Trust based Clustering for Group Key Management

Hamida SEBA

Graphs, Algorithms and Applications

Laboratoire d'InfoRmatique en Image et Systèmes d'information

LIRIS UMR 5205 CNRS/INSA de Lyon/Université Claude Bernard Lyon 1/Université Lumière Lyon 2/Ecole Centrale de Lyon

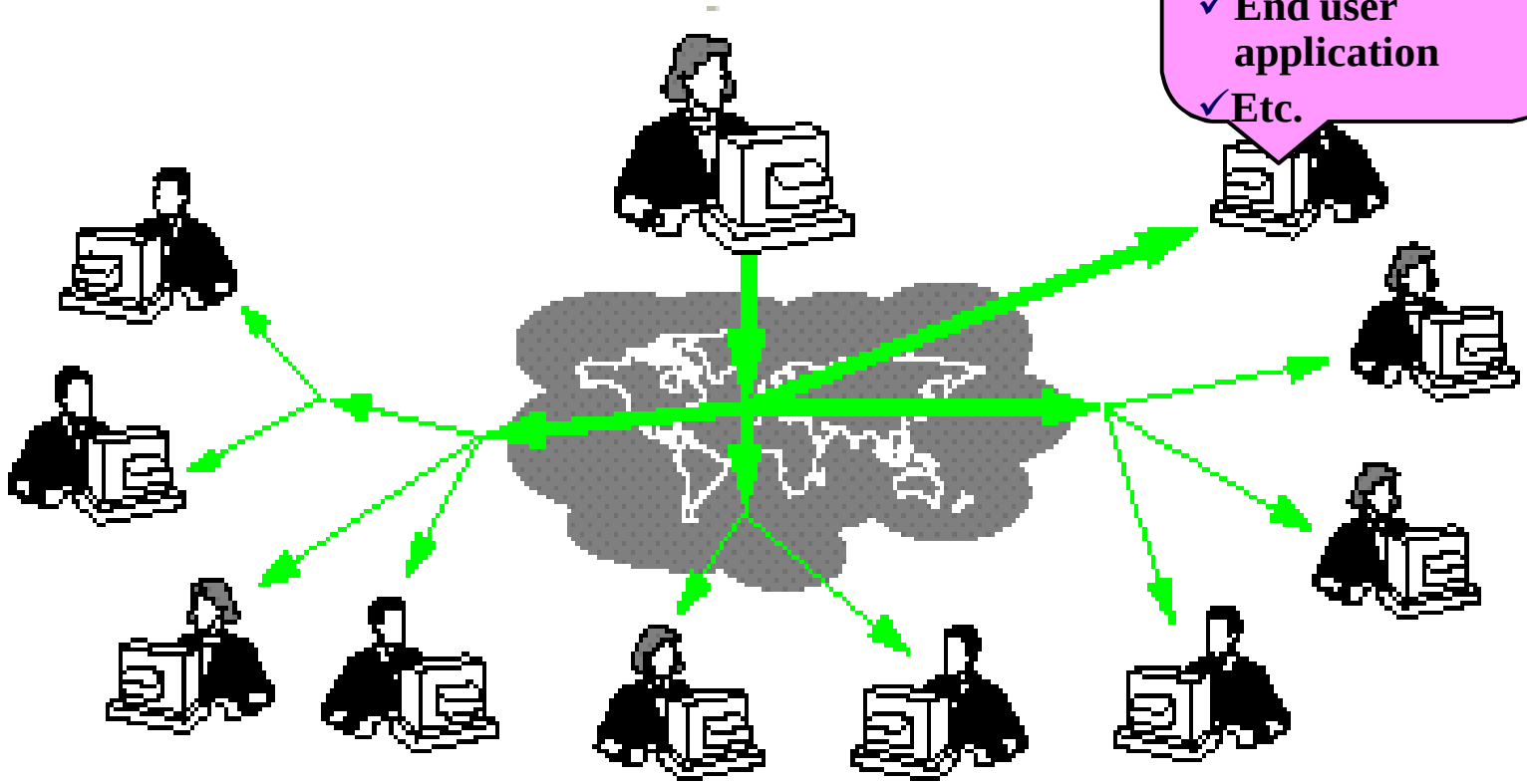
http://liris.cnrs.fr

In this talk:

