SWINBURNE

CENTRE FOR ADVANCED INTERNET ARCHITECTURES

Bittorrent traffic classification

Tung Le, July 2009

Outline

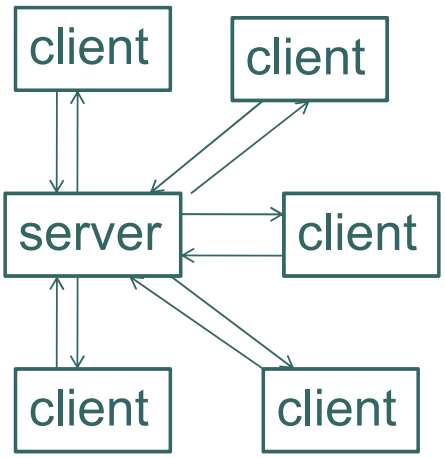
- Bittorrent and FTP overview
- Methodology
- Attribute analyses
- Results
- Limitations
- Conclusion

Purpose

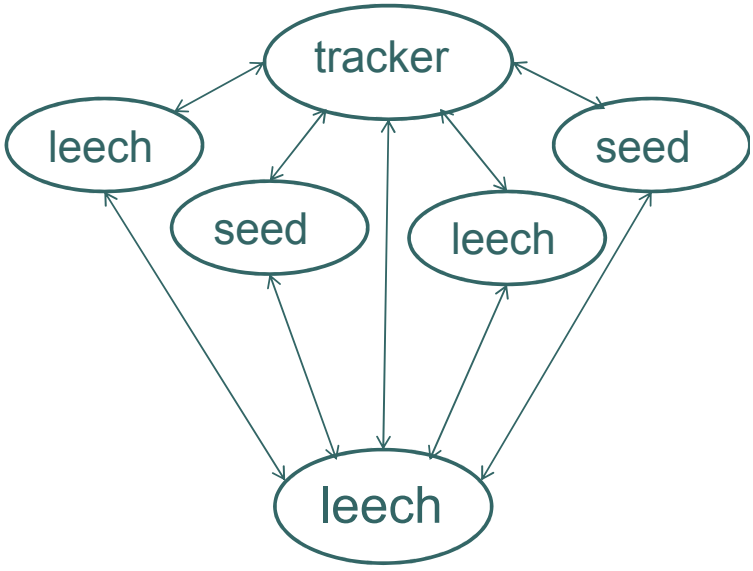
- Project aims
  - Classify Bittorrent traffic
- Why?
  - Network management
  - Lawful interception
- Potential problems
  - Is Bittorrent “similar” to FTP?
- Machine learning based approach

Bittorrent and FTP overview

- FTP



Bittorrent



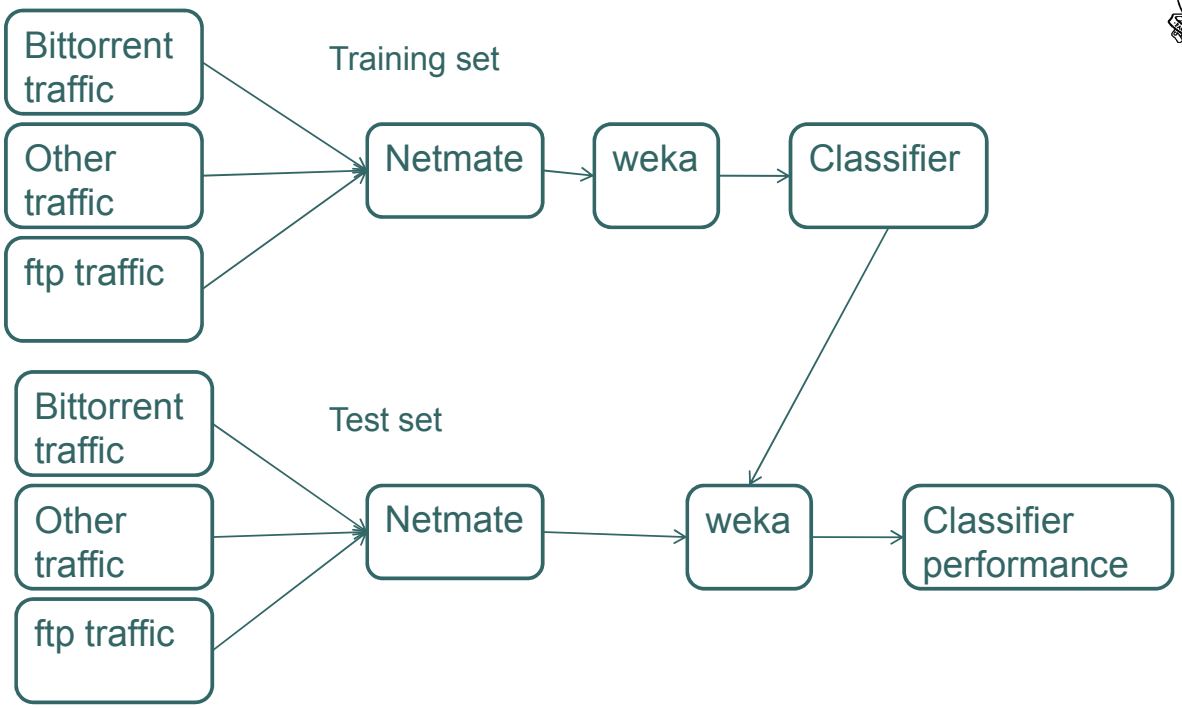
Classification issues

- Difficult to classify Bittorrent
  - Random port number
  - The protocol changes
- What about
  - Deep packets inspection
  - Legal issues when read data of packets

