P2P Filesharing Population Tracking Based on Network Flow Data

Arno Wagner, Thomas Dübendorfer, Lukas Hämmerle, Bernhard Plattner

Contact: arno@wagner.name

Communication Systems Laboratory

Swiss Federal Institute of Technology Zurich (ETH Zurich)

Outline

1. Motivation
2. Setting
3. The PeerTracker
4. Validation by Polling
5. Comments on Legal Aspects
6. Conclusion

Contributors

- Philipp Jardas, ”P2P Filesharing Systems: Real World NetFlow Traffic Characterization”, Barchelor Thesis, ETH Zürich, 2004
- Lukas Hämmerle, ”P2P Population Tracking and Traffic Characterization of Current P2P file-sharing Systems”, Master Thesis, ETH Zürich, 2004
- Roger Kaspar, ”P2P File-sharing Traffic Identification Method Validation and Verification”, Semester Thesis, ETH Zürich, 2005

PDFs available from
`http://www.tik.ee.ethz.ch/~ddosvax/sada/`

Motivation

- P2P traffic forms a large and dynamic part of the overall network traffic
- Identification of P2P traffic allows P2P anomaly detection
- Identification of P2P traffic allows better analysis of other traffic

The DDoSVax Project

`http://www.tik.ee.ethz.ch/~ddosvax/`

- Collaboration between SWITCH (www.switch.ch, AS559) and ETH Zurich (www.ethz.ch)
- Aim (long-term): Near real-time analysis and countermeasures for DDoS-Attacks and Internet Worms
- Start: Begin of 2003
- Funded by SWITCH and the Swiss National Science Foundation

DDoSVax Data Source: SWITCH

The Swiss Academic And Research Network

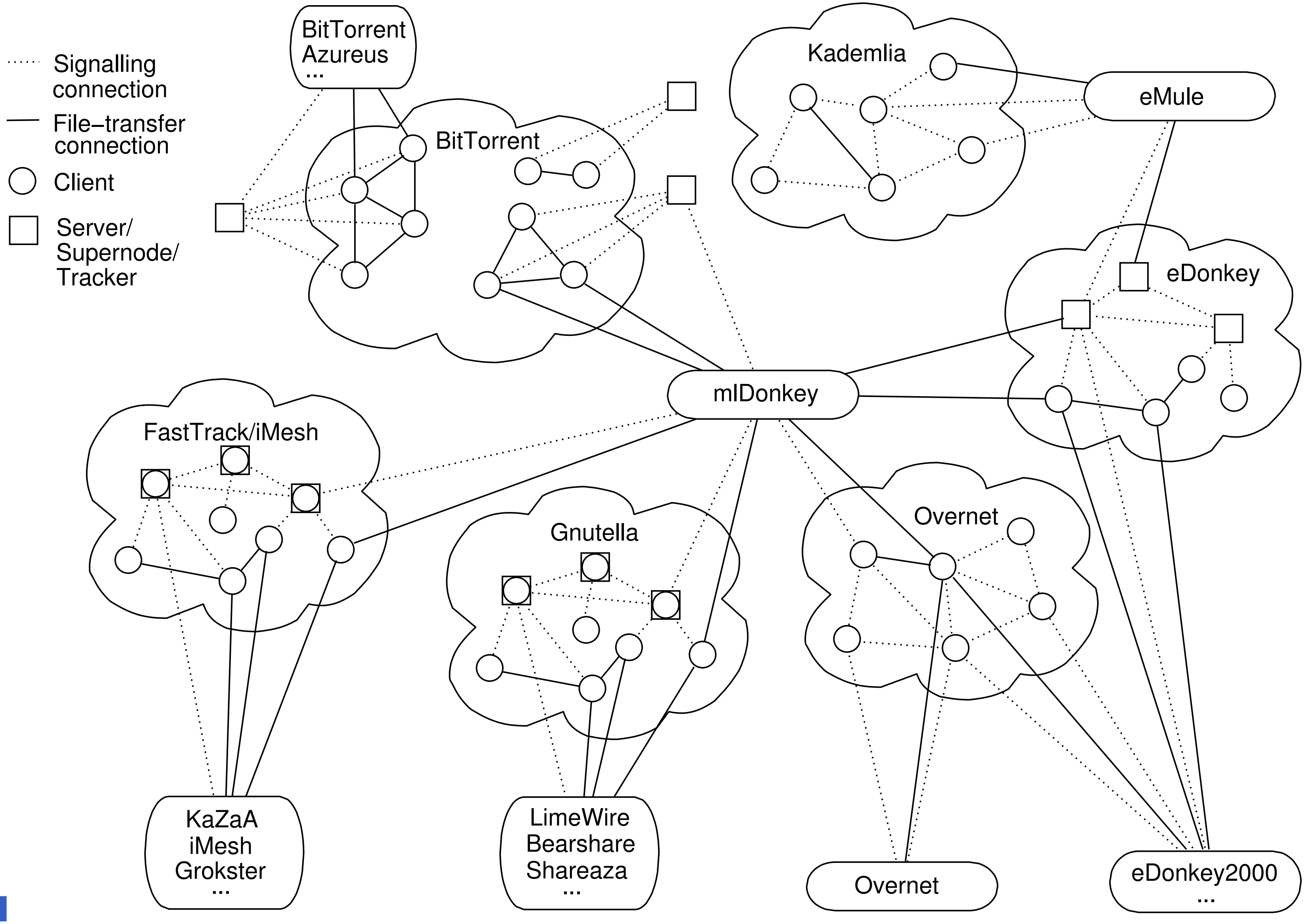
- .ch Registrar
- Links most Swiss Universities and CERN
- Carried around 5% of all Swiss Internet traffic in 2003
- Around 60.000.000 flows/hour
- Around 300GB traffic/hour
- Flow archive (unsampled) since May 2003

