

WEB APPLICATION ATTACK STATISTICS

Q3 2017

POSITIVE TECHNOLOGIES

CONTENTS

Introduction..... 3

Results at a glance 4

Web application attacks: statistics 5

 Attack types 5

 Attack trends 11

Conclusions..... 13

INTRODUCTION

This report provides statistics on attacks performed against web applications during the third quarter of 2017. Sources of data are pilot projects involving deployment of PT Application Firewall, as well as Positive Technologies' own PT AF installations.

The report describes the most common types of attacks as well as the objectives, intensity, and time distribution of attacks. It also contains industry-by-industry statistics. With this up-to-date picture of attacks, companies and organizations can monitor trends in web application security, identify the most important threats, and focus their efforts during web application development and subsequent protection.

To obtain more consistent results, automated vulnerability scanners (such as Acunetix) have been excluded from the data used here. The example attacks presented in this report have been manually verified to rule out false positives.

Protection data for Positive Technologies itself has been classified under the IT sector for reporting purposes.

RESULTS AT A GLANCE



Approximately
1 out of 2 attacks

is aimed at accessing data



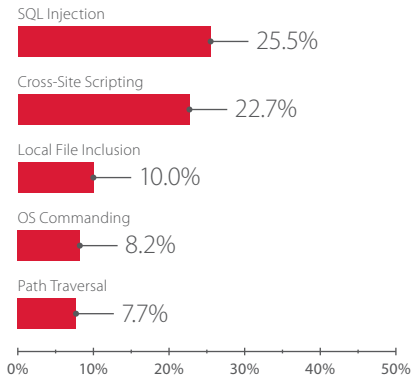
30% of attacks

target users

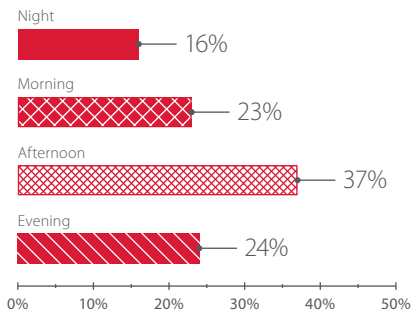


4,321

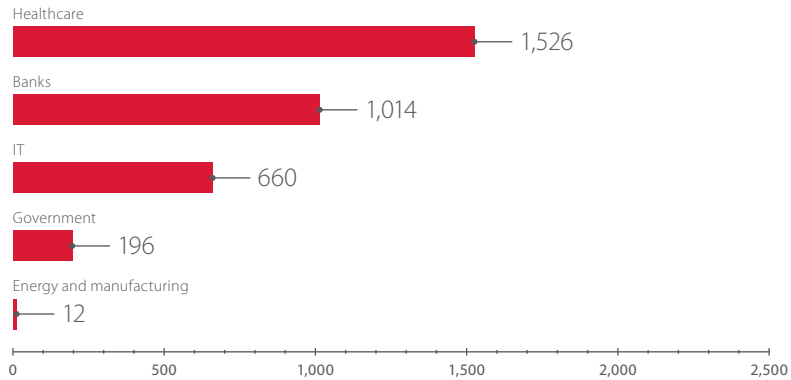
highest number of attacks
on a company in a single day



Most frequent attacks



Distribution of attacks by time of day



Average number of attacks per day per company

WEB APPLICATION ATTACKS: STATISTICS

Attack types

The most common attack in the third quarter of 2017 was SQL Injection. This type of attack is used to obtain unauthorized access to sensitive information and run OS commands. In second place was Cross-Site Scripting, which targets users of web applications. As predicted in the report for the previous quarter, these two attack types continue to account for nearly half of all attacks on the web applications in our dataset. Local File Inclusion, which aims to run arbitrary code on a targeted server, has risen to third place. In addition, OS Commanding has doubled in frequency compared to the previous quarter. These dangerous attacks have the potential to provide attackers with full control over the server hosting a web application.

