ISSN (Online) 2394-2320

International Journal of Engineering Research in Computer Science and Engineering (IJERCSE)
Vol 6, Issue 11, November 2019

Sybilguard I: Prevention and Detection of Attacks Mobile ADHOC Networks

[1] Apoorva A, [2] MadhuriYashoda, [3] T.Somasekhar
[2] Assistant Professor, [3] Associate Professor & Head
[1][2][3] Dept. of CSE, BIT Institute of Technology, Hindupur

Abstract: - **Real-time crowd sourced maps, such as Waze provide timely updates on traffic, congestion, accidents, and points of interest. In this paper, we demonstrate how lack of strong location authentication allows creation of software-based Sybil devices that expose crowd sourced map systems to a variety of security and privacy attacks. Our experiments show that a single Sybil device with limited resources can cause havoc on Waze, reporting false congestion and accidents and automatically rerouting user traffic. More importantly, we describe techniques to generate Sybil devices at scale, creating armies of virtual vehicles capable of remotely tracking precise movements for large user populations while avoiding detection. To defend against Sybil devices, we propose a new approach based on co-location edges, authenticated records that attest to the one-time physical collocation of a pair of devices. Over time, co-location edges combine to form large proximity graphs that attest to physical interactions between devices, allowing scalable detection of virtual vehicles. We demonstrate the efficacy of this approach using large-scale simulations, and how they can be used to dramatically reduce the impact of the attacks. We have informed Waze/Google team of our research findings. Currently, we are in active collaboration with Waze team to improve the security and privacy of their system.**

1. INTRODUCTION

CROWDSOURCING is indispensable as a real-time data gathering tool for today's online services. Take for example map and navigation services. Both Google Maps and Wazeuse periodic GPS readings from mobile devices to infer traffic speed and congestion levels on streets and highways. Waze, the most popular crowdsourced map service, offers users more ways to actively share information on accidents, police cars, and even contribute content like editing roads, landmarks, and local fuel prices. This and the ability to interact with nearby users made Waze extremely popular, with an estimated 50 million users when it was acquired by Google for a reported $1.3 Billion USD in June 2013. Today, Google integrates selected crowdsourced data (e.g. accidents) from Waze into its own Maps application.

Unfortunately, systems that rely on crowdsourced data are inherently vulnerable to mischievous or malicious users seeking to disrupt or game the system [1]. For example, business owners can badmouth competitors by falsifying negative reviews on Yelp or TripAdvisor, and FourSquare users can forge their physical locations for discounts [2], [3]. For location-based services, these attacks are possible because there are no widely deployed tools to authenticate the location of mobile devices. In fact, there are few effective tools today to identify whether the origin of traffic requests are real mobile devices or software scripts.

The goal of our work is to explore the vulnerability of today's crowdsourced mobile apps against Sybil devices, software scripts that appear to application servers as "virtual mobile devices."1 While a single Sybil device can damage mobile apps through misbehavior, larger groups of Sybil devices can overwhelm normal users and significantly disrupt any crowdsourced mobile app. In this paper, we identify techniques that allow malicious attackers to reliably create large populations of Sybil devices using software. Using the context of the Wazecrowdsourced map service, we illustrate the powerful Sybil device attack, and then develop and evaluate robust defenses against them.