- Group based applications
- Security of Group communication
- Group key management
- Trust based clustering for group key management

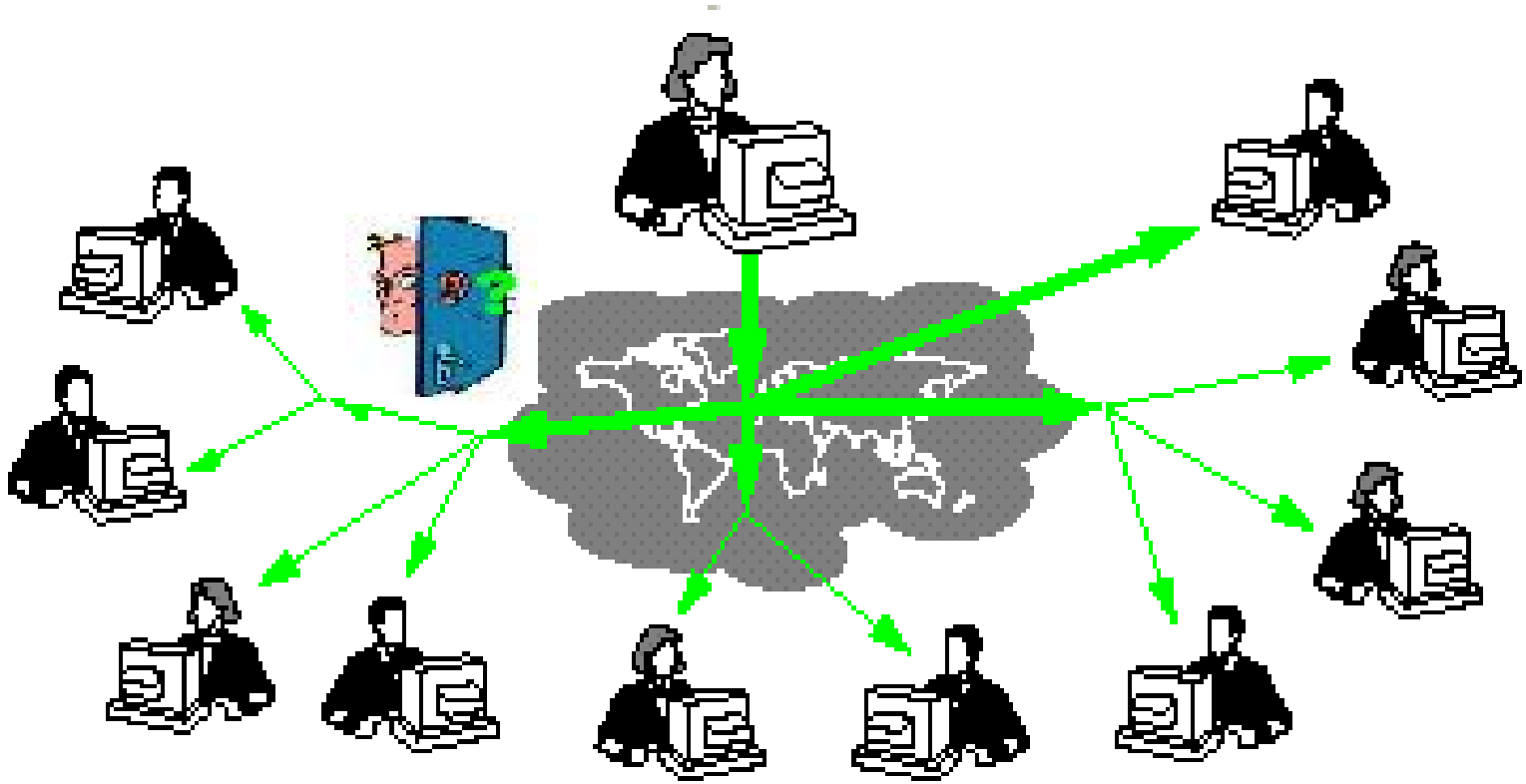
Group based applicatio

- ✓Process
- ✓Web service
- ✓Agent
- ✓End user application
- ✓Etc.



