IBM Software
Thought Leadership White Paper

December 2014

Financial malware explained

Explore the lifecycle of fraudulent transactions and how to take action against emerging threats

Introduction

Financial malware—that is, malicious software designed to enable fraudulent transactions—is a growing concern for line-of-business executives, heads of retail and commercial banking, leaders of global compliance operations, and risk management officers worldwide. Fraudsters rely upon increasingly sophisticated techniques to steal the credentials of online banking customers, and then reuse them to take over the victim's account and perform fraudulent transactions such as transferring money to new destinations. And often, the victim is not aware that anything is amiss.

What's more, today's cybercriminals are aware of the fraud prevention technologies deployed by most financial institutions, and they design attacks to circumvent these controls. For example, fraudsters can bypass various forms of authentication by using social-engineering campaigns to convince users to divulge their credentials online. Transaction anomaly detection and device ID approaches can also be highly inaccurate, generating a large number of false positive alerts that can overwhelm IT resources and may negatively impact the user experience.

This white paper will examine the lifecycle of a fraudulent transaction, including the tactics that cybercriminals use to infect victims' machines, harvest credentials and execute fraudulent transactions. It will then look at how IBM® Security Trusteer® solutions provide a comprehensive managed service that holistically addresses new risks and automatically responds to the evolving threat landscape.

The lifecycle of a fraudulent transaction

Malware allows fraudsters to execute fraudulent transactions in numerous ways. Two examples are:

1. *Account takeover (ATO)*—In this type of manual attack, fraudulent transactions are performed by fraudsters from a criminal device after obtaining the victim's credentials.
2. *Automated transaction systems (ATS)*—In this automated attack, genuine transactions are initiated by the victim and then altered on the fly by malware. The malware changes the designated payee or transaction sum without the victim's knowledge and steals funds from the victim's account.

Fraudsters use *social engineering* techniques to encourage victims to perform various actions—from downloading malicious software onto their computer to giving away their personal information. Fraudsters manipulate victims into clicking on links to what they believe are legitimate websites or applications, but actually download malware automatically and invisibly onto their machines instead.

Once the malware is installed, fraudsters often get victims to divulge their personal information by leading them through a false identification process. Victims may even receive a notice that they will be locked out of their account if they don't confirm their identity, but the false "identification" or "confirmation of identity" process actually steals their personal information and sends it directly to the fraudster.

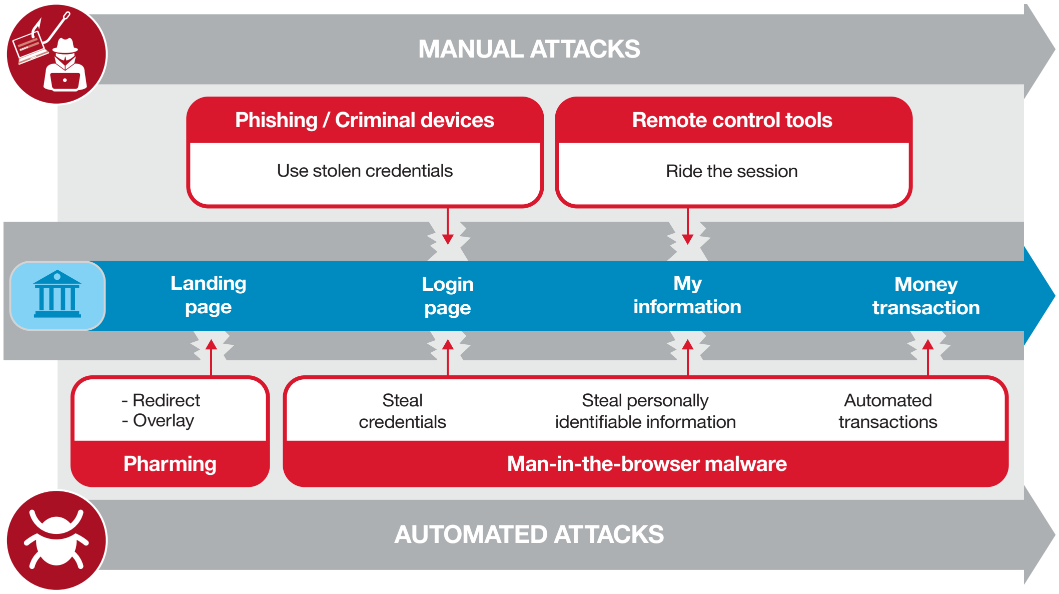
Social engineering is a key fraud factor; without it, fraudsters have little chance of succeeding. The typical lifecycle for malware-originated fraud involves a three-step process:

1. Infecting victims' machines
2. Harvesting credentials (as in the case of account takeover fraud)
3. Executing fraudulent transactions

Infecting victims' machines

To infect machines, fraudsters need to distribute the malware—requiring a distribution vector. The distribution vector can be any means of reaching a large number of people, including:

- **Email spam**: In this common distribution vector, an end user receives an innocent looking email, which contains a link to download the malware onto the machine or includes the malware as an attachment.
- **Malvertising**: Fraudsters sometimes pay for legitimate advertisement space in order to plant malicious ads with embedded JavaScript. The malicious JavaScript is structured in such a way that it uses an existing exploit or redirects the user to the exploit kit to install the financial malware. In addition, fraudsters may use *bots* to advertise links to malicious software. A bot is a piece of software that can run automated tasks, such as planting advertisements in various legitimate sites. These ads download the malware onto the machine without the user's knowledge.
- **Infection services (downloaders)**: Infecting machines is a difficult task, so fraudsters have also developed a specific type of malware that is used to distribute other malware. These *infection services* are available for purchase by other fraudsters. Alternatively, fraudsters may use an existing botnet, already infected in the past, to distribute new types of malware. These botnets are known as *downloaders*.
- **Exploits**: Fraudsters often use security vulnerabilities in browsers or other desktop software to infect a machine. They "exploit" these vulnerabilities to introduce a piece of code called a *payload*, which then installs the desired malware on the machine without the victim's knowledge.



Harvesting credentials

Once machines are infected with the malicious application, the fraudster can use a variety of tools or methods to harvest victims' credentials. These can include:

- **Screen captures**: It is relatively simple for fraudsters to take screenshots of end users' machines without their knowledge. The fraudster can leverage simple functions such as "Print screen" and send the screenshots to the malware's command-and-control (C&C) server. Screen captures are typically used by fraudsters to bypass virtual keyboards, which are used by some banks as a security mechanism against keylogging.
- **Keylogging**: A keylogger is a piece of software that records users' keystrokes without their knowledge and then transmits that information to the C&C server. Fraudsters can perform keylogging with a range of techniques, including hooking into the keyboard application programming interfaces (APIs) or obtaining root access to the kernel.
- **Session hijacking**: Many online banking applications use cookies to specify how long an online banking session is valid before users need to re-authenticate. In the case of session hijacking, the malware copies this cookie and allows the fraudster to use it while the session is still valid.
- **Pharming**: Pharming is the action of redirecting the end-user's browser to a malicious site pretending to be a legitimate banking site. This is done by altering the domain name server (DNS) configuration of targeted applications, either by changing the IP addresses of certain sites locally or by exploiting a DNS server's vulnerability. In addition, a relatively new form of pharming changes the DNS configuration of the targeted application in a local router. In this case, the router redirects the end-user's browser to the malicious site. This method has the added benefit of affecting all the machines that use this router, instead of having to infect every machine.
- **Web injections**: With web injection, fraudsters "inject" code to alter the content of web forms and pages, thus enabling the theft of end-user credentials. Malware can use different web injection techniques against each targeted URL. These are defined in the configuration files of the malware and determined based on the data that is required to commit the fraudulent transaction.
- **Man-in-the-middle (MitM) attacks**: This broad term refers to a third-party's capability to listen to or intercept victims' private and confidential information.
- **Man-in-the-browser (MitB) attacks**: In this type of attack, the end-user's browser communicates with the online banking application. The online banking application sends encrypted information that is decrypted in the user's browser; then, the data is rendered and displayed in the browser. The malware hooks into the browser and can capture the banking information before it is displayed in the browser or just before the encrypted information is sent to the bank. This type of attack requires that the malware is compatible with many browser types and versions. Web injections are an example of MitB attacks that exploit the browser to inject the relevant code into the online banking form.
- **Overlay attacks/fake forms**: Overlay attacks launch a fake form on the user's machine while asking for sensitive information from the end user; once submitted, the information is sent to a C&C server. Most overlay attacks launch the fake form when the end user accesses a targeted application, deploying the form on top of a legitimate website. The form cannot be closed or minimized until the credentials have been submitted. This is different than web injection, session hijacking and pharming attacks, which do not involve the end-user's browser.
- **Remote-control tools**: With remote-access software, fraudsters can control an end-user's machine as if they have physical access to that system. This provides fraudsters with full access to the infected machine, including the ability to read local files and use the end-user's browser to execute fraud from the end-user's machine. This type of attack is particularly effective at evading device ID security measures as fraud is executed from the legitimate customer's machine.