While our experiments and defenses are designed with Waze (and crowdsourced maps) in mind, our results generalize to a wide range of mobile apps. With minimal modifications, our techniques can be applied to services ranging from Foursquare and Yelp to Uber, YikYakand Pokemon Go, allowing attackers to cheaply emulate numerous virtual devices with forged locations to overwhelm these systems via misbehavior. Misbehavior can range from falsely obtaining coupons on Foursquare/Yelp, gaming the new user coupon system in Uber, imposing censorship on YikYak, to cheating in the game play of Pokemon Go. We believe our proposed defenses can be extended to these services as well. We discuss broader implications of our work in Section IX. Sybil Attacks in Waze: In the context of Waze, our experiments reveal a number of potential attacks by Sybil devices. First is simple event forgery, where devices can generate fake events to the Waze server, including congestion, accidents or police activity that might affect user routes. Second, we describe techniques to reverse engineer mobile app APIs, thus allowing attackers to create lightweight scripts that effectively emulate a large number of virtual vehicles that collude under the control of a single attacker. We call Sybil devices in Waze "ghost riders." These Sybils can effectively magnify the efficacy of any attack, and overwhelm contributions from any legitimate users. Finally, we discover a significant privacy attack where ghost riders can silently and invisibly "follow" and precisely track individualWaze users throughout their day, precisely mapping out their movement to work, stores, hotels,gas station, and home. We experimentally confirmed the accuracy of this attack against our own vehicles, quantifying the accuracy of the attack against GPS coordinates. Magnified by an army of ghost riders, an attacker can potentially track the constant whereabouts of millions of users, all without any risk of detection. Defenses: Prior proposals to address the location authentication problem have limited appeal, because of reliance on widespread deployment of specialized hardware, either as part of physical infrastructure, i.e., cellular base stations, or as modifications to mobile devices themselves. Instead, we propose a practical solution that limits the ability of Sybil devices to amplify the potential damage incurred by any single attacker. We introduce collocation edges, authenticated records that attest to the one-time physical proximity of a pair of mobile devices. The creation of collocation edges can be triggered opportunistically by the mapping service, e.g., Waze. Over time, collocation edges combine to form large proximity graphs, network structures that attest to physical interactions between devices. Since ghost riders cannot physically interact with real devices, they cannot form direct edges with real devices, only indirectly through a small number of real devices operated by the attacker. Thus, the edges between an attacker and the rest of the network are limited by the number of real physical devices she has, regardless of how many ghost riders are under her control. This reduces the problem of detecting ghost riders to a community detection problem on the proximity graph (The graph is seeded by a small number of trusted infrastructure locations). Our paper includes these key contributions: • We explore limits and impacts of single device attacks on Waze, e.g., artificial congestion and events. • We describe techniques to create light-weight ghost riders, virtual vehicles emulated by client-side scripts, through reverse engineering of the Wazeapp's ommunication protocol with the server. • We identify a new privacy attack that allows ghost riders to virtually follow and track individualWaze users in realtime, and describe techniques to produce precise, robust location updates. • We propose and evaluate defenses against ghost riders, using proximity graphs constructed with edges representing authenticated collocation events between pairs of devices. Since collocation can only occur between pairs of physical devices, proximity graphs limit the number of edges between real devices and ghost riders, thus isolating groups of ghost riders and making them detectable using community detection algorithms..

2. EXISTING SYSTEM

In existing system, hackers easily can act as source node and sends message to destination. Destination receives wrong message from hackers. Destination believes that its correct message from source. Destination receives the wrong information from hackers.

Messages are passed from sender to destination (receiver) without any security. Message header holds source node information which sends the message to receiver. Hackers can easily change that header information and sends to destination.

2.1 Disadvantages

Destination gets the wrong information from hackers or malicious user. There is no any server to detect hackers. Header information may be hiding by malicious user. Source node does not get any response from destination while hackers get that source information.

3. PROPOSED SYSTEM

In this proposed system, hackers can not act as source, because one centralized server is maintaining to check authentication of source. This centralized server is sybilguard. It blacks unauthorized users or hackers. Sybilguard is maintaining source node information and header information of message. It checks the users using that details whether they are attackers or normal user. Hacker's information has not been transferred to destination. Destination has not been receiving any attacker information.

3.2 ADVANTAGES

Sybilguard is maintained to detect the attackers who are all act as source node. It deletes that wrong information from hackers and indicates that they are attackers. Hackers' information has not transferred to receiver.

Sybilguard act as the centralized server to all users. It handles the message transmission between those users. Each user has to register individually. Those user informations are stored in centralized server and find the attackers using that information.

4. MODULES

4.1 TOPOLOGY CONSTRUCTION

Topology construction is designed to construct one topology with available nodes. Register all nodes which are involved to transfer the data to some other nodes. Depends upon total nodes, topology will be constructed.

Topology construction module allows you to construct node path. If already exits, it will not allow to construct that same path. All nodes are mentioned in topology construction. User can't modify node information after construction.

