POSTER: Covert Channel Based on the Sequential Analysis in Android Systems*

Jun-Won Ho
Dept. of Information Security
Seoul Women's University
Seoul, South Korea
jwho@swu.ac.kr

KyungRok Won, Jee Sun Kim
Dept. of Information Security
Seoul Women's University
Seoul, South Korea
rafaelawon@swu.ac.kr,kjsrlawltjs@swu.ac.kr

ABSTRACT

Due to the wide spread of android smartphones, different types of attacks have emerged against android systems and accordingly many researches have been accomplished in the android security. In particular, a variety of covert channels have been recently developed in android systems. They are usually built up by utilizing physical media and distinct characteristics of systems in the literature. To the best of our information, however, we do not find out any research work establishing covert channels in android systems on basis of the sequential analysis, which is a kind of statistical decision theory. This is mainly because the sequential analysis has been conventionally treated as defense technique in terms of security. In contrast to this common application of the sequential analysis, we discover a new covert channel based on the sequential analysis in android systems. The key idea of newly devised covert channel is to harness the sequential analysis in order to encode (resp. decode) private information bits to (resp. from) multiple sequences of randomly selected data. Through simulation, we demonstrate that our developed covert channel works efficiently and thus it could be substantial threat to android systems.

KEYWORDS

covert channel, sequential analysis, android

1 INTRODUCTION

Covert channels are considered to be stealthy passages through which private information can be leaked from systems in concealed manner. Moreover, attacker can exploit covert channel to propagate attack control messages to malicious entities. In this sense, covert channels are substantial threat against the normal functions of systems and thus the considerable number of researches on them have been achieved in variety of systems. In particular, covert channels are harmful in android systems from the perspective that they could be used to leak private information from android smartphones and thus could menace the security of android ecosystem. In order to defend against covert channels in android systems, it is thus very imperative to discover as many kinds of covert channels as possible that could operate in android systems. In order to fulfill this need, a diversity of covert channels have been proposed in the literature. Lalande et al. [3] proposed several android covert channels based on the task list, process priority, and screen state. Novak et al. [4] designed various android covert channels with using physical media such as ultrasound, flash, vibration, camera, speaker, and accelerometer. Qi et al. [5] exploited user behavior for covert channel establishment.

Although these related work cover covert channels based on various features of systems, user behavior, and physical media in android smartphones, they do not consider covert channels rooted on the sequential analysis. This is because the sequential analysis [7] has been prevalently used as statistical decision process and attack detection mechanism such as mobile sensor replica detection [1], port scan detection [2], and even covert channel detection [6], leading to exclusion from the stepping stones for covert channel establishment. In contrast to these conventional applications of the sequential analysis, we discover a new covert channel that harnesses the Sequential Probability Ratio Test (SPRT) in the sequential analysis [7]. For the SPRT-based covert channel, we make use of the fact that many android applications generally utilize that diverse types of variable sensory and GPS data provided by android systems. More specifically, the SPRT-based covert channel makes the SPRT encode (resp. decode) private information bits to (resp. from) multiple sequences of randomly chosen data. We believe that the SPRT-based covert channel expands the extent of possible covert channels in android systems, contributing to the research of covert channel detection. We describe the details of our newly devised covert channel and present the simulation results of it in the following sections.

2 SPRT-BASED COVERT CHANNEL ESTABLISHMENT

Most android smartphones are equipped with diverse sensors such as accelerometer, ambient light and proximity sensors, gyroscope, compass, and GPS systems. Hence, substantial number of android applications use a variety of variable sensory and GPS data from these sensors and GPS systems. For covert channel establishment, although we can use any types of variable sensory and GPS data provided by android systems, we focus on the location data consisting of latitude and longitude in this paper. This is mainly because many users make use of android applications sending location information to server in order to obtain location-related services.

Indeed, if we use a series of location data for covert channel development, it will seem to be straightforward to design covert

*This work was supported by the National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIP) (No. 2016R1C1B1014126).

CCS '17, October 30-November 3, 2017, Dallas, TX, USA

ACM ISBN 978-1-4503-4946-8/17/10.
https://doi.org/10.1145/3133956.3138847

channel in such a way that each location information is used to encode and decode a single private information bit. However, this method has the weakness that it is relatively easier to reveal the private information bits from location values than using multiple location information. To pacify this limitation, we propose the SPRT-based covert channel in which the SPRT encodes (resp. decodes) a single private information bit to (resp. from) a sequence of randomly chosen location values and thus the number of location values used for encoding and decoding processes are randomly determined, leading to being difficult to unveil private information bits from location information. Furthermore, the SPRT makes a decision with a small amount of location information and hence it will require little time to perform encoding and decoding processes. This will be considerable benefit for the attacker in the sense that the likelihood of being detected will be diminished.

