Breaking Tor Anonymity

Victor Farbman and Maxim Trosman
Under guidance of Amichai Shulman
Background – TOR

- How it works

Diagram:

- Alice
  - Step 1: Alice's Tor client obtains a list of Tor nodes from a directory server.
  - Dave
  - Jane
  - Bob
Background – TOR

Step 2: Alice's Tor client picks a random path to destination server. Green links are encrypted, red links are in the clear.
Background – TOR

Step 3: If at a later time, the user visits another site, Alice’s Tor client selects a second random path. Again, green links are encrypted, red links are in the clear.
Low-Cost Traffic Analysis of Tor
- by Steven J. Murdoch and George Danezis.
- In the Proceedings of the 2005 IEEE Symposium on Security and Privacy, May 2005

Locating Hidden Servers
- by Lasse Øverlier and Paul Syverson.

Low-Resource Routing Attacks Against Tor
- by Kevin Bauer, Damon McCoy, Dirk Grunwald, Tadayoshi Kohno, and Douglas Sicker.
To break the anonymity of the Tor user that accesses our server, using modest means.
We don’t allow the user to access our server without a kind of a “handshake”, during which we break his anonymity.

Rationale: after an indication of user’s malicious behavior we want to detect who he is by breaking his anonymity.
Our Solution

- **Assumptions:**
  - The Entry Node in the Tor circuit is compromised and under our control.
  - The Server is under our control.

- **Used Programs:**
  - **C#** (first implementation of servers)
  - **Python** (simple implementation of servers)
  - **WireShark** (network activity tracing)
  - Etc.
Our solution is based on analysis of browsing traffic between TOR client and server.

The analysis is done on TOR circuit’s Entry Node.
Our Solution - Network Model

User → Compromised Entry Node → Middle → Exit → Our Server
Our Solution

- *Time based approach*

The server responds to user's requests by intentionally delaying its response packets in a predetermined pattern.
Our Solution – Time Based Example

- **Time Sequence Pattern:**
  \{ 1.5, 3, 2, 5 \}

- About 83% matching.

- False positive 25 times.
HTTP Refresh based approach

The server responds to user's requests with data containing an http meta refresh tag.


Our Solution – HTTP Refresh Based Example

- Refresh Each 2 sec.
- 9 Times.
- About 59% matching.
- False positive 13 times
According to received results we can see that the time sequence solution provides the best match rate. In order to increase it we need to research further in the following directions:

- Dependency between the length and complexity of the time sequence and the match rate.
- Influence of quality of connection on the match rate.
- We can improve our results by using smart trace analysis (Heuristics, Machine Learning, Neural Networks Learning).
Protection

- Random small time delay in data transmission on each circuit’s node.
- Integrated detection of time/packet patterns.
- Aggregation/breaking of data in bigger/smaller packets.
- User side solution: don’t route a single session to a single server through a specific Tor circuit
Conclusions

- Proposed ways to break TOR user anonymity do not require significant computational or network overhead, but can provide useful results.

- There is a room for improvement and refinement on our methods by working along aforementioned lines.
Thank you for listening!

Questions?