iSocial

Security, Privacy and Trust in DOSNs: State-Of-The-Art Approaches and Open Challenges

Panagiotis Ilia

Institute of Computer Science
Foundation for Research and Technology – Hellas (FORTH)

<pilia@ics.forth.gr>

FORTH Institute of Computer Science

MARIE CURIE ACTIONS PEOPLE

Online Social Networks

Facebook, MySpace, Google+, Flickr, Twitter, Tumblr, Orkut ...

- **Facebook**: **1b** active users in October 2012.
  1.11b in March 2013.
- **Google+**: **500m** registered users in May 2013 (launched in 2011).
  235m active users per month.
- **Twitter**: **500m** registered users (2012).
  340m "tweets" per day.
  1.6b search queries per day.

OSNs are **Web-based services**

Oriented on people and their interests (Human-centric)

- Connections are based on real-life relationships.
- Users generate and publish their content (posts, photos, chat)
- Users establish groups based on common interests

However…

Most **OSNs** follow the **Centralised Architecture**

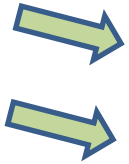
Security Issues:

- Untrusted service providers
- The servers of the providers are **information silos**
- Disclosure of user's personal information
  - To third parties for revenue by advertisement
  - By accident/by malicious users (hackers)
- **Censorship** over user's data

Decentralization is promising..

Benefits:

- Privacy of users – Personal Information
- Data ownership – Intellectual Property



Users:
Manage, store and share their data

Also

- High performance
- Fault tolerance
- High scalability (with low cost)

Security issues, objectives and open challenges in DOSNs

- **User Privacy** [1]
- **Authentication**

  Impersonation and **Defamation attacks**
  Profile Cloning and **Sybil attacks**.
- **Confidentiality**

  Man-In-The-Middle attacks (MITM)
  Controlled **Information sharing** of users' data [1]
- **Availability**

  Denial of Service (DoS) and **Black Hole Attacks**.
- **Spam and malware**

[1]: Presentation from the University of Insubria

- **Web-based decentralised OSNs**
  - Diaspora
  - Friend-of-a-Friend (FOAF)
- **Peer-to-Peer (P2P) OSNs**
  - Safebook
  - PeerSoN
  - Vis-à-Vis
  - DECENT

1. Diaspora

A network of independent **Diaspora servers** (pods).

- Users deploy their own Diaspora server -or- use existing servers.
- Sharing groups ("aspects") -- Communication via **posts** (**public-private**)
- Bi-directional connection -- User's profile is **replicated** on friend's server .
- ***"Push"*** design: New posts are **pushed** to friend's servers
- **HTTPS** - **Encrypted** and **authenticated** communication

1. Diaspora

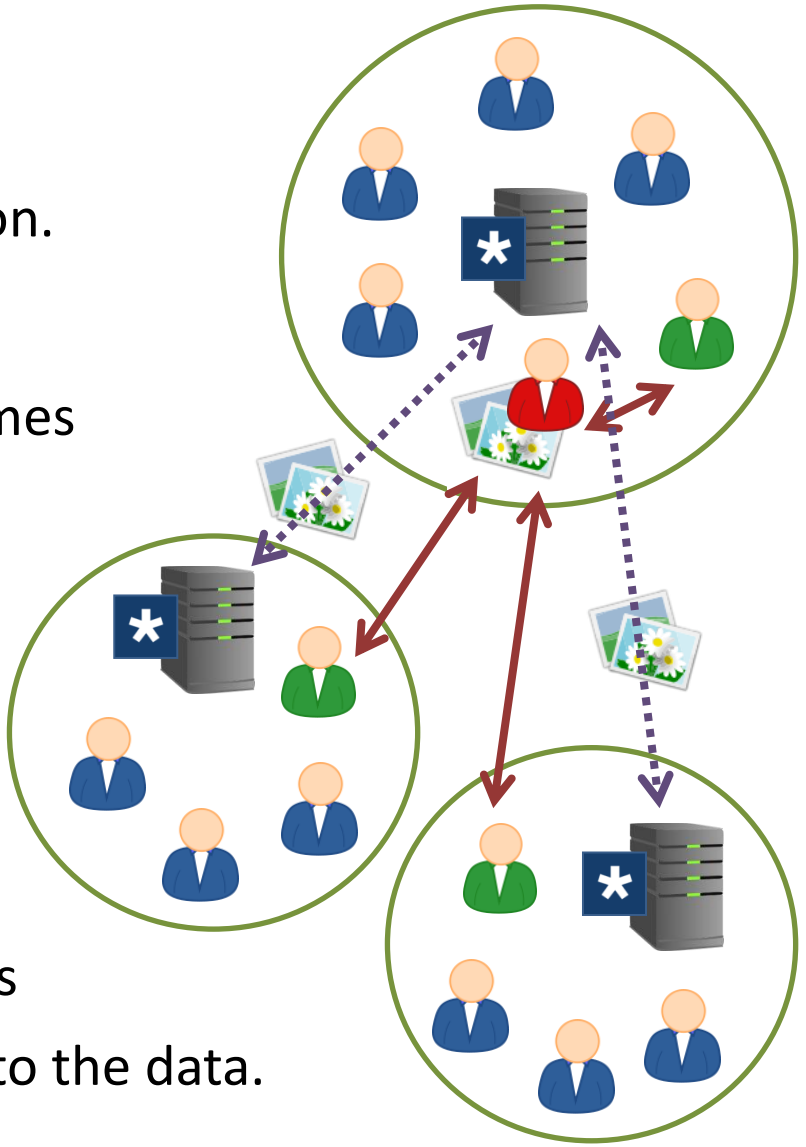
+ Encrypted and Authenticated communication.
+ Prevent the Man-In-The-Middle attack
+ Weak notion of anonymity by using usernames

