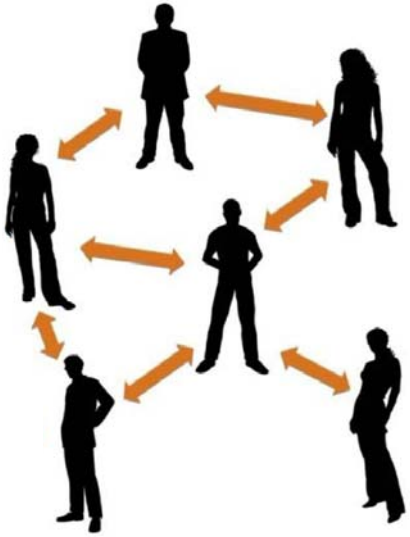
Poisoning the Kad Network

Or: Is P2P technology ready for „the next step“?

Thomas Locher (IBM)
David Mysicka (ETHZ)
Stefan Schmid (T-Labs)
Roger Wattenhofer (ETHZ)

Make It Distributed!



- Decentralization promises

  - scalability: „the (resource) cake grows with more participants“
  - robustness: no single (central) point of failure

Are today‘s p2p systems mature enough for the „next step“?

...ilic and „self-organizing“

Good principles for future Internet

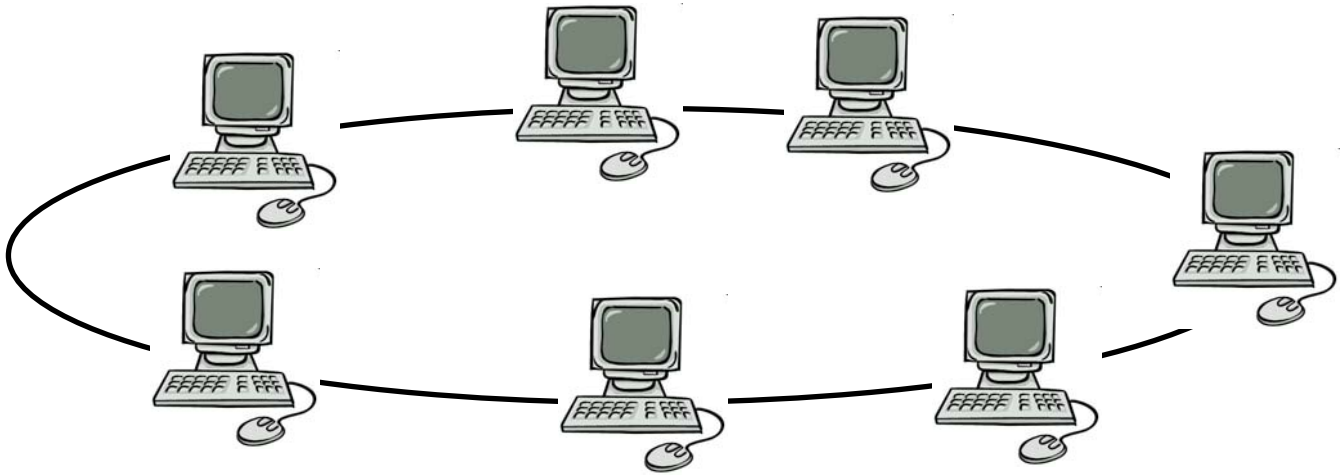
- organize DNS, data centers, etc. according to this paradigm?
- shift today? Relatively less traffic, more in the background?

Case Study: Kad

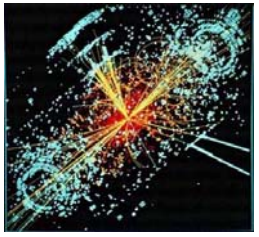
The Kad network (accessed with eMule for example):

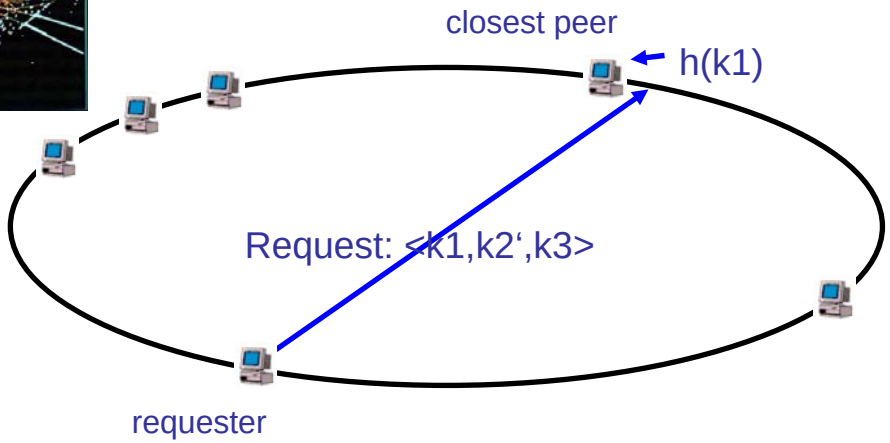
One of the few really distributed („modern and structured“) p2p systems in use!

