vector 😞
  - Multi-Input Functional Encryption (MIFE) 🙂

[Goldwasser-Gordon-Goyal-Jain-Katz-Liu-Sahai-Shi-Zhou - Eurocrypt ’14 - EPrint 2013/727 - EPrint 2013/774]

IP-FE: Concrete Security?

IP-FE: from c = **E**($\boldsymbol{x}$) and dk$_{\boldsymbol{y}}$, for n-vectors $\boldsymbol{x}$ and $\boldsymbol{y}$, one gets $\boldsymbol{x}.\boldsymbol{y}$

- n different keys reveal $\boldsymbol{x}$
- for the indistinguishability between two sets of vectors,
  the adversary is not allowed to ask keys that trivially tell them appart.
  ⇒ if n vectors in the sets, the adversary cannot ask any key!

IP-MIFE: from c_1 = **E**($\boldsymbol{x}_1$), …, c_n = **E**($\boldsymbol{x}_n$) and dk$_{\boldsymbol{y}}$, one gets $\boldsymbol{x}.\boldsymbol{y}$

- if no ordering: one immediately gets $n!$ linear relations on $\boldsymbol{x}$
- even with ordering, if public-key encryption: mix-and-match attack

IP-FE: Too Many Messages/Keys?

IP-FE with Helper:

[Dupont-P. - AsiaCCS '17]

- from $c = \mathbf{E}(\boldsymbol{x})$ and $\mathsf{dk}_{\boldsymbol{y}}$, for n-vectors $\boldsymbol{x}$ and $\boldsymbol{y}$, one must ask an helper
- the helper
  - learns as few as possible about the input
    (possibly the ciphertext, the function, the user, etc)
  - limits the number of answers (according to a bound on the inputs)
  - learns nothing about the output
- whereas there are additional interactions
  - no much leakage of information to the helper
  - more reasonable security model 🙂

IP-MIFE: Mix-and-Match Attacks?

IP-MCFE

[Chotard-Phan-P. - Work in progress]

Multi-Client Functional Encryption with Private Encryption:

- Senders have secret encryption keys ek_i to generate $c_i = \mathbf{E}(i,\lambda,\boldsymbol{x}_i)$ for a label λ
- From $c_1, \ldots, c_n$, for the same label λ, and $\mathsf{sk}_{\boldsymbol{y}}$, one gets $\boldsymbol{x}.\boldsymbol{y}$

- Multi-User Inputs
- Mix-and-match attacks avoided by private encryption
- More reasonable security model 🙂

FE: More Applications

The Graal in Privacy: Machine Learning on Encrypted Data

- One has access to a HUGE **encrypted** *labeled training data*
- Functional Encryption outputs the *prediction function* **in clear**

- No information leaked about the training data?
- No more than in the prediction function…
  but the latter may leak a lot about training data
  with model inversion attacks [Fredrickson-Lantz-Jha-Lin-Page-Ristenpart - Usenix Security ’14]
  even just from black-box prediction queries!

Conclusion

- **Functional Encryption**
  - Ideal functionalities on encrypted data
  - But unlimited access
- **In practice**
  - The ideal functionality leaks a lot!
  - Queries should remain under *some* control
  - Or answers should be noisy (differential privacy)