POSITIVE TECHNOLOGIES

Vulnerabilities in online banking applications

2019

Contents

Introduction

This report draws on the 2018 work of Positive Technologies experts in security analysis of web applications for online banking. Based on their findings, the statistics in this document highlight the most common security issues with online banks and compare the results to 2017.

The findings indicated here do not necessarily reflect the current state of other companies in the same sector. Rather, this information is intended to promote a better understanding among information security specialists of the most relevant issues in a particular sector, as well as assist in timely detection and remediation of vulnerabilities.

Executive summary

Most online banks contain critical vulnerabilities. Among tested online banks, 61 percent have a poor or extremely poor protection level.

All online banks are at risk. Every tested online bank had vulnerabilities with potentially serious consequences. For instance, fraud and theft of funds were possible in 54 percent of applications.

Two-factor authentication (2FA) is poorly implemented. Two-factor authentication mechanisms at 77 percent of online banks contained flaws.

Off-the-shelf solutions are less vulnerable. On average, solutions purchased from vendors contained three times fewer vulnerabilities than software developed by banks on their own.

Production systems are just as vulnerable as testbed systems. In most cases, both types of systems contain at least one critical vulnerability.

Trends

The proportion of high-risk vulnerabilities is steadily decreasing. In 2016, 36 percent of vulnerabilities were critical. In 2017 this number fell to 32 percent, and in 2018 critical vulnerabilities were only 15 percent of the total.

Insufficient Authentication is lessening in relevance as a critical vulnerability. The share of online banks where important operations could be performed without logging in has fallen every year, and at last, in 2018 we could not find any application still having this problem. But on many systems, highly important operations are still carried out without 2FA.

Personal data of clients and sensitive bank information are at risk at every tested online bank. Every year we see an increase in the share of systems at risk of unauthorized access to sensitive bank information and clients' personal data. In 2018, this number reached its maximum: this threat was found at each tested online bank.

Overall statistics

Every online bank was at risk of unauthorized access to sensitive bank information and clients' personal data. Fraud and theft were possible at 54% of online banks

- Implement OAuth 2.0 correctly
- Follow RFC 6749 security recommendations
- Use whitelists to protect from redirect_uri spoofing

Attackers can use a number of vulnerabilities to gain unauthorized access to clients' personal data and, in some cases, sensitive bank information such as account statements and payment orders. Every online bank analyzed in 2018 had at least one vulnerability enabling such access. This threat is particularly relevant for applications harboring authentication and authorization mechanism flaws. Online banking developers often make errors in implementing single sign-on (SSO) based on the OAuth 2.0 protocol, which can lead to interception of credentials sent via an insecure protocol and session hijacking by an attacker.

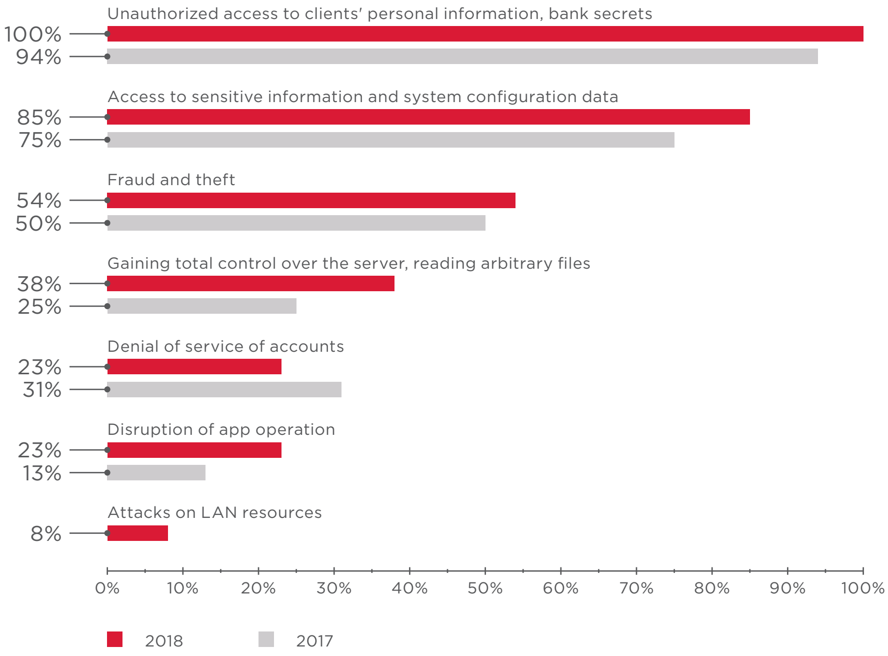

Figure 1. Potential impact of attacks on online banks (percentage of online banks)

