Exploring RNNs for Analyzing Zeek HTTP Data

Daniel Andrews
daniel.andrews2@westpoint.edu
United States Military Academy
West Point, NY

Jennifer Behn
jennifer.behn@westpoint.edu
United States Military Academy
West Point, NY

Danielle Jaksha
danielle.jaksha@westpoint.edu
United States Military Academy
West Point, NY

Jinwon Seo
jinwon.seo@westpoint.edu
United States Military Academy
West Point, NY

Madeleine Schneider
madeleine.schneider@westpoint.edu
United States Military Academy
West Point, NY

James Yoon
james.yoon@westpoint.edu
United States Military Academy
West Point, NY

Suzanne J. Matthews
suzanne.matthews@westpoint.edu
United States Military Academy
West Point, NY

Rajeev Agrawal*
rajeev.k.agrawal@erdc.dren.mil
U.S. Army E.R.D.C.
Vicksburg, MS

Alexander S. Mentis*
alexander.mentis@westpoint.edu
United States Military Academy
West Point, NY

ABSTRACT

Cyber vulnerabilities pose a threat across systems in the Department of Defense. Finding ways to analyze network traffic and detect malicious behavior on a network will help keep these systems safe. This poster looks at the data collection techniques, model creation, and results of building a recurrent neural network to classify incoming traffic as normal or malicious. Additionally, it considers how the information will be best portrayed on a GUI to network administrators. The model's initial accuracy is 83.45% when trained on 500,017 connections. With increased accuracy, this tool may be used by the Department of Defense to help defend its networks.

CCS CONCEPTS

• **Security and privacy → Intrusion/anomaly detection and malware mitigation**.

KEYWORDS

Recurrent Neural Networks, Zeek, Bro

ACM Reference Format:
Daniel Andrews, Jennifer Behn, Danielle Jaksha, Jinwon Seo, Madeleine Schneider, James Yoon, Suzanne J. Matthews, Rajeev Agrawal, and Alexander S. Mentis. 2019. Exploring RNNs for Analyzing Zeek HTTP Data. In *Hot Topics in the Science of Security Symposium (HotSoS), April 1–3, 2019, Nashville, TN, USA.* ACM, New York, NY, USA, 2 pages. https://doi.org/10.1145/3314058.3317291

1 INTRODUCTION

Across Department of Defense (DoD) networks, immense amounts of network traffic is logged daily. Specifically, the Defense Research Engineering Network (DREN) provides robust, high-capacity, and low-latency connectivity between DoD Supercomputing Resource Centers (DSRCs) and user sites. The DREN also supports both the DoD's scientific research & development and test & evaluation missions. The Cybersecurity Environment for Detection, Analysis, and Reporting (CEDAR) contains a variety of cybersecurity sensors, which constantly monitor and record real time network activity on the DREN. Unusual activities on the network generate alerts, which are manually examined by human analysts and annotated as malicious or normal. In this poster, we discuss some preliminary work in applying deep learning techniques to automatically classify alerts as malicious.

Most researchers that have explored the application of deep learning techniques in the cybersecurity domain have either used the KDDCup'99 dataset or its variant NSL-KDD [7]. For example, Appruzese et al [2], compare the usefulness of deep learning and shallow learning for malware detection. While their results show shallow learning outperforming deep learning in their testing environment, we note that their work employs a Feedforward Neural Network (FNN). In contrast, we are studying the use of a Recurrent Neural Network (RNN). The Self-taught Learning (STL) [5] project, based on a sparse auto-encoder and soft-max regression, employs the NSL-KDD dataset and converts it into 2-class (Normal and Anomaly), 5-class (Normal and 4 different attack categories) and 23-class (Normal and 22 different attack categories) output. Their results indicate an F-measure value up to 96%. Prior work by CEDAR [6] explored the use of a simple 3-layer neural network on HTTP, DNS, and connection data. Other prior work [4] explores the use of regexes to analyze Bro HTTP logs for malicious uniform resource identifiers (URIs). The work discussed in this poster focuses on the application of an RNN to HTTP data gathered from Zeek (previously named Bro) sensor logs.

2 METHODS

Figure 1 gives an overview of our proposed system. The graphic user interface (GUI) consists of a "Start/Stop" button that will initiate the network intrusion detection model, which is written in Keras[3] and TensorFlow[1]. Once the process is triggered, the model runs

*corresponding authors

Permission to make digital or hard copies of part or all of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for third-party components of this work must be honored. For all other uses, contact the owner/author(s).
HotSoS, April 1–3, 2019, Nashville, TN, USA
© 2019 Copyright held by the owner/author(s).
ACM ISBN 978-1-4503-7147-6/19/04.
https://doi.org/10.1145/3314058.3317291

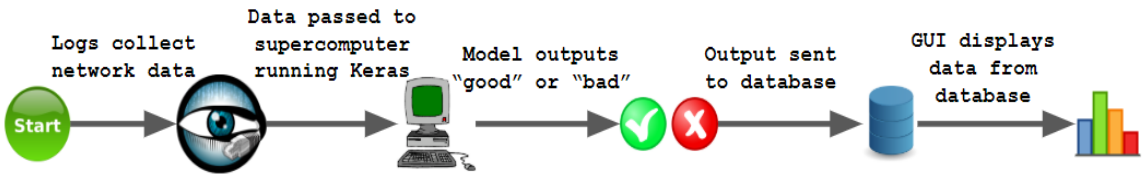


Figure 1: Overview of our automatic classification system

on a SQL database that is constantly querying new data from the HPC Architecture for Cyber Situational Awareness (HACSAW) API managed by CEDAR. HACSAW is responsible for gathering Zeek and other sensor log information from the DREN. The data comes in from HACSAW, is analyzed by the model, and is inputted into the SQL database. If a network alert is detected, it is classified as malicious or normal, and then shown on the web page.

