S. Wendzel et al. (Hrsg.): SICHERHEIT 2024,
Lecture Notes in Informatics (LNI), Gesellschaft für Informatik, Bonn 2024 289

A Privacy-Preserving Architecture for Collaborative Botnet Detection

Leo V. Dessani[1]

Abstract:

Detecting communication with command and control (C2) servers and outbound attacks from internal bots (botnet traffic) is critical for network operators. Detection of botnet traffic is typically done by analyzing communication patterns in their own networks. We hypothesise that cooperation between different network operators can improve the detection of botnet traffic, as a larger amount of traffic can be examined. However, network operators do normally not want to share their traffic with others for privacy reasons. We therefore present a privacy-preserving architecture for collaborative botnet detection. To this end, network operators interested in detecting botnet traffic share traffic from their own networks by using a Threshold Multi-Party Private Set Intersection (T-MP-PSI) protocol to ensure that shared traffic details, such as IP addresses, are only disclosed if they occur on a minimum number of networks. We present the main results from a preliminary evaluation of the architecture based on publicly available benchmark data sets. The evaluation shows that our architecture contributes to the detection of botnet traffic, but that a high number of false positives also occur. However, this high number can be reduced by pre-processing measures. We also present further options for evaluating the architecture.

Keywords: botnet detection; anomaly detection

1 Introduction

Botnets are a very effective method of attacking publicly available third-party IT systems. Botmasters, the operators of botnets [WSZ08], use infected computers (bots) to attack these systems and use command and control (C2) servers to control and instruct the bots. The detection of communication with known C2 servers is already possible using Botnet Control Lists (BCL). However, detecting unknown C2 servers and outbound attacks from internal bots (botnet traffic) is difficult when analyzing the traffic of a single network that has only occasional communication with C2 servers, or where individual outbound connections contribute to a large-scale attack. We hypothesize that combining traffic from multiple network operators will lead to better botnet traffic detection results. This assumption is plausible because the volume of botnet traffic may be (significantly) larger when multiple networks are considered. However, network operators are usually unwilling to disclose some or all of their traffic. Therefore, we present a privacy-preserving architecture for

[1] CISPA Helmholtz Center for Information Security and Saarland University, Saarland Informatics Campus, Saarbrücken, Germany leo.dessani@cispa.de

doi: 10.18420/sicherheit2024_022

collaborative botnet detection. To this end, we use a Threshold Multi-Party Private Set Intersection (T-MP-PSI) protocol that considers elements of the network operators' data sets exactly if they occur in at least a minimum number of the network operators' data sets, specified by a threshold. Our focus is on describing the collaborative part of the architecture, supplemented by an initial evaluation and a look at future evaluations.

2 Related Work

Related work can be divided into privacy-preserving and collaborative approaches for botnet traffic detection and T-MP-PSI protocols, which are described below.

2.1 Privacy-preserving and Collaborative Approaches for Botnet Traffic Detection

Stevanovic et al. [St12] propose a collaborative approach that obtains traffic from various sources, such as packet sniffers, anti-virus-software, and application programs. Thus, the collaborative aspect is based on capturing traffic from as many sources as possible, rather than multiple network operators working together as in our approach. *Guerid et al.* [GMS13] describe a privacy-preserving approach based on bloom filters that detects domain-flux botnets. *Joshi et al.* [JBD17] evaluate the effectiveness of different community algorithms for detecting P2P botnets. While these algorithms contribute specifically to the detection of certain botnets, our approach contributes to the detection of any type of botnet where bots communicate with C2 servers.

2.2 T-MP-PSI Protocols

T-MP-PSI protocols have been proposed by Kissner et al. [KS04], Badrinarayanan et al. [Ba21a], Bay et al. [Ba21b], Yu et al. [Yu22], and Mahdavi et al. [Ma20]. We use the protocol of Mahdavi et al. for our architecture because it is the most efficient one compared to the others, considering elements of the network operators' data sets in the intersection exactly if they occur in at least a minimum number of the network operators' data sets, specified by a threshold $t \in \mathbb{N}$. It then informs the network operators which of their elements have appeared in at least $t-1$ other network operators. The protocol has $O(1)$ communication rounds, a communication complexity of $O(nptk)$ and a computational complexity of $O\left(m\left(p\log\frac{n}{t}\right)^{2t}\right)$, where $n \in \mathbb{N}$ corresponds to the size of each data set, $p \in \mathbb{N}$ to the number of network operators and $k \in \mathbb{N}$ to the number of key holders. According to the included oblivious pseudorandom secret sharing (OPR-SS) protocol, no keyholder responsible for creating shares of the traffic information contained in the data sets learns any information about them or the shares, and thus cannot infer the traffic of the network operators. The network operators do not learn the secret key used and therefore

cannot infer the traffic of other network operators. The selected T-MP-PSI protocol does not require a TTP.

