Online Social Networks

- Revolutionized the way people interact
- Hundreds of millions of users across the world
- Huge collection of personal information

- The lack of user privacy:
  - Users are not in control of their private data. The social network provider has full access to the user’s data.
  - Not enable a user to set fine-grained policies for access control
  - e.g. No policy can be defined for comments.
  - Network provider’s constantly changing and oblique privacy policies.

Contributions

- Design: a decentralized OSN architecture that:
  - Provides flexibility in data management through OOD;
  - Uses an appropriate and advanced cryptographic scheme
  - Supports efficient access revocation
  - Fine-grained policies on each piece of data:
  - Combines confidentiality, integrity, and availability by using the functionalities of a DHT.

- Prototype: We develop a prototype of DECENT (the wall and newsfeed functionalities)
  - and evaluate its performance through simulation and experiments on PlanetLab.

Requirements

- Functional Model:
  - To provide a flexible, general model of operations such as posting content and viewing, commenting on, we define
  - A container object that has two components:
    - The main content
    - A list of comments/annotations, represented as references to other container objects.

- Security Requirements:
  - Confidentiality: Content should be accessible to only those who are authorized.
  - Integrity: Content should remain authentic. Note that storage nodes are untrusted and may try to perform unauthorized updates to the stored data.
  - Availability: User content should remain available, even if the owner is offline, and despite potential malicious attempts to destroy the data.
  - Flexible Policies: Fine grained access e.g., “(friend AND co-worker) OR family”
  - Relationship Privacy: Relationships between users should remain hidden from third parties that may have no relationship with the object owner.

- Distributed Hash Table (DHT):
  - Objects in DECENT are stored in the DHT using the object ID as the key.
  - To ensure availability despite node churn and malicious attacks, several replicas of an object are maintained.
  - Write policy prevents malicious users from creating modifications that will be accepted by the readers, as they cannot produce a correct signature.
  - DECENT DHT supports an append request, which is used to add a comment reference to an existing object.

DECENT is a decentralized OSN, which employs a DHT to store and retrieve data objects created by their owners. Each object is encrypted to provide confidentiality.

Access Policies:
- Read policy: an attribute-based policy that describes the attribute combination required for a user to decrypt an object’s data.
- Write policy: an identity-based policy, which generally is set to the owner of the object.
- Append policy: describes who may add a comment/annotation to the object. It is also an attribute-based policy.

Cryptographic Protection:
- Each user becomes a key authority, issuing different encryption keys to social contacts who are authorized.
- Attribute-based encryption (ABE)
  - A public-key encryption scheme where each encrypted item is associated with a policy.
  - A key can decrypt an encrypted item if its set of attributes satisfies the item’s policy.
- Hybrid encryption mode: the message is encrypted with a randomly chosen symmetric encryption key.
- Supports immediate revocation by the use of the EASIER scheme [1].
- Two extensions on EASIER scheme:
  1. Threshold secret sharing can be used to split the proxy functionality among several randomly selected nodes;
  2. The same experiments on 15 PlanetLab nodes to get an idea of DECENT’s performance in a real deployment.

System Architecture

- DECENT is a decentralized OSN, which employs a DHT to store and retrieve data objects created by their owners. Each object is encrypted to provide confidentiality.
- Access Policies:
  - Read policy: an attribute-based policy that describes the attribute combination required for a user to decrypt an object’s data.
  - Write policy: an identity-based policy, which generally is set to the owner of the object.
  - Append policy: describes who may add a comment/annotation to the object. It is also an attribute-based policy.
- Cryptographic Protection:
  - Each user becomes a key authority, issuing different encryption keys to social contacts who are authorized.
  - Attribute-based encryption (ABE)
    - A public-key encryption scheme where each encrypted item is associated with a policy.
    - A key can decrypt an encrypted item if its set of attributes satisfies the item’s policy.
  - Hybrid encryption mode: the message is encrypted with a randomly chosen symmetric encryption key.
  - Supports immediate revocation by the use of the EASIER scheme [1].
  - Two extensions on EASIER scheme:
    1. Threshold secret sharing can be used to split the proxy functionality among several randomly selected nodes;
    2. The same experiments on 15 PlanetLab nodes to get an idea of DECENT’s performance in a real deployment.

Implementation and Evaluation

- Cryptographic schemes:
  - EASIER for ABE,
  - AES for symmetric encryption,
  - DSA for signatures,
  - RSA to encrypt the write policy signature key.
- The underlying DHT:
  - FreePastry with Euclidean network topology was used for simulation,
  - Kademlia [2], for the experiments on PlanetLab

Simulation

- The simulation was run on a peer-to-peer network of 10 000 nodes.
- Measure the performance of viewing a user’s newsfeed and wall with varying numbers of status messages, posts, and comments.

Experiments on PlanetLab

- The same experiments on 15 PlanetLab nodes to get an idea of DECENT’s performance in a real deployment.

Future Work

- Adding features to DECENT
- Improving performance and resilience through optimized cryptographic techniques, caching, and replication.

References