Mirai Botnet Attack IoT Devices via CVE-2020-5902

blog.trendmicro.com/trendlabs-security-intelligence/mirai-botnet-exploit-weaponized-to-attack-iot-devices-via-cve-2020-5902/

July 28, 2020

Update as of 10:00 A.M. PST, July 30, 2020: Our continued analysis of the malware sample showed adjustments to the details involving the URI and Shodan scan parameters. We made the necessary changes in this post. We would like to thank F5 Networks for reaching out to us to clarify these details.

Following the initial disclosure of two F5 BIG-IP vulnerabilities on the first week of July, we continued monitoring and analyzing the vulnerabilities and other related activities to further understand their severities. Based on the workaround published for CVE-2020-5902, we found an internet of things (IoT) Mirai botnet downloader (detected by Trend Micro as Trojan.SH.MIRAI.BOI) that can be added to new malware variants to scan for exposed Big-IP boxes for intrusion and deliver the malicious payload.

The samples we found also try to exploit recently disclosed and potentially unpatched vulnerabilities in commonly used devices and software. System administrators and individuals using the related devices are advised to patch their respective tools immediately.

Routine

As previously reported, the security bug involves a remote code execution (RCE) vulnerability in the management interface of BIG-IP known as the Traffic Management User Interface (TMUI). After analyzing the published information, we noticed from the mitigation rule in Apache httpd that a possible way to exploit this vulnerability involves a HTTP GET request containing semicolon character in the URI. In a Linux command line, a semi-colon signals to the interpreter that a command line has finished, and it is a character the vulnerability needs to be triggered. To further our analysis, we tested if an IoT botnet author can add a scanning capability to existing and/or new malware variants via this Yara rule:

Yara rule to check for malware Figure 1. Yara rule to check for malware

While the rule used for testing seems simple, it allowed us a broad range of malware, Python, or Ruby proofs of concept (PoC) to check against. From the disclosure date of July 1, we found the first sample of an ELF file compiled to an MIPS architecture on July 11 that identified two addresses: hxxp[:]//79[.]124[.]8[.]24/bins/ (identified as the disease vector) and hxxp[:]//78[.]142[.]18[.]20 (identified as the command and control (C&C) server). A common pattern with IoT malware like Mirai is finding different files hosted in one domain with different extensions meant to attack different architectures. Upon checking the host, we found the following files:

Table 1. Files hosted in the C&C

Hash	File
acb930a41abdc4b055e2e3806aad85068be8d85e0e0610be35e784bfd7cf5b0e	fetch.sh
037859323285e0bbbc054f43b642c48f2826924149cb1c494cbbf1fc8707f942	sora.arm5
55c4675a84c1ee40e67209dfde25a5d1c1979454ec2120047026d94f64d57744	sora.arm6
03254e6240c35f7d787ca5175ffc36818185e62bdfc4d88d5b342451a747156d	sora.arm7
204cbad52dde24ab3df41c58021d8039910bf7ea07645e70780c2dbd66f7e90b	sora.m68k
3f8e65988b8e2909f0ea5605f655348efb87565566808c29d136001239b7dfa9	sora.mips
15b2ee07246684f93b996b41578ff32332f4f2a60ef3626df9dc740405e45751	sora.mpsl
0ca27c002e3f905dddf9083c9b2f8b3c0ba8fb0976c6a06180f623c6acc6d8ca	sora.ppc
ecc1e3f8332de94d830ed97cd07867b90a405bc9cc1b8deccec51badb4a2707c	sora.sh4
e71aca778ea1753973b23e6aa29d1445f93dc15e531c706b6165502d6cf0bfa4	sora.x86

Looking into the IP addresses further, we learned that since June, it had already been used to deploy IoT malware, including other Mirai variants.

The SORA file names have been previously identified as a Mirai variant that can be used for brute-force attacks and the abuse of other vulnerabilities for RCE and unauthorized control and management of devices. Meanwhile, fetch.sh is a shell script with the following content:

fetch.sh shellscript Figure 2. fetch.sh shellscript
fetch.sh connects to http[:]//79[.]124[.]8[.]24/bins/sora.{architecture} to download and execute the applicable malicious binary named "sysctl". Simultaneously, fetch.sh also creates cron jobs to enable automatic execution of the downloaded binary.

Creating cron jobs Figure 3. Creating cron jobs
The script uses the iptables tool to drop any packets to popularly used transmission control protocol (TCP) ports such as default ports for Telnet, Secure Shell (SSH), and the device web panel (HTTP). This may have two different implications:

- No other malware will have direct access to exposed services in the infected device
- The device owner will not be able to access the management interface

This is also reminiscent of implications cited in our recent research paper for the control of IoT devices currently connected.