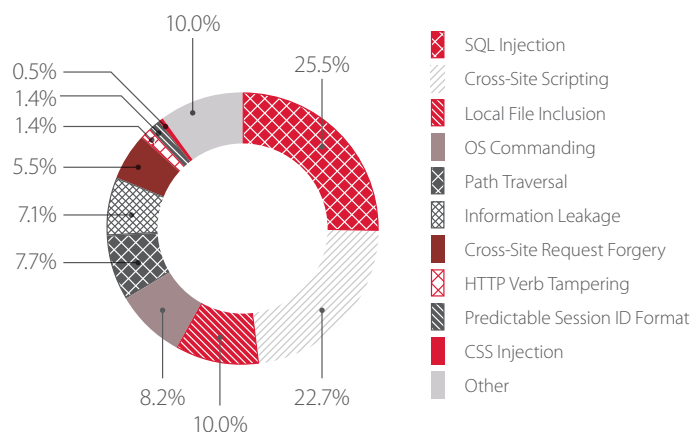


Figure 1. Web application attacks: types

The distribution of attacks by degree of risk (as classified by PT AF) is shown in the following graph.

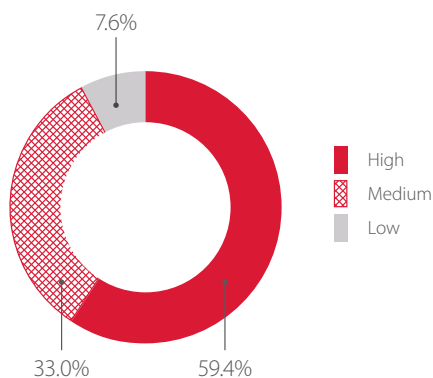


Figure 2. Distribution of web application attacks, by degree of risk

These statistics can be used to paint a detailed picture of the web application attacks experienced by different sectors. Sectors covered in depth in this report for Q3 are healthcare, energy and manufacturing, banks, IT, and government.

Healthcare

Web application attacks on healthcare differed significantly from those in the previous quarter. The primary cause is that the applications in this quarter's report are intended for different purposes than those in the previous quarters. Most of the web applications in the healthcare category this quarter are used to provide information; in other words, they do not handle private data or patient medical records. Given the absence of sensitive information of interest to criminals, it is no surprise that SQL Injection fell considerably compared to the previous quarter (from 46% to 2.9%). Meanwhile, other attacks increased, particularly OS Commanding and Arbitrary Code Execution (to around 50%). Local File Inclusion was one common method used to perform such attacks. In such attacks, a criminal can obtain full control over a web application and alter its content, shut it down, or use it to spread malware. For example, October 24 marked the start of distribution of Bed Rabbit ransomware, which involved media websites hacked previously.¹ When users accessed the page of a hacked media web application, they were invited to download a fake Adobe Flash Player installer. If the user continued with the download, the installer infected the system. Similar use of legitimate websites to spread malware en masse may affect the healthcare sector as well, since like government and media sites, they are popular and trusted by users.

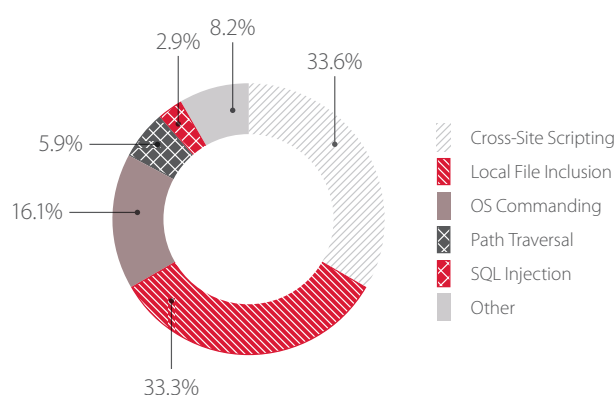


Figure 3. Types of attacks on web applications of healthcare institutions

Energy and manufacturing companies

Attacks on the web applications of manufacturing and energy companies continue to be targeted and deliberate in nature. Attackers generally are attempting to obtain sensitive information about the targeted system in preparation for future attacks. As noted in the prior reports this year, attackers use these web applications as a jumping-off point for penetrating the internal infrastructure of such companies in order to access industrial control and SCADA systems. The ultimate goal is the ability to disrupt operations at the target company. The most common attacks include ones aimed at OS Commanding and obtaining control over the server for subsequent attacks on internal infrastructure. As shown in Positive Technologies research, 77 percent of all attack vectors identified in 2016 that were capable of breaching the corporate network perimeter leveraged vulnerabilities in web applications.²

