Abstract—The smallest element in a botnet is a bot. The behavior of a bot can change dynamically based on the decision of the botmaster. Commonly driven by profit, bots are expected to be profitable. If an infected bot does not fulfill the expectations, the botmaster can instruct the bot to switch its behavior to serve a better purpose. This paper presents a detailed analysis of a network traffic capture of a machine originally infected by a Gamarue variant. The analysis will uncover the behavior of the bot during the initial infection, inactivity period, delivery of a new payload and the following switch of behavior of the bot. The paper will analyze the infection in detail, including the horizontal brute-forcing activity affecting thousands of WordPress websites. The goal of the paper is to show a concrete example of a bot performing brute-forcing, analyze it, identify the mechanisms used and indicators of compromise that will help detect it.

I. INTRODUCTION

Botnets are always recruiting new bots to increase their size and be more profitable. It is a common practice that researchers execute malware in sandboxed environments in order to study the botnet’s behavior. When the sandboxing time is short, about a few minutes, it is only possible to capture the initial behavior of the bot. This initial behavior is very important as it gives critical information about the botnet that can be immediately used for detecting the threat. However, when the sandboxing time is longer, such as days or months, it is possible to acquire knowledge not only from the particular bot and the botnet actions in that period of time, but it is also possible to have an insight on the underlying strategies behind a botnet’s actions.

Gamarue is a complex and modular botnet, also known as Andromeda. It has been studied thoroughly and new reports on its behavior are being published continuously. As a modular botnet, its behavior can dynamically change. The botnet is capable of deploying new modules and behaves in ways that are still unknown. Usually these different behaviors cannot be captured by current sandbox solutions that execute malware for short periods of time. It is only when a malware is run for extended periods of time that their more complex capabilities can be observed.

In 2015, as part of our research of malware behavior through long term sandboxing experiments, we infected several virtual machines (VMs) with a particular Gamarue sample[1]. Every VM was run for a specific amount of time. The shortest sandbox time was 4 hours and the longest sandbox time was 45 days. After each infection stopped, the dynamic behavior of the malware was analyzed.

This paper will present a detailed network capture analysis of a 30 days Gamarue infection, covering the different stages of the infected bot, the observed infrastructure of the botnet and how different command and control (C&C) channels were used for different purposes. Our work also presents an analysis of a previously unknown behavior of the Gamarue botnet: the capability to perform an horizontal brute-forcing of WordPress sites in a highly aggressive and automated fashion.

II. INFECTION TIMELINE

The malware infection experiment lasted thirty days, from May 12, 2015 to June 12, 2015. Two command and control channels were identified in the capture:

1) Gamarue C&C
2) Brute-forcing module C&C

The bot started communicating with the Gamarue C&C immediately after the binary was executed on the VM. The bot was idle the first 10 days of the infection, there was an active communication with the C&C servers but no additional traffic...
was seen during this period. After this long inactivity time, the bot was instructed to start downloading and deploying new payloads into the infected machine as shown on Figure 1.

The bot attempted to download 13 files from 2 different servers in the 30 days infection experiment. The first 4 downloads didn’t produce any change on the behavior of the bot. However, the fifth payload (http://allbestcounter.ru/chb.exe) triggered a second infection on the already infected host. A comparison of the network traffic of each C&C channel is shown in Figure 2, where it is visible how both remained active until the end of the experiment.

Gamarue is the primary infection that controls the activity of the bot and is independent of the second module’s C&C seen on the host. During the thirty days the primary C&C remained active, the bot attempted to communicate with three different servers:

- 188.190.114.99 (w4gvlw4kjbrvshkvb3d.ru)
- 93.95.98.50 (okiijlijlili.eu)
- 166.78.144.80 (f34234f234f2sdcsv.info)

The communication to the C&C servers was done through the HTTP protocol, by sending and receiving encrypted data. An example of a successful HTTP connection to the C&C is shown on Figure 3, where it is possible to observe how the bot is sending encrypted data and the C&C server is also responding with encrypted data. There is also additional key information visible on the Figure, such as the specific User-Agent used on the request “Mozilla/4.0” and the small amount of data sent and received in each query. Figure 4 shows an HTTP connection example to an unavailable server, where the bot is sending encrypted data but the server is responding with a “404 Not Found” error page.