Executing fraudulent transactions

In the case of account takeovers, once fraudsters obtain the victim's credentials, they can execute a fraudulent money transfer to a mule account. A *mule* is someone who allows his or her bank account to be used to transfer funds that were obtained illegally. Money mules are often recruited online for what they think is legitimate employment, and they are unaware that they are participating in criminal activity. The money is transferred from the mule's account to the fraudster's account, typically in another country.

In contrast, automated malware can perform fraudulent transactions on the fly during the user's online banking session—without stealing the end-user's credentials. When the user logs onto a targeted site, the malware initiates a money transfer to a mule account while the session is active and without the knowledge of the end user. This type of fraudulent transaction is hard to implement since it requires the actions of the user and fraudster to be fully coordinated. However, the fraud is also hard to detect since the end user initiates a genuine online banking session from a legitimate device.

Layered protection against malware threats

IBM Security Trusteer solutions offer multiple layers of protection across devices and the fraud lifecycle. Hundreds of organizations and more than 270 million users rely on Trusteer solutions to protect their web applications and mobile devices from online threats such as malware and phishing attacks. Unlike other fraud controls, Trusteer solutions are based on a cybercrime prevention platform that leverages real-time, dynamic fraud intelligence. As a result, Trusteer solutions can adapt to the latest cybercrime risks and help protect against emerging threats.

To provide comprehensive fraud protection, Trusteer solutions are delivered as a turnkey, managed service. These solutions process tens of thousands of malware attack attempts every day into Crime Logic—a unique, compact and actionable footprint of cybercrime targets and tactics. New Crime Logic is automatically integrated into Trusteer products to promptly detect and block attacks on protected endpoints. These security measures are transparent to customers; institutions can fight fraud without disrupting the online banking experience.

What's more, Trusteer solutions deliver actionable intelligence with real-time risk alerts that enable institutions to tailor a timeline and response for specific threats. For example, a high-risk alert may explain that malware is active, targeting a specific web application, and designed to tamper with transactions (or steal user credentials). Organizations can design an automated workflow to respond to the specific threat.

Appendix: The architecture of financial malware

Typically, financial malware is constructed of several components:

- **The malicious executable (the dropper)**: This executable installs the malicious software on the machine and, thus, enables the criminal activity. The .exe file installs some form of persistency that enables the malware to continue running even after a system reboot or removal attempts.
- **Configuration files**: These binary-format files determine the targeted application and the malware behavior in each of these applications—for example, the injected fields for each targeted application. The initial malware installation contains default configuration files; however, these files can be updated at any time by the fraudster using a C&C server. The configuration files will determine many aspects of the malware behavior; for example:
  - **C&C server information**, such as the IP/server name of the C&C server.
  - **Web injection**, that is, the content that is injected or modified per URL on the web page. This portion of the configuration file can include thousands of records.

- **Back-connect configurations**, which make the fraudster's traffic appear as though it originates from a victim's IP address, making the fraudster's traffic seem legitimate.
- **Web filters**, designed to prevent outbound communication from the machine—for example, blocking communication from the machine to the anti-virus management servers to disable updates or preventing the browser from logging out of the session—all for the purpose of session hijacking.
- **DNS redirect configuration**, which can block a specific site by redirecting the domain name to an invalid IP address.
- **Grabs**, designed to gather information from a web page; for example, by saving a submitted HTML form or taking a screenshot.

