



Attacks on web applications: **2018 in review**

Contents

Introduction	2
Executive summary	3
Overall statistics	3
Statistics by sector	5
IT	6
Financial institutions	7
Transportation companies	8
Hospitality and entertainment	9
Government institutions	10
Education and science	11
Conclusions	12
About the research	13

Introduction

As seen in our [study of 2018 cyberthreats](#), web application hacking is one of the most frequent attacks on both organizations and individuals. Hacked sites can be used for a multitude of things: distributing malware, stealing data, posting ads or forbidden information, committing fraud, or penetrating an internal network. In this report, we have turned the spotlight on the main threats to modern web resources, based on web attack statistics collected by PT Application Firewall. More details on the dataset can be found at the end of this document. In addition, we dive into attack statistics for particular sectors. This granularity may assist security specialists in assessing the risks relevant to corporate websites.

Executive summary

All sites, regardless of sector, are attacked on a daily basis. In the case of a targeted attack, it is possible to correlate the different stages and reconstruct the attack chain. The greatest daily number of these chains was found on the sites of financial institutions (151), followed by the sites of transportation companies (135) and hospitality and entertainment (114).

The three most common attacks on websites have remained the same for many years: SQL Injection, Path Traversal, and Cross-Site Scripting (XSS). The situation becomes more interesting when we break out statistics at the per-sector level.

The share of Information Leakage attacks has doubled. This is especially evident with government sites, where such attacks make up 67 percent of the total. Sites of government institutions are also still exposed to dangerous attacks aimed at gaining server control and stealing database information.

Attacks on sites of IT companies were mostly aimed at obtaining data and taking control of the application. Meanwhile, financial institutions suffered primarily from attacks on their clients, the most common attack being XSS (29% of all attacks on finance sites). Hospitality and entertainment companies and educational institutions were hit by similar types of attacks.

Overall statistics

The most common attacks change only minimally from year to year: SQL Injection, Path Traversal, and XSS still top the list. Combined, they account for more than half of all detected cyberattacks on corporate web resources. Why? There are two main reasons. One is that these attacks work. Our [study](#) shows that three quarters of sites are vulnerable to XSS attacks. Half of web applications have access control issues and one third are susceptible to code injection. The other factor is that such attacks are easy to perform. They can be carried out by low-skilled hackers, sometimes even automatically, using publicly available software.

The only difference from the prior year is the order of the attacks in the top three. Naturally, the choice of which applications to include in the dataset has a large effect on the order of these attacks in the list. If a particular application does not support user data input, then instead of trying to subvert the application's logic through malicious input, hackers will try other attack methods.

Analysis of 2018 attacks revealed a few curious trends. Among them: the share of Information Leakage attacks doubled, especially in the context of government sites. At the same time, the share of XSS attacks fell significantly. Despite all this, we cannot conclude that attacks on clients are losing popularity. The change is more likely due to an increase in other attacks, such as Path Traversal. The following data will demonstrate that across sites and sectors, XSS still retains its place near the top of the list of most common attacks.

Vulnerability scanning and even attacks are a daily occurrence for any site connected to the Internet. Scans do not necessarily imply that someone is trying to hack a site. Malefactors may be simply checking accessible addresses for vulnerable systems, or testing a new script on a wide range of addresses.