Require a minimum amount for currency conversions and carefully check the formula used to calculate the final amount

Don't pass serialized objects via parameters that can be easily forged by an attacker, or else digitally sign them and verify on the server side

Fraud and theft are most often possible due to errors in operating logic. For instance, so-called currency rounding attacks if applied iteratively can lead to significant losses for banks. The vulnerability is well known and exists because of an error in rounding during repeated conversion back and forth between currencies.

Along with critical vulnerabilities such as Arbitrary Code Execution or Deserialization of Untrusted Data, our specialists sometimes found an interface on the bank's server with the address of the bank's internal network. Knowing this address, a malefactor can attack corporate infrastructure.

61% of online banks have a poor or extremely poor protection level

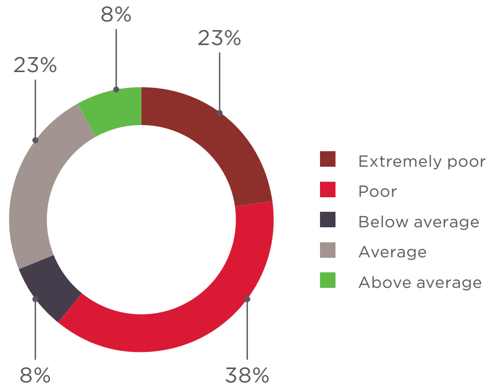

Figure 2. Security level of online banks (percentage of online banks)

The percentage of critical vulnerabilities has fallen by more than by half compared with 2017

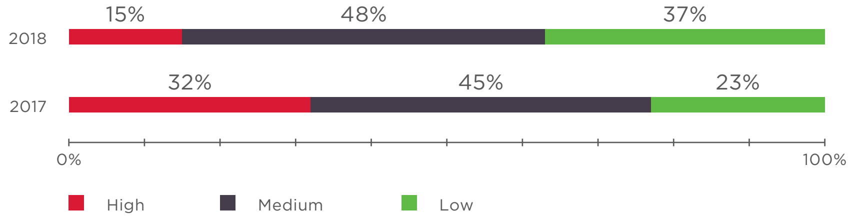

Figure 3. Vulnerabilities by severity level (percentage of vulnerabilities)

The average number of vulnerabilities in a single online bank nearly doubled compared with 2017, but the average number of critical vulnerabilities per bank remained stable

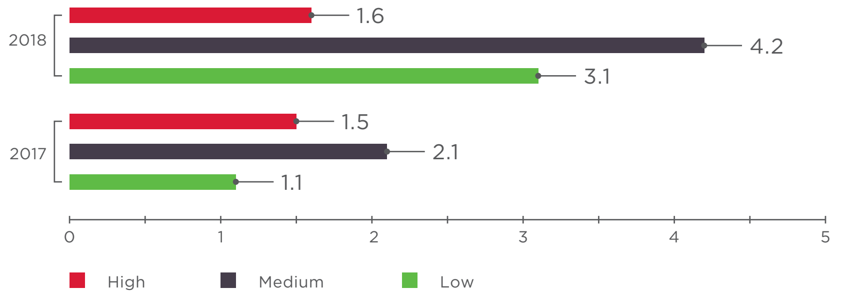

Figure 4. Average number of vulnerabilities per online bank

- Require one-time passwords for all critical actions
- OTPs should have a short life time (maximum two minutes) and be tied to the action with an additional random parameter unique to the action identifier

Let's consider some of the vulnerabilities found by our experts in more detail. In 2018, none of the tested online banks suffered from Insufficient Authentication, and Insufficient Authorization was found much less frequently than in the year prior.

By contrast, flaws in implementation of two-factor authentication mechanisms have taken the lead. For instance, some online banks do not require one-time passwords (OTPs) for critical operations (such as authenticating or changing credentials), or set password lifetimes that are excessively long. In our opinion, these gaps occur when banks consider themselves forced to choose between safety and ease of use. The need to enter one-time passwords many times during a single session may be frustrating for users.