How the SPRT is adapted to establish covert channel for private information transmission from a trojan android application to a trojan server is described as follows. Let us consider a trojan android application that transmits its location information to the trojan server in order to get the location-related services. We first assume that trojan app owns private information leaked from android systems. We also assume that the attacker can obfuscate this trojan app to avoid the static analysis detection and can install the obfuscated version of trojan app in android smartphone. This assumption is achievable because the attacker can develop his own obfuscation mechanism for the evasion from static analysis detection. Moreover, we assume that trojan app employs random selection process of location information periodically obtained from android systems. In this process, trojan app sends its location information to the trojan server with a certain probability. This assumption is reasonable in the sense that trojan app users will not likely have difficulty in acquiring seamless location-related services as long as the time period for location information collection is maintained as reasonably small. Additionally, this random selection strategy has advantage of reducing communication costs incurred by location information transmission to the trojan server.

2.1 Encoding Private Information Bits

Let us first consider the encoding process for a private information bit. When obtaining the first pair of latitude and longitude (A_1, O_1), trojan app accepts (A_1, O_1). Each time obtaining a pair of latitude and longitude (A_i, O_i) $(i \geq 2)$, trojan app performs *random location selection process* in the following manner. If trojan app wishes to encode private information bit 1 and both $|A_i - A_{i-1}| \geq \delta_a$ and $|O_i - O_{i-1}| \geq \delta_o$ hold, it selects (A_i, O_i) with probability p_s $(p_s > 0.5)$, where δ_a and δ_o are pre-configured thresholds. If trojan app wishes to encode a private information bit 0 and both $|A_i - A_{i-1}| \geq \delta_a$ and $|O_i - O_{i-1}| \geq \delta_o$ do not hold, it selects (A_i, O_i) with probability p_s. If all of the above two conditions do not hold, it selects (A_i, O_i) with probability p_u $(p_u < 0.5)$.

Trojan app performs the SPRT with only pairs (A_k, O_k) and (A_{k+1}, O_{k+1}) that are chosen by random location selection process $(k \geq 1)$. We first define the kth sample D_k as a Bernoulli random variable such that $D_k = 1$ if $|A_{k+1} - A_k| \geq \delta_a$ and $|O_{k+1} - O_k| \geq \delta_o$ hold. Otherwise, $D_k = 0$. Given the success probability c of the Bernoulli distribution, $c = \Pr(D_k = 1) = 1 - \Pr(D_k = 0)$ holds. In the SPRT, null hypothesis (H_0) indicates that the value of private information bit is 0 and alternate hypothesis (H_1) indicates that the value of private information bit is 1. Under these definitions of H_0 and H_1, we preconfigure c_0 and c_1 $(c_0 < c_1)$ such that the likelihood of H_0 (resp. H_1) acceptance increases if $c \leq c_0$ (resp. $c \geq c_1$) holds.

By the definition of the SPRT [7], given a sequence of $D_1, \ldots, D_j$ $(j \geq 1)$, the log-probability ratio Q_j on j samples is given by

$$Q_j = \ln \frac{\Pr(D_1, \ldots, D_j | H_1)}{\Pr(D_1, \ldots, D_j | H_0)}$$

In the sense that location information provided by android systems is usually independent of each other, it is reasonable that D_k is assumed to be independent and identically distributed. Under the i.i.d. assumption, we define E_j as the number of times that $D_k = 1$ in the j samples. We also denote α' (resp. β') as a false positive rate (resp. a false negative rate) that is configured by user. If trojan app wants to encode private information bit 1 and $E_j \geq \frac{\ln \frac{1-\beta'}{\alpha'} + j \ln \frac{1-c_0}{1-c_1}}{\ln \frac{c_1}{c_0} - \ln \frac{1-c_1}{1-c_0}}$ holds, the SPRT accepts H_1 and thus encoding private information bit 1 is completed. If trojan app wants to encode private information bit 0 and $E_j \leq \frac{\ln \frac{\beta'}{1-\alpha'} + j \ln \frac{1-c_0}{1-c_1}}{\ln \frac{c_1}{c_0} - \ln \frac{1-c_1}{1-c_0}}$ holds, the SPRT accepts H_0 and thus encoding private information bit 0 is completed.

