NOracle: Who is communicating with whom in my network?

Patrick Kalmbach
Technical Unviersity of Munich
Munich, Germany

David Hock
Infosim GmbH & Co. KG
Würzburg, Germany

Fabian Lipp
Infosim GmbH & Co. KG
Würzburg, Germany

Wolfgang Kellerer
Technical Unviersity of Munich
Munich, Germany

Andreas Blenk
Technical Unviersity of Munich
Munich, Germany

ABSTRACT

This demo presents `NOracle`: a system using Stochastic Block Models (SBMs) to infer structural roles of hosts and communication patterns of services in networks. `NOracle` can be used with existing monitoring systems to analyze and visualize networks in an online manner or be used to analyze stored traces. Network operators can use SBMs to monitor and verify network operation, detect possible security issues and change-points. To showcase this, `NOracle` combines the production-grade network management solution `StableNet` with an SBM based anomaly detection and network visualization module. `StableNet` provides network flow statistics in real-time from actual devices. The SBM extracts roles and communication patterns live from the data provided by `StableNet`. The result can help to reason about communication behaviors, detect anomalous hosts and indicate changes in the large scale-structure of network communication.

KEYWORDS

Anomaly Detection, Stochastic Block Model, Network Monitoring

1 INTRODUCTION

Answering the questions of "who is communicating with whom in my network" can help operating today's and future self-driving networks in many directions. For instance, knowing the communication pattern of applications in data centers helps improving resource management systems, e.g., speeding up the completion times of distributed data processing applications. In particular, data driven resource management systems for placement and embedding tasks like [1, 2, 15] can use communication patterns as basis for their predictions, and thus help networks to run themselves.

Futerhmore, inferring communication patterns can help detecting security holes, e.g., infected hosts being part of a botnet [3, 6, 14]. A better understanding of the communication behaviors of users and services is crucial to make networks self-driving and thus inevitable for future communication paradigms such as application-aware networking needed for low-latency networks, such as 5G.

SIGCOMM '19, August 19–23, 2019, Beijing, China

ACM ISBN 978-1-4503-6886-5/19/08…$15.00
https://doi.org/10.1145/3342280.3342303

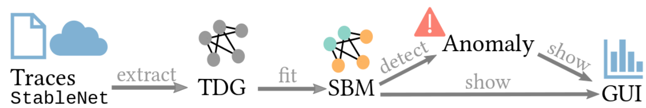


Figure 1: System diagram of `NOracle`. `NOracle` extracts TDGs from different sources (online and offline analysis), sequentially fits SBMs to the data, checks the data for anomalies, and visualizes the result.

Existing solutions often rely on prior knowledge, require unencrypted network traffic, significant computational resources and time or cannot be easily interpreted by technical staff [3, 14]. Such approaches neither work live and ad-hoc (i.e., without a prior information base) nor will they perform efficiently in the future due to the encryption of network traffic and increasing network sizes. However, efficient pattern extraction is a requirement to enable data driven algorithms that unlock the full potential of Software Defined Networking and Network Function Virtualization [10], as well as novel technologies such as re-configurable physical layers [8].

In this demo, we present `NOracle`: A system that analyzes and visualizes network traffic based on Probabilistic Graphical Models (PGMs), fitted to network monitoring data in real-time. PGMs are used in robotics to model complex systems in a principled and understandable fashion [11]. We apply a specific class of PGMs, the so-called Stochastic Block Models (SBMs) [9], in a network scenario: a machine will autonomously learn structural roles and the communication pattern of a network and use this information to detect anomalies. Since SBMs work in an unsupervised fashion, they do not rely on any prior knowledge such as port-to-service mappings and do not required labelled data. SBMs can efficiently be estimated and thus allow `NOracle` to operate online. In addition, `NOracle` relies only on packet header information.

Fig. 1 illustrates our approach: `NOracle` extracts Traffic Dispersion Graphs (TDGs)[1] using IP and TCP header information. A SBM is fitted to each TDG and passed to an anomaly detector. The result of SBM and detector is visualized on a web-based interface illustrated in Fig. 2 and Fig. 3. Modeling traffic as TDG allows `NOracle` to explicitly model and exploit relational data between hosts.

2 `NOracle`: A DATA-DRIVEN APPROACH

`NOracle` uses Stochastic Block Models (SBMs) [9] to separate hosts into meaningful groups. A SBM is a PGM that represents a parametric probability distribution over graphs [9]. The model encodes

[1]In a TDG nodes correspond to IP addresses and edges to flows/communication between these addresses.