- Teleconferencing
- Collaborative work
- Replicated databases
- Distributed interactive simulation
- E-learning
- Etc.

Security of Group Communication



Prevention

- Confidentiality
- Authentication
- Integrity
- Non-repudiation

Confidentiality- Key Management

- **Solution= Encryption**
  - **Symmetric Key : shared between the sender and the receivers.**
  - **This key is called : the group key**

Main issue : how to compute and distribute keys?

GROUP KEY MANAGEMENT

- **GROUP KEY: a secret quantity known only to current group members**

- **BACKWARD SECRECY**
  - **Any subset of group keys cannot be used to discover previous group keys**

- **FORWARD SECRECY**
  - **Any subset of group keys cannot be used to discover subsequent keys**

- **THE GROUP KEY MANAGEMENT PROTOCOL MUST UPDATE THE GROUP KEY (REKEY)**

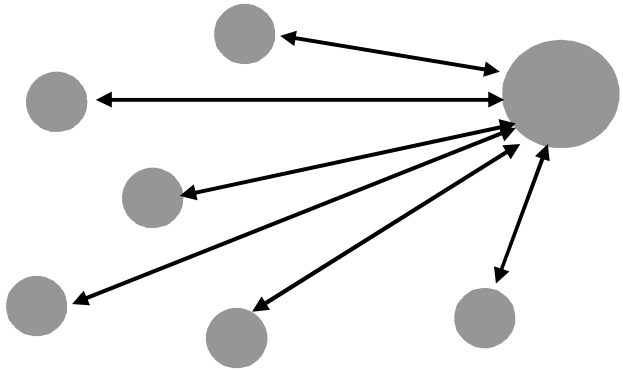
A new group member can not read data exchanged before he joins the group

An excluded member can not read data exchanged after he leaves the group

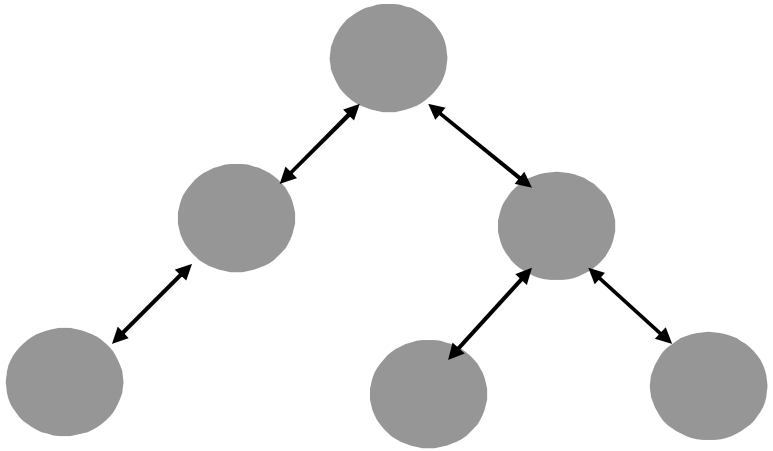
MODELS OF GROUP KEYS (1)

