New Metrics for Reputation Management in P2P Networks

D. Donato$^1$, M. Paniccia$^2$, M. Selis$^2$, C. Castillo$^1$, G. Cortesi$^3$, S. Leonardi$^2$

1. Yahoo!Research Barcelona – Catalunya, Spain
2. Università di Roma “La Sapienza” – Rome, Italy
3. Consorzio Università Industria, Radiolabs, University of Rome, Tor Vergata – Rome, Italy
1. P2P networks and reputation
2. Preliminaries
3. Threat models
4. Metrics
5. Evaluation
<table>
<thead>
<tr>
<th>#</th>
<th>Section</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>P2P networks and reputation</td>
</tr>
<tr>
<td>2</td>
<td>Preliminaries</td>
</tr>
<tr>
<td>3</td>
<td>Threat models</td>
</tr>
<tr>
<td>4</td>
<td>Metrics</td>
</tr>
<tr>
<td>5</td>
<td>Evaluation</td>
</tr>
</tbody>
</table>
A type of network in which each workstation has equivalent capabilities and responsibilities. This differs from client/server architectures, in which some computers are dedicated to serving the others.
P2P networks features

- ✓ Resource sharing: bandwidth, storage space, and computing power
- ✓ Information sharing
- ✓ Lack of central authority
- ✗ Lack of guarantee and certification of the shared resources
Downside

The open and anonymous nature of P2P networks opens doors to manipulation of the services (information) provided.
Downside

The open and anonymous nature of P2P networks opens doors to manipulation of the services (information) provided.

The open and anonymous nature of P2P networks makes it difficult to calculate reliable quality metrics for peers and objects.
Reputation management is used to:

- Describe the performance of peers in the network
- Describe how reliable they are
Reputation management is used to:

- Describe the performance of peers in the network
- Describe how reliable they are

Such mechanisms should be robust against malicious peers.
Starting point

EigenTrust
We start with EigenTrust [Kamvar et al., 2003], an algorithm designed for reputation management in file sharing application over p2p networks. The main idea is to combine this algorithm with metrics of reputation computed using techniques recently introduced for detecting and demoting Web Spam.
Contribution

- We adapt Truncated PageRank [Becchetti et al., 2006], Estimation of Supporters [Palmer et al., 2002] and BadRank in reputation management
- We introduce a number of new threat models
- We test existing and new threat models in a simulated environment
- We show that our combined approaches perform better than EigenTrust alone in reducing the amount of inauthentic downloads
Applications of EigenTrust for reputation management

P2P networks (using a DHT to record transaction outcomes – never allow a peer to do its own evaluation), but also online communities
Definition of local trust in EigenTrust

We define a local trust value $s_{ij}$ as

$$s_{ij} = sat(i, j) - unsat(i, j).$$

In order to avoid malicious peers to assign arbitrarily high local trust values, it is necessary to normalize them. The normalized local trust value is $c_{ij}$ is defined as follows:

$$c_{ij} = \frac{\max(s_{ij}, 0)}{\sum_j \max(s_{ij}, 0)}.$$
Hypothesis

Peers who are honest about the files they provide are also likely to be honest in reporting their local trust values.
Global trust

The idea of transitive trust, inspired by PageRank [Page et al., 1998], leads to a system where trust values propagate through paths along the network.
PageRank can be expressed as a weighted summation of paths of varying lengths

\[ S = \sum_{t=0}^{\infty} \frac{\text{damping}(t)}{N} P^t. \]

\( t \): the lengths of the paths.
\( \text{damping}(t) \): decreasing function of \( t \).
\( P \): row-normalized citation matrix
Truncated PageRank

Proposed in [Becchetti et al., 2006]. Idea: reduce the direct contribution of the first levels of links:

\[
\text{damping}(t) = \begin{cases} 
0 & t \leq T \\
C \alpha^t & t > T
\end{cases}
\]
Truncated PageRank

Proposed in [Becchetti et al., 2006]. Idea: reduce the direct contribution of the first levels of links:

\[
damping(t) = \begin{cases} 
0 & t \leq T \\
C\alpha^t & t > T 
\end{cases}
\]

☑ No extra reading of the graph after PageRank
Estimation of supporters

Becchetti et al., 2006 shows an improvement of ANF algorithm [Palmer et al., 2002] based on probabilistic counting [Flajolet and Martin, 1985]. After $d$ iterations, the bit vector associated to any page $x$ provides information about the number of supporters of $x$ at distance $\leq d$. This algorithm can be used to estimate the number of different peers contributing to the ranking of a given peer.
Estimation of supporters

[Becchetti et al., 2006] shows an improvement of ANF algorithm [Palmer et al., 2002] based on probabilistic counting [Flajolet and Martin, 1985]. After \( d \) iterations, the bit vector associated to any page \( x \) provides information about the number of supporters of \( x \) at distance \( \leq d \). This algorithm can be used to estimate the number of different peers contributing to the ranking of a given peer.
BadRank

If a page links to another page with a high BadRank, then also this page should be considered a page with negative characteristics. The difference with respect to PageRank is that BadRank is not based on the evaluation of inbound links of a web page but on its outbound links.

\[ br(i) = d \sum_{i \to j} \frac{br(j)}{\text{indeg}(j)} + (1 - d)e(i) \]

computed on the graph of negative evaluations
Reputation Management
Metrics
Donato et al.
P2P networks and reputation
Preliminaries
Threat models
Metrics
Evaluation
Network Models

Transaction Network
A link from a node (peer) $i$ to a node $j$ is inserted every time $i$ downloads a file from $j$. Each link is weighted with a positive value if the downloaded file was authentic, negative otherwise.