3 Architecture Concept

The architecture assumes that p network operators (indexed by $i \in \mathbb{N}$) are willing to share data sets $\mathbb{L}_i$ with network connections from their own networks, where $|\mathbb{L}_i| \geq 1$. The collaborating network operators agree on a realistic time window in which they capture traffic.

3.1 Traffic

Each network operator i has non-private traffic in its network, i.e. incoming or outgoing traffic with a remote host that is not within the network. From this traffic, only the IP, TCP and UDP headers are analyzed using our architecture, allowing each network operator i to collect the following information I about the traffic on its network to form the data sets $\mathbb{L}_i$: **(a)** Timestamp of the first packet, **(b)** IP address of the remote host (IPv4 or IPv6), **(c)** direction (incoming/outgoing), **(d)** source and destination port, **(e)** transport layer and application layer protocol, and **(f)** transmission duration. Not all of these details are necessarily used as input to the T-MP-PSI protocol; the details to be used are selected by the network operators with a mode according to subsection 3.2. Because TCP is connection-oriented, TCP segments can be combined into TCP connections to obtain information that is not immediately apparent from the TCP headers (e.g. the transmission duration). Although UDP is not connection-oriented, all UDP segments are also combined based on IP address and port to ensure that UDP segments belonging to a UDP communication do not occur more than once to avoid redundant information. As many segments as possible should be captured, but there is no predefined number because the selected T-MP-PSI protocol can handle different set sizes.

3.2 Modes

Next, the network operators agree on a mode when executing the architecture by selecting $m_1, ..., m_q \in I$ (with $q \in \mathbb{N}$) to define how fine-grained the network operators' IP, TCP or UDP headers are compared when calculating the private set intersection.

3.3 Pre-processing

After all $\mathbb{L}_i$ are captured, each network operator i **(a)** removes all headers belonging to unsuspicious IP addresses that it trusts[2] and **(b)** locally matches the remaining IP addresses with one or more BCLs and removes the belonging headers if an IP address is listed. On the one hand, these steps reduce the amount of data to be analyzed, resulting in a shorter runtime of the T-MP-PSI protocol. On the other hand, the number of false positives is reduced. Finally, the T-MP-PSI protocol is executed using the final data sets $\mathbb{L}_i$ as input.

4 Preliminary Evaluation Results and Discussion

Ring et al. [Ri19] provide an overview of 34 traffic data sets for the development of anomaly-based intrusion detection systems. Nine of these data sets contain traffic with C2 servers and some of them also outbound attacks (in addition to benign traffic). In a first step, we evaluated our architecture based on two of the data sets containing traffic with C2 servers, but no outbound attacks (Botnet 2014 [Be14] and IoT-23 [GPE20]). The reasons for this selection are as follows: Firstly, not every botnet necessarily performs attacks on publicly available third-party IT systems; some also steal information or perform other actions on the bots. However, communication between the bots and the C2 servers does take place in any case. Secondly, the communication between the C2 servers and the bots normally takes place before the bots start attacking. We simulated the execution of the T-MP-PSI protocol with $p = 100$, which is a realistic number of network operators who want to detect botnet traffic on their networks, by randomly distributing the traffic of the data sets among them. This random distribution is necessary because the data sets did not contain any useful information about the structure of the networks in which they were captured. Thus, in this first analysis, we consider the traffic from each data set captured in one coherent network as collaboratively collected data until we obtain better data sets originating from multiple networks. Due to the high runtime of the T-MP-PSI protocol given in subsection 2.2, we divided the network operators into groups of ten network operators each and executed the T-MP-PSI protocol with different modes and the thresholds 2, 3 and 4 within each group. The main results of this analysis were:

- The vast majority of connections to C2 servers were detected, but there were also a large number of false positives.
- Higher thresholds result in fewer false positives.
- Some IP addresses of C2 servers were not detected correctly (false negatives).

[2] The traffic captured by each network operator usually contains a large number of such segments (e.g. traffic from trustworthy VPN, email, SIP and DNS servers). The network operator is able to identify such segments because it usually knows the IP addresses of the trustworthy servers that its network frequently communicates with.

The high number of false positives is due to the fact that effective pre-processing was not possible for the two data sets, as no information about trusted IP addresses was available. It is therefore evident that pre-processing is an important step in our architecture because it helps reduce false positives. At the same time, it becomes clear that older data sets and data sets without information on the structure of the networks in which they were recorded are not suitable for architecture evaluation.

The reason for the false negatives is that communication with some IP addresses of C2 servers are very rare and therefore do not reach the chosen threshold. One reason for this may be that botmasters communicate with their bots very infrequently to avoid detection. In particular, if they have infected devices and are holding them for upcoming attacks, there will be little C2 traffic.