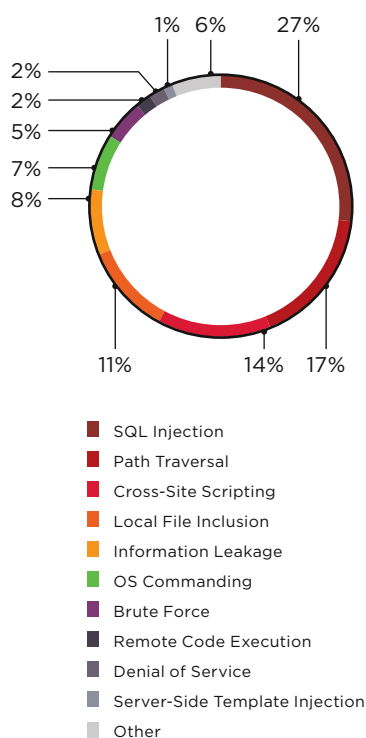


Figure 1. Top 10 web application attacks

To get information or gain access to a specific site, an attacker will have to do more than just a handful of scans, which otherwise might appear to be one-off events. These efforts are needed to figure out which vulnerabilities are present in the victim's system. So in the case of a targeted hacking attack, events—even ones occurring at different times—are assembled into a chain. A single chain can contain dozens, hundreds, or even thousands of events a day. The ability to detect the whole chain of related events enables localizing the threat and protecting resources effectively. Using the capabilities of PT Application Firewall, we were able to find connections between attacks and put them together. This allowed us to calculate and compare the average number of targeted attack chains on sites of companies from various sectors.

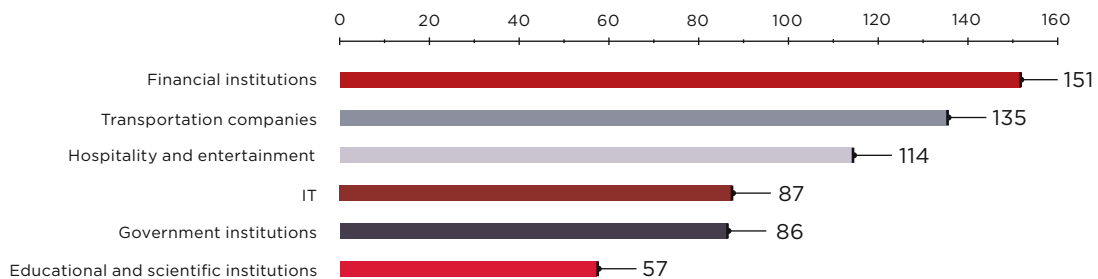


Figure 2. Average number of targeted attacks per day on a single web application

To make attack detection and blocking more difficult, hackers can hide the IP address of the request source, User Agent data, and other identifiers. These techniques are easily performed with the help of automated scripts and traffic proxying through third-party servers. These proxy servers can be in far-flung countries, leaving defenders in the dark about attackers' real location. Attack attribution becomes more difficult and, without additional data, futile. By way of illustration, in our study we recorded a total of over 12,000 attack sources from 90 countries.

Statistics by sector

Now let's see **which attacks were typical for web applications depending on the target company's line of business.**

IT

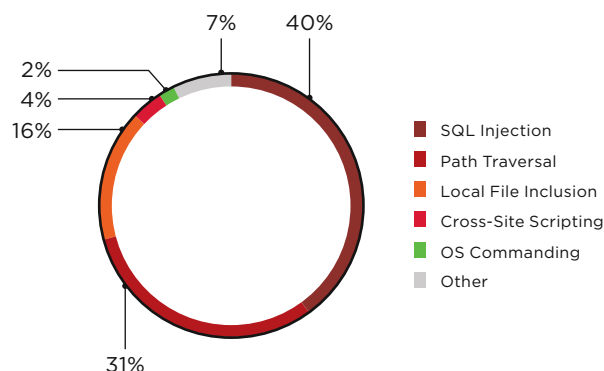


Figure 3. Top five attacks on web applications of IT companies

The overwhelming majority of attacks on IT companies were aimed at obtaining information. Generally, this means SQL Injection and Path Traversal. The first of these allows a hacker to get information from the application database (such as user credentials, personal data, and information about the database and its tables) and even run commands on the server. If successful, this attack can often completely compromise the site and yield control of the server. Using Path Traversal, the attacker can view the contents of directories that should not be accessible to a regular user, even one who is logged in. These directories can contain such important data as configuration files for the software and OS, or site source code. For this attack, the perpetrator needs to know (or bruteforce) the correct name and path to a file of interest. This is why such Path Traversal is usually not limited to a single attempt, so we can find a whole chain of similar events in the protection console. Most often, hackers attempt to read the file `/etc/passwd`, which is used to store information about users on Linux systems. In the screenshots that follow, you can see the attacker trying to find this file with the help of dots and slashes (which are shown as `%2f` in URL Encoding format) in GET requests, in an attempt to move "up" a directory in the file system.

```
1 GET /... source=..%2fetc%2fpasswd%00 HTTP/1.1
2 Accept: Text/Html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
3 Connection: keep-alive
4 User-Agent: Mozilla/5.0 (Windows NT 10.0; WOW64; Rv:50.0) Gecko/20100101 Firefox/50.0
5 Accept-Language: en-us,en;q=0.5
6 Referer: http://... .com/... source=..%2fetc%2fpasswd%00
7 Accept-Encoding: gzip
```

