TrustMe: Anonymous Management of Trust Relationships in Decentralized P2P Systems

Aameek Singh
aameek@cc.gatech.edu

Ling Liu
lingliu@cc.gatech.edu

College of Computing
Georgia Institute of Technology
Outline

- Introduction of concepts
  - Trust based P2P systems
  - Review of existing work
- Anonymity – Why is it essential ?
- TrustMe – Protocol details
- Security Analysis
- Experimental Results
- Conclusions and Future Work
Introduction – Use of Trust

- Open and anonymous nature invites malicious behavior – sharing harmful content, viruses
- Decentralized solutions are required
- Trust based reputation metrics
  - Measure the trustworthiness of a peer
  - Dynamically assign a trust value based on peer reviews
- What reputation metrics to use – Trust Model
- How to access and secure their use – Access Protocol
Desired Features

- **Security**
  - Trust values are securely accessed and transmitted
  - No malicious attacks on peers giving reviews

- **Reliability**
  - Querying peer gets the correct reply in spite of presence of malicious peers

- **Accountability**
  - Way to hold a peer accountable for its feedback
Trust Based P2P Systems - Review

- Various trust models are available
- Scarce work on access protocols
- Polling based protocol – Cornelli et al
  - Every peer before interacting with another peer broadcasts a trust query for that peer
  - All peers that have interacted with that peer send their votes which are combined locally
  - Public Key Cryptography used to secure
Issues with current approaches

- No persistence
  - Users not currently logged on cannot participate
  - Extremely prone to simple malicious group activity
- Tedious decision making
  - Require to wait for all replies and confirmations
- No anonymity
  - Peers giving reviews cannot remain anonymous
  - Fear of retaliation and external attacks
TrustMe

- Persistent - uses global trust values
  - All peers after interacting, review and file a report
  - All reviews combined to give a single value based on the trust model used
  - The trust values are hosted at another peer
    - Each peer has a Trust Holding Agent (THA) Peer

- Secure and Anonymous
  - Complete anonymity to both the querying peer and THA peer

- Fast decision making
  - A single reply message is enough to make a decision
TrustMe - Phases

- Query
  - Broadcast a *trust query* for another peer (say Peer A)

- Reply
  - Peer A’s THA peer replies with its trust value

- Interaction
  - If trustworthy, querying peer interacts with Peer A and collects proof-of-interaction

- Report
  - The querying peer reviews Peer A’s performance and files a report
TrustMe – Infrastructure I

- **Secure Bootstrap Server (BS)**
  - Entry point for peers to enter the network
  - Acts as a *kind* of certification authority – helps only in pseudo-identification of peers
  - Possesses a private-public key pair \(<P_{BS}, B_{BS}>\)
  - \(B_{BS}\) is publicly available to all peers – network parameter

- **Each Peer**
  - Possesses two pairs of private-public keys \(<P_i, B_i>\) and \(<P'_i, B'_i>\)
  - BS assigned ID: \(BID_i = P_{BS}(“Valid Node” | B'_i)\)

- **BS maintains a list of active peers**
Peer Join – When Peer i joins the network

- Bootstrap server needs to assign a THA peer (say Peer x)
  - Chooses a peer randomly from the list of active peers
  - Creates a new private-public key pair < SP<sub>i</sub>, SB<sub>i</sub> >
    - Only the THA peer will have the knowledge of SP<sub>i</sub>
    - Used for secure transmission of trust values for the reply and the report phase
  - Securely transmits < ID<sub>i</sub>, B<sub>i</sub>, SP<sub>i</sub>, SB<sub>i</sub> > to Peer x
    - Broadcast a message of the format
      \[
      \text{BID}_x | P_{BS}(\text{BID}_x | B'_x(\text{ID}_i | B_i | SP_i | SB_i))
      \]
    - Only BS can generate and only Peer x can read
TrustMe – Query & Reply

- Peer typically will have a list of offering peers
- Message Format:
  \[ ID_i | ID_j | ID_k | ID_l | \ldots \]
- For any peer \( i \) being queried, its THA peer should reply with its trust value. Need to ensure
  - Reply can only be sent by the THA peer
  - Reaches destination un-tampered
- Message Format:
  \[ ID_i | B_i | SB_i | SP_i(TV | TS | BID_x | P'_x(TS)) \]
**TrustMe – Reply**

ID_i | B_i | SB_i | SP_i(TV | TS | BID_x | P'_x(TS))

- All peers can read it
- Only THA peer can send it (encryption with SP_i)
- Cannot be replayed (use of timestamp TS)
- Use of BID_x ensures accountability
  - Can be used to identify malicious THA peers
- Use of P'_x(TS) ensures nobody can use somebody else’s BID_x
- B_i and SB_i are used by querying peer
TrustMe – Interaction

- No peer can file a report without interacting
  - Prevents malicious report-filing
- If Peer i and Peer j interact, they exchange messages
- Peer j gets $P_i(TS \mid B_j \mid ID_j)$ and Peer i gets $P_j(TS \mid B_i \mid ID_i)$
  - Prevents replay
  - Cannot be generated in a fake manner
  - Ensures only the correct peer can file a report
TrustMe – Report

- Need to make sure that only the THA peer can read the report (*secret ballot*)

- Only a peer that actually interacted with Peer i can file a report for Peer i

- Message Format:

  \[
  \text{ID}_i | \text{SB}_i(\text{“Report”} | V | \text{B}_j | P_j(P_i(\text{TS} | \text{B}_j | \text{ID}_j)))
  \]

  
  Proof-of-interaction

- THA peer updates rating by Peer j and updates TV
TrustMe – Analysis I

- Manipulating Reply Messages
  \[ \text{ID}_i | \text{B}_i | \text{SB}_i | \text{SP}_i(TV | TS | BID_x | P'_x(TS)) \]

  - Malicious THA peer
    - Maintain K THA peers and select based on majority vote
    - Blacklist malicious THA peers based on BID_x
  
  - Malicious non-THA peers
    - Offering peers include their B_i and SB_i as part of the initial offer

- Manipulating Proof-of-interaction - \( P_i(TS | B_j | \text{ID}_j) \)
  - Using fake keys
    - Easily verifiable
TrustMe – Analysis II

- Why use two pairs \(<P_i, B_i>\) and \(<P'_i, B'_i>\)
  - \(<P'_i, B'_i>\) used only while acting as a THA peer
  - Prevents mapping of public key to identifier after prolonged monitoring of the network

- Peer Leave – Whenever Peer \(i\) leaves the network
  - Create a new THA peer for peers it was responsible for
  - Its trust information is dumped after it is not accessed for some time
TrustMe – Benefits

- Security and Anonymity
- Reliability
- Accountability
- Persistence
- Fast decision time
- Ease of contribution
  - A single report is required
TrustMe – Experimental Results I

Effect of Persistence

- Non-persistent systems can report highly misleading values
- Having as little as 10 malicious peers acting together can rate the peer being untrustworthy, even when it is not
TrustMe – Experimental Results II

- Cost and Response Times

- TrustMe costs more because of more broadcasts
- Cost varies little with increase in number of THA peers
- Caching improves response times
- Increase in number of THA peers also improves response times
Conclusions and Future Work

- Anonymous trust management possible
- TrustMe provides secure and reliable access to trust values in a decentralized P2P system
- Compatible with existing Gnutella style systems

- Use symmetric key based broadcast authentication protocols like TESLA
- Design variants with different levels of anonymity and security
Thanks