With the assistance of CEDAR operators, we selected 11 features from our HTTP log data to use in our models. These selected features relate to the size, origin, and packet type of each log event. We hypothesize that features relating to the size of the entry (Request body length, and Response body length) may show some correlation between large data transfers and malicious activity. We use features related to the origin (Country of origin, User agent string, and Cookie) because they may indicate traits of users that pose a threat. Features related to the type of packet involved (URI type, Request data type, and HTTP verb) can reveal information about the client's action. The remaining features (URI length, Severity, and Destination port) do not fall into those three categories but were identified by operators as being potentially useful. The severity feature is a machine-generated feature that guesses how severe an interaction was, but is not always accurate. One-hot encoding for categorical features yielded a final set of 25 features.

3 EXPERIMENTAL SETUP & PRELIMINARY RESULTS

We used the DoD supercomputer, Topaz, for data collection, training, and testing. Using the HACSAW API, we gathered 1,461,238 labeled Zeek log events. By nature, the raw log data has a disproportionately large amount of normal data. We preprocessed the dataset to balance the ration of malicious to normal events by selecting a random subset and sorting by time. The data was further manipulated by scaling quantitative features and one-hot encoding categorical features. This yielded a final dataset of 625,022 events. We used 90% of the data for training, and 10% for testing.

We ran experimentation against two models. The first is a multilayer perceptron (MLP), one of the most basic feed forward neural networks. Our second, and more accurate model was a RNN. A unique feature of RNNs is their ability to "remember" information that they have been given previously. As malicious actions don't tend to occur with only one connection, this would give the ability to see patterns in connections and more accurately spot malicious activity. The current model has two long short term memory (LSTM) layers, three dropout layers, and two dense layers. The first LSTM layer has 25 nodes, corresponding to the 25 features. The final dense layer is two nodes for the two possible classes. The other internal layers were determined as a result of experimentation. Our current results are shown in Table 1.

Table 1: Preliminary Results

Model	Accuracy	Precision	Recall
MLP	77.06%	25.69%	24.35%
RNN	83.45%	47.58%	40.21%

The RNN currently yields an accuracy of 83.45% compared to the MLP's accuracy of 77.06%. We will continue to experiment with different RNN model structures, and explore smarter ways to utilize the benefits of RNNs and time series data by grouping blocks of alerts to train on.

ACKNOWLEDGMENTS

We are grateful to the DoD High Performance Computing Modernization Program (HPCMP) for a grant of time on DoD supercomputer Topaz. The opinions expressed in this work are solely of the authors and do not reflect those of the U.S. Military Academy, the U.S. Army, or the Department of Defense.

REFERENCES

[1] Martín Abadi, Ashish Agarwal, Paul Barham, Eugene Brevdo, Zhifeng Chen, Craig Citro, Greg S. Corrado, Andy Davis, Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Ian Goodfellow, Andrew Harp, Geoffrey Irving, Michael Isard, Yangqing Jia, Rafal Jozefowicz, Lukasz Kaiser, Manjunath Kudlur, Josh Levenberg, Dan Mané, Rajat Monga, Sherry Moore, Derek Murray, Chris Olah, Mike Schuster, Jonathon Shlens, Benoit Steiner, Ilya Sutskever, Kunal Talwar, Paul Tucker, Vincent Vanhoucke, Vijay Vasudevan, Fernanda Viégas, Oriol Vinyals, Pete Warden, Martin Wattenberg, Martin Wicke, Yuan Yu, and Xiaoqiang Zheng. 2015. TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems. http://tensorflow.org/ Software available from tensorflow.org.

[2] Ferretti Guide Marchetti Apruzzese, Colajanni. 2018. On the effectiveness of machine and deep learning for cyber security. https://ieeexplore.ieee.org/abstract/document/8405026

[3] François Chollet et al. 2015. Keras. https://keras.io.

[4] S. Deaton, D. Brownfield, L. Kosta, Z. Zhu, and S. J. Matthews. 2017. Real-time regex matching with apache spark. In *2017 IEEE High Performance Extreme Computing Conference (HPEC)*. 1–6. https://doi.org/10.1109/HPEC.2017.8091063

[5] Ahmad Javaid, Quamar Niyaz, Weiqing Sun, and Mansoor Alam. 2016. A Deep Learning Approach for Network Intrusion Detection System. In *Proceedings of the 9th EAI International Conference on Bio-inspired Information and Communications Technologies (Formerly BIONETICS) (BICT'15)*. ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering), ICST, Brussels, Belgium, Belgium, 21–26. https://doi.org/10.4108/eai.3-12-2015.2262516

[6] C. Lorenzen, R. Agrawal, and J. King. 2018. Determining Viability of Deep Learning on Cybersecurity Log Analytics. In *2018 IEEE International Conference on Big Data (Big Data)*. 4806–4811. https://doi.org/10.1109/BigData.2018.8622165

[7] M. Tavallaee, E. Bagheri, W. Lu, and A. A. Ghorbani. 2009. A detailed analysis of the KDD CUP 99 data set. In *2009 IEEE Symposium on Computational Intelligence for Security and Defense Applications*. 1–6. https://doi.org/10.1109/CISDA.2009.5356528