- **Profile availability** is not preserved
- **Unique IDs (**and joining **Invitation)**

  but still vulnerable to

  Impersonation and Sybil attacks
- Data are stored **un-encrypted** on the servers

  The server administrator has access to the data.



2. Friend-of-a-Friend (FOAF)

- **User's Personal web-space** on a **trusted server.**
- Data: Friend-Of-A-Friend (FOAF) file -- Activity log -- Photo Albums.

- **FOAF file:** Metadata for **people, interests, relationships** and **activities**
- **"Web ID"** -- Friend's "Web IDs" are stored in the user's FOAF
- For accessing friend's data → visit **FOAF** to obtain the corresponding **URIs**

- The user (data owner) can define **fine-grained access control policies**

2. Friend-of-a-Friend (FOAF)

Authentication with :

The **OpenID** protocol -or- The **FOAF + SSL** certificates

+ Difficult to perform **Impersonation attacks** as users use their OpenID
+ Encrypted and authenticated communication through the “FOAF + SSL”

- User’s data are stored **unencrypted**.
- The **correctness** of the **FOAF** meta-data is not verified.
- The user’s **FOAF file** is available publicly.
- Users can obtain multiple IDs (**Sybil attack**).

- **Web-based decentralised OSNs**
  - Diaspora
  - Friend-of-a-Friend (FOAF)
- **Peer-to-Peer (P2P) OSNs**
  - Safebook
  - PeerSoN
  - Vis-à-Vis
  - DECENT

1. Safebook

Structured **peer-to-peer** architecture (p2p).