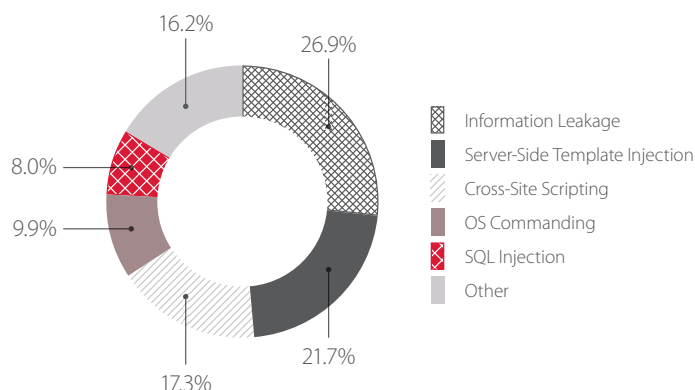


Figure 4. Types of attacks on web applications of energy and manufacturing companies

¹ welivesecurity.com/2017/10/24/bad-rabbit-not-petya-back/

² [Security Trends & Vulnerabilities Review: Corporate Information Systems \(2017\)](#)

Banks

Financial gain is the usual motive for attacks on bank web applications. SQL Injection can allow unauthorized access to sensitive information, such as clients' personal data and financial records. Nearly one out of three attacks was aimed at web application users. Such attacks make it possible to steal user credentials or infect users' workstations with malware. The resulting financial losses may affect both clients and banks, not to mention the potential for reputational damage to the latter.

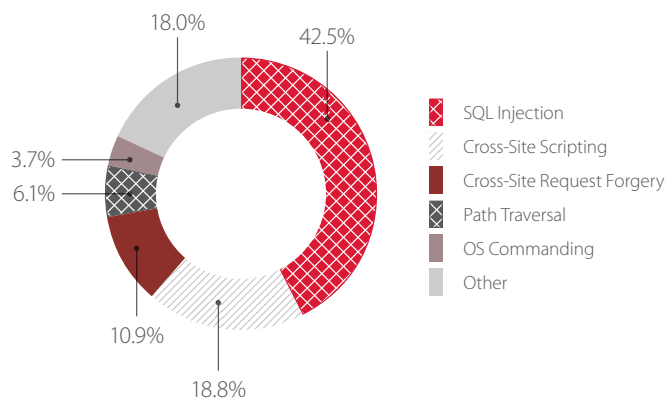


Figure 5. Types of attacks on web applications of banks

IT

Attacks aimed at web applications in the IT sector show little change from the previous quarter. As before, SQL Injection and Cross-Site Scripting account for approximately half of attacks. These attacks are aimed primarily at obtaining sensitive information and user credentials, which can be used to access systems belonging to the target company. For example, a successful Cross-Site Scripting attack would allow a criminal to obtain a user's cookie, which would grant access to materials on the company's portal for partners. Such portals often contain sensitive information of high value to attackers. Attackers are also highly interested in the credentials of privileged users, who have administrator rights and often use the same account for accessing multiple web applications and portals. Therefore SQL Injection and Cross-Site Scripting attacks may be aimed specifically at obtaining the credentials of web application administrators. Access to partner portals and other web applications threatens to compromise the reputation, and possibly bottom line, of targeted IT companies.

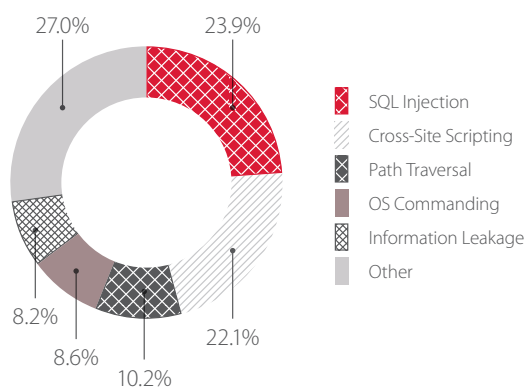


Figure 6. Types of attacks on web applications of IT companies

Government

Most of the government web applications in our statistics are intended either for handling personal data or for providing information and news to the public. One out of five attacks is aimed at unauthorized access to sensitive information; one out of two targets the users of web applications. Hackers take advantage of the fact that most users of these sites are not security-savvy. The main aim of attackers is to steal users' personal information.

