Scam Booter

CPEN442

October 11

Zoy Huang (20026150), Ryan Koon (11062149), and Wendy Zhou (41378150)
Department of Electrical and Computer Engineering
University of British Columbia
Vancouver, Canada

***Abstract*—**

I. Introduction

A. What is the problem that we addressed?

We addressed the issue of social engineering from technical support scams, which has caused an increasing trend of reported financial losses [1]. We proposed a solution that protects users with little knowledge about Windows' administrative tools. The asset at risk is the victim's bank account. The vulnerability is the victims lack of knowledge about Administrative tools on Windows. The threats are people pretending to be legitimate technicians and deceiving users about the state of their computer.

B. Why is this problem important?

According to Microsoft, there were 153,000 technical support scam reports worldwide in 2017, a 24% growth from the previous year [3, Fig 1]. In the same year, the Internet Crime Complaint Center (IC3) received approximately 11,000 technical support complaints that totals to a loss of almost $15 million. That was an "86% increase in losses from 2016" [1].

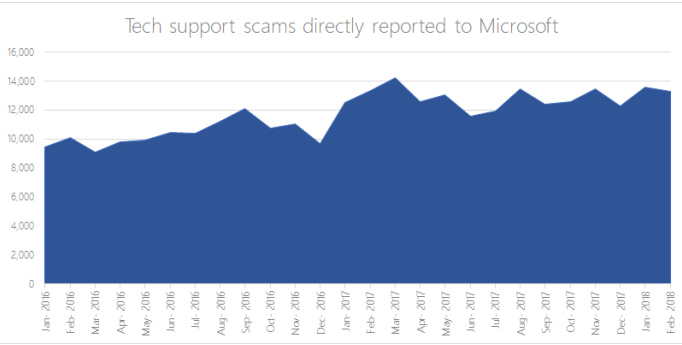

Fig. 1. Number of tech support scams reported to Microsoft

A study [2] conducted in 2014 discovered 1,688,412 unique visitors to scam sites and estimated a loss of at least $9.7 million. With this trend, it appears that technical support scams are not going away even with efforts to stop them [2, p. 8].

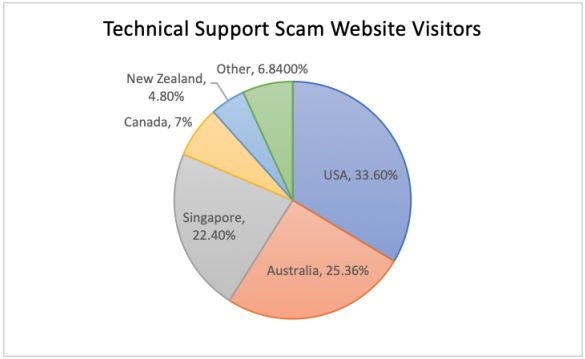

Fig. 2. Tech support scam website visitors by country

C. Summary of the Designed System

Social engineering attacks thrive on human error based on an inadequate knowledge of the system being used. A novice user does not have the knowledge to recognize fallacies provided by the technical support scammer, nor do they have working knowledge of Windows tools used to manipulate them.

However, notably, scammers follow similar sequences of events and use similar techniques to convince their victims. The most common techniques are shown in [2, Fig 3].

Technique	% Calls
Stopped Services/Drivers	67
Event Viewer	52
Specific Virus Explained	50
System Information	47
Action Center	40
Fake CMD Scan	40
Netstat Scan	40
Installed/Running Programs	35
Browsing History/Settings	27
Downloaded Scanner	17
Reliability/Performance	15
Other (Temp, Registry)	13

Fig. 3. Techniques used by support scammers in order to convince their victims of a malware infection

We defined these events and techniques as "suspicious behavior". For example, the use of the administrative tool, "Event

Viewer", qualifies as suspicious behavior. Our solution protects users that lack knowledge about the tools and techniques used in technical support scams by detecting these events on their behalf.

We addressed the problem by using the Windows API to build a Windows application that detects the suspicious behaviors. If a scam is detected, the application forcefully terminates any remote connections and informs the user of the event. We also explored options to prevent the application from being terminated. An option was to create a Windows service that manages the lifecycle of the application. There are other methods to make it even more difficult to shut down the application. However, those methods would likely be flagged by an antivirus solution. To ensure our application is compatible with antivirus programs, we uploaded our application binaries to VirusTotal to be analyzed by 68 antivirus scanners.

D. Summary of Related Works

ROBOVIC, short for Robotic Victim, was a tool created in a study that collected data about technical support scams. It crawled the web to find websites, its visitors, and phone numbers used for these kinds of scams [2, Fig 2]. The investigators concluded that "the vast majority of AV users are likely not going to be protected against technical support scams" [2, p.7]. On average, 64% of 1624 malicious TLDs (Top Level Domains) were only detected by 3.25 AV engines out of 68 engines [2, p.7]. Phone applications on Android detected "less than 1% of the 1,581 scammer-operated phone numbers" [2, p.8]. On average it took 44 days for a phone number to be reported as a part of a scam. Even worse, some mobile applications associated scam numbers with positive reviews and legitimate businesses such as Dell and McAfee. Microsoft has been making a wide range of improvements by "enhancing antivirus, email, URL blocking, and browser security solutions". TeamViewer has been displaying prompts to warn users about technical support scams. In terms of user education, attempts such as public service announcements have been made. However, they were ineffective being on specific sites that were not known by the general population [2, p.13].

E. Summary of the Methodology

We constructed a test suite that contains various forms of unique and documented technical support scams. For each scam, we made many similar attacks against a system protected by our solution. We assessed the effectiveness of our solution based on the number of detected attacks in the test suite.