Statistical approach

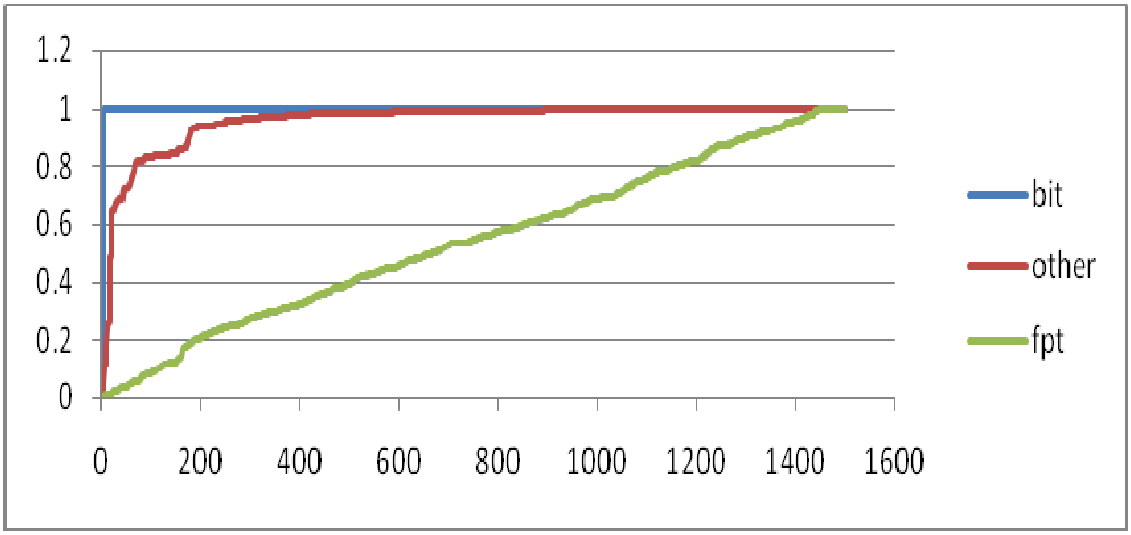
- Group packets into flows
- Calculate per-flow statistics
  - Eg Number of packets, maximum packet length, duration, active time…
- Use machine learning techniques to recognize Bittorrent
- Focus on packet length – why?