Machines form a virtual ID ring (plus some hypercube links...):



How to Find a File? Kad Keyword Request



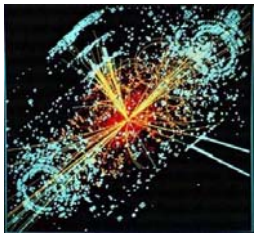


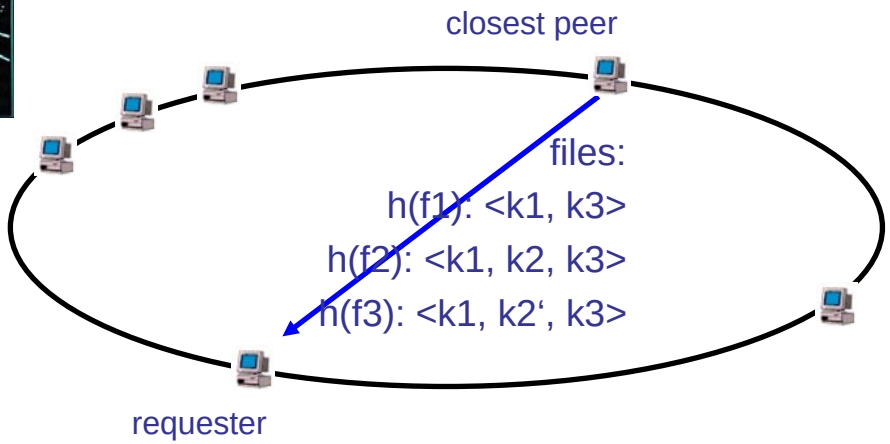
„Higgs CERN Trace“?

Lookup only with first keyword in list.

Key is hash function on this keyword, will be routed to peer with Kad ID closest to this hash value.

How to Find a File? Kad Keyword Request

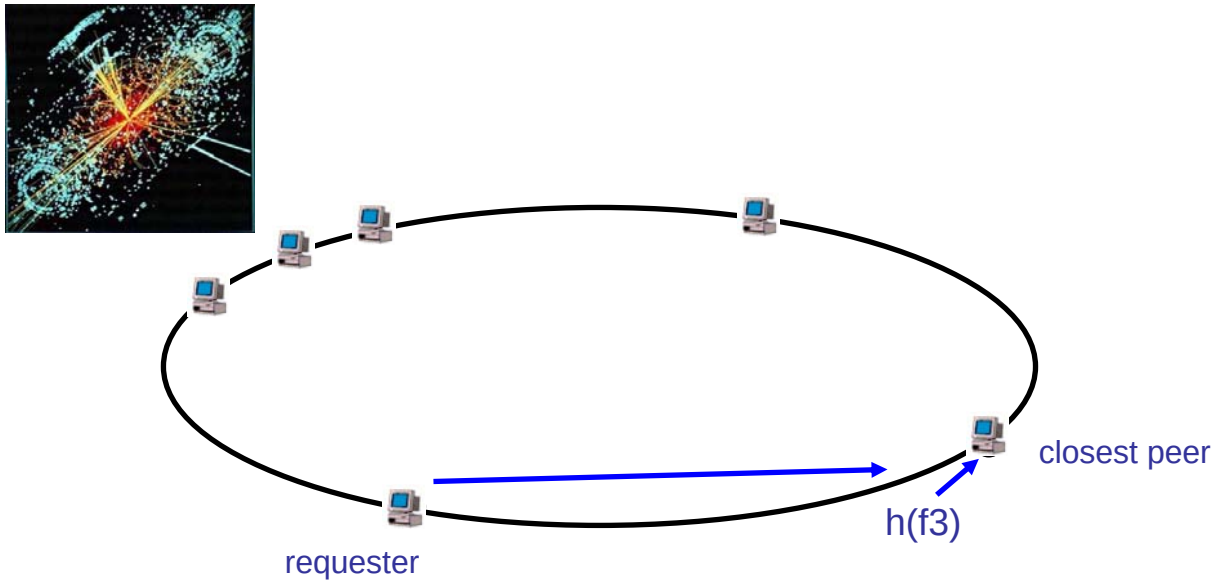




Peer responsible for this keyword returns different sources together with keywords.