Thanks to ease of implementation and the opportunity to save on SMS messages with OTPs, online banking systems nowadays frequently use adaptive authentication mechanisms as part of a risk-based authentication model. But unavoidably, disabling even some security features in favor of convenience increases the risk of fraud. If there is no need to confirm an operation with a one-time password, the attacker no longer needs access to the victim's phone, and a password that expires after a long time is more easily bruteforced.

The share of attacks where a hacker can affect operating logic increased to 31 percent in 2018 (versus 6% in 2017). Most likely, this is due to the increased number of vulnerabilities in in-house applications. As shown later in this report (Figure 11), in 2018 the share of such vulnerabilities was 59 percent, while in the prior year it was 39 percent.

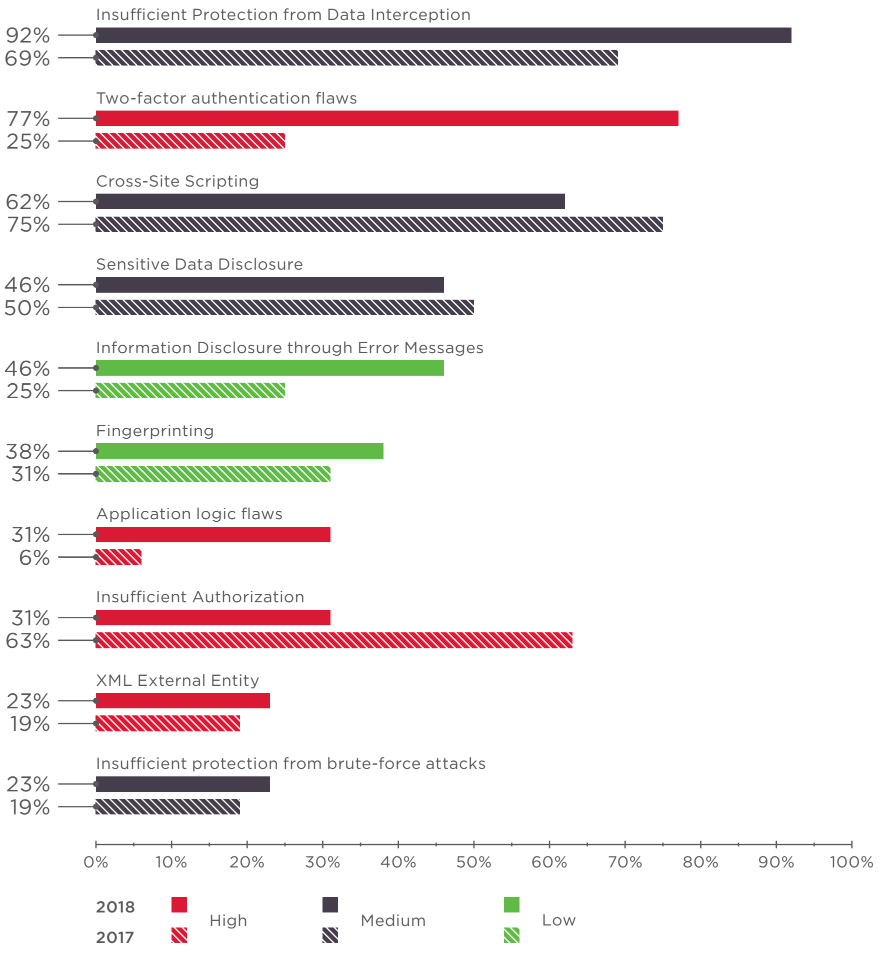

Figure 5. Top online banking vulnerabilities (percentage of online banks)

If the application does not use HSTS, and cookie parameters are not protected with the Secure and SameSite flags, an attacker can intercept a user's session ID and gain access to the user's account and bank information

To prevent interception of sensitive data and attacks on users, modern browsers support a number of mechanisms. To name a few:

- HTTP Strict Transport Security (HSTS) is a mechanism for forcing connections via the secure HTTPS protocol. The mechanism is activated by the Strict-Transport-Security header in the HTTP response of the server.
- HTTP Public Key Pinning (HPKP) is a technology that prevents connection to a web server if a hacker has spoofed the SSL certificate. This mechanism is activated by the Public-Key-Pins header.
- Content Security Policy (CSP) is a mechanism ensuring protection from attacks involving content injection, such as Cross-Site Scripting. This mechanism is activated by the Content-Security-Policy header.
- X-Content-Type-Options is a header for protecting a user's browser from attacks that spoof the MIME type of content.
- X-Frame-Options is a header to protect from Clickjacking .