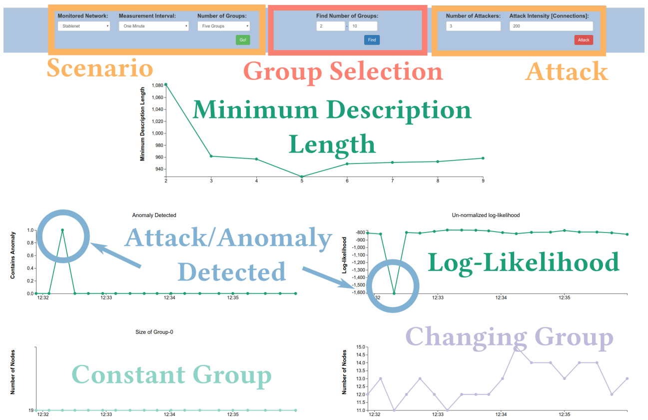
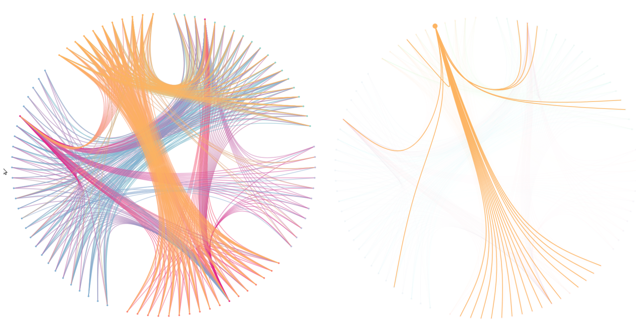


Figure 2: Different time series obtained with `NOracle`, indicating the presence of an anomaly, the goodness of fit of the underlying model and the group sizes.



(a) Group Structure **(b) Node Communication**

Figure 3: Large scale structure of monitored network and communication of one individual node.

high-level relations, details are filled-in by estimating model parameters from data. SBMs have already proven their potential in generating synthetic IP-to-IP communication for simulations [5].

The SBM has three different types of parameters: the number of communication groups k, the node-to-group assignment z, and the number of expected edges $\theta_{r,s}$ between two groups r and s. The probability of a graph G with nodes $\mathcal{V}$ and edges $\mathcal{E}$ is then:

$$P(G \mid \theta, z) = \prod_{i<j} \mathrm{Poi}_{\theta_{z_i, z_j}}(A_{i,j}) \prod_{i} \mathrm{Poi}_{\theta_{z_i, z_i}}(A_{i,i}).$$

A is the adjacency matrix and $A_{i,j}$ the number of edges between nodes i and j. The procedure of `NOracle` is then as follows: given a TDG G created with data from `StableNet` or other sources, `NOracle` uses Maximum Likelihood to find z and θ. Parameter k is provided by the user or can be estimated from data using the Minimum Description Length (MDL) principle [13] in our demo. We represent TDGs as unweighted graphs, i.e., $A_{i,j} \in \{0, 1\}$. In the future, we plan to extend `NOracle` to include edge weights as shown in [7] and node metadata [12] to boost model accuracy.

While the model can work completely unsupervised, human knowledge can still improve the overall system performance drastically. For instance, by roughly knowing the services inside a network, a system administrator can help to faster bootstrap the system or choose a more suitable value for k. We will showcase both examples in our demo.

3 DEMO

The demo presents how `NOracle` can (1) infer the communication structure of applications and (2) based on this information detect anomalous hosts, i.e., hosts infected with malware or generally with a suspicious communication pattern.

Scenario. The demo considers three scenarios: (1) synthetic graphs with known structure, (2) a campus network with more than 5 000 hosts and (3) an enterprise network with more than 100 hosts. For all scenarios, network traffic, i.e., the packet level traces or netflow data are fed into `NOracle`. The data can be evaluated for different parameter settings. For instance, the demo shows how network hosts are grouped for different k, exposing structural roles of nodes (e.g., client vs. server). Moreover, the demo shows how live grouping of hosts can help to detect hosts with abnormal behavior, e.g., hosts which suddenly change their communication pattern when infected with malware.

Network data. For (1) we use synthetic data with planted groups. This data is generated using a SBM with pre-set parameters. *The demonstration shows how known structural roles can be identified in an completly unsupervised fashion.*

For (2), the demo uses the publicly available data set "CTU13 Corpus 9". The data set contains the trace of a campus network with known infected hosts. Those hosts are manually infected with the Neris malware by the authors [4]. *The demonstration shows that `NOracle` can detect the malicious bots shortly after the malware becomes active.*

Data for (3) is taken live with the network management system `StableNet` from a remote enterprise network testbed located in Würzburg, Germany. The enterprise network provides a testbed for trying out network management operations — it consists of more than 100 devices. `StableNet` is the core part "glueing" all together, i.e., it fetches networking data from all devices and makes it available. *Here, the demo shows how a network operator can inspect the communication behavior of the users and services live at run-time.* For example, it is possible to select the number of communication groups. Using `NOracle`'s GUI illustrated in Fig. 2, a network operator/administrator can investigate the evolution of the network over time, or investigate details of the communication structures within or between groups illustrated in Fig. 3. Clients that should be blocked from the outside world should not show any communication with "external" groups. Again, human knowledge is useful or even required to finally infer the semantic meaning of the communication groups.