Positive Opinion Network
A link is inserted from a node $i$ to a node $j$ only after the download of authentic files.

Inverse Network
The transpose of the positive opinion network.
Threat Model A (individuals) and B (collective)
They provide good files sometimes
Threat Model D

Have a set of nodes providing good ratings for them
Threat Model G - malicious smart model

Sometimes give ratings to the rest of the network
Threat Model H - malicious smart model with camouflage

Sometimes provide authentic files and ratings to the rest of the network
Encourage peers to provide ratings about other peers

**Require:** EigenTrust score vector $ET$, Inverse EigenTrust score vector $I$

1. if $I[i] > 0$ then
2. return $ET[i]$
3. else
4. return 0
5. end if
Encourage peers to provide many ratings about other peers

**Require:** EigenTrust score vector $ET$, Inverse EigenTrust score vector $I$, threshold $tr = \sum_i \frac{ET[i]}{N}$

1: if $I[i] \geq tr$ then
2: return $ET[i]$
3: else
4: return 0
5: end if
Malicious peers receive positive values from the other members of the coalition (malicious and spy). This means that the most of the trust mass is propagated starting from nodes at few hops of distance.

\textbf{Require:} Eigentrust score vector $ET$, Truncated PageRank vector $P$, threshold $tr$

1: \textbf{if} $P[i] \geq tr$ \textbf{then}
2: \hspace{1em} \textbf{return} $ET[i]$
3: \textbf{else}
4: \hspace{1em} \textbf{return} 0
5: \textbf{end if}
Malicious peers supporters necessarily belong to the same coalition. This means that a malicious peer obtain an high reputation because of the great number of supporters at short distance from it.

The Bit Propagation algorithm can be used to perform an analysis of the connectivity of the transition network in order to detect local anomalies.

**Require:** EigenTrust score vector $ET$, Bit Propagation vector $BP$, threshold $tr$

1. if $BP[i] \geq tr$ then
2. return $ET[i]$
3. else
4. return 0
5. end if
Badness

Propagating badness

If $i$ trusts $j$ and $j$ distrusts $k$ then, with high probability, also $i$ should regard $k$ as untrustworthy. We can define the Global Badness as:

$$\text{neg}T = D^\top T$$

where $D$ is the normalized negative opinion matrix and $T$ is the EigenTrust Rank. Each peer $i$ has a global Badness given by

$$\text{neg}T_i = \sum_{j=1}^{n} \text{neg}C_{ji} \times T_j$$
Average BadRank for models A-D

Average BadRank after 25 and 50 cycles.
The badness is able to differentiate between good and malicious peers but it does not help in discovering spies.

We measure dishonesty:

\[ \text{dishonesty}_i = \sum_{j \in P} \text{neg}T_j \]

where \( P \) is the set of peers that \( i \) have given positive ratings. The dishonesty is high for all those peers which give good ratings to peers with high badness.
Average dishonesty for models A-D

Average Dishonesty after 25 and 50 cycles.
Settings

- 100 good peers
- 5 pre-trusted peers
- probability to supply corrupted files equals to 2% for good peers
Settings

- 100 good peers
- 5 pre-trusted peers
- Probability to supply corrupted files equals to 2% for good peers

Evaluation

We consider the average ratio between the number of inauthentic downloads and the total number of downloads.
Inauthentic downloads for threat model D (malicious and spies) and threat model G (plus smartness)
Threat models A (individuals) and B (collective)

EigenTrust, E. + TruncatedPR, E. + badness + dishonesty
Threat model C (camouflage) and D (spies)

EigenTrust, E. + TruncatedPR, E. + badness + dishonesty
Threat model G (smart) and H (smart+camouflage)

EigenTrust, E. + TruncatedPR, E. + badness + dishonesty
Variant: provide bad files, but be honest

Threat model A', C'
Variant: provide bad files, but be honest; combined attacks

Threat model $D+A', D+C'$
What’s next

- We have discussed several threat models and tools
What’s next

- We have discussed several threat models and tools
- Find more general threat models (not easy!)
What’s next

- We have discussed several threat models and tools
- Find more general threat models (not easy!)
- Propose more tools that increase the cost of attacks and/or make them less successful
What’s next

- We have discussed several threat models and tools
- Find more general threat models (not easy!)
- Propose more tools that increase the cost of attacks and/or make them less successful
- Propose techniques that can adapt to different environments (e.g.: learn how hostile is the network currently, behave accordingly)
What’s next

- We have discussed several threat models and tools
- Find more general threat models (not easy!)
- Propose more tools that increase the cost of attacks and/or make them less successful
- Propose techniques that can adapt to different environments (e.g.: learn how hostile is the network currently, behave accordingly)
What’s next

- We have discussed several threat models and tools
- Find more general threat models (not easy!)
- Propose more tools that increase the cost of attacks and/or make them less successful
- Propose techniques that can adapt to different environments (e.g.: learn how hostile is the network currently, behave accordingly)

Thank you!

PR0 - Google’s PageRank 0 Penalty.

Using rank propagation and probabilistic counting for link-based spam detection.

Probabilistic counting algorithms for data base applications.

In *WWW*, pages 640–651.