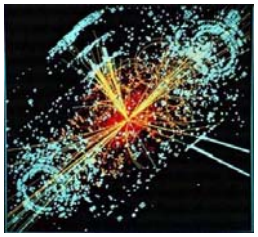
Remark: only those files with entries that include remaining keywords of request are returned.

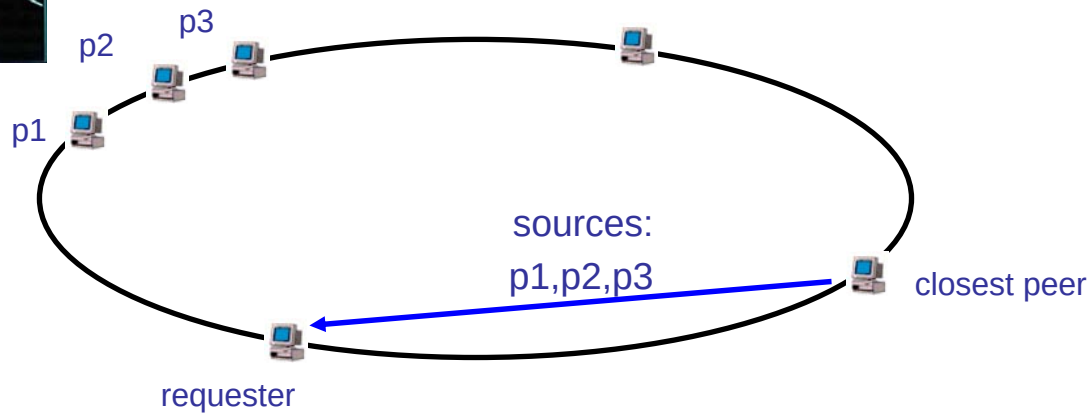
How to Find a File? Kad Source Request



Peer can use this hash to find peer responsible for the file
(possibly many with same content / same hash)

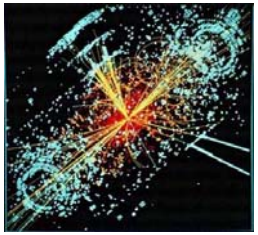
How to Find a File? Kad Source Request

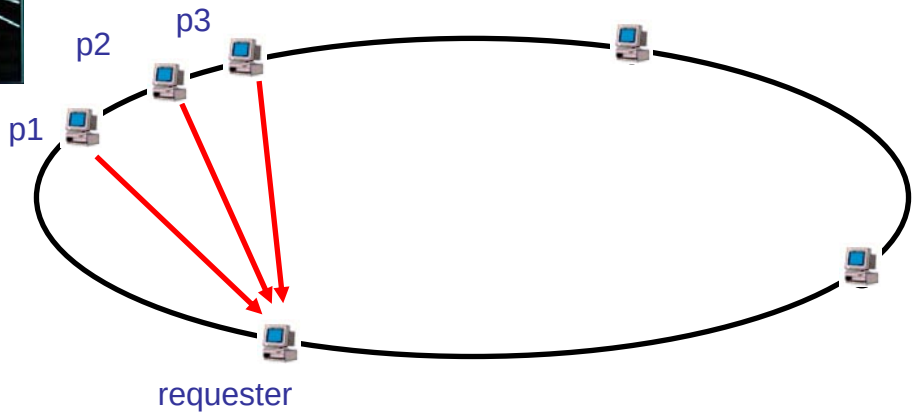




Peer provides requester with a list of peers storing a copy of the file.

