Spam Filtering and *Global* Social Email Networks

Vwani Roychowdhury

Department of Electrical Engineering, UCLA

Joint work with:

Joseph S. Kong, Oscar Boykin, Behnam Rezaei, Nima Sarshar

UCLA

Motivation

- In the previous presentation, we see that the *personal* email network can be harnessed to filter spam.
- This naturally begs the question: can the *global* social email network, which reaches hundreds of millions of users worldwide, be harnessed to help fight spam?

2

Presentation Overview

UCLA

Collaborative Spam Filter Design

Related Work

Email Network as a Distributed Platform

Scalable Percolation Search

System Implementation and Design

Simulation Results

Malicious Infiltration and Trust Schemes

3

Collaborative Spam Filter Design

- A collaborative spam filter uses the collective memory of, and feedback from, the users to reliably identify spams.
- There are three main challenges facing collaborative spam filter design:

1. *How to find and connect users?:*
2. *How to make the search scalable?*:
3. *Who to trust*?

Existing Design and Solutions

- SpamNet: central server design, server-based search, seniority trust scheme.
- Vipul's Razor (open source version of SpamNet).
- SpamWatch: distributed hash table (DHT) P2P network design, approximate text addressing (ATA) search scheme. [Zhou03]

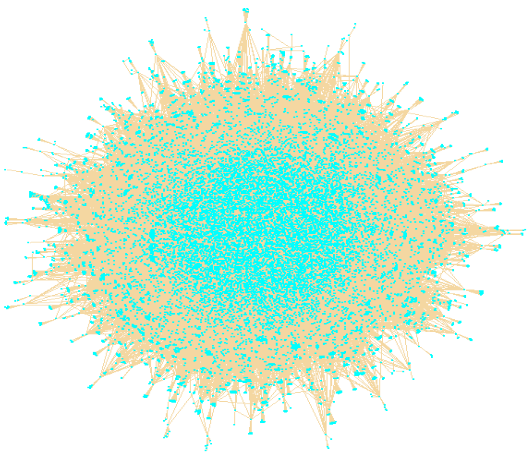
UCLA

Harnessing the Email Network

- The basic idea: instead of using dedicated central servers or P2P systems, we use our latent social email network as the distributed spam filter platform.
- All queries and communications are exchanged via background email through personal contacts.
- Motivation: we should exploit our existing network to find and connect users.

Topology of Email Networks

- Email network:
  nodes = email addresses;
  edges = email exchanges in either direction.
- Email network data [Ebel02]:
  56,969 nodes, 84,190 edges
- PL degree distribution:
  $P(k) \propto k^{-1.81}$
- We can utilize the scalable *percolation search algorithm* on this naturally power-law network to search.



- Real-world email network [Ebel02]

7

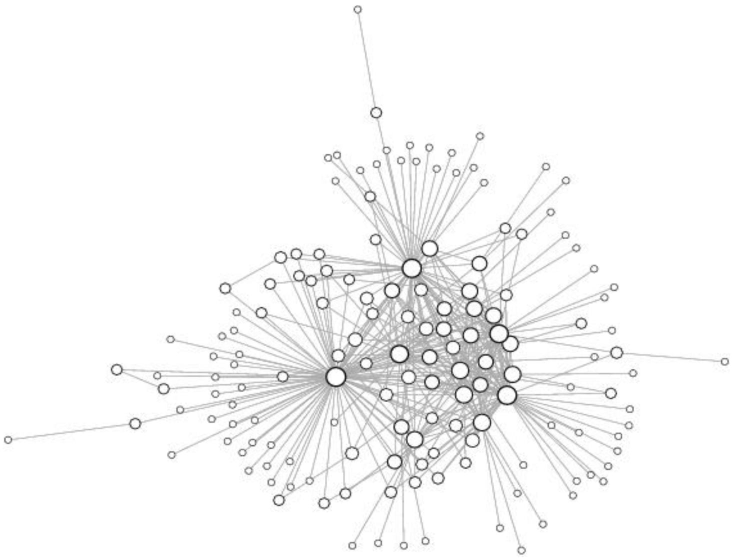
Percolation Search [Sarshar04]

- N. Sarshar, P. Boykin, and V. Roychowdhury. Percolation search in power law networks: Making unstructured peer-to-peer networks scalable. In *Proceedings of the 4th IEEE international conference on peer-to-peer computing*, 2004.
- Developed in the UCLA Complex Networks Lab led by Prof. Roychowdhury.
- Provably solves the search scalability problem in unstructured P2P networks.
- Even rare items can be found with probability 1, while keeping the traffic scalable ($O(\log^2 N)$ for the optimal network topology).
- Offers comparable performance to Distributed Hash Table (DHT) object location schemes but in an unstructured network.