ACKNOWLEDGEMENT

This work is part of a project that has received funding from the European Research Council (ERC) under the European Unions Horizon 2020 research and innovation program (grant agreement No 647158 - FlexNets).

REFERENCES

[1] Andreas Blenk, Patrick Kalmbach, Stefan Schmid, and Wolfgang Kellerer. 2017. o'zapft is: Tap Your Network Algorithm's Big Data!. In *Big-DAMA*. Los Angeles, CA, USA. https://doi.org/10.1145/3098593.3098597

[2] Andreas Blenk, Patrick Kalmbach, Patrick van der Smagt, and Wolfgang Kellerer. 2016. Boost Online Virtual Network Embedding: Using Neural Networks for Admission Control. In *CNSM*. https://doi.org/10.1109/CNSM.2016.7818395

[3] Sudipta Chowdhury, Mojtaba Khanzadeh, Ravi Akula, Fangyan Zhang, Song Zhang, Hugh Medal, Mohammad Marufuzzaman, and Linkan Bian. 2017. Botnet detection using graph-based feature clustering. *Journal of Big Data* 4, 1 (12 May 2017), 14. https://doi.org/10.1186/s40537-017-0074-7

[4] S. García, M. Grill, J. Stiborek, and A. Zunino. 2014. An empirical comparison of botnet detection methods. *Computers & Security* 45 (2014), 100 – 123. https://doi.org/10.1016/j.cose.2014.05.011

[5] Patrick Kalmbach, Andreas Blenk, Markus Klügel, and Wolfgang Kellerer. 2017. Generating Synthetic Internet- and IP-Topologies using the Stochastic-Block-Model. In *AnNet*. https://doi.org/10.23919/INM.2017.7987411

[6] Patrick Kalmbach, Andreas Blenk, Stefan Schmid, and Wolfgang Kellerer. 2018. Poster abstract: Themis: A data-driven approach to bot detection. In *IEEE INFOCOM 2018 - INFOCOM WKSHPS*. 1–2. https://doi.org/10.1109/INFCOMW.2018.8406870

[7] Patrich Kalmbach, Lion Gleiter, Johannes Zerwas, Andreas Blenk, and Wolfgang Kellerer. 2018. Modeling IP-to-IP communication using the Weighted Stochastic Block Model. In *ACM SIGCOMM 2018 Conference Posters and Demos*, ACM (Ed.). Budapest, Hungary, 1–3. https://doi.org/10.1145/3234200.3234245

[8] Patrick Kalmbach, Johannes Zerwas, Péter Babarczi, Andreas Blenk, Wolfgang Kellerer, and Stefan Schmid. 2018. Empowering Self-Driving Networks. In *SelfDN 2018*, ACM (Ed.). ACM Press. https://doi.org/10.1145/3229584.3229587

[9] Brian Karrer and M. E. J. Newman. 2011. Stochastic blockmodels and community structure in networks. 83, 1, Article 016107 (Jan 2011), 016107 pages. https://doi.org/10.1103/PhysRevE.83.016107 arXiv:physics.soc-ph/1008.3926

[10] Wolfgang Kellerer, Patrick Kalmbach, Andreas Blenk, Arsany Basta, Martin Reisslein, and Stefan Schmid. 2019. Adaptable and Data-Driven Softwarized Networks: Review, Opportunities, and Challenges. *Proc. IEEE* 107, 4 (Apr 2019), 711 – 731. https://doi.org/10.1109/JPROC.2019.2895553

[11] Daphne Koller and Nir Friedman. 2009. *Probabilistic Graphical Models: Principles and Techniques - Adaptive Computation and Machine Learning*. The MIT Press.

[12] M. E. J. Newman and Aaron Clauset. 2016. Structure and inference in annotated networks. *Nature Communications* 7 (June 2016), 11863. https://doi.org/10.1038/ncomms11863

[13] Tiago P. Peixoto. 2012. Entropy of stochastic blockmodel ensembles. *Phys. Rev. E* 85, 5 (May 2012), 056122. https://doi.org/10.1103/PhysRevE.85.056122

[14] Liron Schiff, Ofri Ziv, Manfred Jaeger, and Sterfan Schmid. 2018. NetSlicer: Automated and Traffic-Pattern Based Application Clustering in Datacenters. In *Big-DAMA '18*. ACM, New York, NY, USA, 21–26. https://doi.org/10.1145/3229607.3229614

[15] Johannes Zerwas, Patrick Kalmbach, Carlo Fuerst, Arne Ludwig, Andreas Blenk, Wolfgang Kellerer, and Stefan Schmid. 2018. Ahab: Data-Driven Virtual Cluster Hunting. In *2018 IFIP Networking Conference (IFIP Networking) and Workshops*. 1–9. https://doi.org/10.23919/IFIPNetworking.2018.8696399