How to Find a File? Kad Download



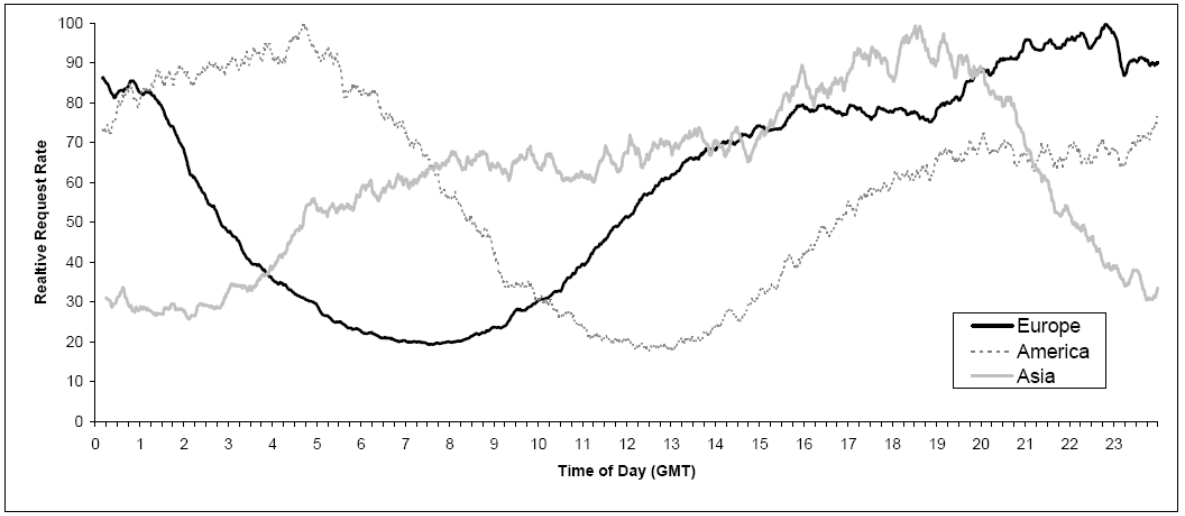


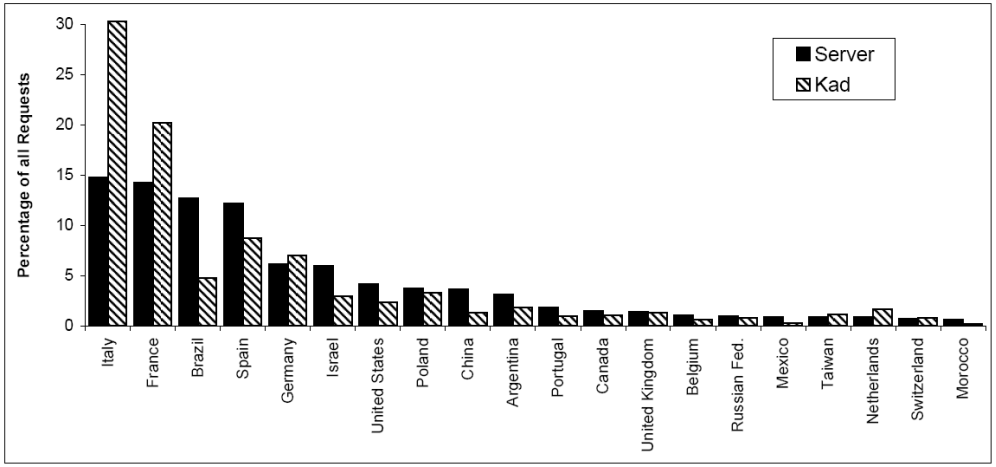
Eventually, the requester can download the data from these peers.

Some Facts: Measurements (wrt GMT)

Kad activity: evening

(weakness that ID is choosable: spy at 14 positions)