UCLA

Bond Percolation Theory

- Consider the random network below (153 nodes, 366 edges).

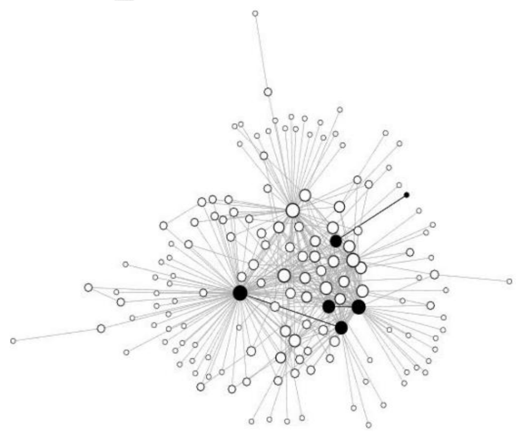


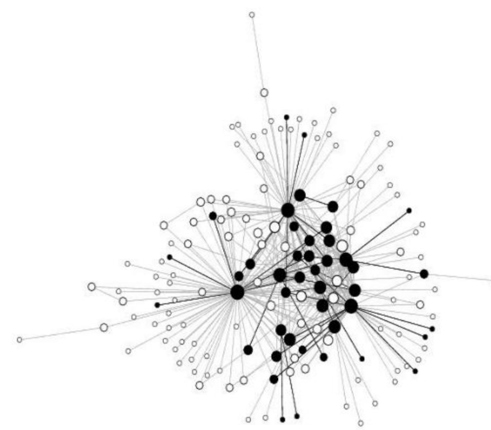
9

Bond Percolation Theory

- Bond percolating with probability p means to keep each edge with probability p.

- $p = .0144$
- $p = .0898$





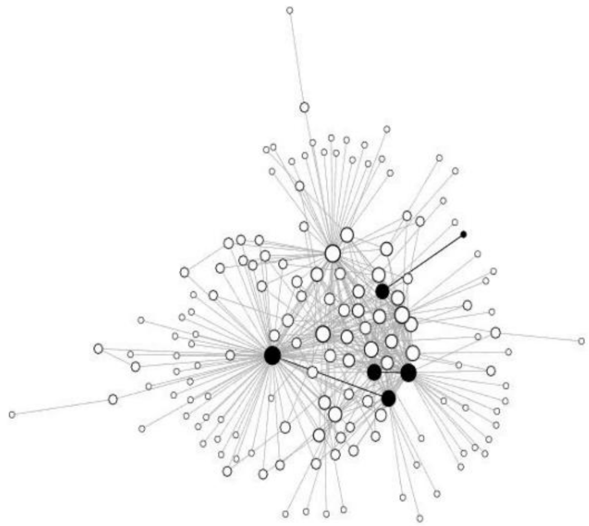
10

Percolation Threshold

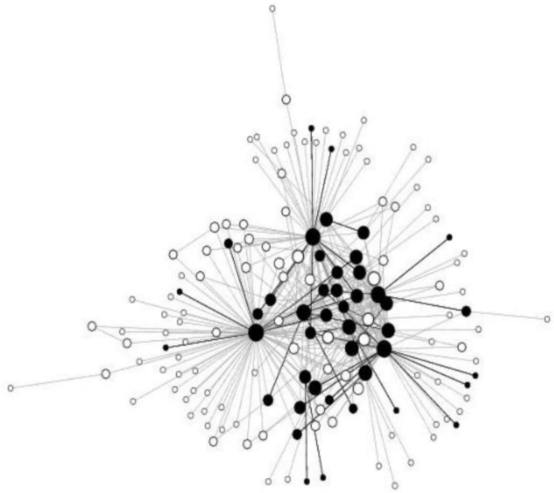
- The percolation threshold p_c of a network is defined as followed:
- for a network $p_c \approx \frac{<k>}{<k^2>}$, $p_c \approx .0359$ for our sample network.
- if $p < p_c$, the percolated network consists of small and disconnected components;
- if $p > p_c$, a giant connected component (GCC) emerges.