GROUP KEY DISTRIBUTION

- One party generates a secret key and distributes it to others



Pairwise model



Hierarchical model

(tree of members or third parties

Group member

No key generation

Key node

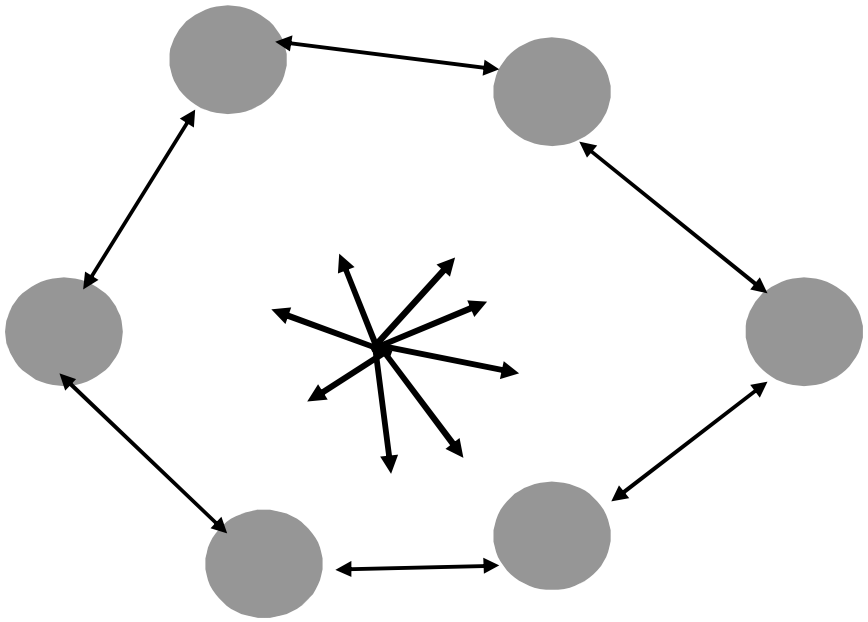
Group member

Does key generation

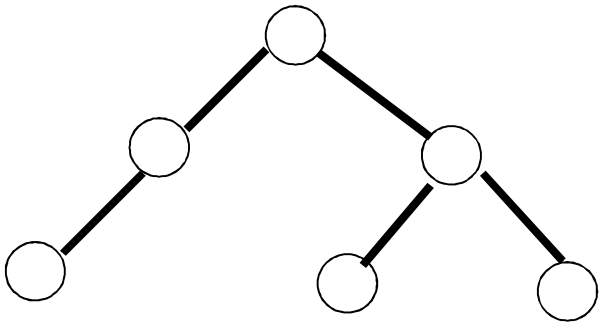
MODELS OF GROUP KEYS (2)

GROUP KEY AGREEMENT

- Secret key is derived jointly by two or more parties
- Key is a function of information contributed by each member
- No party can pre-determine the result



No pre-determined structure



Distributed Tree of keys

(maintained by each member)