In a second step, we extended the architecture to detect outbound attacks in the IoT-23 data set and evaluated it again.

The main results are:

- The number of false positives is reduced because some outbound attacks were classified as such in the first step.
- The architecture can also be used to detect outbound attacks.

Without proper pre-processing, the data set is not really suitable for a meaningful evaluation in the second step either.

The results presented only apply to the selected data sets and are not representative. Therefore, generalizations are not possible at this time.

These initial results show the following limitations of the architecture: it classifies many identical benign segments as suspicious (false positives) if the benign IP addresses the users are communicating with cannot be identified in advance and removed by pre-processing. In addition, there are botnets that use fast-flux techniques to quickly switch between C2 servers with different IP addresses. Our architecture does not classify the different IP addresses as suspicious if the threshold is not reached by the IP address change.

5 Conclusion and Future Work

In this paper, we presented a privacy-preserving architecture for collaborative botnet detection and provided initial evaluation results. The initial results show that the majority of connections to C2 servers were correctly detected. At the same time, however, they also show that older data sets and data sets without information about the structure of the networks in which they were captured generate numerous false positives and some false negatives and are therefore not suitable for architecture evaluation. We plan to evaluate

the architecture with historical passive DNS data and actual real-world data to provide more meaningful results. This will include both live network data and data about active C2 servers. In addition, we plan to investigate existing SMPC protocols to see how they can be optimized for the collaborative detection of botnet traffic.

Bibliography

[Ba21a] Badrinarayanan, Saikrishna; Miao, Peihan; Raghuraman, Srinivasan; Rindal, Peter: Multi-Party Threshold Private Set Intersection with Sublinear Communication. In: IACR International Conference on Public-Key Cryptography. Springer, pp. 349–379, 2021.

[Ba21b] Bay, Aslı; Erkin, Zekeriya; Hoepman, Jaap-Henk; Samardjiska, Simona; Vos, Jelle: Practical Multi-Party Private Set Intersection Protocols. IEEE Transactions on Information Forensics and Security, 17:1–15, 2021.

[Be14] Beigi, Elaheh Biglar; Jazi, Hossein Hadian; Stakhanova, Natalia; Ghorbani, Ali A: Towards Effective Feature Selection in Machine Learning-Based Botnet Detection Approaches. In: 2014 IEEE Conference on Communications and Network Security. IEEE, pp. 247–255, 2014.

[GMS13] Guerid, Hachem; Mittig, Karel; Serhrouchni, Ahmed: Privacy-Preserving Domain-Flux Botnet Detection in a Large Scale Network. In: 2013 Fifth International Conference on Communication Systems and Networks (COMSNETS). IEEE, pp. 1–9, 2013.

[GPE20] Garcia, Sebastian; Parmisano, Agustin; Erquiaga, Maria Jose: , IoT-23: A labeled dataset with malicious and benign IoT network traffic (Version 1.0.0) [Data set], 2020.

[JBD17] Joshi, Harshvardhan P; Bennison, Matthew; Dutta, Rudra: Collaborative botnet detection with partial communication graph information. In: 2017 IEEE 38th Sarnoff Symposium. IEEE, pp. 1–6, 2017.

[KS04] Kissner, Lea; Song, Dawn: Private and Threshold Set-Intersection. School of Computer Science, Carnegie Mellon University, 2004.

[Ma20] Mahdavi, Rasoul Akhavan; Humphries, Thomas; Kacsmar, Bailey; Krastnikov, Simeon; Lukas, Nils; Premkumar, John A; Shafieinejad, Masoumeh; Oya, Simon; Kerschbaum, Florian; Blass, Erik-Oliver: Practical Over-Threshold Multi-Party Private Set Intersection. In: Annual Computer Security Applications Conference. pp. 772–783, 2020.

[Ri19] Ring, Markus; Wunderlich, Sarah; Scheuring, Deniz; Landes, Dieter; Hotho, Andreas: A Survey of Network-based Intrusion Detection Data Sets. Computers & Security, 86:147–167, 2019.

[St12] Stevanovic, Matija; Revsbech, Kasper; Pedersen, Jens Myrup; Sharp, Robin; Jensen, Christian Damsgaard: A Collaborative Approach to Botnet Protection. In: International Conference on Availability, Reliability, and Security. Springer, pp. 624–638, 2012.

[WSZ08] Wang, Ping; Sparks, Sherri; Zou, Cliff C: An Advanced Hybrid Peer-to-Peer Botnet. IEEE Transactions on Dependable and Secure Computing, 7(2):113–127, 2008.

[Yu22] Yu, Xiaopeng; Li, Fagen; Zhao, Wei; Dai, Zhengyi; Tang, Dianhua et al.: Multiparty Threshold Private Set Intersection Protocol with Low Communication Complexity. Security and Communication Networks, 2022, 2022.