Figure 4. Attempt to read `/etc/passwd` by moving one directory up from the current one

```
Скачать
1 GET /... source=../../../../..%2fetc%2fpasswd%00 HTTP/1.1
2 Accept: Text/Html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
3 Connection: keep-alive
4 User-Agent: Mozilla/5.0 (Windows NT 10.0; WOW64; Rv:50.0) Gecko/20100101 Firefox/50.0
5 Accept-Language: en-us,en;q=0.5
6 Referer: http://... .com/... source=../../../../..%2fetc%2fpasswd%00
7 Accept-Encoding: gzip
```

Figure 5. Attempt to read `/etc/passwd` by moving six directories up from the current one

Both of these attacks are extremely dangerous. If the vulnerable application processes user data, information leaks are inevitable. According to [our statistics](#), 46 percent of all attacks on IT companies in 2018 targeted web resources, and 40 percent of attacks were aimed at data theft. So IT companies need to pay attention to protecting their sites and securing the information they process. If a site is located inside the corporate infrastructure (instead of being hosted separately), the company needs to minimize the risk of attacker access to the local network that could result from a compromise of the web application.

Financial institutions

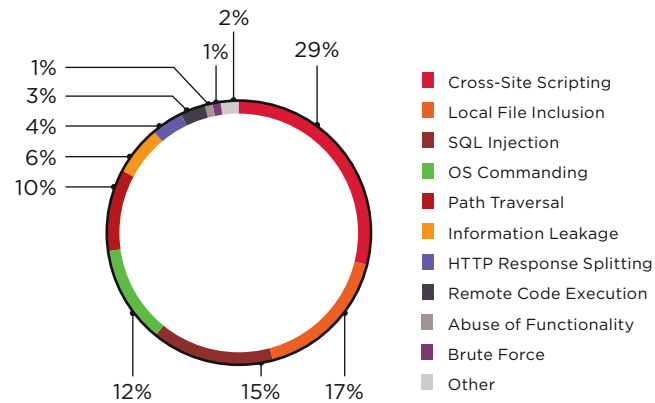


Figure 6. Top 10 attacks on web applications of financial institutions

Risk-wise, web applications of financial institutions are different from the rest. With banking services, users can manage their finances, pay bills, add to savings, get loans, and transfer funds. But even if we leave e-banking out of the equation, the consequences of a site compromise are still quite high. For instance, if the official bank site is used to distribute malware or stage phishing attacks, clients will be the first to suffer. Attacks on clients, such as XSS, head the list of attacks on web applications of financial institutions. XSS was one of the most common [web application vulnerabilities](#) in 2018.

For the financial institution itself, the risk of an attack on clients is less about financial losses per se, and more about damage to reputation. So when a bank performs a security assessment or penetration testing, the most dangerous vulnerabilities are the first to get fixed. This makes it critical to use a web application firewall (WAF): such protection will keep users secure even if the application itself is vulnerable.