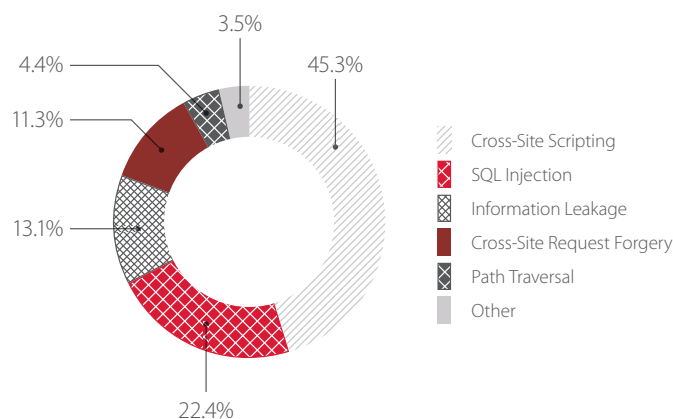


Figure 8. Types of attacks on web applications of government institutions

Average number of attacks per sector

Healthcare took the brunt of attacks in the outgoing quarter, recording the highest number of events per day of any sector. The number of attacks increased compared to the prior quarter in part because the particular sites in our dataset are used to provide healthcare information for large parts of a country where PT AF was piloted.

The significant fall in the number of attacks on government web applications is likely caused by the prevalence in this quarter's dataset of local government web applications, which receive fewer visitors than the government web applications studied in the previous quarter. Local web applications are of less interest to hackers, who are interested in maximizing the payoff of an attack and striking as many victims as possible. Regional web applications have more users and more sensitive information than local web applications; as such, attacks on the web applications of regional governments have the potential to cause greater damage.

As in the previous quarter, the average number of attacks on manufacturing companies is below two dozen, a low number that is consistent with our hypothesis regarding the targeted nature of attacks on this sector. Attackers try to remain as stealthy as possible in order to achieve their objective: access the target's internal network and laterally develop the attack to penetrate industrial control systems. This makes each attack exceptionally dangerous, with potential consequences ranging from interruption or disruption of operations to severe accidents, environmental harm, and loss of life.

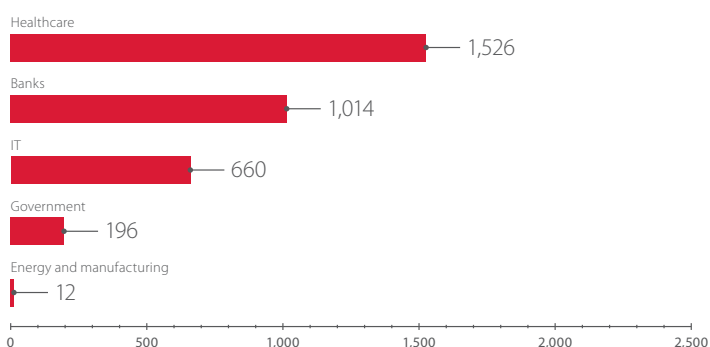


Figure 7. Average number of attacks per day, by sector

Attack examples

Results of a pilot PT AF deployment in the healthcare sector showed a high number of Local File Inclusion attempts over a period of several days. For example, after noon on July 9, attackers tried from dozens of IP addresses to run arbitrary code on the targeted server. Local File Inclusion occupies second place in our rating of the most common attacks on healthcare web applications, and third place in our rating of the most common attacks across all pilot projects.

ATTACKS					
0 to 40 of 260 available for paging					
EVENT_SEVERITY	EVENT_TAG.NAME	EVENT_DESCRIPTOR...	POLICY_NAME	MATCHED.VARIABLE...	TIMESTAMP
high	Local File Inclusion	A Local File Inclusi...	Default	REQUEST_ARGS....	2017-07-09 22:33:28
high	Local File Inclusion	A Local File Inclusi...	Default	REQUEST_ARGS....	2017-07-09 22:33:26
high	Local File Inclusion	A Local File Inclusi...	Default	REQUEST_ARGS....	2017-07-09 22:32:59
high	Local File Inclusion	A Local File Inclusi...	Default	REQUEST_ARGS....	2017-07-09 22:32:59
high	Local File Inclusion	A Local File Inclusi...	Default	REQUEST_ARGS....	2017-07-09 22:32:58

