Performance Comparison of low-latency Anonymisation Services from a User Perspective

Rolf Wendolsky Dominik Herrmann Hannes Federrath

Department of Business Informatics
University of Regensburg

7th Workshop on Privacy Enhancing Technologies, 2007

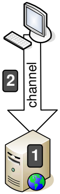
Outline

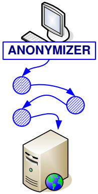
Background and Motivation
- Mix-based Anonymisation Services
- Importance of Performance Evaluation

Obtaining the Sample
- Analysed Metrics and Scenarios
- Automated Performance Assessment

Results
- Performance of Tor and AN.ON
- Influence of Load on Performance

Anonymisation Services Protect the Sender's Privacy by Relaying Traffic Multiple Times





- Purpose
  1) protect users' privacy (at least their IP address) from destination server
  2) prevent service providers from establishing relationship between sender and receiver (traffic analysis)
- Idea
  - users run anonymiser software (acts as proxy server)
  - anonymiser software relays traffic over multiple hops
- But: Relaying causes bottlenecks (performance impact)

Structural Differences Between AN.ON and Tor Might Implicate their Performance

AN.ON	**Tor**
static *mix cascades*	dynamically constructed *circuits*
user selects cascade	client constructs circuits
well-known operators	voluntary node operators
high bandwidth nodes	low and high bandwidth nodes
$\approx$ 10 mix cascades	$\approx$ 1000 onion routers
supports HTTP(S) only	supports various TCP protocols

http://www.anon-online.de/

(http://www.jondos.de)

http://tor.eff.org

Performance is a Critical Feature for Users and Developers

Users

- mostly cannot judge security of anonymisers
- see usability and performance as key features
- tend to avoid slow anonymisers

Anonymisers

- would like to attract as many users as possible
- have to be tuned for high performance

Evaluation allows for assessment of tuning measures

Results might uncover inherent characteristics unknown so far

Questions Answered in this Presentation

- Users are interested in
  - Which service should I use for downloading large files?
  - Which service offers fastest web surfing?
- Developers are interested in
  - How is performance affected by user load?
  - How much is performance affected by structural differences?
  - What performance should we aim for?

Questions Answered in this Presentation

- Users are interested in
  - Which service should I use for downloading large files?
    Tor
  - Which service offers fastest web surfing?
- Developers are interested in
  - How is performance affected by user load?
  - How much is performance affected by structural differences?
  - What performance should we aim for?

Questions Answered in this Presentation

- Users are interested in
  - Which service should I use for downloading large files?
    Tor
  - Which service offers fastest web surfing?
    AN.ON (at least in Europe)
- Developers are interested in
  - How is performance affected by user load?
  - How much is performance affected by structural differences?
  - What performance should we aim for?

Questions Answered in this Presentation

- Users are interested in
  - Which service should I use for downloading large files?
    Tor
  - Which service offers fastest web surfing?
    AN.ON (at least in Europe)
- Developers are interested in
  - How is performance affected by user load?
    heavily
  - How much is performance affected by structural differences?
  - What performance should we aim for?

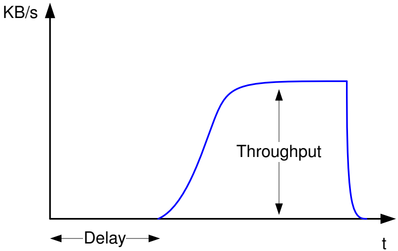
Questions Answered in this Presentation

- Users are interested in
  - Which service should I use for downloading large files?
    Tor
  - Which service offers fastest web surfing?
    AN.ON (at least in Europe)
- Developers are interested in
  - How is performance affected by user load?
    heavily
  - How much is performance affected by structural differences?
    not so much as you would think
  - What performance should we aim for?