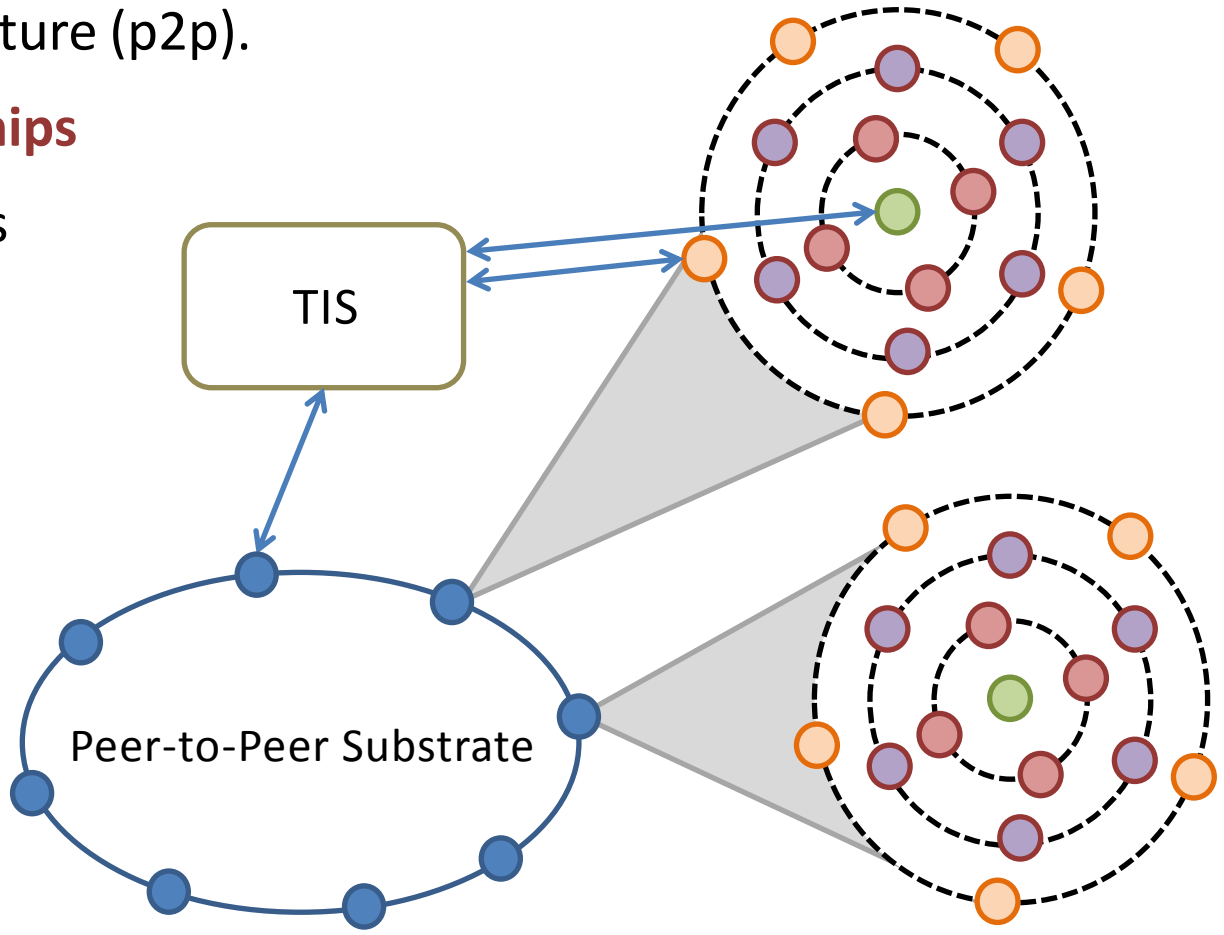
Leverages user's **trust relationships**

Multi-hop routing among friends

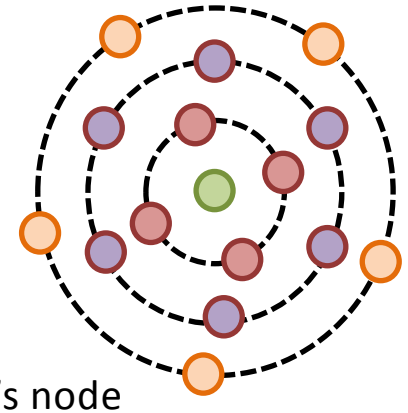
Matryoshkas

Peer-to-Peer substrate (DHT)

Trusted Identification Service



1. Safebook



Matryoshka (user-based view of the system)

User's **full profile** is replicated at the inner nodes.

Access to data → multi-hop through the Matryoshka

Peer-to-Peer Substrate (global view of the system)

All the nodes are organized in a DHT.

Outer nodes are registered as matryoshka's entry-points.

Trusted Identification Service

Provides ***unique*** and ***uncorrelated*** **identifiers** --- the respective Certificates

1. Safebook

- Data Encryption + Authentication → Public Key Cryptography (PKC)
- Access Control to profile attributes → Group-based encryption - respective keys

\+ **Anonymity** similar to "onion routing" - based on **social trust relationships**.

\+ Matryoshka structure - suitable for collaboration among the users

\+ Prevent **Impersonation** and **Sybil Attacks** (unique and unforgeable ID from TIS)

\- **Profile availability** is high but **not 24/7 guaranteed**

\- The level of **anonymity** depends on the spanning factor (less performance)

\- No mechanism for detecting **spam** and **malware distribution**.