Percolation Threshold

$p = .0144 < p_c$



$p = .0898 > p_c$



12

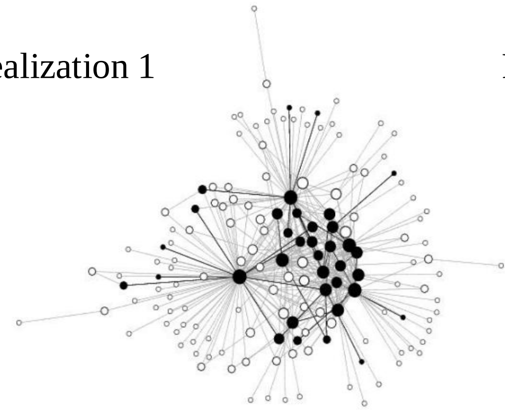
Percolation Search

UCLA

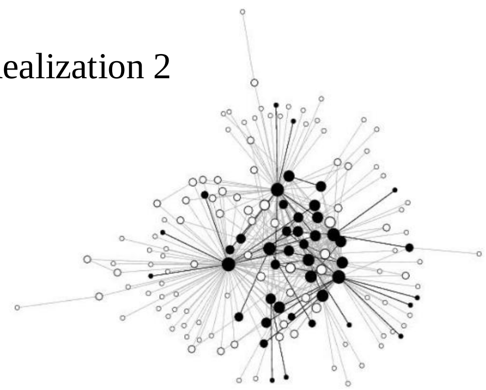
- For $p > p_c$, the high degree nodes almost suredly remain in the giant connected component of the percolated network.