Figure 9. Local File Inclusion attempts on July 9, displayed in the PT AF interface

```
Raw request
1 GET /scripts/index.php?PathPrefix=../../../../../../../../../../../../../../../../boot.ini HTTP/1.1
2 Connection: Keep-Alive
3 Host:
4 Pragma: no-cache
5 Cache-Control: no-cache
6 User-Agent:
7 Accept: image/gif, image/x-bitmap, image/jpeg, image/pjpeg, image/png, */*
8 Accept-Language: en
9 Accept-Charset: iso-8859-1,*;utf-8
10
11
```

Figure 10. Example of a request used in a Local File Inclusion attempt, displayed in the PT AF interface

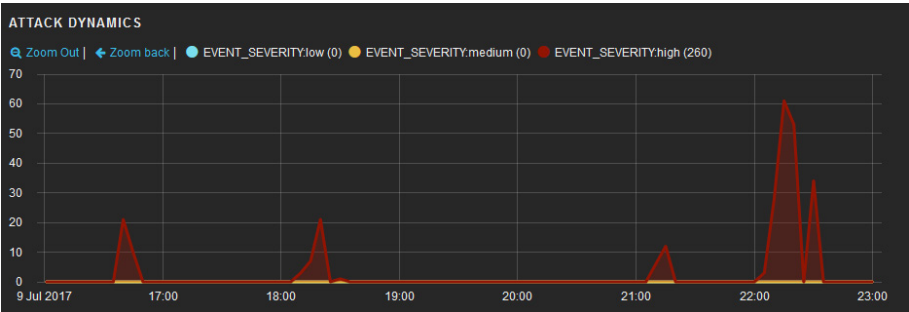


Figure 11. Graph of Local File Inclusion attempts on July 9, displayed in the PT AF interface

In another pilot project, PT AF detected an attack chain aimed at obtaining information by exploiting vulnerability [CVE-2017-9798](#) in the Apache web server. The vulnerability and relevant attack are known as Optionsbleed. An attacker can use the OPTIONS HTTP method to access fragments of memory that contain residual data from requests handled by the current process from other clients of a co-host. The vulnerability and its exploit were published on September 18. The first exploitation attempt recorded by PT AF was just three hours later, with many more on the following day. The ease of exploitation meant that hackers could make use of the vulnerability almost immediately after it was published. Moreover, a public exploit was developed and published very soon after.³

```
Raw request
1 OPTIONS / HTTP/1.1
2 Host:
3 X-Orator-IP-Source:
4 X-Orator-TCP-Info: 51927, 70000, 35000
5 X-Forwarded-For:
6 Content-Length: 0
7 Accept-Encoding: gzip, deflate
8 Accept: */*
9 User-Agent:
10
11
```

Figure 12. Example of a request used in an Optionsbleed attack, displayed in the PT AF interface

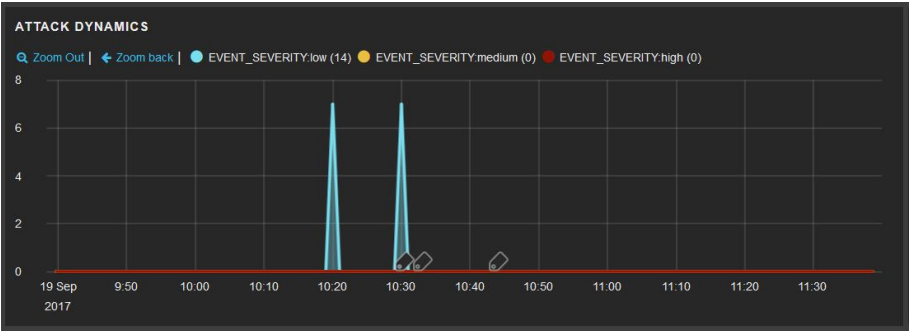


Figure 13. Graph of Optionsbleed attack attempts on September 19, displayed in the PT AF interface

³ [rapid7.com/db/modules/auxiliary/scanner/http/apache_optionsbleed](#)

On September 19, during a ten-minute window multiple Optionsbleed attacks were detected and correlated in real time by PT AF in two distinct attack chains.