- The Secure flag requires that cookies must be transmitted only via HTTPS—if the flag is not set (by selecting "true" for the requireSSL property), cookies could be intercepted
- Setting the SameSite attribute to Strict mode prevents cookies from being sent to third-party sites and protects against Cross-Site Request Forgery

- Use the Public-Key-Pins and Strict-Transport-Security headers
- Prohibit use of out-of-date browser versions and browsers that allow trusting forged certificates

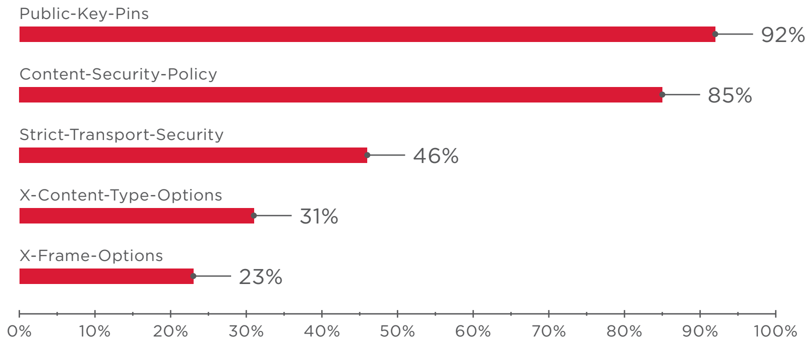

Figure 6. Applications without proper server headers (percentage of online banks)

Comparison of in-house and off-the-shelf applications

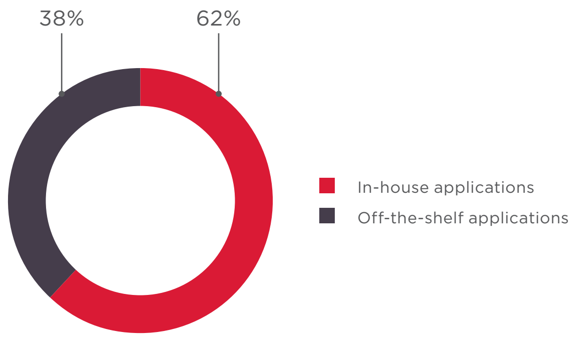

Figure 7. Types of online banks

Online banking systems developed by banks are more vulnerable than off-the-shelf solutions

The average number of vulnerabilities in in-house applications is three times more than in software from vendors

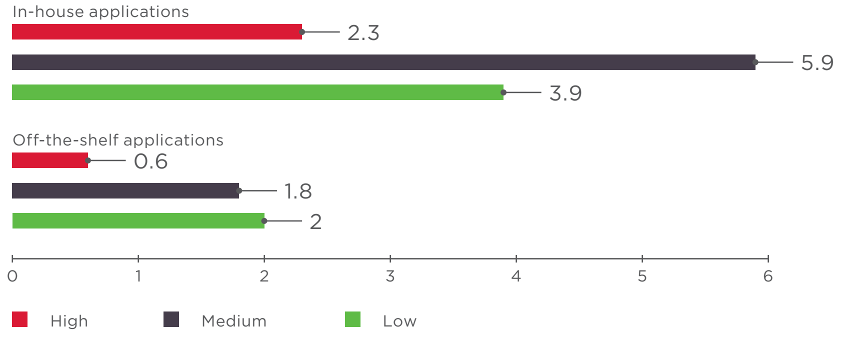

Figure 8. Average number of vulnerabilities per application

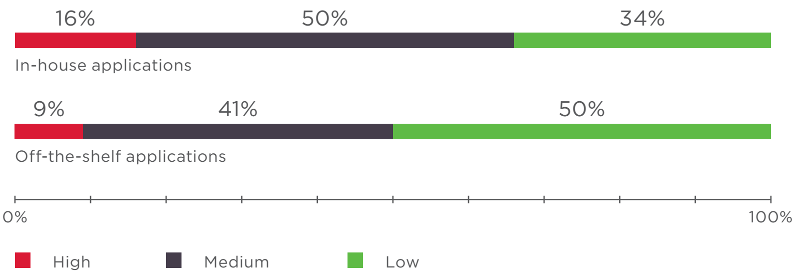

Figure 9. Vulnerabilities by severity level (percentage of vulnerabilities)

Identified vulnerabilities were divided into the following groups:

- Vulnerabilities in web application code (errors made by the developer)
- Errors in security mechanism implementation (unlike vulnerabilities in code, these bugs appear during the design stage)
- Configuration flaws

The first group includes flaws such as Cross-Site Scripting and SQL Injection. Insufficient Brute-Force Protection and Insufficient Authorization are examples of vulnerabilities in protection mechanisms. The most common configuration flaws are Disclosure through Error Messages or Web Server HTTP Header Information Disclosure.

