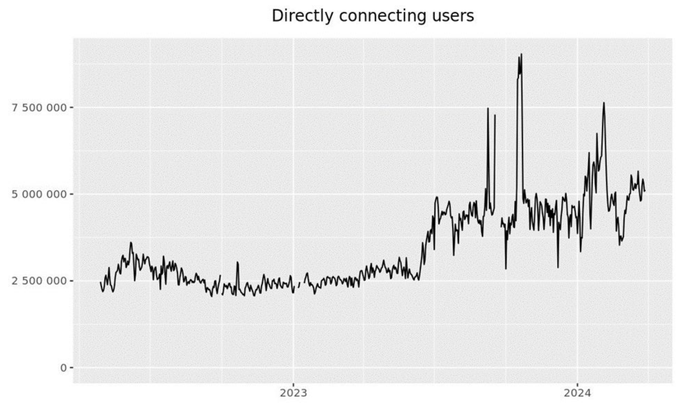
Abstract of Tor User De-Anonymization: Client-Side Originating Watermark with LSTM Autoencoder Detection

Daniel Brown & Natalija Vlajic

Department of Electrical Engineering and Computer Science, York University, Toronto, Canada

INTRODUCTION

Tor has historically been one of the most popular anonymity browsers on the market, with signs of a steady growth in user population. Unfortunately, Tor's services have also attracted the attention of **cyber criminals** who seek anonymity while performing various **illegal actions** online.



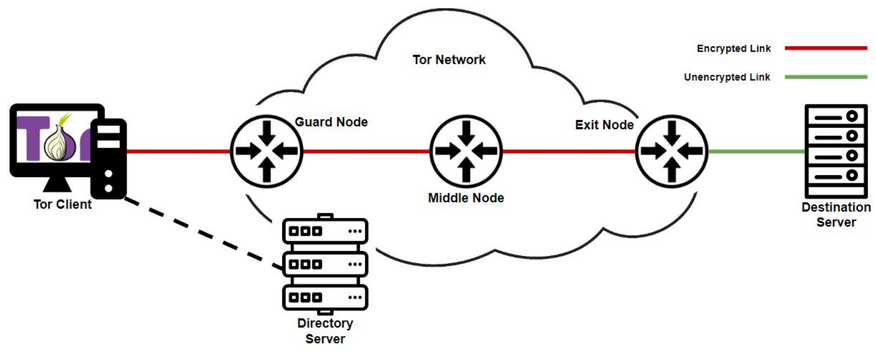
Source: Tor Metrics

Traditional traffic-watermark techniques for Tor user de-anonymization are generally implemented through utilization of **server-side originating watermark (SSOW)**. However, the effectiveness of **SSOW** is often hindered by significant amounts of traffic noise that accumulates along Tor's communication pathways. In this paper, we outline the performance and key ideas behind our novel user de-anonymization technique that utilizes **client-side originating watermark (CSOW)**.

TOR TOPOLOGY

Tor is an overlay network that is built on top of the **Transmission Control Protocol (TCP)** and utilizes multiple layers of cryptographic protection.

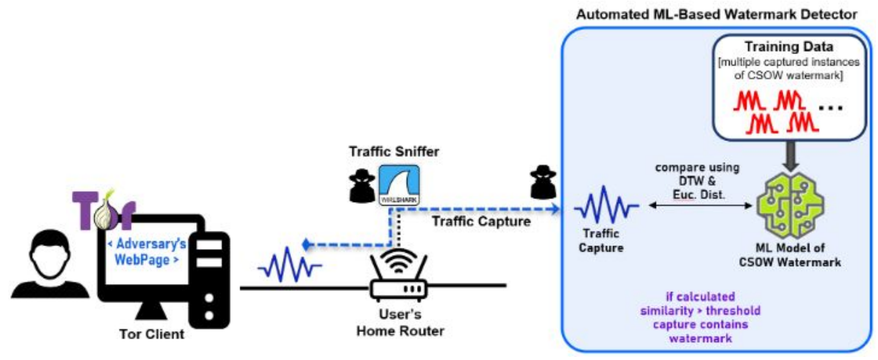
Tor creates circuits from a **user/proxy** to a **destination server**, which in most cases consists of two **relay nodes** and one **guard node**.



LSTM-BASED CSOW

For simplicity purposes, we originally assumed that extracting the **CSOW** from one single traffic capture will provide accurate watermark detection and user de-anonymization results (given it does not pass the Tor network). However, we do recognize that in some real-world situations, the Original **CSOW** itself could potentially be impacted by traffic noise.

Therefore, for us to build a truly effective and robust **CSOW** user de-anonymization framework, it would be necessary to incorporate into this framework an automated module capable of building a well-established noise-resistant watermark 'profile' from (not only one but) a number of observed watermarks. Hence, we introduced a **Long Short-Term Memory (LSTM) Autoencoder** to our approach.



LSTM Autoencoder is trained on "normal" traffic (watermark) pattern only. Subsequently, the trained LSTM model is presented both normal and anomaly test data. This allows us to evaluate how well the model is able to both detect the **CSOW** in a randomly presented traffic instance, as well as to differentiate the learned **CSOW** profile from other webpages' traffic.

