PSD2 - WILL IT BE A GOLD RUSH FOR CYBERCRIMINALS?

By Pedro Fortuna, CTO at Jscrambler (www.jscrambler.com)

PSD2 (Revised Payment Service Directive) went live earlier this year. This new EU directive effectively allows a different set of businesses to compete with the banks for access to customer data - no more monopoly for the banks when it comes to customer account information and payment services.

PSD2 allows banking customers (consumers and businesses) to use third-party providers (TPPs) to manage their finances. For example, you could use Facebook or Google to pay your bills or make a P2P transfer. Banks are obliged to provide TTPs with access to their customers' accounts through open APIs (application program interfaces). This will allow third parties to build financial services on top of banks' data and infrastructure. Customers now face the challenge of relying on both banks AND 3rd party providers to safeguard their sensitive data and also be confident in the measures that TPPs take to collect the information they need. It's time to think about what new security measures need to be put into place..

PROTECTING THE CLIENT-SIDE

Banks have traditionally been targets of Man-in-the-Browser (MitB) attacks: by allowing access to customer data, TPPs will also be a target of such attacks. The integration between TPPs and the incumbent financial services providers' backends will be via Application Programming Interfaces (APIs), which greatly increases the opportunities for hackers to attack.

Traditionally, security teams have focused their resources on perimeter security - in other words, protecting everything that runs inside the firewall. However, when data passes through an open API to a client, it is extremely vulnerable to attack; there is no way to control the client device, whether that be a mobile or a browser. Such attacks manifest themselves in different ways. In a man-in-the-middle attack (MitM), an attacker sits secretly between two parties who believe they are directly communicating with each other. For example, somebody checking their bank balance by connecting from their device to a bank's application is vulnerable to a MitM style attack. This type of fraud is becoming more and more commonplace.

MitB attacks start with viruses (Trojans) that infect devices or web browsers. Then they silently log the activities of the user, potentially manipulating them by taking advantage of the ability to modify web pages, modify transaction content or insert additional transactions. All of this is invisible to both the user and the owner of the web application, allowing the scenario of the attack to be his own interface.

In the UK, users of Barclays, Royal Bank of Scotland, HSBC, Lloyds Bank and Santander have previously been targeted by cyberthieves using the Dyre banking trojan. Malicious emails were sent over a number of days, from spam servers worldwide, inviting users to download an archive containing a malicious .exe file posing as personal financial information.

And even the most cautious users can be harmed through a browser extension that has the authorization to modify web pages and may be storing information that users are typing or submitting via forms or other components i.e. banking information or credit card details. Regardless of the attack vector, hackers can potentially see everything you type on your device if you have your device or browser infected.

ADOPT THE CORRECT APPROACH

To comply with the European Banking Authority's tough new standards, Banks and TPPs need to adopt an "outside the firewall" mind-set and approach to security. The best way to protect client-to-server communications against MitM is to incorporate the correct levels of shielding to the applications and APIs that run outside the firewall. A screening / surveillance service that monitors every page served to end-users is needed. If something suspicious is spotted and the user is under attack, the application backend is immediately notified, allowing near real-time reaction from the application to the attempted fraud - and potentially preventing many others..

Try and investigate the client-side threats that your applications are facing so that you can react in real-time to these threats. Consider a solution that can bulletproof the client-side of your bank webpages, for example. Such solutions can automatically detect which behaviour is expected and which is not. This helps to identify sessions where malicious activity is taking place so that system administrators can be alerted. Your solution should be able to detect code injections on browser pages that are not being served by your institution. ■

ABOUT THE AUTHOR

Pedro Fortuna is a co-Founder and CTO of Jscrambler, where he leads the application security research activities and lays out the technical vision for all the products developed by the company. Pedro holds a degree in Computing Engineering and an MSc in Computer Networks and has more than a decade of experience researching and working in the application security area. He is a regular speaker at cyber security conferences and software development events, including multiple-time speaker at OWASP events. His research interests lie in the fields of Application Security, Reverse Engineering, Malware, and Software Engineering. Pedro is also the author of several patents in application security.

Pedro can be reached online at pedro.fortuna@jscrambler.com and at the company website www.jscrambler.com