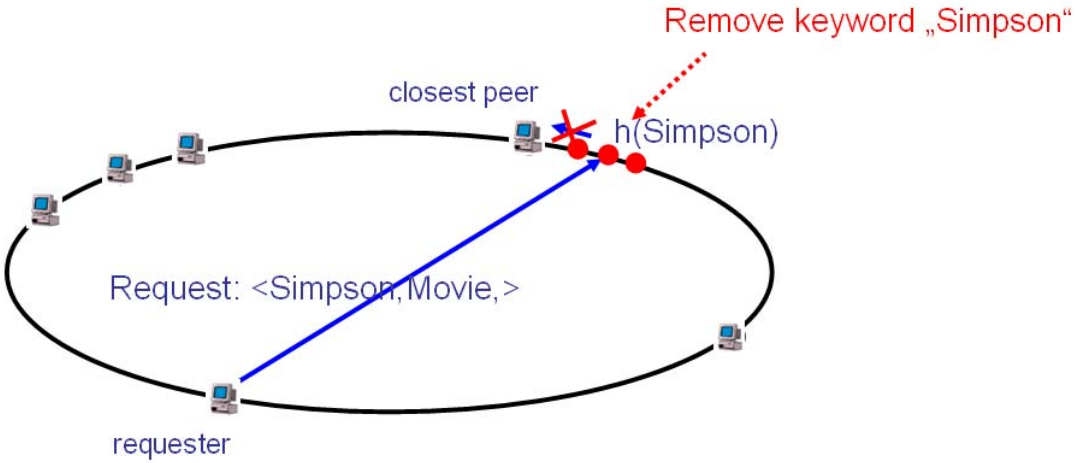
eMule: eDonkey vs Kad

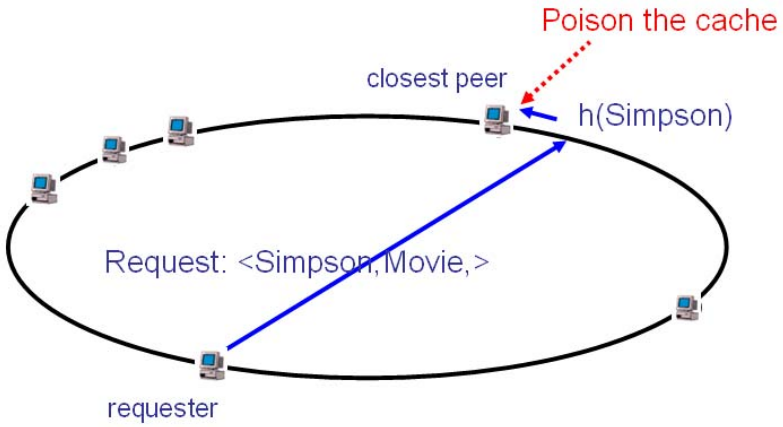
- In Kad, the distribution is more concentrated.
- In particular, it is quite popular in European countries.

How Robust is Kad?

Several Weaknesses Today (1)

- Censorship Node Insertion Attack (choose ID):