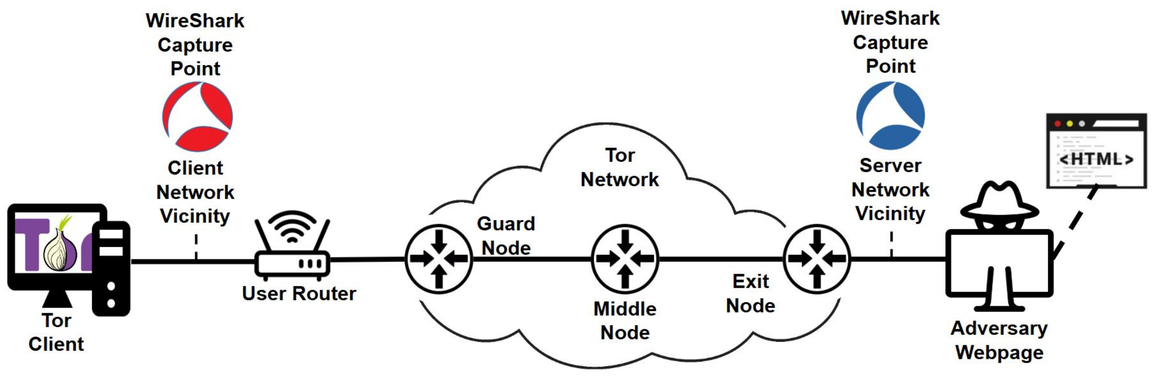
Our **LSTM CSOW model** is able to achieve an accuracy of **93%** (correctly detecting our watermark out of **200 samples**) with **0 false-positives** versus Torben's **91%** accuracy (out of **100 samples**) with **0 false-positives**, both models assume **ideal conditions**.

SSOW Confusion Matrix	
91/100 **(91%)**	9/100 **(9%)**
0/100 **(0%)**	0/100 **(0%)**

CSOW Confusion Matrix	
185/200 **(93%)**	15/200 **(7%)**
0/200 **(0%)**	0/200 **(0%)**

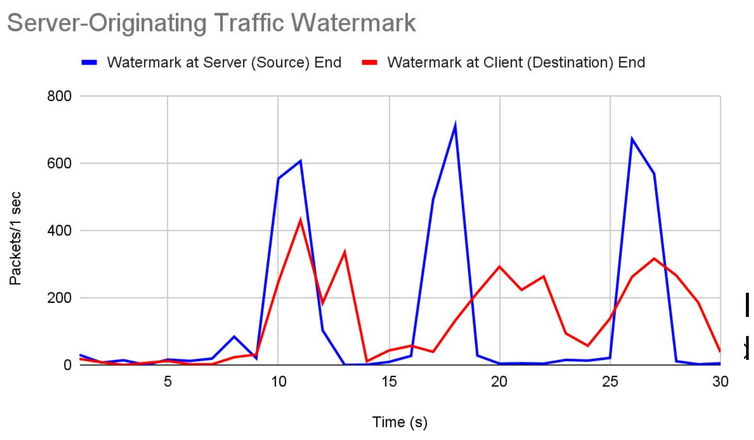
SERVER-SIDE WATERMARK PERFORMANCE

Server-Side Originating Watermarks (SSOW) often experiences significance levels of traffic noise when passing through the **Tor network** in order to reach the the **watermark detector** in the **Client Network Vicinity (CNV)**.



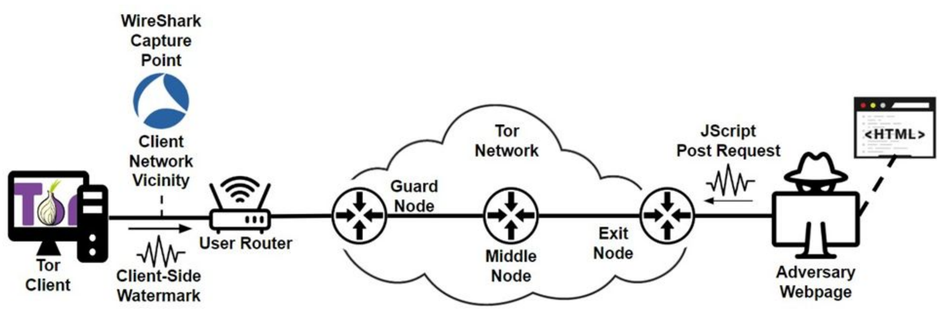
The chart below shows two captures, in **blue** we observe the **intended watermark** as seen in the **Server Network Vicinity (SNV)**. In **red** we observe the same 3-spike watermark when captured in the **Client Network Vicinity (CNV).**

The detrimental impact of the network noise on the look of this watermark is very evident, ultimately distorting watermark recognition.



CLIENT-SIDE WATERMARK PERFORMANCE

The advantage of our novel **Client-Side Originating Watermark (CSOW)**, is that using **JavaScript Post Requests** we are able to inverse the flow of the watermark. This allows us to generate a minimally distorted watermark within **CNV** as we are able to accurately detect it before it passes the **Tor network**.



Not only does this method solve the issue from **Tor** induced traffic noise, but also creates far more robust and consistent traffic spikes compared to the watermarks of the **SSOW approach**.