Methodology



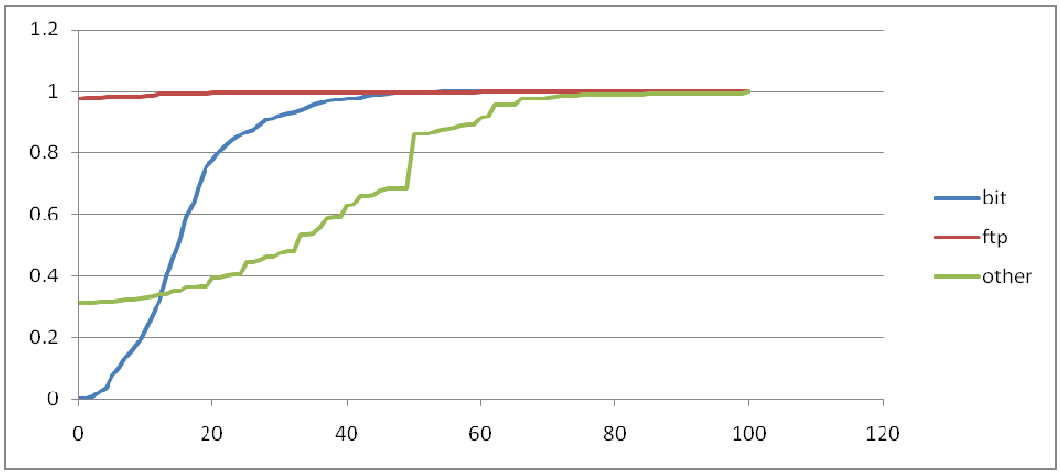
Attribute analyses

- Minimum payload



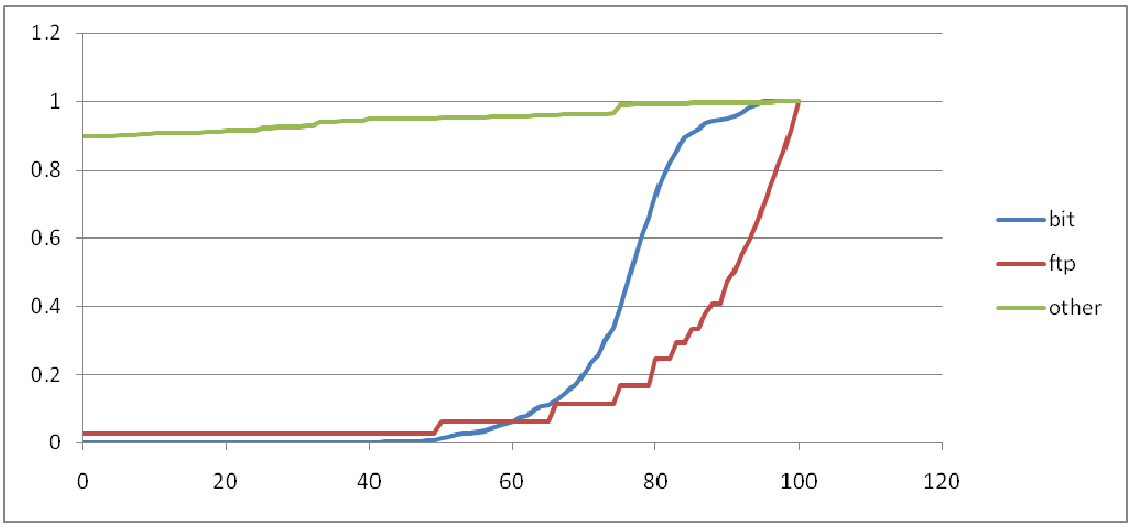
Attribute analyses

- Ratio of small packets



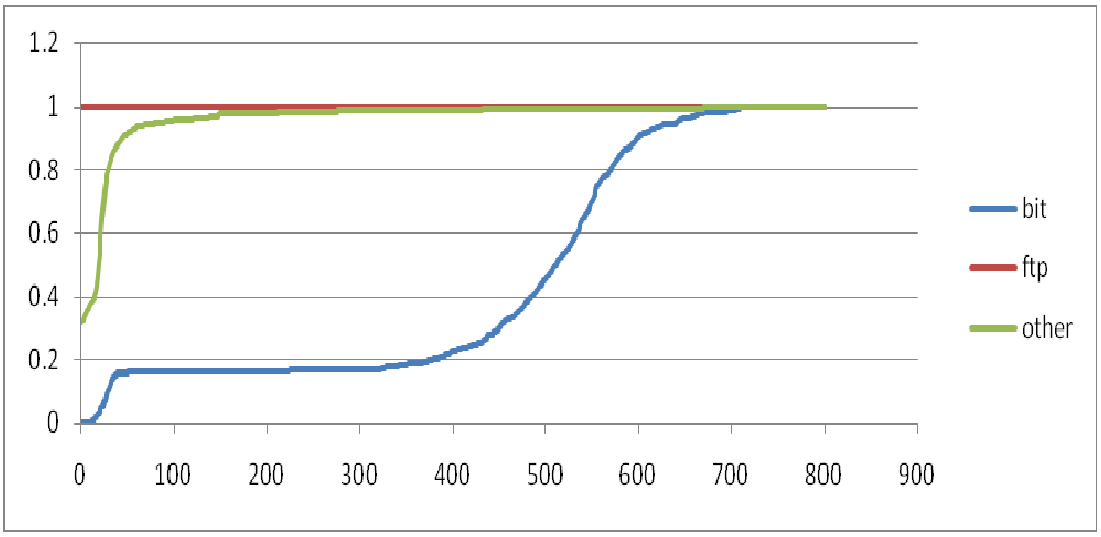
Attribute analyses

- Ratio of big packets



Attribute analyses

- Smaller Standard deviation payload



Bitorrent vs ftp

- Both Bittorrent and ftp have big packets
  - ftp has big packets in one direction
  - Bittorrent has big packets in both directions
- Unique packet length
  - Bittorrent interest packets and unchoke packets have fixed payload 5 bytes
- Bittorrent uses same port to control and send data
  - ftp uses different port

Results

- Single feature results

Attribute name	Other	Bitorrent	Ftp
Min pld	96.6	100	99.5
Small ratio	95.4	94.8	98.2
Big ratio	96.9	58.9	98.2
std	98.1	68.5	100

Results

- Multiple feature results

Attribute name	Other	Bittorrent	Ftp
Min pld, big ratio	99.6	100	99.7
Big ratio, std	98.6	99.6	100
Min pld, small ratio, std	99.5	96.8	100
Min pld, small ratio, big ratio	99.6	100	99.7
Small ratio, big ratio, std	99.5	98.4	100
Min pld, small ratio, big ratio, std	99.7	100	100

Limitations

- Need a lot of packets in each flow
  - At least 50 packets in each direction
- The classifier was not tested with the Bittorrent flows between seed and leech
- May only works with TCP Bittorrent
  - µtorrent actually uses Bittorrent over UDP

Future work

- Classify on subflows
- Add more features?

Conclusion

- Bittorrent traffic can be classified effectively based on payload statistics of TCP packets
  - Minimum payload
  - Small packets ratio
  - Big packets ratio
  - Std payload
- Some limitations
  - Need a lot of packets each flow
  - Bittorrent between seed and leech
  - Bittorrent over UDP

Thanks to

- Grenville
- Jason
- Philip
- Amiel
- Sebastian
- Mattia