Questions Answered in this Presentation

- Users are interested in
  - Which service should I use for downloading large files?
    Tor
  - Which service offers fastest web surfing?
    AN.ON (at least in Europe)
- Developers are interested in
  - How is performance affected by user load?
    heavily
  - How much is performance affected by structural differences?
    not so much as you would think
  - What performance should we aim for?
    latency below 4 seconds

Analysed Metrics for Performance Evaluation



Relevant performance data:

- Network latency (delay)
- Network bandwidth (throughput)

Desirable:
Number of concurrent users to estimate load of services

Measured in small intervals over long period of time

Evaluated Systems and Scenarios

HTTP performance measured for 4 systems:

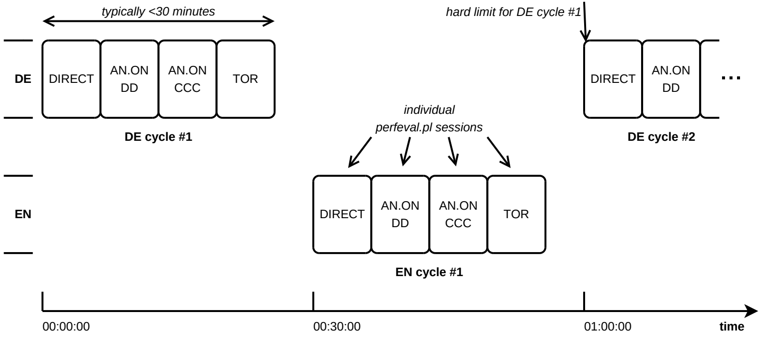
DIRECT	Direct download as benchmark
DD	AN.ON cascade #1 (default cascade)
CCC	AN.ON cascade #2 (has to be selected manually)
TOR	Tor client with Privoxy local proxy server

For each system 4 distinct scenarios were evaluated:

- Usage pattern: web surfing (WEB) and downloads (DL)
- Region: URLs from Germany (DE) and US (EN)

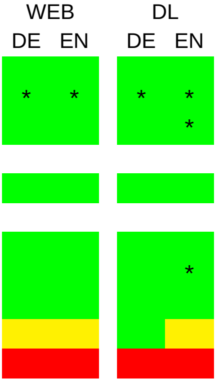
Automated Performance Assessment

- Simulation of a web browser with Perl (ParallelUA)
- 258 hourly tests over a period of over 10 days



Our Analyses Indicate Overall High Data Quality

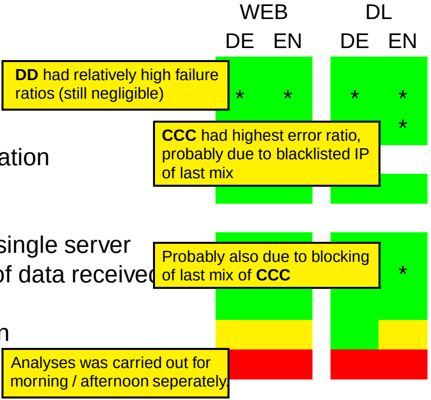
- External factors
  - Network downtimes
  - Service failures
  - Server errors
- Bias introduced by observation
  - Overlapping tests
- Performance fluctuations
  - Evaluation biased by single server
  - Comparable amount of data received
- HTTP redirects
- Limited internet connection
- Dependancy on daytime



- No influence expected (green)
- Small influence assumed (yellow)
- Non-negligible influence expected (red)

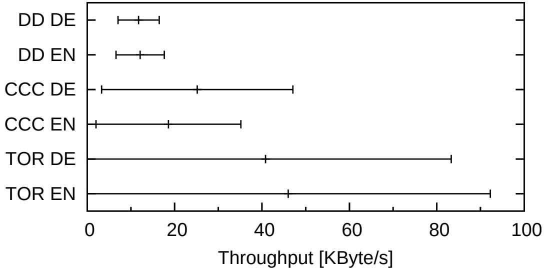
Our Analyses Indicate Overall High Data Quality

- External factors
  - Network downtimes
  - Service failures
  - Server errors
- Bias introduced by observation
  - Overlapping tests
- Performance fluctuations
  - Evaluation biased by single server
  - Comparable amount of data received
- HTTP redirects
- Limited internet connection
- Dependancy on daytime



