Detection of obfuscated Tor traffic with unsupervised neural networks

An MPhil project proposal

Chongyang Shi (*cs940*), Christ's College

Project Supervisor: TBC

Abstract

Censorship-circumventing Tor network traffic can be disguised as regular TLS-encrypted traffic by the pluggable transport tool meek *[1], utilising the* domain fronting *technique. This proposed project seeks to develop a model for identifying such obfuscated Tor traffic from regular network traffic with the use of unsupervised neural networks. In comparison with existing detection techniques featuring supervised machine learning [2, Sec. 6], unsupervised neural networks have the potential of achieving faster and more adaptive detection capabilities. The resulting detection model could contribute to improvement of meek's counter-classification capabilities.*

1 Introduction, approach and outcomes

Tor is a popular tool for anonymised and censorship-resistant network communications. While it is trivial for a network node in a privileged position to detect and block non-obfuscated Tor traffic [3, Tb. 6] in a process called *traffic classification*, Tor provides a set of *pluggable transport* tools which clients can use to conceal their connections to a Tor bridge node from such censors. An arms race between state-sponsored censors and pluggable transport developers in traffic obfuscation has been going on for many years [4].

Among pluggable transports, two classes of techniques currently exist to achieve obfuscation of the encrypted traffic: pseudo-random transformation and fronting of other "legitimate" protocols. Techniques in the former class attempt to avoid traffic classification by transforming Tor traffic into pseudo-random data, while those in the latter class transform Tor traffic into the likes of other protocols that will result in too much collateral damage for the censor to block. A number of tools have been developed in each class and deployed with Tor distributions, with pseudo-random transformation represented by ScrambleSuit [5] and Format-Transforming Encryption (FTE) [6], and fronting represented by meek [1].

Obfuscation techniques can generally be evaluated on two metrics: the transmission performance after obfuscation, and the indistinguishability of obfuscated traffic in the eye of a censor. For the purpose of censorship-circumvention, interests are usually concentrated on the latter. There has been a number of independent distinguishability evaluations on the aforementioned tools [2] [7] [8], utilising both entropy-based analysis and machine learning (ML) analysis on timing and features of obfuscated packets. Consensus reached by past studies suggest that fronting techniques represented by meek performs significantly better than pseudo-random transformation techniques against entropy-based attacks. However, meek can still be vulnerably to ML-based attacks, as demonstrated by Wang *et. al.* [2, Sec. 6].

Therefore, the primary objective of the proposed project is to apply unsupervised neural network learning to attempt to identify meek-obfuscated Tor traffic from regular encrypted HTTPS traffic produced by the TLS protocol, which now accounts for more than half of all web traffic [9]. While supervised machine learning through trained decision tree classifiers have been utilised effectively by Wang *et. al.* [2, Tb. 8] with a high true positive rate, the supervised classifiers only worked well when trained and tested on traces from the same network environment, as Dixon *et al.* [10] pointed out, due to the inevitable overfitting caused by the singular source of training data. The use of unsupervised neural networks will allow the classifier to adapt to changing network environments more easily, permitting potential application on a greater scale. Unsupervised neural networks may also be less computationally expensive when compared with supervised classifiers, as they will not require a target dataset in training.

The proposed project intends to apply two well-studied artificial neural networks designed for pattern recognition purposes: the self-organising map (SOM) [11] and the adaptive resonance theory (ART) network [12]. Various existing implementations of these neural networks are available for research use, and new implementations can also be developed. Whatever the implementation used, the goal is to achieve comparable or better detection performance than the aforementioned studies; which is to say, achieving a high true-positive rate (identifying obfuscated traffic traces) and a low false-positive rate (not misidentifying non-obfuscated traffic as obfuscated), both of which are desirable to a censor.

A secondary objective is to adapt a functional implementation for the primary objective to run on a piece of specialised machine learning hardware such as Google's TensorFlow, in order to achieve very high levels of performance in detecting obfuscated traffic normally unachievable on conventional computer systems. While the notable past effort by Wang *et. al.* in ML-based detection showed it being not as efficient as entropy-based attacks [2, Sec. 8], successful completion of this secondary objective will demonstrate the possibilities of performing machine learning-based detection of meek-obfuscated traffic in real-time or near-real-time.

Traffic traces used in this proposed study could be sourced from anonymised internet traces available for research purposes, injected with self-generated meek traffic traces, as conducted by Wang *et. al.* [2, Sec. 3]. Alternatively, for traffic traces more resembling real network conditions, human volunteers can be invited to browse the internet in a monitored environment where meek is in use on some of the network clients, subject to ethical review approval.

2 Workplan (500 words)

Project students have approximately 28 weeks between the submission of the proposal, and the submission of the dissertation. Essay students have approximately 14 weeks. This section should account for what you intend to do during that time. One approach would be to divide the time into two-week chunks, and describe the work to be done (and, as relevant, milestones to be achieved) in each chunk. You should leave one chunk for writing an essay or two chunks for writing a project dissertation. You should leave 1 chunk for contingencies.

References

[1] D. Fifield *et al.*, "Blocking-resistant communication through domain fronting," *Proceedings on Privacy Enhancing Technologies*, vol. 2015, no. 2, pp. 46–64, 2015.

[2] L. Wang *et al.*, "Seeing through network-protocol obfuscation," in *Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security*. ACM, 2015, pp. 57–69.

[3] T. Bujlow *et al.*, "Independent comparison of popular dpi tools for traffic classification," *Computer Networks*, vol. 76, pp. 75–89, 2015.

[4] S. Khattak *et al.*, "Systemization of pluggable transports for censorship resistance," *arXiv:1412.7448*, 2014.

[5] P. Winter *et al.*, "Scramblesuit: A polymorphic network protocol to circumvent censorship," in *Proceedings of the 12th ACM workshop on Workshop on privacy in the electronic society*. ACM, 2013, pp. 213–224.

[6] K. P. Dyer *et al.*, "Protocol misidentification made easy with format-transforming encryption," in *Proceedings of the 2013 ACM SIGSAC conference on Computer & communications security*. ACM, 2013, pp. 61–72.

[7] A. Houmansadr *et al.*, "The parrot is dead: Observing unobservable network communications," in *Security and Privacy (SP), 2013 IEEE Symposium on*. IEEE, 2013, pp. 65–79.

[8] Q. Tan *et al.*, "Towards measuring unobservability in anonymous communication systems," *Journal of Computer Research and Development*, vol. 52, no. 10, p. 2373, 2015.

[9] G. Gebhart, "We're halfway to encrypting the entire web," 2017. [Online]. Available: https://www.eff.org/deeplinks/2017/02/were-halfway-encrypting-entire-web

[10] L. Dixon *et al.*, "Network traffic obfuscation and automated internet censorship," *IEEE Security & Privacy*, vol. 14, no. 6, pp. 43–53, 2016.

[11] T. Kohonen, "The self-organizing map," *Neurocomputing*, vol. 21, no. 1, pp. 1–6, 1998.

[12] G. A. Carpenter and S. Grossberg, "The art of adaptive pattern recognition by a self-organizing neural network," *Computer*, vol. 21, no. 3, pp. 77–88, 1988.