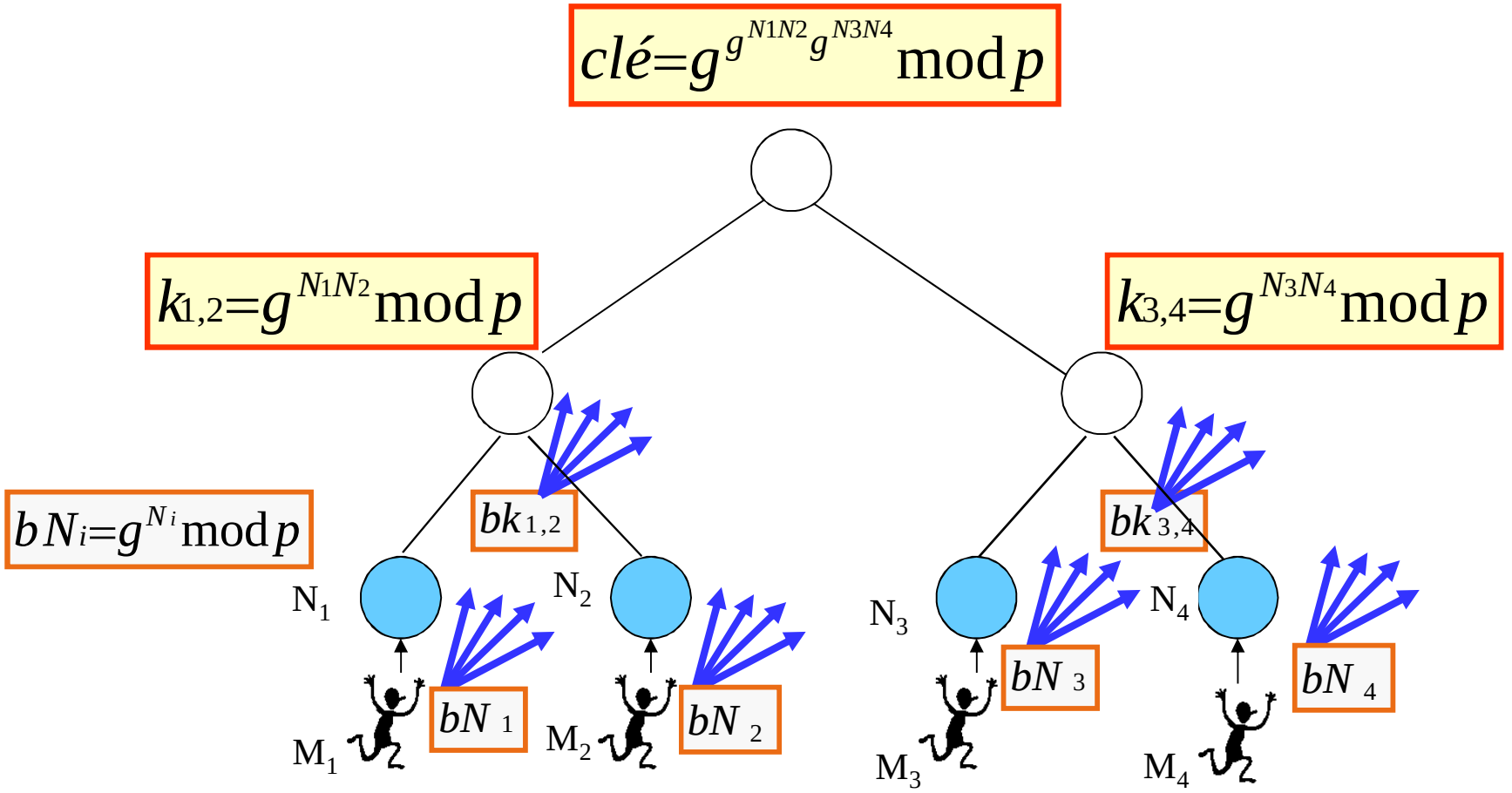
Group member

Does key generation

Key node

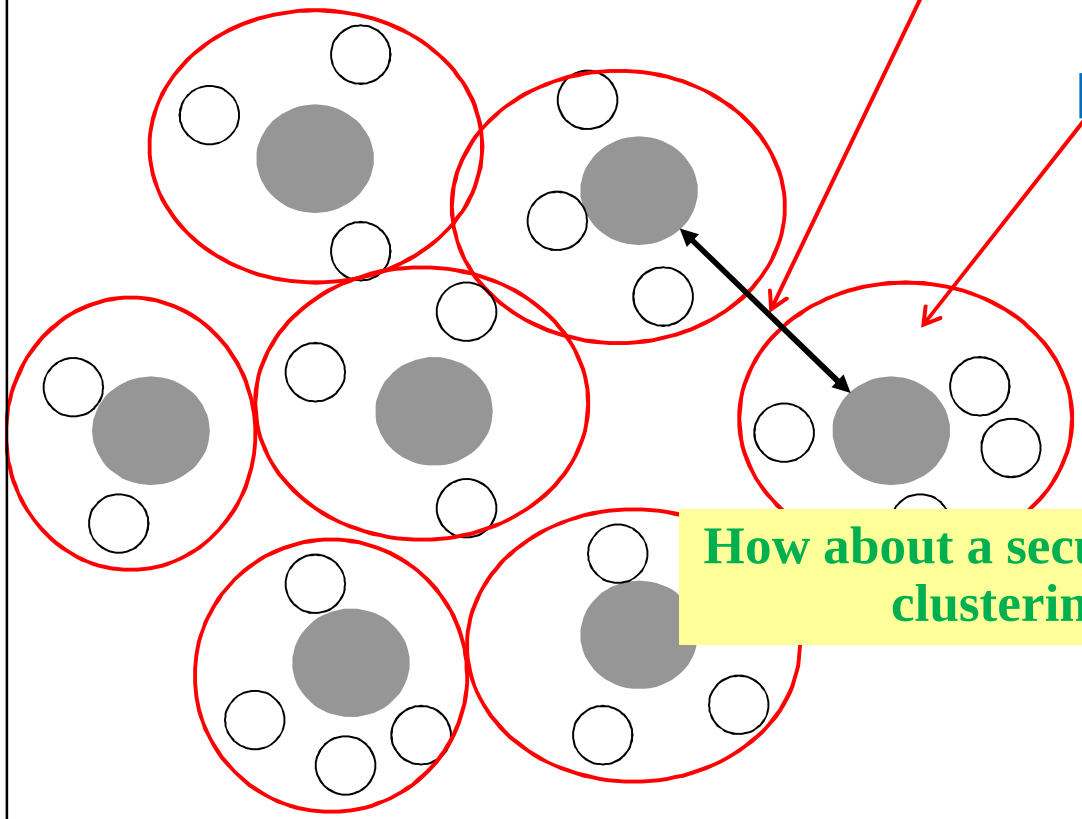
Group Key computation: an example

Protocole de Perrig et al.,2000