Most vulnerabilities, both in off-the-shelf solutions and in in-house apps, are web application code vulnerabilities. But while vendors are more likely to make an error during the design stage, vulnerabilities in in-house solutions tend to occur during coding.

The vast majority of vulnerabilities, in both vendor and in-house apps, are code vulnerabilities

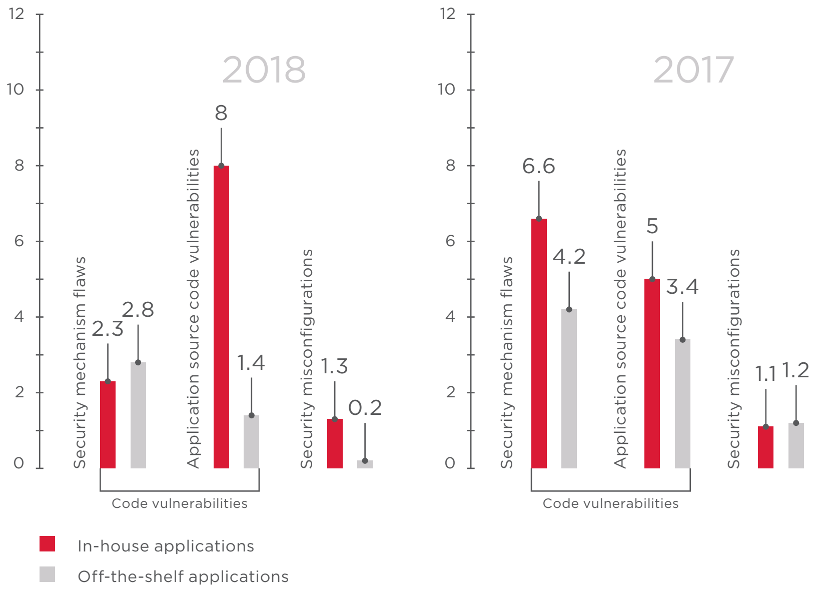

Figure 10. Average number of vulnerabilities per application

Companies developing online banking systems are more concerned with functionality than security: 75% of vulnerabilities in off-the-shelf solutions are flaws in protection mechanisms

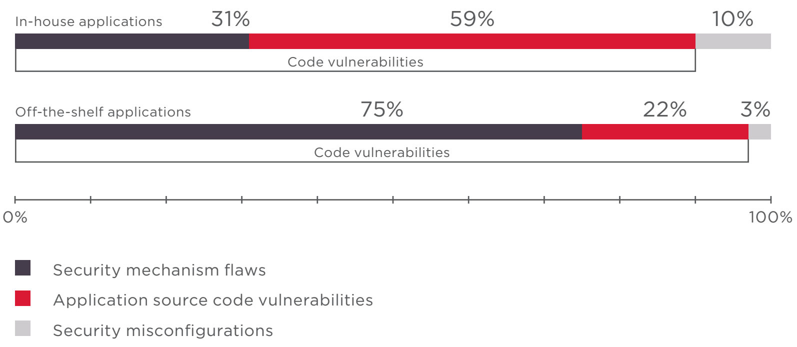

Figure 11. Vulnerabilities by category (percentage of vulnerabilities)

Comparison of production and testbed applications

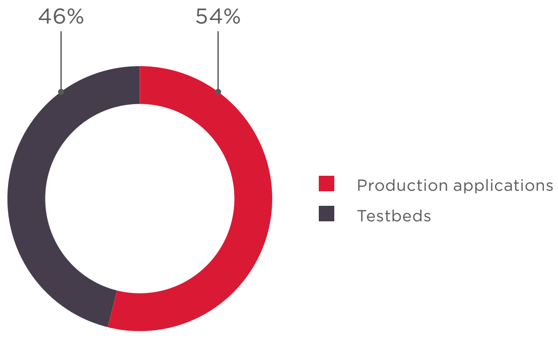

Figure 12. Shares of testbed versus production systems among tested applications

Production systems contain about the same number of vulnerabilities as testbed systems

