

# MANTIS

An anonymity preserving,  
searchable, P2P network.

Christopher Soghoian, Steve Bono  
Johns Hopkins University

# The RIAA Attacks!

- The RIAA has been filing lawsuits left and right – against sharers of 1000+ files.
- This is having a chilling effect.
- People either stop trading files (hah!), or switch to leech mode (more likely).
- Networks cannot survive if everyone leeches.

# You're helping illegal filesharers?

- There are several non-infringing uses of our technology.
- Those stuck behind the great firewall of china.
- Dissident radio networks.
- Diebold source-code sharers.
- Country Music Fans (explain).

# Pre-requisites

- For a P2P network to survive, it has to meet the following requirements:
  - 1. Sharers must remain anonymous at all times.
  - 2. It must be searchable.

# Pre-req's continued...

- 3. It must not be excessively expensive (CPU/Bandwidth). Note that this threshold is different for clients/servers/peers.
- 4. Peers are extremely sensitive to bandwidth usage (they're selfish bastards).
- 5. It must be fast.

# Initial Designs/MUTE

- We originally came up with an idea very similar to MUTE ([mute-net.sourceforge.net](http://mute-net.sourceforge.net)).
- Simply put, MUTE insures that your search requests/responses and file transfers all get sent through other clients on the network before going to you. However, still check MUTE out.

# Mute continued..

- This gives you plausible deniability, as you can just claim to be forwarding traffic for another host.
- This design has a few problems....

# MUTE Problems

- .Avi files are large.
- Peers are selfish.
- Transferring a large movie through several peers will annoy them, and take ages to complete. Your download speed will be that of the slowest peer on the link.

# Mantis - Nodes

- Mantis nodes have 3 simultaneous roles: client, server and peer.
- Each node maintains connections to 5+ other nodes.
- Node discovery can either involve a directory server, a web page of some kind, or a napkin.

# Mantis - searches

- When a node wishes to search for a file/service, it passes that search request on to its neighbors.
- They in turn pass it on to everyone they know..
- This repeats for a while.. (~ 5 hops), until nodes start dropping the search request (probabilistic dropping vs. TTL).

# Mantis - searches

- When a node receives a search request, and it indeed has the file, it sends a response back up the tree.
- Search requests/responses carry unique ID's, which all nodes use to keep track of where to forward search replies, and other communications.

# Mantis - Connections

- Using the combination of the uniquely generated search and response ID's, both parties are able to establish a connection over the link.
- No node knows the final destination of a message. Just the next hop. In theory, that next hop could be the destination. But he'll deny it.

# File Transfers

- A client has searched for something, gotten a few interesting responses, and now wishes to download one of the files.
- The client shares his IP and a listening port with the server.

# And now for the fun bit!

- The server now sends the file to the client over a direct UDP channel to the client (i.e not passed through all the other nodes).
- He spoofs his source address when doing so.
- Control data (to arrange for retransmission of lost packets, etc) is sent over the anonymous back channel of other nodes.

# The fun bit continued.

- Middlemen nodes only have to pass search requests/responses and control data.
- 2 DSL speed hosts can communicate through a sea of modem users, and still transfer a file at high speed.
- The server remains anonymous.

# Issues?

- Q: Can't middlemen learn of the client's IP/port.
- A: Not usually. Communication between nodes is encrypted at every hop, with an additional layer of encryption between client and server.

# Issues

- A middle-man can man in the middle the connection, but there is nothing we can do to stop this. PKI's don't work when everyone is anonymous.
- We don't aim to protect the client, we just try to make it a bit difficult to learn his identity.

# Issues

- Q: Are you subject to Timing Attacks?
- A: We hope not. Nodes use random-ish delays before returning data, which allows them to claim that they are just forwarding a message for someone else, and are not the originator.

# Issues?

- Q: If ISP's start egress filtering, and don't allow spoofed packets out, aren't you SOL?
- A: Good Question!
- Sliding Egress detection protocol.
- Servers can choose how much of their ip they are willing to make public.

# Egress issues

- With solid egress, you can't pretend to be an AOL user, but you can at least hide amongst the other users on your ISP's network.
- If the RIAA forces ISP's to implement widespread egress filtering, we all win anyway, as DDoS will be made much harder.