- **Functionality enablers**: These software components enable malware to execute the desired functionality. They can be legitimate software used for malicious purposes or components developed by fraudster. For example, screen-capturing software or keyloggers can be built-in or downloaded on demand from the C&C server.
- **C&C server components**: These server-side components communicate with the malware client to send information or retrieve updates. Capabilities typically include:

- Reporting on the size and distribution of infected machines, also known as the botnet
- The ability to distribute commands to specific bots or all bots
- The ability to distribute configuration file updates or software upgrades
- Management of harvested credentials

The C&C architecture generally takes one of three forms:

- The botnet is managed by one or two central servers. The information flows from all the bots to the central servers.
- Peer-to-peer (P2P) communication occurs between all the bots. In this case, all the information flows between different bots directly to the fraudster; this in turn helps the fraudster to obfuscate the machines that are being used to manage the botnet. This use case is relatively new and was seen in a malware known as "Zeus game over." This C&C architecture makes it much more difficult to track down the fraudster or rescue harvested information.
- Domain Generation Algorithm (DGA) structures are used to periodically generate a large number of domain names that can be used as meeting points with their controllers. The large number of potential points makes it difficult for law enforcement to effectively shut down botnets since infected computers will attempt to contact some of these domain names every day to receive updates or commands.

For more information

To learn more about fighting financial malware with IBM Security Trusteer solutions, please contact your IBM representative or IBM Business Partner, or visit: **ibm.com**/security

About IBM Security solutions

IBM Security offers one of the most advanced and integrated portfolios of enterprise security products and services. The portfolio, supported by world-renowned IBM X-Force® research and development, provides security intelligence to help organizations holistically protect their people, infrastructures, data and applications, offering solutions for identity and access management, database security, application development, risk management, endpoint management, network security and more. These solutions enable organizations to effectively manage risk and implement integrated security for mobile, cloud, social media and other enterprise business architectures. IBM operates one of the world's broadest security research, development and delivery organizations, monitors 15 billion security events per day in more than 130 countries, and holds more than 3,000 security patents.

Additionally, IBM Global Financing can help you acquire the software capabilities that your business needs in the most cost-effective and strategic way possible. We'll partner with credit-qualified clients to customize a financing solution to suit your business and development goals, enable effective cash management, and improve your total cost of ownership. Fund your critical IT investment and propel your business forward with IBM Global Financing. For more information, visit: **ibm.com**/financing

© Copyright IBM Corporation 2014

IBM Corporation
Software Group
Route 100
Somers, NY 10589

Produced in the United States of America
December 2014

IBM, the IBM logo, and ibm.com are trademarks of International Business Machines Corp., registered in many jurisdictions worldwide. Other product and service names might be trademarks of IBM or other companies. A current list of IBM trademarks is available on the web at “Copyright and trademark information” at ibm.com/legal/copytrade.shtml

Trusteer is a trademark or registered trademark of Trusteer, an IBM Company.

This document is current as of the initial date of publication and may be changed by IBM at any time. Not all offerings are available in every country in which IBM operates.

THE INFORMATION IN THIS DOCUMENT IS PROVIDED “AS IS” WITHOUT ANY WARRANTY, EXPRESS OR IMPLIED, INCLUDING WITHOUT ANY WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND ANY WARRANTY OR CONDITION OF NON-INFRINGEMENT. IBM products are warranted according to the terms and conditions of the agreements under which they are provided.

The client is responsible for ensuring compliance with laws and regulations applicable to it. IBM does not provide legal advice or represent or warrant that its services or products will ensure that the client is in compliance with any law or regulation.

Statement of Good Security Practices: IT system security involves protecting systems and information through prevention, detection and response to improper access from within and outside your enterprise. Improper access can result in information being altered, destroyed or misappropriated or can result in damage to or misuse of your systems, including to attack others. No IT system or product should be considered completely secure and no single product or security measure can be completely effective in preventing improper access. IBM systems and products are designed to be part of a comprehensive security approach, which will necessarily involve additional operational procedures, and may require other systems, products or services to be most effective. IBM does not warrant that systems and products are immune from the malicious or illegal conduct of any party.

♻ Please Recycle