MODELS OF GROUP KEYS (3)

- **Hybrid Solutions**
  - **Cluster based.**



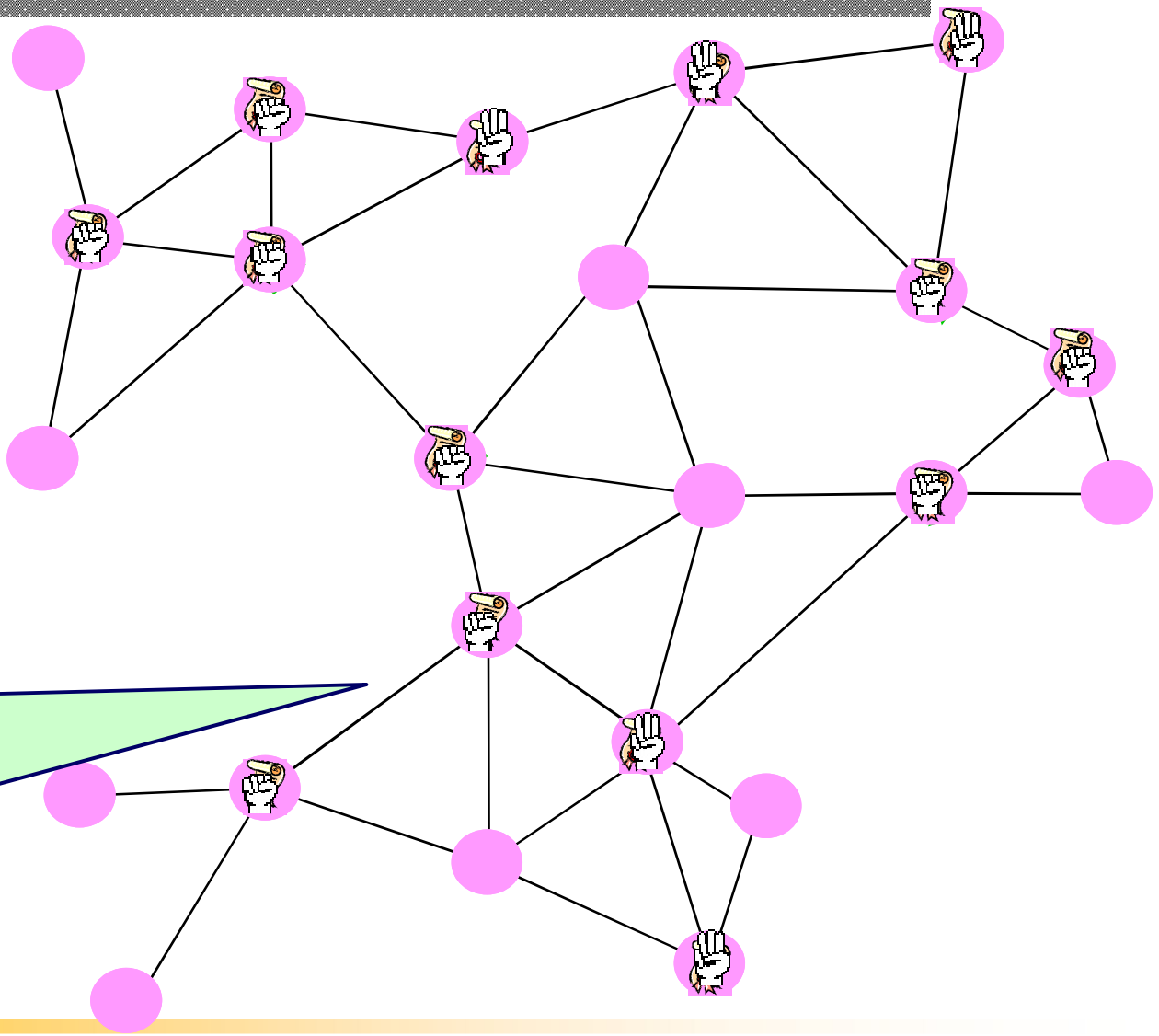
- How to construct/maintain clusters?
- How to compute inter-cluster keys and intra-cluster keys?