ALERTS			
ALERT_SEVERITY	STATUS	ALERT_NAME	TIMESTAMP
	finished	OPTIONSBLEED Attack	2017-09-19 10:30:32
View: Basic / Advanced / Raw			
Field	Value		
Description	Alert name: OPTIONSBLEED Attack (https://blog.fuzzing-project.org/60-Optionsbleed-HTTP-OPTIONS-method-can-leak-Apaches-server-memory.html)		
Severity summary	High: 0 Medium: 0 Low: 7		
Date	Start: 2017-09-19 10:30:32 End: 2017-09-19 10:32:38		
Correlation	59c0c64605367c18e174a1e5 Default:37.48.115.230.1505806232		
Policy	Default		
Status	FINISHED		

Figure 14. Example of one of the detected Optionsbleed attack chains, displayed in the PT AF interface

ATTACKS					
EVENT_TAG.NAME	EVENT_SEVERITY	EVENT_DESCRIPTION	POLICY_NAME	MATCHED_VARIABLES	TIMESTAMP
OPTIONSBLEED	low		Default	REQUEST_METHOD	2017-09-19 10:30:28
OPTIONSBLEED	low		Default	REQUEST_METHOD	2017-09-19 10:30:29
OPTIONSBLEED	low		Default	REQUEST_METHOD	2017-09-19 10:30:30
OPTIONSBLEED	low		Default	REQUEST_METHOD	2017-09-19 10:30:33
OPTIONSBLEED	low		Default	REQUEST_METHOD	2017-09-19 10:30:35
OPTIONSBLEED	low		Default	REQUEST_METHOD	2017-09-19 10:30:36
OPTIONSBLEED	low		Default	REQUEST_METHOD	2017-09-19 10:30:37

Figure 15. Optionsbleed attacks comprising a single attack chain, displayed in the PT AF interface

The risk of Optionsbleed attacks is limited, because even a successful attack obtains only several bytes of memory, which is unlikely to yield much sensitive information. Moreover, the issue affects only co-hosted systems (when sites belonging to multiple users/companies are hosted on the same sever) and only if the .htaccess file sets the Limit directive for an HTTP method not globally registered on the server. To prevent such attacks, vulnerable versions of the Apache web server must be updated.

As in the prior quarter, attempts were also made to perform OS Commanding with the help of [CVE-2017-5638](#), a recently published vulnerability affecting the Apache Struts framework.

```

Raw request
1 GET / HTTP/1.1
2 Host:
3 Connection: keep-alive
4 User-Agent:
5 Content-Type: multipart/form-data; (#dm=@ognl.OgnlContext@DEFAULT_MEMBER_ACCESS).(#mem
berAccess=(#memberAccess=#dm):{(#container=#context['com.opensymphony.xwork2.ActionContext.container']).(#ognlUtil=#container.getInstance().com.opensymphony.xwork2.ognl.OgnlUtil@clear()).(#ognlUtil.getExcludedPackageNames().clear()).(#ognlUtil.getExcludedClasses().clear()).(#context.setMemberAccess(#dm))}).(#cmd='echo CVE-2017-5638 check!').(#iswin=(@java.lang.System@getProperty('os.name')).toLowerCase().contains('win')).(#cmds=(#iswin?{'cmd.exe','/c',#cmd}:{'/bin/bash','-c',#cmd})).(#p=new java.lang.ProcessBuilder(#cmds)).(#p.redirectErrorStream(true)).(#process=#p.start()).(#ioc=@org.apache.struts2.ServletActionContext@getResponse().getOutputStream()).(@org.apache.commons.io.IOUtils@copy(#process.getInputStream(),#ros)).(#ros.flush())}}
6

```

Figure 16. Request aimed at exploiting vulnerability CVE-2017-5638, displayed in the PT AF interface

Attackers try to stay informed about the newest published vulnerabilities and take advantage of the fact that many systems of interest do not have the most up-to-date updates installed. As a result, many attacks attempt to exploit recently discovered vulnerabilities, as happened with Optionsbleed. This phenomenon underscores the importance of updating all web application components as soon as such updates become available in order to stay ahead of attackers.