Transportation companies

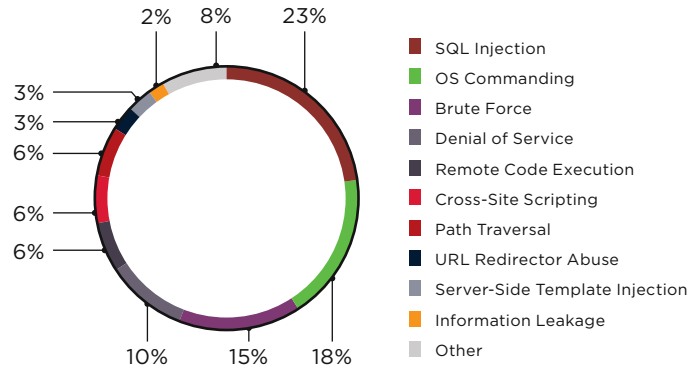


Figure 7. Top 10 attacks on web applications of transportation companies

These days, transportation companies offer more than just timetables and pricing information on their websites. Usually they include support for online payments, such as purchasing tickets. This attracts hackers interested in users' payment cards. A vivid example of one such attack is [theft of client data from British Airways](#) in August and September of 2018. Attackers hacked the site and modified a script by adding their own code (called a JS sniffer). This resulted in theft of data, including card information, for approximately 380,000 clients.

To manage a feat like that, the attacker needs to either take control of the server or gain access to the web application administration page. Our study shows that more than 50 percent of attacks, including common bruteforcing or vulnerabilities in obsolete software, can compromise the websites of transportation companies in some way. The following picture shows how attackers tried using an [exploit](#) for a known vulnerability in a popular framework to seize control of the server.

```
1 GET /signin.action HTTP/1.1
2 Accept-Encoding: gzip;q=1.0,deflate;q=0.6,identity;q=0.3
3 Accept: */*
4 User-Agent: Mozilla/5.0 (Windows NT 6.3; WOW64; Trident/7.0; rv:11.0) like Gecko
5 Host:
6 Content-Type: %({{nike='multipart/form-data'}}).{{dm=@ognl.OgnlContext@DEFAULT_MEMBER_ACCESS}}.{{# memberAccess?{{# memberAccess=dm}}:({{container=#context['com.opensymphony.xwork2.ActionContext.container']}}.{{ognlUtil=#container.getInstance(@com.opensymphony.xwork2.ognl.OgnlUtil@class)}}.{{ognlUtil.getExcludedPackageNames().clear()}}.{{ognlUtil.getExcludedClasses().clear()}}.{{context.setMemberAccess(dm)}}).{{cmd='ping -c 3 '}} || ping -n 3 '}}.{{#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}}.{{#ros=@org.apache.struts2.ServletActionContext@getResponse().getOutputStream}}.{{@org.apache.commons.io.IOUtils@copy(@process.getInputStream(),#ros)}}.{{#ros.flush}}}}}}}
```

Figure 8. Attempt to exploit vulnerability CVE-2017-5638 in the Apache Struts 2 framework

Any company providing the ability to pay online must consider and minimize such risks.

Hospitality and entertainment

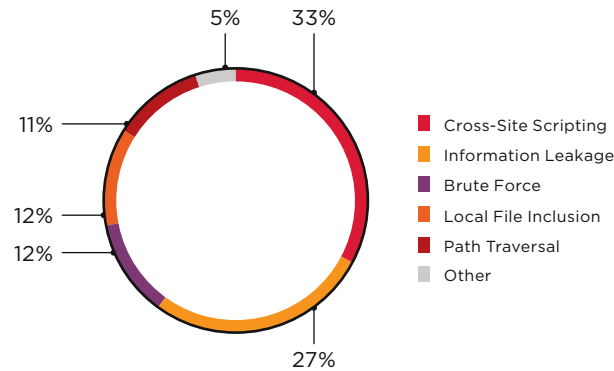


Figure 9. Top five attacks on web applications of hospitality and entertainment companies

Hospitality and entertainment sites typically emphasize convenience. For instance, hotel sites contain a booking form for the client to enter personal data. And surely any visitor wants the site to have an easy-to-use payment form. This means that if a sniffer is injected into the site's code, hackers can strike it rich with clients' payment information. In this sense all web applications of this sort are oriented toward clients, and user data must be treated as a valuable asset.