Group member
Does key generation
Simple node

Trust-based Clustering

Know each other: Establish trust/distrust relations

- **Log and analyze interactions**
- **Give scores**



Good interaction

Bad interaction

Application dependent

Peer to peer network:

Nodes: promiscuous mode

Forward packet: +

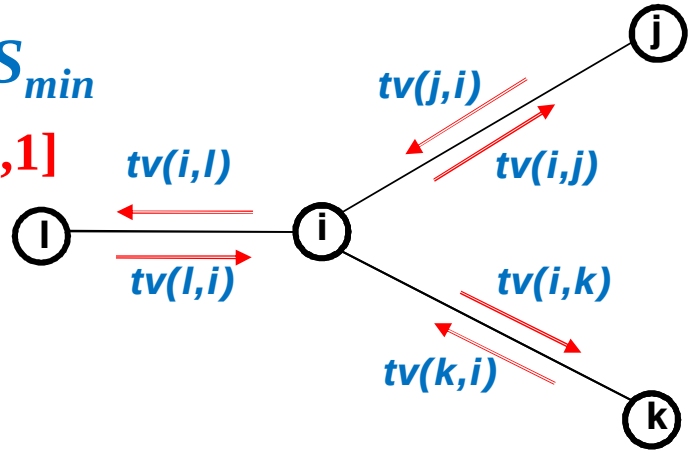
Black hole attack: -

Recommendations, etc.

Trust-based Clustering

- **Two trust thresholds : S_{max} and S_{min}**

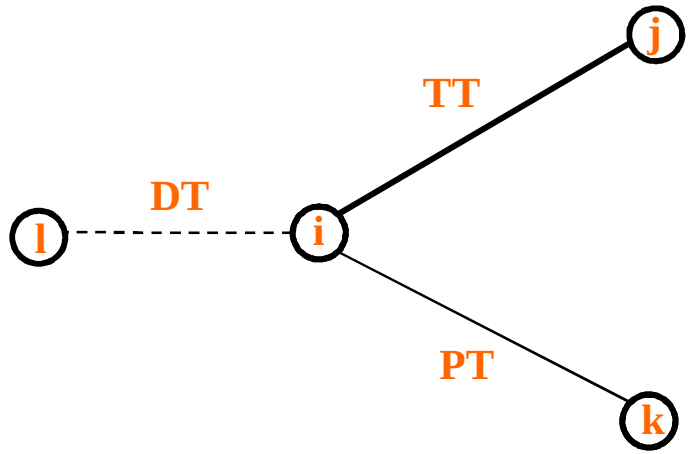
  [1, S_{min} S_{max},1]



  - **Total trust (TT)**
    - *tv(i,j) and tv(j,i)* $\in [S_{max},1]$
  - **Partiel Trust (PT)**
    - *tv(i,j)* $\in [S_{max},1]$ and *tv(j,i)* $\in [S_{min}, S_{max}]$
    - *tv(i,j)* $\in [S_{min}, S_{max}]$ and *tv(j,i)* $\in [S_{max},1]$
    - *tv(i,j)* and *tv(j,i)* $\in [S_{min}, S_{max}]$
  - **Distrust (DT)**
    - *tv(i,j)* and *tv(j,i)* $\in [-1, S_{min}]$