$p = .0898 > p_c$ $p = .0898 > p_c$

Realization 1



Realization 2

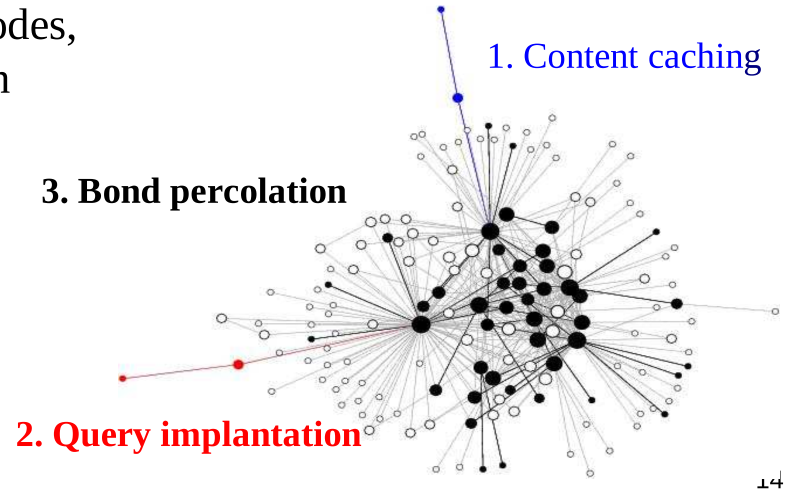


13

Percolation Search

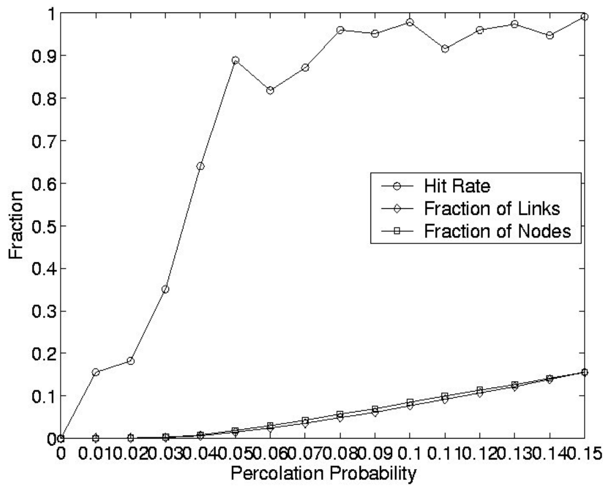
UCLA

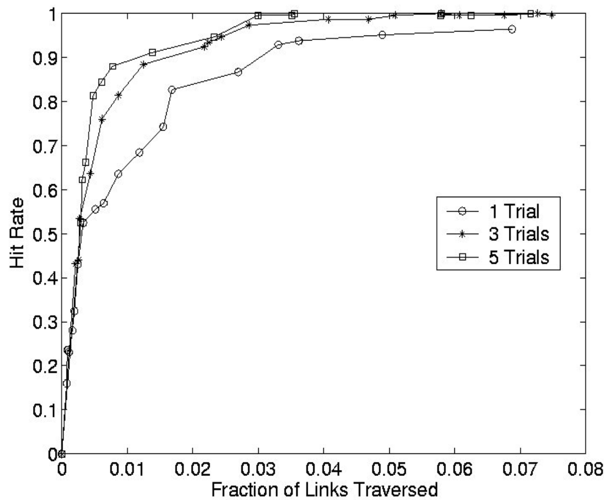
- The idea: start queries from high degree nodes AND cache contents at high degree nodes; high degree nodes can find each other by percolation.
- To find high degree nodes, do a random walk with log(N) hops.



14

Percolation Search on Email Net





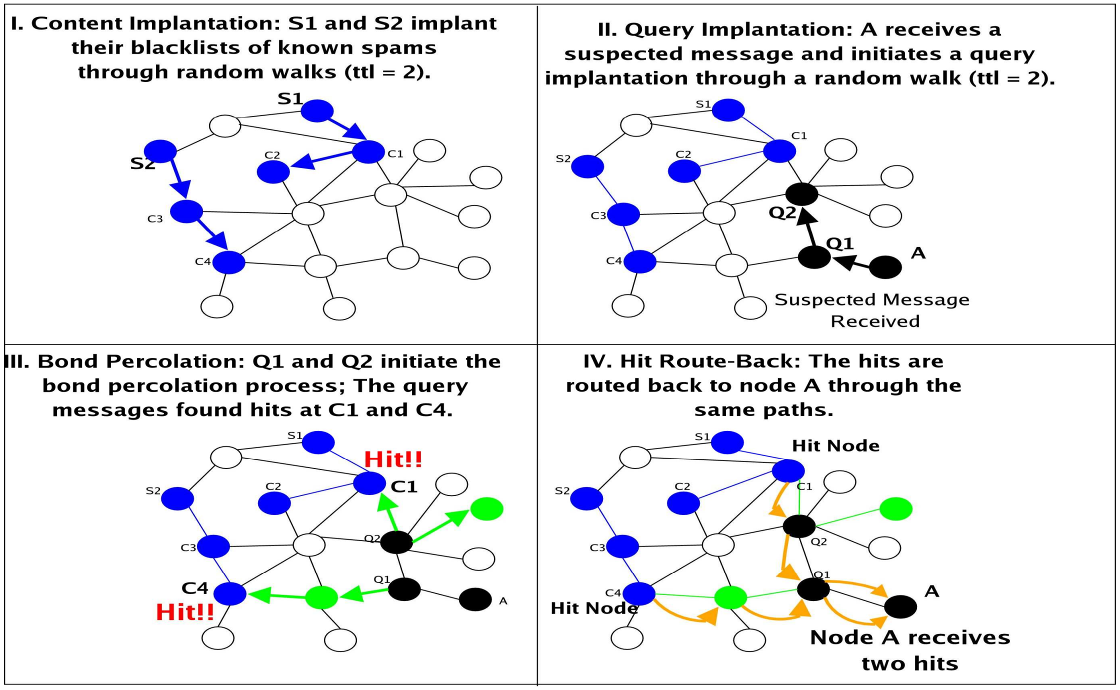
Simulating percolation search on real-world email network: only one unique copy of the desired content exists on the network of 56,969 nodes. A node is chosen uniformly randomly to search for this unique content.