What is a ”Flow”?

In the DDoSVax-Project: CISCO NetFlow v5.
Basically unidirectional aggregated header information:

- Source and destination IP
- Protocol
- Source and destination port (TCP and UDP)
- Packet and byte count
- Start and end time stamp, 1ms accuracy
- Some routing information
- No payload information whatsoever

P2P Networks Considered



⋯ Signalling connection

— File–transfer connection

○ Client

□ Server/ Supernode/ Tracker

BitTorrent
Azureus
...

BitTorrent

Kademlia

eMule

eDonkey

mlDonkey

FastTrack/iMesh

Gnutella

Overnet

KaZaA
iMesh
Grokster
...

LimeWire
Bearshare
Shareaza
...

Overnet

eDonkey2000
...

PeerTracker Algorithm

Idea: ”Traverse” P2P network from seeds

Seeds:

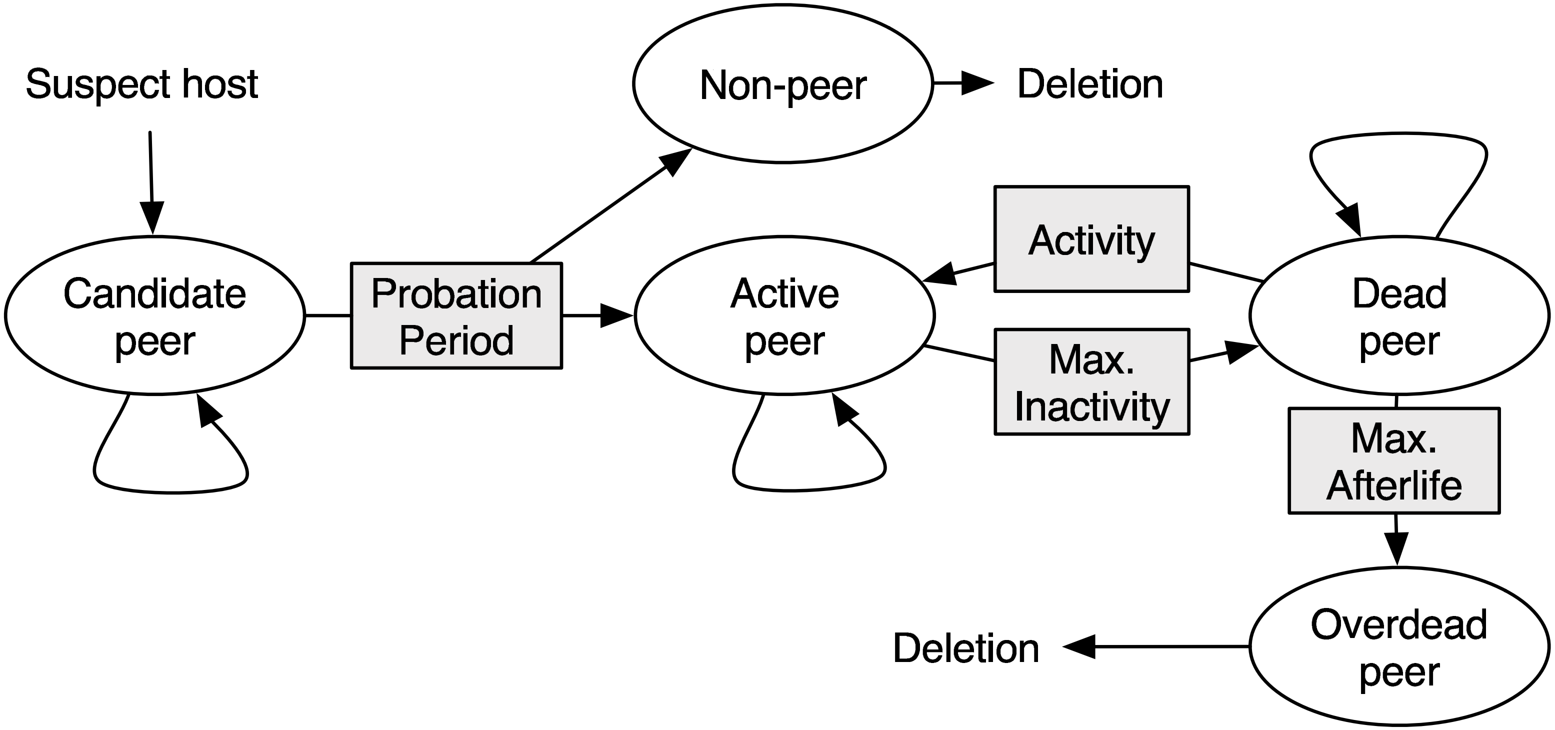
- Seeds are peers using default ports (TCP and UDP)
- Keep a pool of peers for each network
- Add hosts that communicate with the pool
- Remove hosts that are idle

