A Practical Analysis on Mirai Botnet Traffic

Getoar Gallopeni, Bruno Rodrigues, Muriel Franco, Burkhard Stiller
Communication Systems Group CSG, Department of Informatics IfI, University of Zurich UZH
Binzmühlestrasse 14, CH-8050 Zürich
E-mail: [rodrigues,franco,stiller]@ifi.uzh.ch, getoar.gallopeni@uzh.ch

***Abstract*—Distributed Denial-of-Service (DDoS) attacks are one of the biggest threats to the availability of Internet services. Behind these attacks are Botnets, such as Mirai, which exploits default and weak security credentials to take control of the host and spreads itself to other devices. This paper demonstrates a Mirai traffic analysis based on on DNS heavy-hitters streams and Mirai scanning patterns by simulating an attack and the extraction of traffic data. The Mirai Command-and-Control (CnC) traffic as well as its scanning traffic are analyzed in a local Testbed composed of six ASUS Tinker Board devices (Raspberry-Pi like devices) cluster nodes and a MikroTik's RouterOS to route traffic in different internal networks. In addition to the analysis of traffic flow patterns a real-time mitigation is demonstrated in the experiments.**

I. Introduction

Since smart devices in private households often seek convenience and are not treated equally impactful as personal computers, their potential security issues are typically overseen [1]. A malware that became famous for exploiting these connected devices' weaknesses (*i.e.*, IoT devices) is called Mirai [2]. Mirai gained notoriety in towards the end of 2016 after several large-scale attacks. It comprises malware that scans through ranges of IP addresses looking for devices to attack [2], and once an IP found, a set of default credentials is used in order to log in and take control over the device. This procedure is repeated until the attacker has set up a network of controlled devices.

Whenever a successful connection is established, the IP address of the new device victim is sent to a reporting server along with the successful credentials. Then, the victim receives the malware from the loader and continues the spreading behavior until an attack is requested by the Command-and-Control (CnC) [3]. Unlike most complex Peer-to-Peer (P2P) systems, the behavior of the default Mirai botnet is comprehensible by analyzing its code components. Sinanovic and Mrdovic (2017) [4] performed a comprehensive analysis of the Mirai code. The authors provided a low-level description of the Mirai segments, which is the foundation of how Mirai works and interacts, covering all static aspects and demonstrates the included SQL table structures.

This paper demonstrates an analysis of the control flow channel between a Mirai botmaster and bots extracting communication patterns. A baseline for the Mirai CnC traffic analysis is created by simulating an attack and the extraction of traffic data based on a dedicated cluster hardware composed of six nodes and a manageable router.

Annex to ISBN 978-3-903176-28-7

II. Overview

The traffic analysis is deployed at an access network *e.g.*, in a company where the operator has access to network management, which can passively or actively observe internal and external traffic. Data captured by a router/switch is mirrored to a control server, which performs a retrospective (post-mortem) and real-time traffic inspection. While the first is performed using Wireshark, in which periodic traffic PCAPs are generated, the second takes as input an analysis on the PCAPs files to refine the search for live traffic based on Pyshark. Hence, the monitoring approach aims to detect the presence of Bots *i.e.*, infected devices and their communication with CnC traffic on local networks.

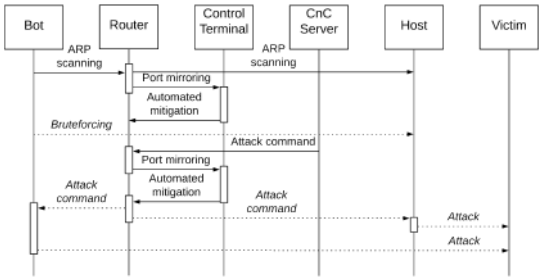

Fig. 1: Traffic Analysis Approach for Different Components

As illustrated in the first step of Figure 1, Bots scan addresses across an allowed range of IP addresses over the Internet that in our setup was configured to the local network. The router mirrors all traffic to a control server, which filters the traffic for analysis. The analysis is based on a supervised algorithm taking as input previously scanned traffic patterns of bot-scanning and communication with the CnC. The following strategies were deployed:

- **Scanning behavior**: can be identified by counting the number of ARP packets sent by Bots in a PCAP file. Thus, thresholds are defined to emit alerts or to blacklist a certain host.
- **CnC traffic**: Bots establish a connection with at least one CnC server via Telnet. Based on the identified communication pattern, the traffic dissector reconstructs the packages based on the packet and frame number. While the former determines how packets relate to each other, the latter establish a chronological order.
- **Attack commands**: packets are marked as 'detected' as soon as the Mirai syntax is identified at a certain position within the payload (*cf.*, Figure 2).