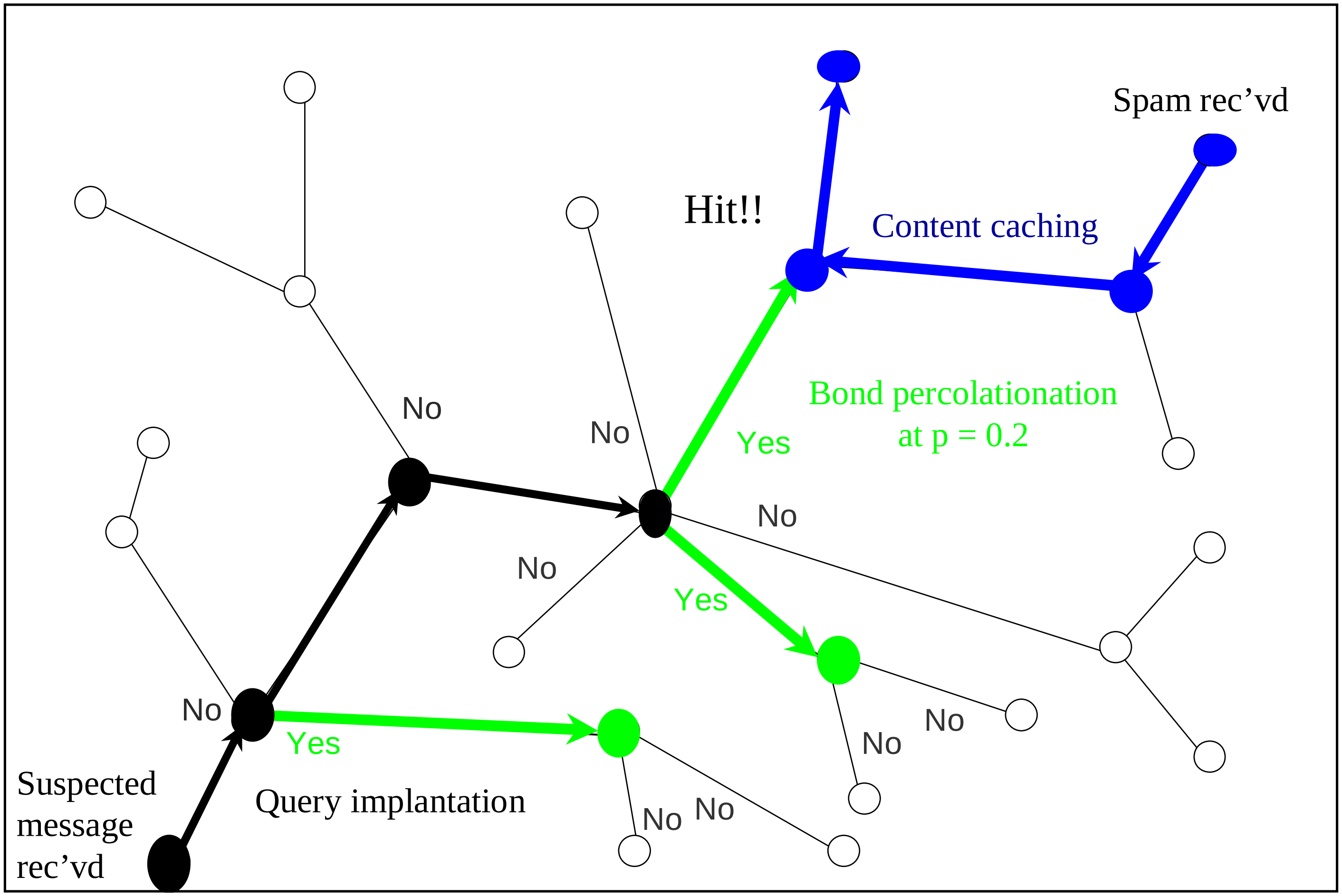
UCLA

Build a Distributed Spam Filter!

- Use the digest-based mechanisms to index spams.
- Use our latent social email network as the distributed platform.
- Use the efficient percolation search protocol to query for suspected spam messages indexed by digests.

System Protocol




Spam rec'vd
Hit!!
Content caching
Bond percolationation
at p = 0.2
Yes
No
No
No
No
Yes
No
Yes
Suspected
message
rec'vd
Query implantation
No
No
No
No

Distributed Client Implementation

- Implement the system as a plug-in to any email client such as Sendmail.
- Any other existing spam filter can be used to do initial filtering.
- All system messages (e.g. queries, hit returns and content caching) can be done via background email exchanges.

Algorithm 1 PROCESS-MAIL(Email E)

```
if DefinitelySpam(E) then
    Mark E as Spam
else if DefinitelyNotSpam(E) then
    Mark E as not Spam
else
    D_e = Digest(E);{Gray SPAM};
    Implant percolation of D_e on a random walk of length l
    Wait(T);
    H_e = HitScore();
    if H_e < threshold then
        Mark E as not Spam
    else
        Mark E as spam
    end if
end if
```

Algorithm 2 Publish-Spam(Email E)