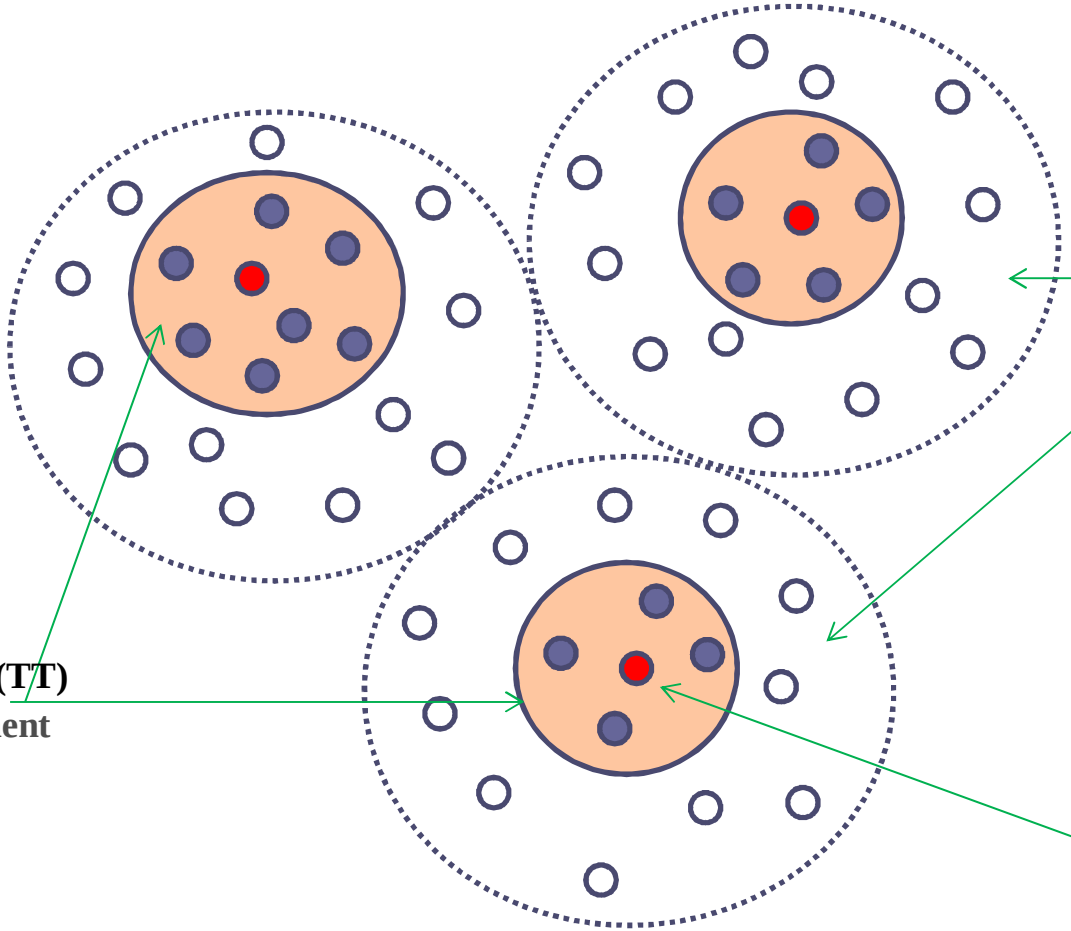
Trust-based Clustering

- **Two trust thresholds : S_{max} and S_{min}**



- **Total trust (TT)**
  - $tv(i,j)$ *and* $tv(j,i) \in [S_{max},1]$
- **Partiel Trust (PT)**
  - $tv(i,j) \in [S_{max},1]$ and $tv(j,i) \in [S_{min}, S_{max}]$
  - $tv(i,j) \in [S_{min}, S_{max}]$ and $tv(j,i) \in [S_{max},1]$
  - $tv(i,j)$ and $tv(j,i) \in [S_{min}, S_{max}]$
- **Distrust (DT)**
  - $tv(i,j)$ and $tv(j,i) \in [-1, S_{min}]$

Trust-based Clustering



Distrusted node

Cluster periphery (key distribution zone)

Cluster core (TT) (key agreement zone)

Clusterheads

Max number of TT relations

Self-stabilizing algorithm:

- **Adaptive**
- **Self-maintaining**

Example