\- **Man-In-The-Middle** and **Black Hole attacks** are very difficult but **feasible.**

2. PeerSoN

Overcoming Internet connectivity problems -- Preserving user's privacy

Two-tier architecture:

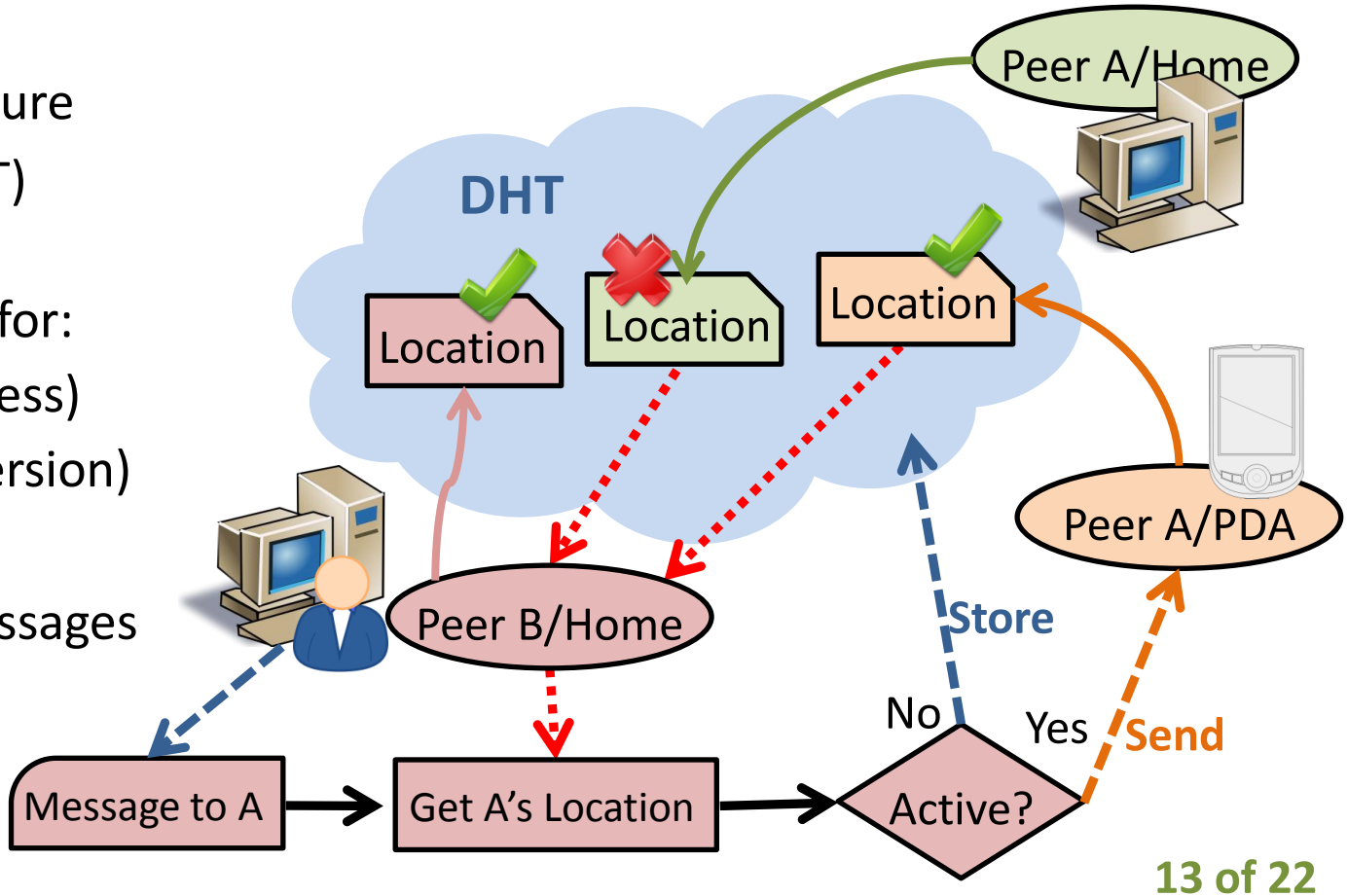
Peer-to-Peer infrastructure

A Lookup Service - (DHT)

The DHT stores **metadata** for:

- User's Location (IP address)
- User's data (Files and version)

It also stores incoming messages if the user is offline.



