Collecting Data while Preserving Individuals' Privacy: A Case Study

YACC 2014

Brief history

2011 : A private company needs a crypto mechanism for anonymizing recorded data from a set of pharmacies.

Goal

Statistical use of these data

Applications

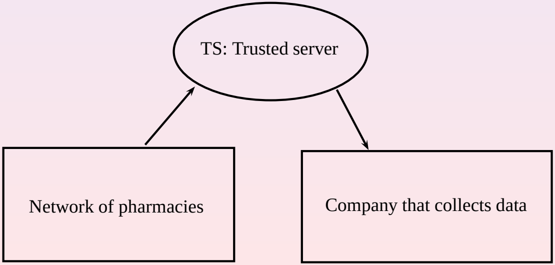
1. Detect outbreaks (influenza, ...)
2. Better understand buying behaviour of the patients
3. Give statistical views on diseases
4. ...

The problem

For the company Data $=$ Money if individual privacy is preserved

Requirements

1. Ensure the individual privacy in accordance with the legislation
2. TPH box in each pharmacy
3. No direct contact between boxes and the company
4. Detect if 2 transactions refer to the same patient



Practical solution

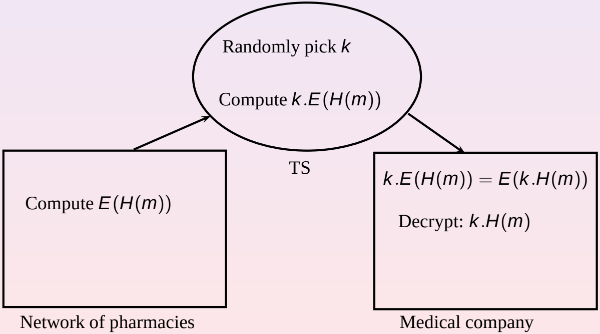
DATA = Header (identity of patients)
Body (medical data)

1. Hash the Header, Body remains the same
2. Drawback: **dictionary attack**
3. How to avoid a dictionary attack ?
4. Use a secret hash function?

Anonymization protocol

1. Anonymization of the pharmacies
   The set of boxes is a Tor network
2. Anonymization of the Header

Anonymization of the record.



Anonymization of the Header

Cryptographic elliptic curve Γ over a prime field $\mathbb{F}_p$.
n (prime number) the number of $\mathbb{F}_p$-rational points of Γ
G the cyclic group of order n of rational points on Γ
P a public generator of G
H a public map-to-point hash function

Setup

- TS picks at random $k < n$ and keeps it secret
- Company picks at random $a < n$ (private key of the company)
- Company computes $Q = aP$ (public key of the company) and transmits it to the network of pharmacies.

Anonymization of the Header

1. A box B draws at random k_1 between 0 and $n-1$. Then B computes

$$P_1 = k_1 P \quad P_2 = H(m) + k_1 Q.$$

   P_1 and P_2 are sent to TS.

2. TS computes, using its secret key k

$$R_1 = kP_1 \quad R_2 = kP_2$$

   and sends R_1 and R_2 to the company.

3. Company computes the anonymous number AN associated to the header

$$AN = (R_2 - aR_1)_x$$

   where $(R_2 - aR_1)_x$ denotes the x-coordinate of the point $R_2 - aR_1$.

Security issues

Privacy in regards to TS:

- identity of the pharmacies
- identity of the patients (Header)

Proposition

Under the assumption that DDH problem is hard on G, TS is not able to distinguish whether two encrypted headers represent the same plaintext header or not.

Proof: ElGamal is IND-CPA in the random oracle model

Security issues

Privacy in regards to the company:
Suppose an attacker knows some identities of clients of the pharmacies and the set of corresponding blinded headers.
Since the blinding value k is fixed, is he able to calculate k?

Generalized Discrete Logarithm of Order s (P_s)

$A = \{A_1, \ldots, A_s\}$ a (non ordered) set of rational points
$kA = \{kA_1, kA_2, \cdots, kA_s\}$.
The problem P_s on Γ_p is the following:
Given A and $A' = kA$, calculate k.

Remarks:

- Knowledge of A and $A' = kA$ is equivalent to knowledge of $B = \complement A$ and $B' = kB = \complement A'$.
  In particular, P_{n-1} is equivalent to P_1 (DLP).
- In our case study, $s << n$ and in practice, $500 \leq s \leq 10^6$.

P_s

Theorem

Suppose $\mathcal{A}(\Gamma_p, s)$ solves P_s in a time bounded by $T(s)$, then it is possible to construct an algorithm which solves DLP on Γ_p in a time bounded by $T(s) + st_0$ where t_0 is the time needed to choose an integer m and to calculate two scalar multiplications on Γ_p.

Proof:

- Let A_1, $A'_1 = kA_1$ be an instance of the DLP
- choose distinct m_i to construct the points $A_i = m_i A_1$ and $A'_i = m_i A'_1$
- We have $A'_i = m_i k A_1 = k m_i A_1 = kA_i$
- $A' := \{A'_1, A'_2, \cdots, A'_s\}$ and $A' = kA$ is an instance of P_s
- Applying $\mathcal{A}(\Gamma_p, s)$ to this instance of P_s, we can obtain k
- We have therefore solved DLP in a time bounded by $T(s) + st_0$

Sizes of parameters

if we had a practical algorithm to solve P_s, s being sufficiently small, then we could solve DLP over Γ_p.

Example

- Curve over $\mathbb{Z}/p\mathbb{Z}$ where p is around 256 bits, then from Weil's bound, the size of n is of order 2^{256} and the best known algorithms to solve DLP need about 2^{128} operations.
- If s is bounded by 10^6 (our case study), then s is negligible compared with 2^{128}.
- Thus, unless breaking the DLP for this size, we cannot obtain an algorithm to solve P_s with a number of operations significantly less than 2^{128}.

Conclusion

We solved a problem which has effectively been encountered in an industrial framework.

- Our protocol has many other possible applications
- Concept of generalized discrete logarithm problem of order s
- Protocol has been implemented in thousands of pharmacies and by students at Polytech