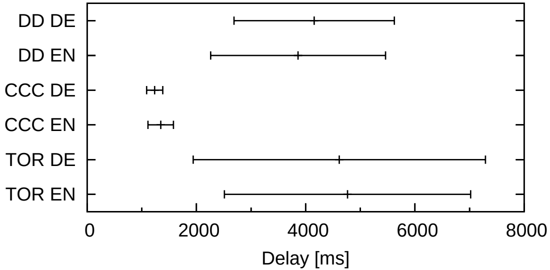
No influence expected
Small influence assumed
Non-negligible influence expected

Offered Bandwidth: Tor Outperforms AN.ON



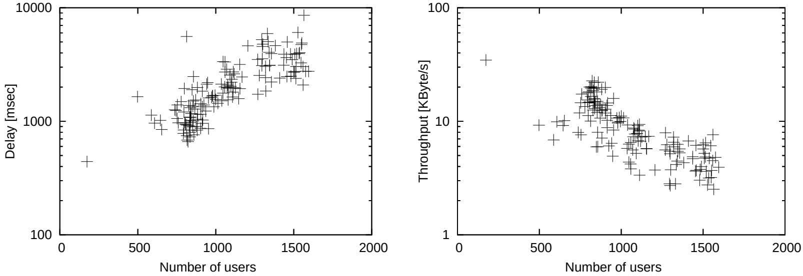
- Lowest throughput: DD (up to 1,700 concurrent users), slightly better: CCC (650 users on avg.)
- Tor with *significantly* more bandwidth
- But: Tor's performance subject to considerable fluctuations

Network Latency: AN.ON Responds Faster



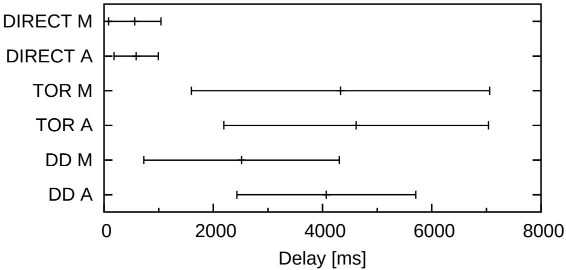
- CCC: lowest latency and very constant quality of service
- Tor and DD with similarly high latencies
- DD/CCC offer *significantly* lower delays than Tor
- All in all, CCC offers best web surfing performance

Significant Correlation Between Load and Performance (observed on DD)



- Regression analysis: *significant* exponential relationship
- Results indicate a large inactive user base on DD cascade
- Accordingly, connected users are no robust measure for anonymity provided; should refer to active users instead!

Intraday Performance Fluctuations Resulting from Different Loads



- Majority of AN.ON users from Europe (number of connected users follows sinusoidal curve)
- Thus, fluctuations on AN.ON due to varying loads
- But Tor also affected (world-wide distributed user base!)

Intraday Performance Fluctuations Resulting from Different Loads (cont.)

Tor: *Significantly* lower delays / higher throughputs during night time (averages differ by 500 milliseconds)

- Is the user base not distributed over the world at all times?
- Are low-latency (= geographically nearby?) nodes preferred for building circuits?
  - might have implications for anonymity (simplifies collusion attack)
  - but: no such node selection strategy in source code
- Currently, no satisfactory explanation available, more data points needed!

Empirically Derived Tolerance Level for Latency

- AN.ON and Tor with similar average delays of 4 seconds
- Users deterred from using the services during times of higher latencies
- Suggestion: 4 seconds as empirically determined tolerance level for low-latency anonymisation systems

Implications for scaling

- Anonymisation services taking up as many users as they can carry
- Tor scales incrementally as more nodes are added (often)
- AN.ON scales by setting up new cascades (seldomly)

Suggestions for Developers

- AN.ON
  - Set up new cascades or upgrade bandwith of existing ones
  - Count only *active users* in client's anonymeter as a better measure for anonymity
- Tor
  - Encourage users to enable *concurrent connections* and *pipelining* in browser to reduce perceived latency
  - Supply estimation of currently connected users for assessment of impact of load on performance and security

Summary

- Performance is critical feature for users – may also have security implications for anonymisation services
- Structural characteristics of the services have (small) impacts on different performance aspects
- Performance primarily affected by load – i.e., services just have to scale to increase performance

- Outlook
  - Perform extended study for long-term analysis
  - Look into Tor’s intraday performance fluctuations