Any data leak or damage to company reputation caused by an attack may significantly hurt growth and competitive standing. Our study confirms that most cyberattacks on hospitality and entertainment sites target either clients or the data processed by the web application. A serious attitude toward security is essential for avoiding lasting harm to reputation. A security-first mindset should be part of the site development process: when creating the technical assignment and accepting code from developers, teams must make security just as important as flashy design, ease of use, and long feature lists.

Government institutions

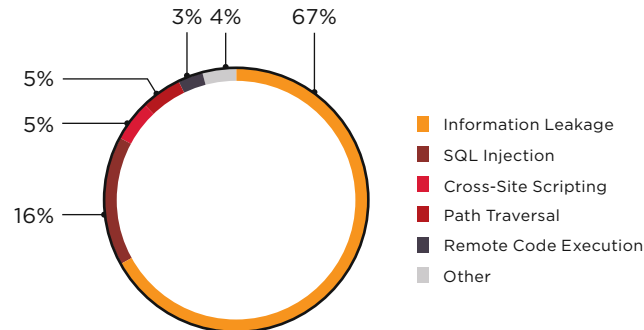


Figure 10. Top five attacks on government web applications

In 2017 we tended to see typical attacks on government sites, primarily Cross-Site Scripting and SQL Injection. In 2018 we saw quite a lot of these attacks (as well as extremely dangerous Remote Code Execution attempts for gaining control of the server). But also we saw Information Leakage attacks designed to obtain information about the web application. There were numerous attempts to access .svn or .git directories, which may store the application's current source code. With access to these files, an attacker can analyze the site for vulnerabilities that usually cannot be found by an average external attacker. In effect, the attacker would have an opportunity to see the site the way its developer did—and see every mistake the developer might have made.

```
1 GET /.git/config HTTP/1.1
2 User-Agent: Mozilla/5.0 (Windows; U; Windows NT 6.1; en-US; rv:1.9.2.12) Gecko/20101026 Firefox/3.6.12
3 Host:
4 Accept: text/xml,application/xml,application/xhtml+xml,text/html;q=0.9,text/plain;q=0.8,image/png,*/*;q=0.5
5 Cache-Control: no-cache
6 Connection: Keep-Alive
7 Keep-Alive: 300
```

Figure 11. Attempt to access .git directory files

```
1 GET /.svn/entries HTTP/1.1
2 Host:
3 Connection: keep-alive
4 Accept: */*
5 User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10_14_2) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/71.0.3578.98 Safari/537.36
6 Accept-Encoding: gzip, deflate, br
7 Accept-Language: zh-CN,zh;q=0.9,en;q=0.8
```

Figure 12. Attempt to access .svn directory files

Insecure storage of directories with data relating to project files and changes is a well-known issue. We mentioned it back in our [2017 study](#), where we gave an overview of the most common methods used by external attackers to infiltrate a local network. Many companies are still vulnerable. To test whether your site is vulnerable, try accessing the following two addresses: <http://example.com/.git/HEAD> and <http://example.com/.svn/entries>. If the application is secure, you will see an error message stating that the page was not found. If no error message appears, act quickly to delete the .git and .svn directories. They are not necessary for site operation, and their presence can be considered an administration error.

Education and science

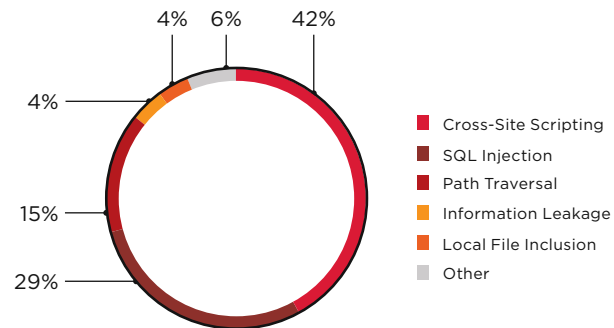


Figure 13. Top five attacks on web applications of educational and scientific institutions

Attacks on educational and scientific institutions follow the overall trends described in the beginning of this report (see Figure 1). In [our 2018 cybersecurity review](#), we noted that hackers attack such sites for more than just information on employees or students, or upcoming exam questions. Valuable data on new research is at stake too. Web application hacking, one of the top five common attack methods, was used in 12 percent of attacks on sites of universities and other educational institutions. When securing intellectual property from theft, always consider the risk of an attack via vulnerable websites.