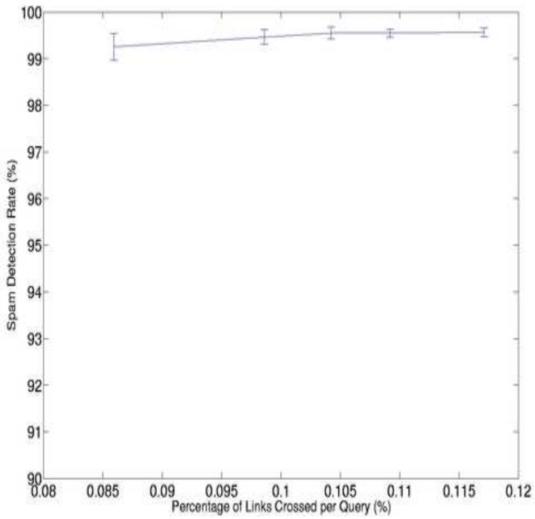
```
D_e = Digest(E);
Implant D_e on a random walk of length l
```

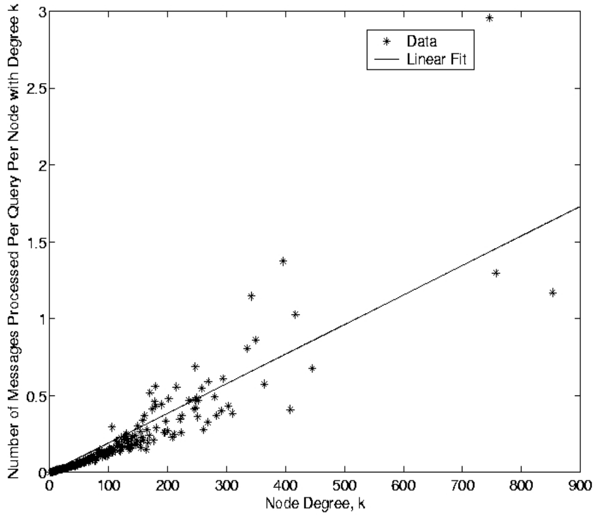
Simulation on Real Email Net

Number of Nodes	56,969
Number of Edges	84,190
Node Degree Distribution	Power-Law (exp = 1.8)
Bond Percolation Threshold	$\frac{<k>}{<k^2>} = .0169$
Time-to-Live (TTL)	50
# of arrivals of the same spam	500
# of hits needed to id spam	2
# of runs	30

20

Simulation Results





Detection rate vs. traffic. **Traffic vs. node degree.**

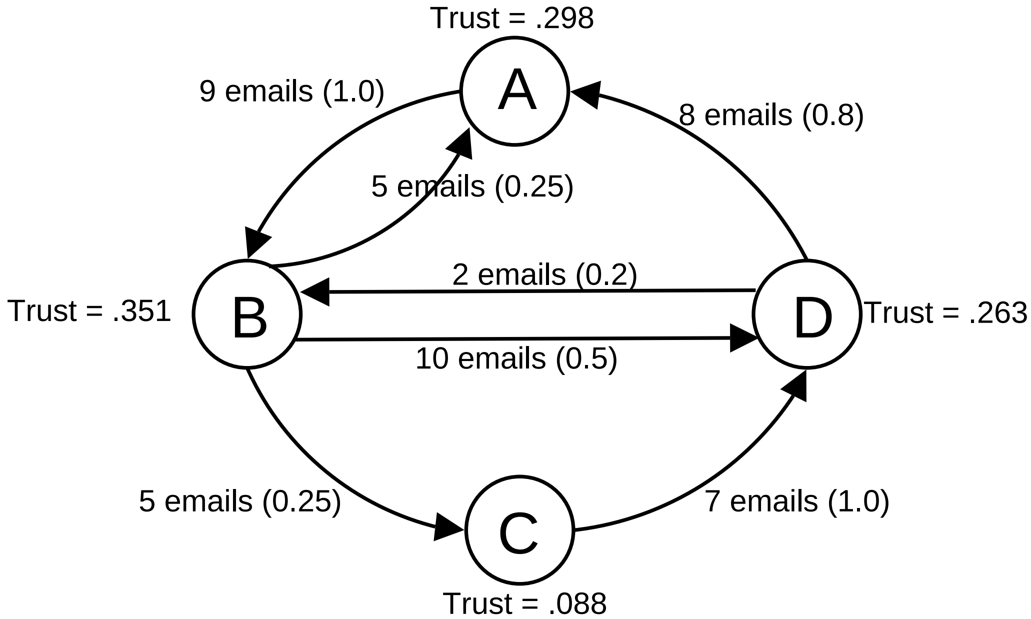
Infiltration and MailTrust

UCLA

- Malicious users will attempt to subvert the system.
- We propose an eigen-based trust scheme.
- The topological structure of the social email networks can be used to assign trust or reputation to individual users (just like PageRank).
- Model each email contact as placing a unit of trust on the recipient.
- Model the whole graph as a discrete time Markov chain.