Default Port Usage

P2P System	Default port usage (TCP)
BitTorrent	70.0 %
FastTrack	8.3 %
Gnutella	58.6 %
eDonkey	55.6 %
Overnet	83.9 %
Kademlia	66.6 %

From 8 day PeerTracker measurement end of 2004

PeerTracker: Hosts State Diagram



Some PeerTracker Details

- Peers in the SWITCH network form ”core”
- External hosts only identified if they contact core
- Different ageing for core and external hosts
- Dead core peers are still contacted from external peers

Validation

The PeerTracker does not look at Payloads
⇒ Large error possible

Validation Approach:

- Run PeerTracker
- Poll found peers immediately
- Compare polling and tracker results

Peer Polling Methods

P2P System	Polling method
FastTrack	Request: `GET /.files HTTP/1.0` Response: `HTTP 1.0 403 Forbidden` <number 1> <number 2> or `HTTP/1.0 404 Not Found/nX-Kazaa-`<username>
Gnutella	Request: `GNUTELLA CONNECT/`<version> Response: `Gnutella` <status>
eDonkey, Kademlia, Overnet	Request: Binary: 0xE3 <length> 0x01 0x10 <MD4 hash> <ID> <port> Response: Binary: 0xE3 . . .
eMule	Same as eDonkey, but replace initial byte with 0xC5.
BitTorrent	Unsolved. Seems to need knowledge of a shared file on the target peer.

Polling Results

P2P System	TCP Connect	P2P-client found
eDonkey, Kademlia, Overnet	50%	41%
Gnutella	53%	30%
FastTrack	51%	41%
Total	51%	38%

Table 1: Positive polling answers

Polling Remarks

- Delay to polling < 10 Minutes
- About 50% unreachable via TCP and port seen by PeerTracker ⇒ likely behind NAT
- Other errors: Peer variation (esp. Gnutella), classification into wrong network, tracker error
- BitTorrent not really pollable

Legal Aspects

(Warning: I am no legal expert)

- Flow data likely not subject to privacy laws (unless attempts to identify people are made)
- PeerTracker does not identify content shared ⇒ output unproblematic, no action needs to be taken
- Identification of heavy hitters unproblematic, since payloads (shared contents) not identified
- Polling is unproblematic, since similar to running a peer (some users get nervous though...)

PeerTracker for Law Enforcement

Situation for CH!

- Massive private file-sharing is done
- Law enforcement is not really interested in copyright infringement
- Illegal contents (child pornography and the like) can be done far better with modified P2P clients

$\Rightarrow$ Not really suitable

Conclusion, Remarks

- Peer identification feasible with flow data
- Nodes with little traffic problematic
- BitTorrent problematic
- P2P filesharing still evolves
- PeerTracker code is available under GPL
- In productive use at ETH Zurich since 2005

Thank You!