Conclusions

The most common attacks are simple and effective: SQL Injection, Path Traversal, and Cross-Site Scripting. Hackers do not necessarily aim at obtaining access to a particular site. Increasingly, they target the personal and payment data of clients. We also see attacks aimed at infecting a site with malware, which allows the hackers to reach more victims and use vulnerable sites for targeted infection in a watering hole attack.

Our results confirm that across all sectors, the site of any company is at risk every day. To secure resources and safeguard reputation with clients, companies must take preventive measures for protection. Our study of corporate infrastructure security revealed that a hacker typically needs only one or two vulnerabilities to penetrate the internal network, and that 75 percent of penetration vectors are based on weaknesses in website security.

We recommend the following measures to protect web applications:

- Use a web application firewall (WAF). This software will secure the web application even if the application contains vulnerabilities. It will also protect the application from new threats that might arise in the future. A WAF can do more than protect from known attacks at the level of application and business logic: it also detects attempts to exploit zero-day vulnerabilities, prevents attacks against users, and analyzes and correlates events in order to build attack chains. Performing these tasks requires advanced implementations of normalization, heuristics, machine learning, and behavioral analysis. Another useful function is interaction with external security information and event management (SIEM) systems and notification to network-level DDoS protection tools. The WAF can be integrated with automated source code analysis tools, which allows using virtual patching to shield a vulnerability from attack until it is fixed in code.
- Perform security assessment of web applications regularly, and fix any vulnerabilities it reveals. White-box testing (when security specialists have access to source code) yields the best results. We recommend this testing at every stage of the site development lifecycle. It should not be one-time occurrence during code acceptance or right before going live.
- Do not use obsolete versions of web servers, operating systems, content management systems, libraries, or other software. Update systems regularly and install the latest patches.
- Log and investigate all incidents for timely identification of the threat source and minimization of risks.

About the research

Positive Technologies is a leading provider of enterprise security solutions for vulnerability and compliance management, incident and threat analysis, and web application protection on the European market. Every year we complete hundreds of pilot projects and deployments of our products on the information systems of leading companies. This report is based on data for 28 web applications protected with PT Application Firewall located on the network border of organizations active in diverse lines of business. The purpose of our study was to demonstrate current trends in attacks on web applications in 2018. The dataset consists of pilot projects where clients consented to use of results for research purposes, as well as from PT Application Firewall data for our company's own sites (which were classified under the IT sector).

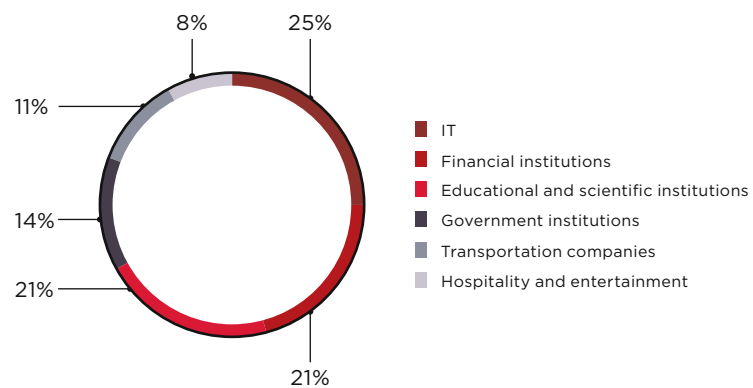


Figure 14. Client snapshot

The study covered a total of around 140,000 attacks. This number does not include incidents involving web vulnerability scanners, which were excluded from consideration. We also excluded the false positives inevitable in any protection system.

Our conclusions are not necessarily indicative of the current state of information security at other companies in the relevant sectors. The purpose of this document is to make information security specialists in various sectors aware of the most important issues in web application security, and to help them define and minimize risks in a timely manner.

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.

ptsecurity.com
info@ptsecurity.com

© 2019 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.