2. PeerSoN

Storage and Availability

- Data is **split into small objects** (files) – and replicated to the requesting nodes
- Parts of data may be unavailable on specific times.
- Space and time limitations for storing messages in DHT (if user is offline)

Privacy and confidentiality

Use both **symmetric** and **asymmetric** cryptography:

- The data is encrypted with a **symmetric key**.
- This symmetric key is encrypted with the **public key of each recipient**.

- Users easily **added but hardly removed** from a group(**re-encryption** is required)

2. PeerSoN

+ Globally Unique User ID – Resistant to the **Man-In-The-Middle** attack.
+ Use of cryptography for preserving privacy and confidentiality
+ Handshake for connection, thus a user can avoid un-wanted data.

- **Data availability** and **freshness** is **not 24/7 guaranteed.**
- Does not leverage on trust relationships of the users.
- **Impersonation** and **Sybil Attacks** are hard but **feasible.**
- Private user information can be inferred from metadata

3. Vis-a-Vis

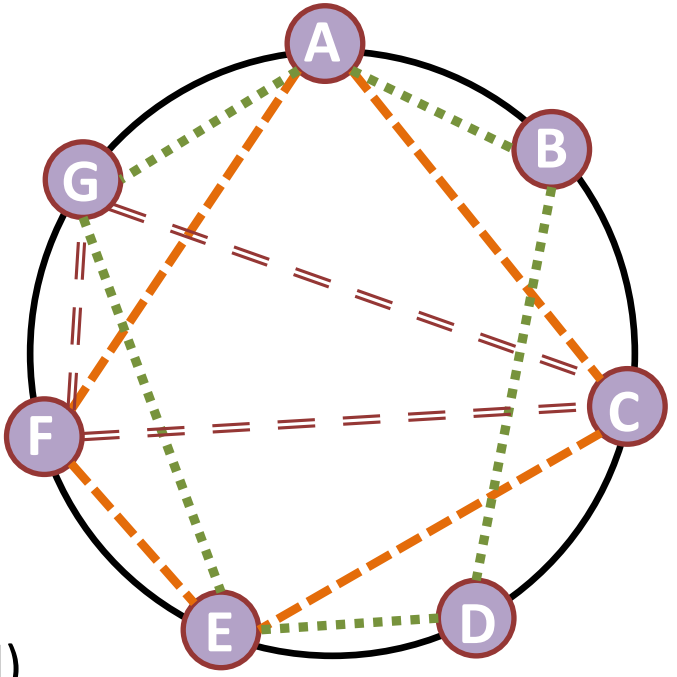
Virtual Individual Server (VIS) → A Virtual machine (acts as a proxy server)

→ **Data storage** and **management**

VISs are organised into **P2P overlay networks**

Each overlay corresponds to **a social group**

- Multiple VISs are connected to form an overlay
- Each VIS belongs to multiple overlay networks



Cloud-based VIS ([+] availability [-] security)

Self-hosted machines (replication and PKI is needed)

3. Vis-a-Vis

Virtual Individual Servers - The cloud-based approach

- ***Restricted* data**: access only to authenticated nodes
  - Diffie-Hellman Shared secret key (on friend addition)
- ***Searchable*** data: Accessible to strangers
  - The user **create groups** as **<descriptor, value> pairs** for each attribute.
- Each group is an overlay P2P network, implemented with a DHT.

  Peers join a group upon approval of existing members.

3. Vis-a-Vis

+ **High availability** due to the cloud-hosted virtual machines.
+ Privacy and confidentiality through secure (encrypted) communication.
+ Open and Close Groups, defined access control policies for each group.

- The data and the shared secret keys are stored **un-encrypted** within the VIS.
- Vis-à-Vis is vulnerable to **malware.** There is no control on execution
- Vulnerable to **Sybil attacks** as an adversary can create multiple VISs.
- Vulnerable to **Impersonation attacks** (no control on created VISs)

4. DECENT

A fully decentralised OSN (**peer-to-peer** architecture).