If trojan app wants to encode private information bit 1 (resp. 0) and $E_j \leq \frac{\ln \frac{\beta'}{1-\alpha'} + j \ln \frac{1-c_0}{1-c_1}}{\ln \frac{c_1}{c_0} - \ln \frac{1-c_1}{1-c_0}}$ (resp. $E_j \geq \frac{\ln \frac{1-\beta'}{\alpha'} + j \ln \frac{1-c_0}{1-c_1}}{\ln \frac{c_1}{c_0} - \ln \frac{1-c_1}{1-c_0}}$) holds, the j th sample is revoked from the SPRT and the SPRT proceeds with new samples. Additionally, the pair of latitude and longitude (A_{j+1}, O_{j+1}) contributing to the jth sample is removed from the random location selection process. The rational behind the exclusion of the jth sample in this case is to prevent private information bit from being incorrectly encoded. If all of the above three conditions do not hold, the SPRT goes on with new samples. The above encoding process is repeatedly applied to a series of private information bits.

During the encoding process, trojan app sends the pairs of latitude and longitude, which are chosen from random location selection process and are fed into the SPRT, to trojan server.

2.2 Decoding Private Information Bits

Each time receiving the pairs of latitude and longitude from trojan app, trojan server decodes private information bits by performing the SPRT on these pairs of latitude and longitude in accordance with the same configuration parameter values, sampling method, and i.i.d assumption as used in the encoding process. If $E_j \geq \frac{\ln \frac{1-\beta'}{\alpha'} + j \ln \frac{1-c_0}{1-c_1}}{\ln \frac{c_1}{c_0} - \ln \frac{1-c_1}{1-c_0}}$ holds, the SPRT accepts H_1 and thus decoding private information bit 1 is completed. If $E_j \leq \frac{\ln \frac{\beta'}{1-\alpha'} + j \ln \frac{1-c_0}{1-c_1}}{\ln \frac{c_1}{c_0} - \ln \frac{1-c_1}{1-c_0}}$ holds, the SPRT accepts H_0 and thus decoding private information bit 0 is completed. If all of the above two conditions do not hold, the SPRT goes on with new samples.

3 SIMULATION STUDY

For the evaluation of our newly proposed covert channel, we write a simple simulation program to emulate the SPRT-based covert

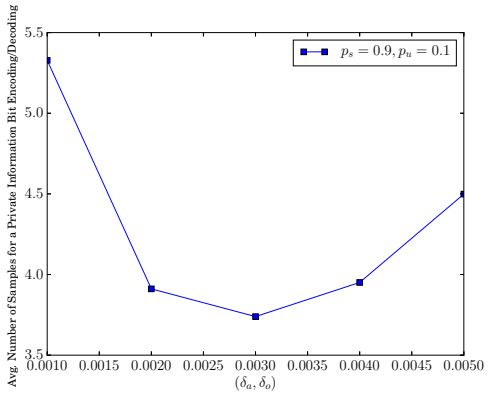

Figure 1: Average number of samples required to encode/decode a private information bit while (δ_a, δ_o) is changed from 0.001 to 0.005.

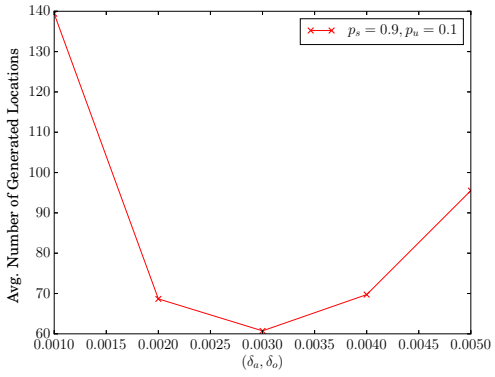

Figure 2: Average number of locations generated per simulation run while (δ_a, δ_o) is changed from 0.001 to 0.005.

channel establishment between trojan app and trojan server. Specifically, the latitude and longitude of trojan app are initially set to 35.0 degree north and 127.0 degree east, respectively. These initial settings represent the latitude and longitude of some area in South Korea. We select a range value uniformly at random in [0, 0.01]. The next pair of latitude and longitude is computed by adding the range value to the current pair of latitude and longitude or subtracting it from the current pair of latitude and longitude. Decision on whether to add or subtract is randomly done. We repeat this process to generate a series of pairs of latitude and longitude. We perform random location selection process and the SPRT with each pair of latitude and longitude.