Attack trends

The statistics collected in Q3 allow reconstructing the distribution of attacks over time. To do so, the following graph depicts types of attacks on web applications for one of the companies included in the dataset. The ten most frequent attack types are shown. For each type, the number of attacks per day was calculated. The graph reflects data for the entire quarter. The results suggest which attack types stood out in terms of the number of requests sent by attackers.

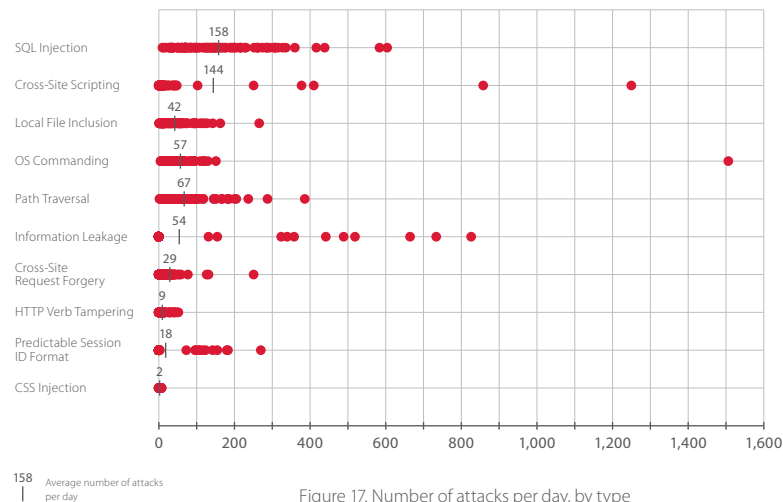


Figure 17. Number of attacks per day, by type

SQL Injection and Cross-Site Scripting stand head-and-shoulders above the others as the most frequent attack types both in this quarter and in previous quarters. Certain days recorded numbers of Cross-Site Scripting attacks that were several times the average daily value for the quarter. There were also long periods when these attacks fell significantly—to less than 20 per day.

SQL Injection and Local File Inclusion attacks remained stable throughout the quarter, generally staying below 400 and 200 per day respectively. This observation is explained by the fact that successfully performing such attacks requires bruteforcing improperly filtered characters or the names of scripts, directories, and files. Therefore a single attack comprises many such attempts, which are correlated by PT AF into a single attack chain.

On one particular day, OS Commanding attacks jumped markedly. This day-long increase in malicious requests was caused by attackers intent on taking control of a certain target's web resources.

Overall, the average number of attacks of other types was less than 70 per day.

Attacks can be broken down by day of the week as well.

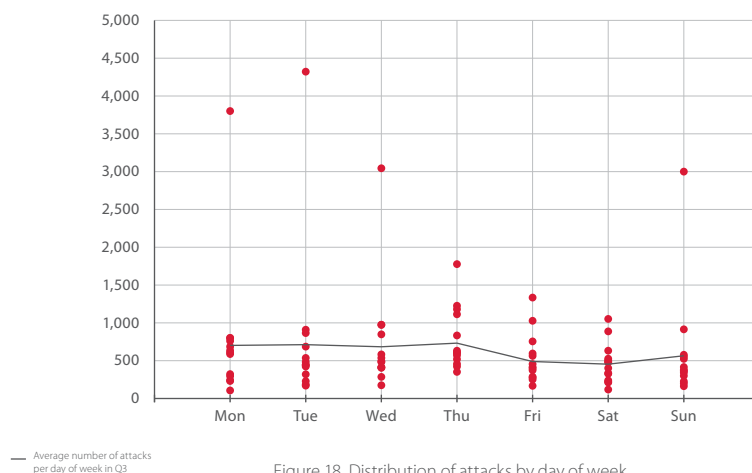
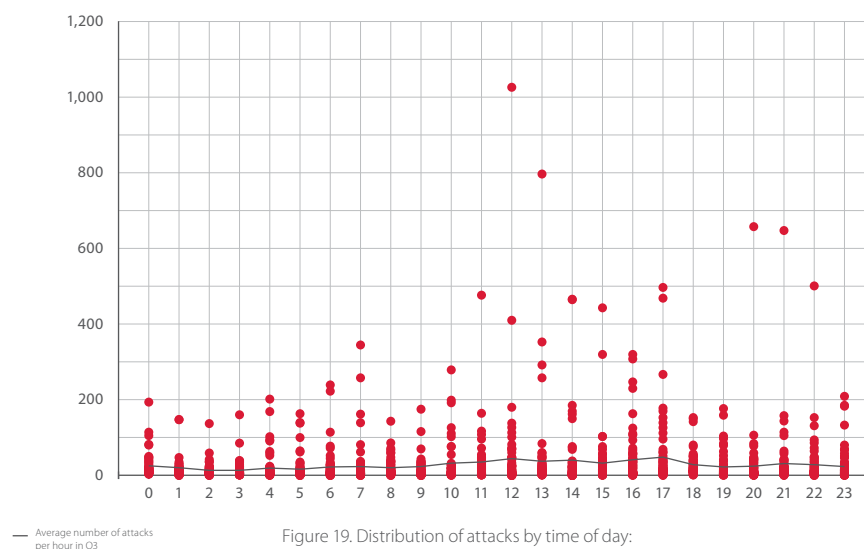