Uses a distributed hash table (DHT) for data storage

Confidentiality, Integrity → **Cryptography**

Availability, Freshness → **Data replication** (with versioning)

Attribute-based Encryption (ABE):

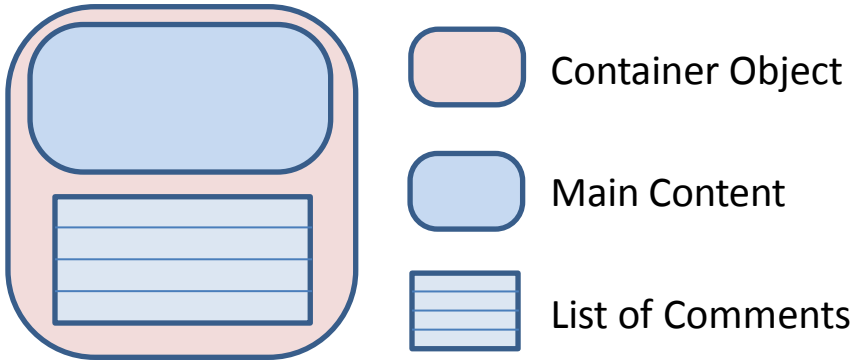
Many decryption keys, each one for **a set of attributes**.

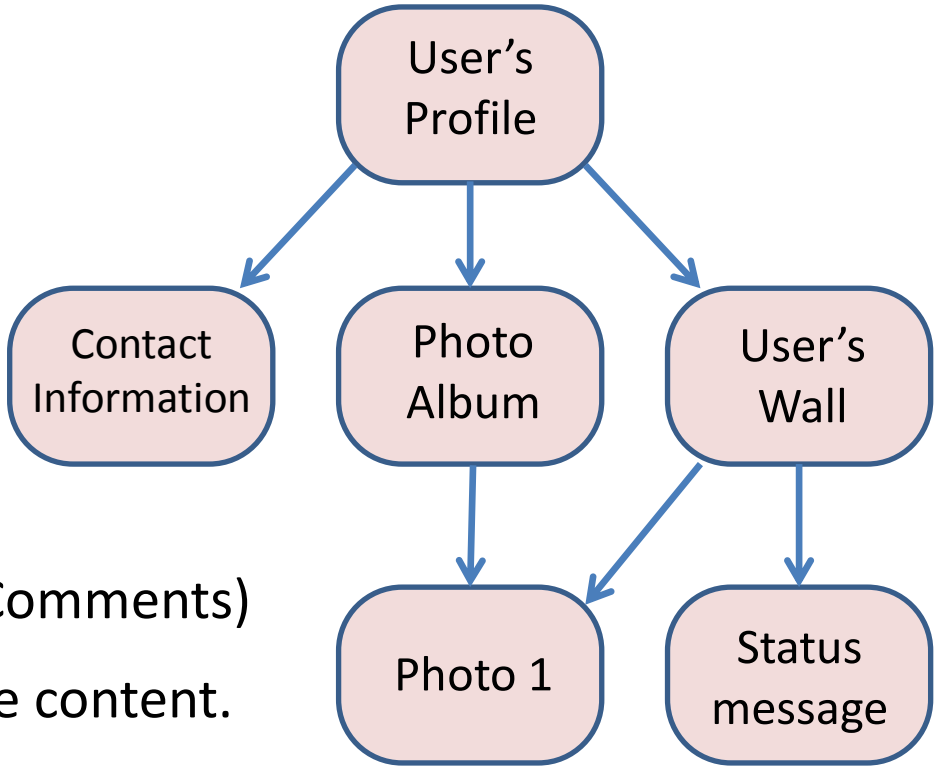
DECENT uses a hybrid approach:

- Objects are encrypted with **symmetric key** cryptography (AES).
- Symmetric keys are encrypted with ABE

4. DECENT

Based on the **Object-Oriented Design (OOD)**





Container objects (Main Content + List of Comments)

Comments can be more restrictive than the content.

The objects has references to other objects

4. DECENT

+ **High availability** due to data replication.
+ The data is stored encrypted, access control with ABE
+ The used DHT is immune to DDOS attacks.

- If the data are replicated only to malicious nodes – availability problem.
- There is no control on **spam dissemination** and malware distribution
- Vulnerable to large scale **Sybil attacks** and **Impersonation attacks.**

SUMMARY

Web-based decentralised OSNs

+ Encrypted and authenticated communication
- Vulnerable to Sybil attacks
- User's data are stored **unencrypted**.

Peer-to-Peer (P2P) OSNs

- **Availability** issues
- **Spam** dissemination and **malware** distribution
- **Sybil attacks** and **Impersonation attacks.**