In simulation program, we set $p_s = 0.9$ and $p_u = 0.1$. We also configure $\alpha' = \beta' = 0.01$ and $c_0 = 0.1$ and $c_1 = 0.9$. The number of private information bits to encode is set to 8 and the value of each private information bit is randomly determined. In addition, we have five distinct configurations of $\delta_a = \delta_o$ ranging from 0.001 to 0.005 in an increase of 0.001. We report the average results of 1000 simulation runs.

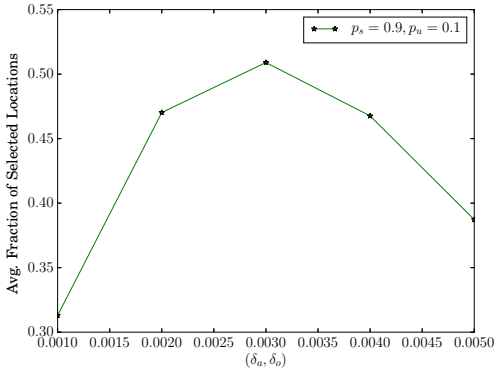

Figure 3: Average fraction of locations selected for the SPRT while (δ_a, δ_o) is changed from 0.001 to 0.005.

As displayed in Figure 1, we perceive that the number of samples required for a private information bit encoding/decoding is below 5.33 on an average in all five configurations of (δ_a, δ_o). This means that a few number of location values are sufficient to encode/decode a private information bit, leading to fast private information bit transmission to trojan server and fast private information bit interpretation in trojan server. As shown in Figures 2 and 3, when $\delta_a = \delta_o = 0.003$, an average number of generated locations and an average fraction of selected locations reaches its minimum and maximum in all five configurations of (δ_a, δ_o), respectively. We infer from this observation that the higher fraction of selected locations contributes to the lower number of generated locations. Moreover, we also see that an average number of samples for a private information bit encoding/decoding reaches its minimum when $\delta_a = \delta_o = 0.003$. This observation signifies that an increase in average fraction of selected locations likely leads to a decrease in average number of samples for a private information bit encoding/decoding.

REFERENCES

[1] Jun-Won Ho, Matthew K. Wright, and Sajal K. Das. 2011. Fast Detection of Mobile Replica Node Attacks in Wireless Sensor Networks Using Sequential Hypothesis Testing. *IEEE Trans. Mob. Comput.* 10, 6 (2011), 767–782.

[2] J. Jung, V. Paxon, A.W. Berger, and H. Balakrishnan. 2004. Fast Portscan Detection Using Sequential Hypothesis Testing. In *IEEE Symp. Security and Privacy*. 211–225.

[3] Jean-François Lalande and Steffen Wendzel. 2013. Hiding Privacy Leaks in Android Applications Using Low-Attention Raising Covert Channels. In *2013 International Conference on Availability, Reliability and Security, ARES 2013, Regensburg, Germany, September 2-6, 2013*. 701–710. https://doi.org/10.1109/ARES.2013.92

[4] Edmund Novak, Yutao Tang, Zijiang Hao, Qun Li, and Yifan Zhang. 2015. Physical media covert channels on smart mobile devices. In *Proceedings of the 2015 ACM International Joint Conference on Pervasive and Ubiquitous Computing, UbiComp 2015, Osaka, Japan, September 7-11, 2015*. 367–378. https://doi.org/10.1145/2750858.2804253

[5] Wen Qi, Yichen Xu, Wanfu Ding, Yonghang Jiang, Jianping Wang, and Kejie Lu. 2015. Privacy Leaks When You Play Games: A Novel User-Behavior-Based Covert Channel on Smartphones. In *23rd IEEE International Conference on Network Protocols, ICNP 2015, San Francisco, CA, USA, November 10-13, 2015*. 201–211. https://doi.org/10.1109/ICNP.2015.40

[6] K. P. Subbalakshmi, Rajarathnam Chandramouli, and Nagarajan Ranganathan. 2007. A Sequential Distinguisher for Covert Channel Identification. *International Journal of Network Security* 5, 3 (November 2007), 274–282.

[7] A. Wald. 2004. *Sequential Analysis*. Dover Publications.