The HTTP requests to each of these servers were identical: a POST request sending the same amount of encrypted data to the resource “/gate.php”. The C&C server on 188.190.114.99 was the only one that seemed to be active: responding to the bot with encrypted data. The other two were inactive or responded with an HTTP error code. Figure 5 shows the average of uploaded content-bytes (without headers) per each C&C server related to Gamarue. The bot’s communication to the servers was highly stable, constantly sending the same amount of data, 61 bytes. The average of downloaded content-bytes (without headers), as seen on Figure 6, shows that the server on 166.78.144.80 was sending zero bytes as response; the server on 93.95.98.50 had a higher daily average of bytes received, which indicates that the server was not available and the response was a typical HTTP not found error.

The behavior of the Gamarue bot during the first 10 days is standard for this botnet. Indicators discussed in this section can be used to identify and detect this type of traffic in the network. The activity associated with the download of new payloads provides new insight on this botnet’s long term activity and potential capabilities. The next section will discuss in depth the behavior of the second infection responsible for the horizontal brute-forcing of WordPress sites.

III. HORIZONTAL BRUTE-FORCING C&C

The second infection on the host was triggered by the download of a new payload after 14 days of the original
infection. The delivered payload doesn’t have a concrete detection signature according to Virus Total, causing a poor threat categorization.

The behavior of this particular module can be summarized as follows:

1) It obtains a list of WordPress targets to attempt to login from the C&C server.
2) It takes the next entry from the list and attempts to login with chosen credentials (explained in section V) in order to gain access.
3) If the login attempt was successful, reports it to the C&C server.
4) If the login attempt was unsuccessful, it will iterate from step 2) until exhausting the retrieved websites.

The C&C communication was not encrypted and sent through HTTP to only one server:

- g.commandocenter.ru (5.8.32.51)

The analysis of the network traffic associated to this C&C corroborates the simplicity of the malware behavior stated above. There are only three different types of HTTP requests made to this server, as exemplified next:

1) http://g.commandocenter.ru/default.aspx
   ?guid=dca94df1f-7eb-487f-ad24-923cd1b49f64&gate=1&good=1&bad=0&unlucky=1&ip=&fn=??
2) http://g.commandocenter.ru/files/2/9d753bd0-33a5-46ac-841d-99d9ace3446.txt ??
3) http://g.commandocenter.ru/col.aspx
   ?t=wp_b&g=1&gid=1 ??

The purpose of the first request is twofold: send a status report to the C&C and retrieve the name of a new file to download. The second request is the one that actually retrieves the file from the server as indicated on the response of the previous request. The file retrieved from the server contains a list of 1,000 WordPress domains, in average, that the bot will attempt to brute-force. The first and second requests are usually in the same TCP session as it can be seen on the example shown in Figure 7.

While the first and second request occur with a high frequency during the brute-forcing activity, the third request is not common. The third request is used to send information to the C&C server about the successful attempts made by the bot. The POST request will send a list of domains that the bot could successfully log into. Figure 8 illustrates an example in which just one website is being reported as a successful attempt. This request will trigger again the first and second requests, sharing the same TCP session: reporting the status of the last brute-forcing attempt and getting a new batch of domains to brute-force as seen in Figure 9.

A breakdown of the amount of traffic associated to each one of these individual behaviors during the whole infection is shown in Figure 10.

The brute-forcing activity was condensed on the first four days after the deployment of the new payload, as it can be observed in Figure 10. The activity of the bot during this
Fig. 8. Example of how the bot is reporting a successful attempt to the C&C. The communication initiates with an HTTP POST request but the actual data is sent as a response to the '100 Continue' message received from the C&C server.