22

Simple Illustration of MailTrust



23

UCLA

Markov Chain (Cont'd)

- Using the example on the previous slide, we can construct the following transition matrix:

$$P = \begin{pmatrix} 0 & 1.0 & 0 & 0 \\ 0.25 & 0 & 0.25 & 0.5 \\ 0 & 0 & 0 & 1.0 \\ 0.8 & 0.2 & 0 & 0 \end{pmatrix}$$

Markov Chain (Cont'd)

- In addition, we must ensure that the Markov Chain is ergodic.
- In email network, this means that nodes must out-links.
- This can be achieved by having nodes with zero out-degree assign uniform trust to a set of pre-trusted nodes who have been carefully picked.

UCLA

Markov Chain (Cont'd)

- In order to find the MailTrust score, compute the steady state probability distribution, which corresponds to the principal eigenvector of *P*.
- Can be done efficiently using the *Power Iteration* method.
- Using this method, the MailTrust score vector for our example are found to be:

.

$$\begin{pmatrix} .298 & .351 & .088 & .263 \end{pmatrix}$$

- The MailTrust scores are computed in a distributed fashion (see a paper by Kamvar et al. ACM WWW 03).

26

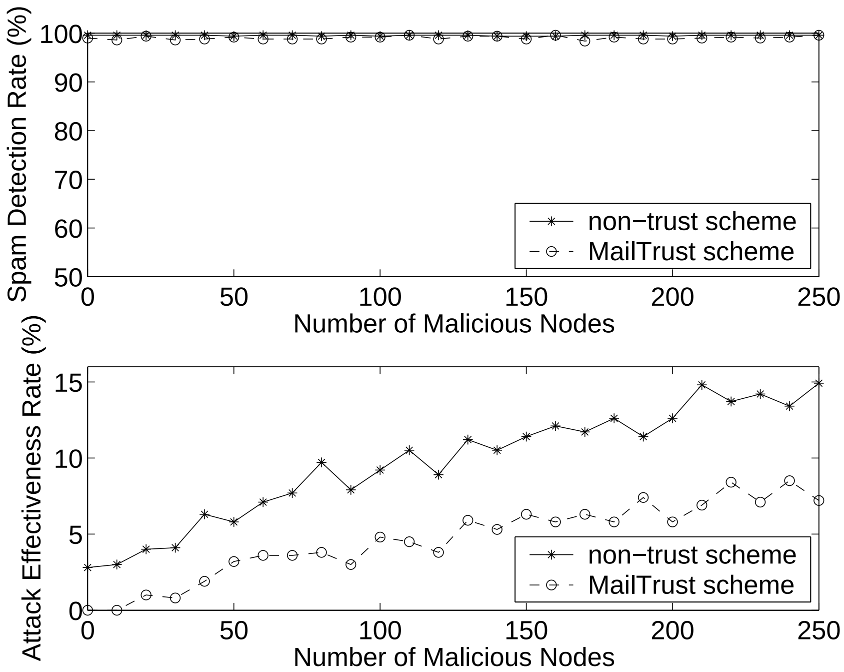
MailTrust Simulation Setup

- A malicious node can subvert the system by introducing blacklists of well-known valid messages into the network.
- Note that this form of attack will only raise the false positive rate of the system and it has no impact on the spam detection rate.
- a small fraction of nodes in the network (250 nodes) will be labelled as malicious nodes and these malicious nodes will blacklist nonspams from popular mailing lists;

MailTrust Simulation Setup (Cont'd)

- The malicious nodes will follow all specifications of the protocol such as forwarding and routing queries and storing cache implants for other nodes.
- The MailTrust scheme results in about 50% improvement in lowering the false positive rate.

MailTrust Simulation Results



29

UCLA

Summary

- We presented a distributed collaborative anti-spam system.
- Can be implemented as a simple plug-in.
- Delivers superior performance while keeping bandwidth cost on the internet extremely low.
- MailTrust scheme is effective in deterring malicious users.
- A more comprehensive version is available on: http://arxiv.org/abs/physics/0504026
- Journal version will soon appear on IEEE Computer.

30