By analyzing the x86 sample of this botnet, we realized its attempts at exploiting vulnerable BIG-IP boxes as it sends a GET request to the victim port 443/TCP (HTTPS):

GET request for exploit of CVE-2020-5902 Figure 4. GET request for exploit of CVE-2020-5902
Given the severity of the flaw, a simple GET request with a "command" parameter to tmshCmd.jsp would be enough to remotely execute a command in an infected device if the ID path is correctly prepended to it.

Other exploits abused

We also found, upon checking the variant further, that it tries to exploit recently disclosed and discovered vulnerabilities in randomly generated targets. Here is the full list of exploits used by this variant:

Table 2. Other exploits used by other samples

Device	Vulnerability	CVE Identification
Apache Kylin 3.0.1	Command Injection Vulnerability	CVE-2020-1956
Aruba ClearPass Policy Manager 6.7.0	Unauthenticated Remote Command Execution	CVE-2020-7115

Big-IP 15.0.0 < 15.1.0.3 / 14.1.0 < 14.1.2.5 / 13.1.0 < 13.1.3.3 / 12.1.0 < 12.1.5.1 / 11.6.1 < 11.6.5.1	TMUI Remote Code Execution	CVE-2020-5902
Comtrend VR-3033	Command Injection	CVE-2020-10173
HP LinuxKI 6.01	Remote Command Injection	CVE-2020-7209
Tenda AC15 AC1900	Remote Code Execution	CVE-2020-10987
Nexus Repository Manager 3	Remote Code Execution	CVE-2020–10204
Netlink GPON Router 1.0.11	Remote Code Execution	N/A
Netgear R7000 Router	Remote Code Execution	N/A
Sickbeard 0.1	Remote Command Injection	N/A

Conclusion and security recommendations

F5 Networks caters to a number of enterprises for networking devices, with BIG-IP as one of the most popular products in use by governments and companies, especially given the abrupt work-from-home arrangements today. It affects a wide range of products and versions, including the most recently released ones close to the vulnerability's disclosure date. With CVE-2020-5902 receiving a rating of 10 in the Common Vulnerability Scoring System (CVSS) v3.0 vulnerability scale, the vulnerability also indicates that the security gap itself is easy to abuse online and automate. Moreover, it does not require credentials or advanced coding skills to exploit.

That said, F5 has already published an informative and detailed mitigation procedure in order to deny requests containing a semi-colon. To add, while the default settings do not expose the management interface to the public, our Shodan scan showed approximately 7,000 exposed hosts online (considering the ones listening on ports 443 and 8443 only). By "exposed" we mean "accessible from the Internet", but not with certainty that the said hosts are vulnerable.

Recognizing the severity of the security flaw, the Department of Defense's Cyber Command issued a tweet three days after the disclosure, advising immediate remediation of the vulnerability. Given the vulnerability's disclosure date and the number of days it took for an exploit to be at large (10 days), it appears as if the malicious actors are paying close attention to the most recent disclosures and reports to come up with their own exploits. While some of these vulnerabilities were only discussed in blog posts and not announced as publicly available exploit codes, these cybercriminals are aware of two things: first, manufacturers have yet to come up with the corresponding patches, and second, system administrators have yet to download and implement the released fixes in the equivalent duration.

System administrators and security teams can protect IoT devices from these types of threats with some of these best practices:

- Ensure that IoT devices' firmware run on the latest versions by constantly monitoring manufacturers' releases.
- Use a virtual private network (VPN) to prevent exposing any management interfaces directly to the internet.
- Employ network segmentation to limit the spread of infections and customize the security settings of devices.
- Ensure that there is a network traffic monitoring and detection system with a good Web Application Firewall (WAF) in place. This is to track baseline and abnormal ranges of usage to protect management interfaces that are accessible online.
- Install a multilayered protection system that can detect, block, and prevent threats such as brute-force attacks that abuse security flaws like these for entry.

Connected devices can also be protected by security software such as the Trend Micro™ Home Network Security and Trend Micro™ Home Network Security SDK solutions, which can check internet traffic between the router and all connected devices as well as help users asses for vulnerabilities.

Indicators of Compromise (IOCs)

Please see the complete list of IoCs here.

Exploits & Vulnerabilities

Based on the workaround published for CVE-2020-5902, we found a Mirai botnet downloader that can be added to new malware variants to scan for exposed Big-IP boxes for intrusion and deliver the malicious payload.

By: Fernando Merces, Augusto Remillano II, Jemimah Molina July 28, 2020 Read time: (words)

Content added to Folio