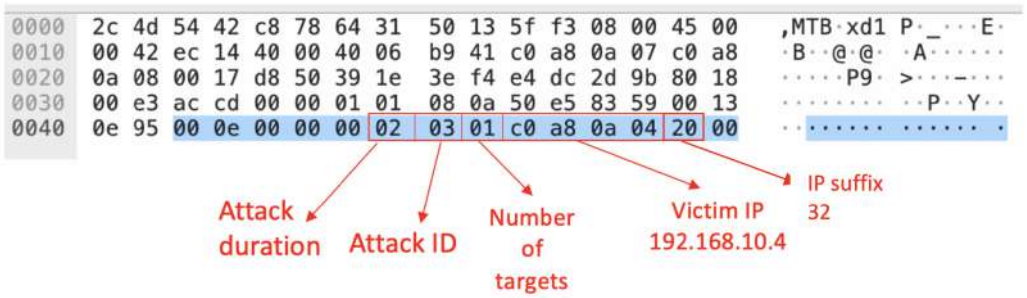

Fig. 2: Parsing an Attack Command in a Hexadecimal Form

Despite obfuscating its traffic, Mirai in its original variant leaves suspicious traits in the network, which includes not only a flood of ARP packets caused by Mirai's scanning mechanism, but also resolved DNS requests, started Telnet sessions and CnC attack commands. While some are clear indicators for Mirai activity, others may just hint to doubtful traffic within the network.

III. Demonstration

The demonstration is implemented as a physical single board computer cluster (*cf.*, Figure 3). The hardware is based on a single MikroTik router running RouterOS (v. 6.42) and six ASUS Tinker Board devices (Raspberry-Pi like devices) to simulate Bots. Two networks are configured 192.168.10.0 and 192.168.20.0 (/24) to evaluate the scanning behavior and prevent switching on the MAC address. Further, it ensures firewall filters and queues to be applied accordingly.

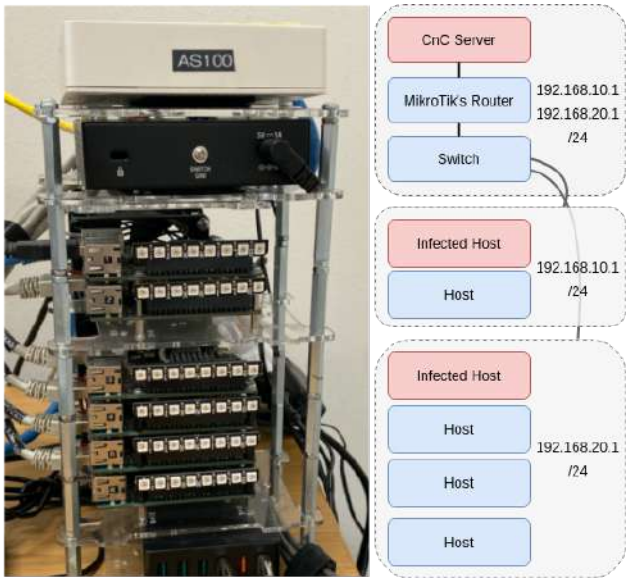

Fig. 3: Testbed Architecture

The control terminal on the CnC will send the attack command to one or more bots as required. Then, bots start the corresponding attack. The victim of the attack is at least one host in the other network. In a first stage, the traffic analyzer learns the Mirai traffic patterns (which is pre-configured as an input in the demonstration) and perform the inspection and command detection in the second stage. Bots within the two networks are triggered to use ten attack vectors at once. In each of the attacks, the command can be detected and intercepted. In all executed tests, only the IP address that started scanning earlier was detected. After the detection, multiple alternatives and services are possible to mitigate the attack, which can be determined by approaches such as [5].

IV. Preliminary Conclusions and Future Work

This demonstration aims to show the audience the practical functioning of one of the major recent Botnets, Mirai. Thus, its traffic pattern for infection from other hosts, as well as communication with the CnC are demonstrated based on an isolated hardware testbed. The analysis was based on the fingerprints of the traffic patterns to perform the detection based on the Mirai traffic signature in real-time. As future work, by following an anomaly-based detection procedure, more general patterns could be learned and applied. This could help to detect other malware and variations of the Mirai source code.

Acknowledgements

This paper was supported partially by *(a)* the University of Zürich UZH, Switzerland and *(b)* the European Union's Horizon 2020 Research and Innovation Program under Grant Agreement No. 830927, the CONCORDIA project.

References

[1] B. Rodrigues, M. F. Franco, G. Paranghi, and B. Stiller, "SEConomy: A Framework for the Economic Assessment of Cybersecurity ," *16th International Conference on the Economics of Grids, Clouds, Systems, and Services (GECON 2019)*. Leeds, UK, Springer, September 2019, pp. 1–9.

[2] M. Antonakakis, T. April, M. Bailey, M. Bernhard, E. Bursztein, J. Cochran, Z. Durumeric, J. A. Halderman, L. Invernizzi, M. Kallitsis *et al.*, "Understanding the Mirai Botnet," *26th USENIX Security Symposium (USENIX Security 17)*, 2017, pp. 1093–1110.

[3] C. Kolias, G. Kambourakis, A. Stavrou, and J. Voas, "DDoS in the IoT: Mirai and Other Botnets," *Computer*, Vol. 50, No. 7, pp. 80–84, 2017.

[4] H. Sinanovic and S. Mrdovic, "Analysis of Mirai Malicious Software," *2017 25th International Conference on Software, Telecommunications and Computer Networks (SoftCOM)*, pp. 1–5, 2017.

[5] M. F. Franco, B. Rodrigues, and B. Stiller, "MENTOR: The Design and Evaluation of a Protection Services Recommender System," *2019 15th International Conference on Network and Service Management (CNSM)*. IEEE, 2019, pp. 1–7.