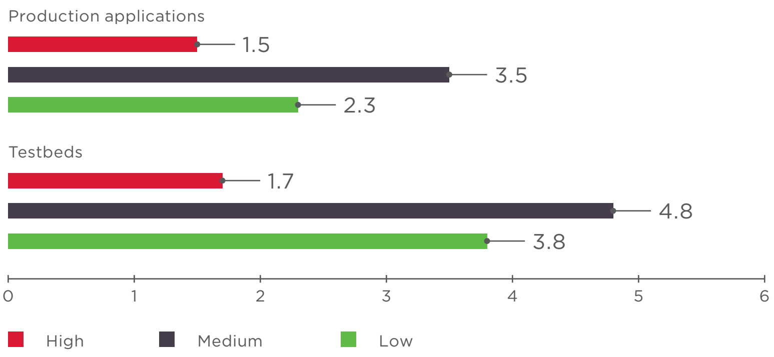

Figure 13. Average number of vulnerabilities per application

Regularly analyze online banking web applications at every stage of development—and for maximum throughness, don't neglect access to source code (white-box testing)

After application security is tested and vulnerabilities have been remediated, at some point the developers get to work again: perhaps to modify or optimize the web application, such as to add new features. Small changes in code may seem harmless from a security standpoint. Testing of these incremental changes is limited to functional testing of the new capabilities, with no new security assessment performed. Yet over time, a significant number of vulnerabilities will appear in the production system—perhaps in numbers comparable to those found during initial security testing.

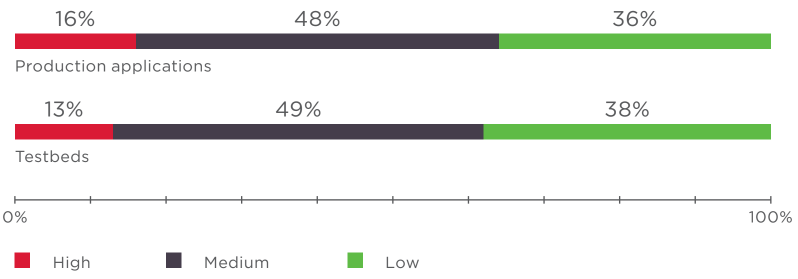

Figure 14. Vulnerabilities by severity level (percentage of vulnerabilities)

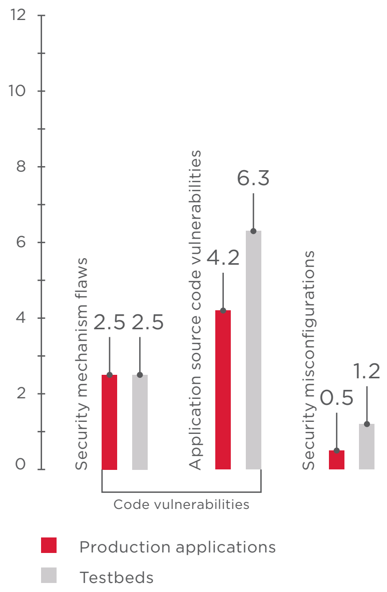

Figure 15. Average number of vulnerabilities per application

Conclusion

The main positive trend in online banking security in 2018 is the reduced percentage of high-severity vulnerabilities. However, the overall security level of online banks remains low.

Without a doubt, one of the most serious potential consequences of an attack is theft of funds. In 2018, this threat was present at 54 percent of online banks. The threat of unauthorized access to clients' data and sensitive bank information was found at every tested bank. In some cases, vulnerabilities allowed escalating the attack up to penetration of the corporate infrastructure.

Off-the-shelf online banking solutions tend to have better security than in-house applications, but their developers make more frequent mistakes in protection mechanisms in the rush to get product functionality out the door.

Changes to code slip by without new testing being performed, ultimately making production systems just as vulnerable as testbed systems. This goes to show that security processes need to be established at every stage of the online bank lifecycle. Implementing a Secure Software Development Lifecycle (SSDLC) prevents a wide range of errors, but still does not eliminate the need for regular assessment of web application security. White-box analysis, because it includes testing of source code, is more effective than gray- and black-box methods. As a preventive measure, we also urge use of a web application firewall (WAF) to prevent exploitation of vulnerabilities caused by code changes.

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