Figure 18. Distribution of attacks by day of week

Web applications were hit by 500 to 700 attacks on average per day, rarely dipping below 200. Attackers search determinedly for any opportunity to profit whether on working days or on weekends. The maximum number of attacks recorded in a single day was 4,321.

During the three-month period covered by this report, the number of daily attacks exceeded 3,000 on four different days.

Trends can be seen both by day of week and time of day. Time of day is the local time of the target.



Attacks tend to intensify in the afternoon and evening. However, spikes may occur at any time of day. As an example, below is a graph of attacks on August 15. Peaks were seen both before and after noon.

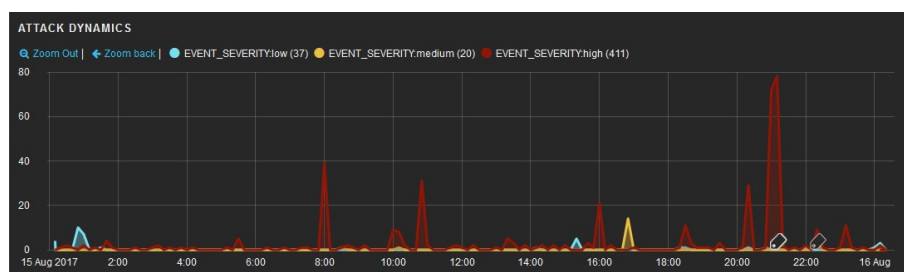


Figure 20. Graph of attack attempts on a targeted company on August 15, displayed in the PT AF interface

There are two main explanations for these findings. Most attacks occur during the afternoon and evening because ordinary web visitors are most active during these hours. By contrast, when active during the night or morning in the target's time zone, attackers are hoping to catch defenders unaware in order to slip by unnoticed. One important protection measure for consistent 24-hour detection and prevention is a web application firewall (WAF).

The functionality and sector of a web application influence the types, days, and times of attacks. But a web application does not have to be specially targeted in order to fall victim. Mass attacks, such as those intended for experimenting with and fine-tuning new exploits, may hit random sites or an entire range of IP addresses. Web application protection and security staff should take these risks into account.

CONCLUSIONS

Web applications of various types continued to tempt criminals in Q3 2017. As seen in previous quarters, a large proportion of attacks were aimed at sensitive information and users of web applications.

Attackers are active 24/7, including on weekends and holidays. They continue to collect information about vulnerabilities, rapidly developing and testing exploits to seize any opportunity. Many companies still fail to quickly update web application components and install necessary patches. The result is that attackers are able to slip through defenses by using already known vulnerabilities. Minimizing the likelihood and possible damage of such attacks requires two main steps: updating software on a timely basis and using proactive solutions such as a web application firewall to detect and prevent attacks on corporate systems.

About Positive Technologies

Positive Technologies is a leading global provider of enterprise security solutions for vulnerability and compliance management, incident and threat analysis, and application protection. Commitment to clients and research has earned Positive Technologies a reputation as one of the foremost authorities on Industrial Control System, Banking, Telecom, Web Application, and ERP security, supported by recognition from the analyst community. Learn more about Positive Technologies at ptsecurity.com.

© 2017 Positive Technologies. Positive Technologies and the Positive Technologies logo are trademarks or registered trademarks of Positive Technologies. All other trademarks mentioned herein are the property of their respective owners.