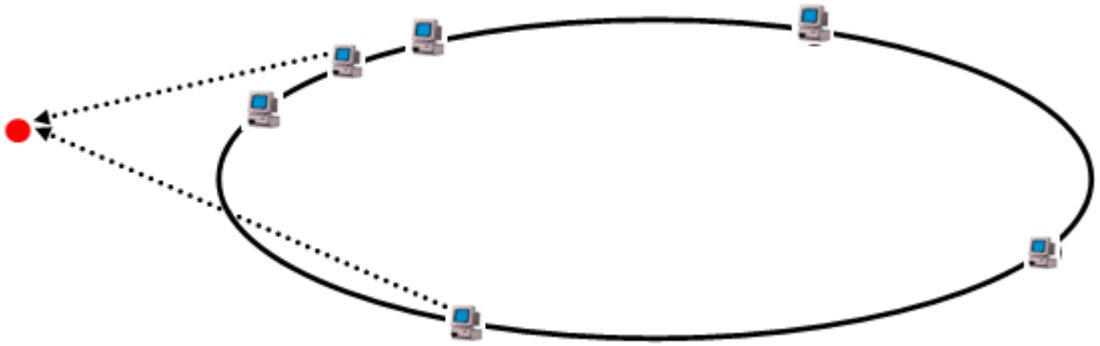


- Censorship Publish Attack: Fill cache of publishing peers with fake entries

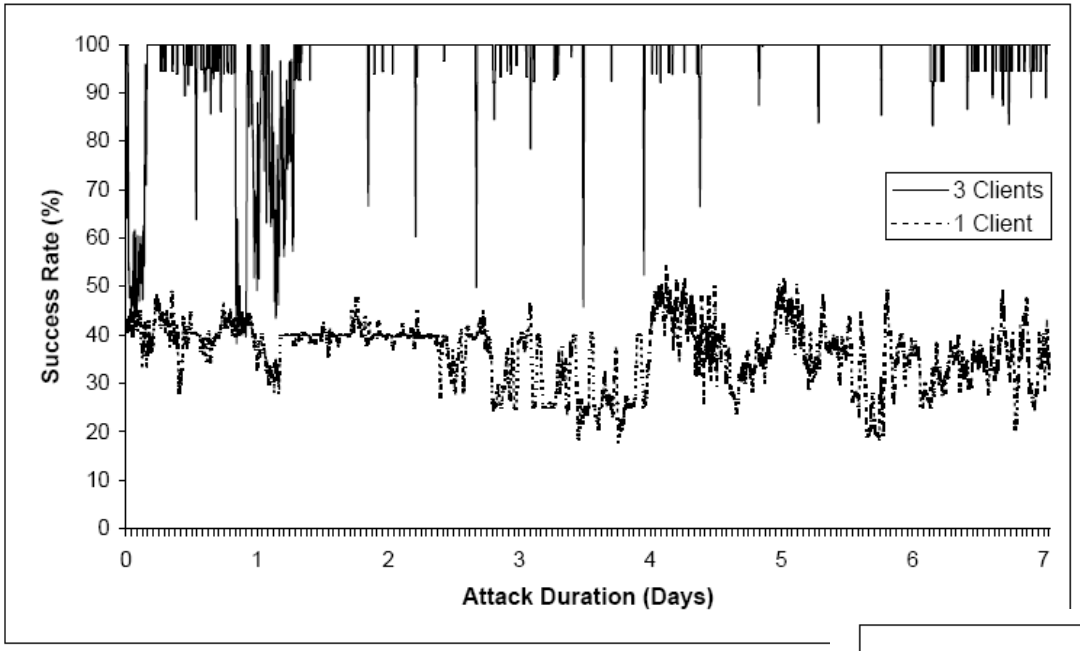
Several Weaknesses Today (2)

- Eclipse Attack: Choose positions / IDs to become only neighbor

- DoS Attack: Direct traffic to arbitrary IP address

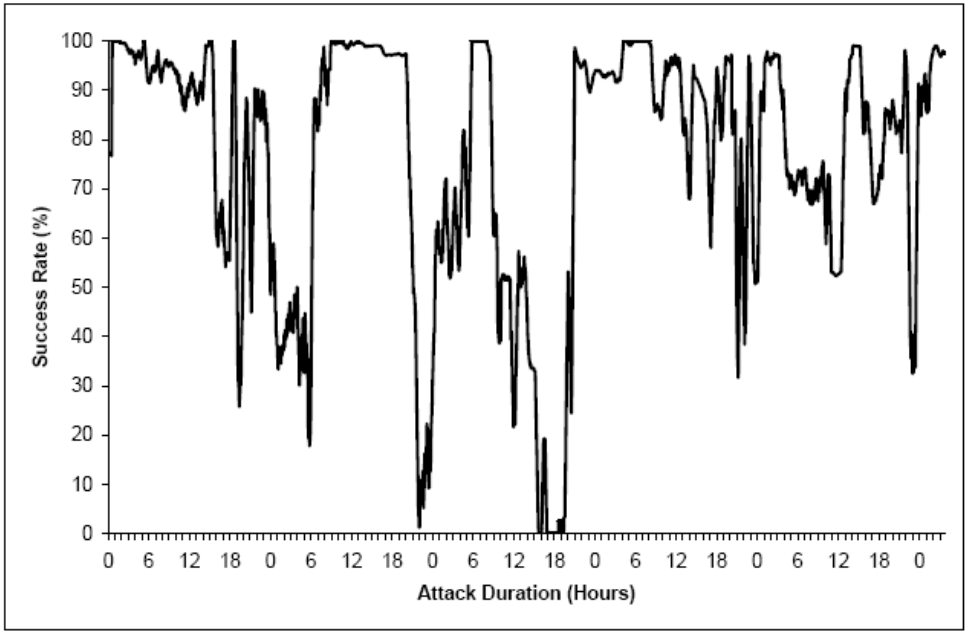


Impact



Node Insertion Attack

Publish Attack



Inherent weakness? Countermeasures?

Countermeasures?

It seems that these attacks can easily be prevented
- important insight: do not accept too much information from same peer!
- do not allow peers to choose their ID!

Idea: Choose overlay ID depending on IP address

But:

- dynamic IP addresses / DHCP? (lose credits?)
- NAT?
- other idea: compute a hash of user-generated data; however, as there are much less than 2^{128} peers in a network, an approximate ID will do the job for a peer insertion attack, and this can be computed efficiently
- attacker may have access to many IP addresses
- etc.

Help From the Theory Side?

A promising approach:

Robust distributed random number generation

(Awerbuch, Scheideler @ Theor. Comput. Sci. 2009)

Verifiable IDs, but many questions remain open (e.g., churn)

धन्यवाद !