Fig. 9. Sequence of requests triggered by a successful attempt report

period was highly aggressive. With more than 40,000 sites attempted per day, the bot attempted to login into 167,066 WordPress sites. Figure 11 shows a general overview of the brute-forcing activity where it is possible to observe the burst on the first 3 days, a considerable reduction on day 4 and a stop of the activity on day 5.

As it was mentioned before, the bot downloaded files from the C&C server containing lists of WordPress sites. Every new file retrieved by the bot contained a list of 1,000 WordPress domains or less. In the four days of activity 300 files were downloaded, containing 219,078 WordPress domains. Every list contained sites in alphabetical order, usually one list contained domains starting with ‘a’, next one with ‘b’ and so on. While it is unknown how these lists of sites were gathered and created, there are clues that seem to indicate they are generated in an automatic fashion. Some of the lists contained what seem to be IP addresses but with only three octets instead of four of them; some lists contained two domain names concatenated, and some lists contained domains starting with ‘.’, which may mean a parsing error or some mistake in the retrieval of those domains.

IV. INSIGHT ON TARGETS

An analysis of the domains targeted by the botnet showed that the brute-forcing activity was not focused in a particular geographic region as illustrated on Figure 12. It was a global brute-forcing attack of WordPress sites located all over the world.

The WordPress domains seem to have been chosen automatically in no particular order. An analysis on the TLD distribution shows that there were domains matching 164 different TLDs. A breakdown of common TLDs targeted by the bot is displayed in Figure 13.

Many of the targeted domains seem unattended, giving advantage to the attackers as there are higher chances that passwords on those sites are left by default and there are less chances that the site will be cleaned up immediately.

V. BRUTE-FORCE LIST OF USER NAMES AND PASSWORDS

As mentioned in the previous section, many of the successful login attempts were because of WordPress sites using a weak user name and password combination. Due to the fact that every login attempt was performed through a HTTP POST request, it was possible to analyze the combinations of user names and passwords the bot used in its activity.

The analysis of login credentials showed that there was no variability on user names. The bot only attempted to login as
'admin' in all the cases. The decision by the attacker was right: WordPress sites have a default 'admin' user and fixing this variable reduces the complexity of the brute-forcing. There is additional information leveraged by the attackers in this aspect, and it is the fact that WordPress is designed to be easy to be used by anyone that increases the success chances as many non-experienced users just use default or easy to guess credentials.

The passwords used by the bot were interestingly distributed in two groups:

- Default credentials (admin,admin): 47.5% of the cases
- Custom passwords: 52.5% of the cases

There is a huge number of unique combinations of user names and passwords used by the bot as alternatives to the default 'admin' password. Figure 14 shows a few examples of these observed unique passwords. The existence of this list of passwords may initially indicate that the bot downloaded such a list from the C&C server. The analysis of the network traffic showed no signs of this request or of any additional download by the bot. Additionally, there is a correspondence between some of the passwords and domain names of targeted WordPress sites. The analysis showed the powerful capacity of a single bot for this activity. It also showed us how this new malware, by massively harvesting vulnerable websites, may be supplying other malicious actors with compromised sites to use as part of their moving infrastructure. The main reason of the success of this type of botnet relies on the human factor: people keeping default user names and passwords or choosing passwords that are easy to guess.

The presented work also served as a concrete example of what type of extra information can be extracted from long term sandboxing experiments and how this uncommon long term captures can give an unexpected insight of the often complex behavior of botnets. We hope it will also serve as motivation for others to experiment as well with long term sandboxing malware captures.

VII. LIST OF IOCS

The following domains and IPs were observed during this investigation:

- 188.190.114.99
- w4gvnlw4kjbrvbshkvbsd.ru
- 93.95.98.50
- okiiijlijili.eu
- 166.78.144.80
- f34234f234f2sdcsv.info
g.commandocenter.ru
- 5.8.32.51

Veronica Valeros, Make It Count: an Analysis of a Brute-forcing Botnet [Short conference paper] (numbering of pages to be confirmed)