4.2 NODE ENTRY

Node entry module describes node authentication. To activate node who are all involved in topology, node should be login into that topology. It does not allow unauthorized node entry. Many nodes can enter into that mentioned topology. Each node can send the messages to their destination after login.

4.3 MESSAGE TRANSMISSION

Each node (source node) can send the data to some other node(destination) which one connected with that source node. While sending message, the source node should mention the header information. Source node can send the data to destination. Destination will receive that message.

4.4 SYBLGUARD

Sybilguard is maintained in this project to detect the attacker. Sybilguard is called as centralized server. Sybilguard does not allow hackers to send the wrong data. It compares node information and header information. If matches, normal user sending the message to destination. Otherwise sybilguard will not allow the hackers to send message. It blocks that data and it provides the attacker information to attacker.

Sybilguard gets node information from its registration. While data transmission, sybilguard will get their header information. This centralized server maintains to find out the attacker details.

5. CONCLUSIONS AND FUTIRE WORK

We describe our efforts to identify and study a range of attacks on crowdsourced map services. We identify a range of single and multi-user attacks, and describe techniques to build and control groups of virtual vehicles (ghost riders) to amplify these attacks. Our work shows that today's mapping services are highly vulnerable to software agents controlled by malicious users, and both the stability of these services and the privacy of millions of users are at stake. While our study and experiments focus on the Waze system, we believe the large majority of our results can be generalized to crowdsourced apps as a group. We propose and validate a suite of techniques that help services build proximity graphs and use them to effectively detect Sybil devices. Throughout this work, we have taken active steps to isolate our experiments and prevent any negative consequence on real Waze users. We also proactively informed Waze team of theses attacks, and worked with them to mitigate the threat.

REFERENCES

[1] N. Stefanovitch, A. Alshamsi, M. Cebrian, and I. Rahwan, "Error andattack tolerance of collective problem solving: The DARPA shredder challenge," EPJ Data Sci., vol. 3, no. 1, pp. 1–27, 2014.

[2] B. Carbunar and R. Potharaju, "You unlocked the Mt. Everest badge on Foursquare! ountering location fraud in geosocial networks," in Proc. MASS, 2012, pp. 182–190.

[3] Z. Zhang et al., "On the validity of geosocial mobility traces," in Proc. HotNets, 2013, p. 11.

[4] J. R. Douceur, "The Sybil attack," in Proc. IPTPS, 2002, pp. 251–260.

[5] S. Cheng, Uber's Terrifying 'Ghost Drivers' are Freaking out Passengers in China. New York, NY, USA: Quartz, Sep. 2016.

[6] Y. Wang, "Ghost drivers are just one of Uber China's problems following DIDI takeover," Forbes, Sep. 2016.

[7] M. Wehner, "How to cheat at Pokémon Go and catch any Pokemon you want without leaving your couch," DailyDot, Jul. 2016.

[8] How to Avoid Getting Banned in Pokemon Go While Location Spoofing, Cydiageeks, San Francisco, CA, USA, Jul. 2016.

[9] V. Goel, "Maps that live and breathe with data," The New York Times, New York, NY, USA, Tech. Rep., Jun. 2013. [Online]. Available: https://www.nytimes.com/2013/06/11/technology/mobile-companiescrave-maps-that-live-and-breathe.html

[10] Google Maps and Waze, Outsmarting Traffic Together, Google Official Blog, Google, Mountain View, CA, USA, Jun. 2013.

[11] GenyMotion Emulator. Accessed: Jun. 2016. [Online]. Available: http://www.genymotion.com

[12] Monkeyrunner. Accessed: Jun. 2016. [Online]. Available: https://developer.android.com/studio/test/monkeyrunner/index.html

[13] B. Reed, "Google Maps becomes Google's second 1 billion-download hit," Yahoo! News, Jun. 2014.

[14] Charles Proxy. Accessed: Jun. 2016. [Online]. Available: http://www. charlesproxy.com

[15] D. Sounthiraraj, J. Sahs, G. Greenwood, Z. Lin, and L. Khan, "SMVHUNTER: Large scale, automated detection of SSL/TLS man-in-themiddle vulnerabilities in Android Apps," in Proc. NDSS, 2014. [Online]. Available: http://dx.doi.org/10.14722/ndss.2014.23205