F. List of Contributions

II. Related Work

III. Adversary Model

Although there are various techiniques that are used by technical support scammers, we developed a generalized model by analyzing the patterns of technical support scams. This model facilitated our understanding of the problem and helped us to design our system.

A. Objectives

The objective of the adversaries is to convince the victim that their computer is infected by a virus. After gaining the trust of the victim, the adversaries make the victim to pay them for fixing non-existent problems. The adversaries are scammers who want to make profits by deceiving inexperienced Windows users.

B. Initial capabilities

The adversaries can call users, claiming to be from well-known companies. The adversaries can also set up malicious websites or advertisements that tell the user to call technical support numbers.

C. Capabilities during the attack

The adversaries ask a user to download remote access software and gain access to their computer. They can open administrative tools to show system "errors", which are logs of normal operation. They can install malwares on the user's machine, causing the machine crash.

IV. System Design

A. Detecting a remote connection

Detecting an active remote desktop connection is not as trivial as checking a flag. For Windows, that is only possible if the application is using the Microsoft Remote Desktop Protocol, which operates on specific ports. However, the remote desktop applications used by technical support scammers typically establish connections on port 80 and 443, the standard ports used by web clients to communicate with a machine. Also, these ports are only used to establish a connection, the resulting port is not the same for each session as it is assigned by the machine serving the desktop. Other non-standard ports are used by remote desktop applications, but they differ between software, they may not be documented, or the ports they use will change depending on network conditions. Such a moving target will result in a high rate of false-positives. Instead of analysing network traffic, our application analyses raw input data using the Windows Desktop API [4]. Every time a mouse click occurs, regardless of the application in focus, our application will check whether a handler to the device is provided in the raw input header. If so, it is a local device that is attached to the computer. If no handler is provided, it is an input made from an active remote connection to the computer. Using this method, we can distinguish between clicks from a locally connected mouse and clicks from a remote desktop application.

B. Detect the use of administrative tools

As illustrated in [2, Fig 2]., scammers use specific Windows programs during an attack. 67% of scammers start services.exe to show users a list of stopped services. 52% calls to Event Viewer, 40% calls to netstat.exe and 40% calls to commond prompt. Since the calls to these tools are closely related to a scam in progress, we decided to monitoring the use of

these administrative tools. To implement this, our application registers a WMI (Windows Management Instrumentation) event handler to receive notifications of the start of a process [6]. We check if the launched process includes services.exe, eventvmr.exe, mmc.exe, cmd.exe or netstat.exe.

C. Analysing keystrokes

To improve the accuracy of detecting a scam in progress, we log keystrokes when the Windows Command Prompt is in focus. Scammers frequently execute certain commands to trick the user into believing that there is malware on their computer. 40% of scammers employ a "Fake CMD scan" by running a command to list the tree of files, folders, and subfolders at a certain path [2, p.7]. Our application will capture keystrokes that occur while the command prompt is outputting information. We will look for common words and phrases typed in such as "Virus detected". We employ this strategy because users are often tricked into believing that the system produced the output not knowing that user inputs are displayed after a command is done executing. To capture keystrokes we use a global hook open source library that will allow our application to learn about keystrokes regardless of the application in focus [5].

D. Terminating remote access applications

A fail-safe strategy to terminate the connection between the user and the scammer is to stop all network traffic. However, this will affect all other applications that require network connectivity which arguably, makes our application appear to be malware. To prevent such degradation to the user experience, we terminate common remote desktop applications used by technical support scammers instead [2, Fig 4]. Our application runs with elevated privileges to allow us to terminate remote access applications with elevated privileges and to counter process renaming done via a file rename. We access the file metadata to determine the original name of the application rather than relying solely on the process name.

Remote Administration Tool	Websites	Scammer abuse
LogMeIn Rescue	www.support.me www.lmi1.com www.logmein123.com	60%
CITRIX GoToAssist	www.fastsupport.com	21%
TeamViewer	www.teamviewer.com *Scammer-controlled*	12%
Other	www.anydesk.com www.gethelp.us www.supremocontrol.com	7%

Fig. 4. Remote access tools used by technical support scammers.

E. Self restarting

V. System Prototype

VI. Evaluation

A. Evaluation Methodology

References

[1] Federal Bureau of Investigation, "TECH SUPPORT FRAUD", https://www.ic3.gov/media/2018/180328, Mar. 28, 2018 [Oct. 6 2018]

[2] N. Miramirkhani, O. Starov, and N. Nikiforakis, "Dial One for Scam: A Large-Scale Analysis of Technical Support Scams," in Proceedings 2017 Network and Distributed System Security Symposium, 2017

[3] Erik Wahlstrom, "Teaming up in the war on tech support scams", https://cloudblogs.microsoft.com/microsoftsecure/2018/04/20/teaming-up-in-the-war-on-tech-support-scams/, April 20, 2018 [Oct. 6 2018]

[4] "C# Get Mouse handle (GetRawInputDeviceInfo)", Stack Overflow, 2018. [Online]. Available: https://stackoverflow.com/questions/14584280/c-sharp-get-mouse-handle-getrawinputdeviceinfo. [Accessed: 09- Nov- 2018].

[5] "rvknth043/Global-Low-Level-Key-Board-And-Mouse-Hook", GitHub, 2018. [Online]. Available: https://github.com/rvknth043/Global-Low-Level-Key-Board-And-Mouse-Hook. [Accessed: 09- Nov- 2018].

[6] "Receiving a WMI Event", Microsoft, 2018. [Online]. Available: https://docs.microsoft.com/en-us/windows/desktop/WmiSdk/receiving-a-wmi-event#event-consumers. [Accessed: 09- Nov- 2018].