DITA-NCG: Detecting Information Theft Attack Based on Node Communication Graph

Zhenyu Cheng[1], Xiaochun Yun[4] ✉, Shuhao Li[1,2,3], Jinbu Geng[1], Rui Qin[1], and Li Fan[1]

[1] Institute of Information Engineering, Chinese Academy of Sciences
[2] School of Cyber Security, University of Chinese Academy of Sciences
[3] Key Laboratory of Network Assessment Technology, Chinese Academy of Sciences
[4] National Computer Network Emergency Response Technical Team/Coordination Center of China
{chengzhenyu, lishuhao, gengjinbu, qinrui, fanli}@iie.ac.cn, yunxiaochun@cert.org.cn

Abstract. The emergence of information theft poses a serious threat to mobile users. Short message service (SMS), as a mainstream communication medium, is usually used by attackers to implement propagation, command and control. The previous detection works are based on the local perspective of terminals, and it is difficult to find all the victims and covert attackers for a theft event. In order to address this problem, we propose DITA-NCG, a method that globally detects information theft attacks based on node communication graph (NCG). The communication behavior of a NCG's node is expressed by both call detail record (CDR) vectors and network flow vectors. Firstly, we use CDR vectors to implement social subgraph division and find suspicious subgraphs with SMS information entropy. Secondly, we use network flow vectors to distinguish information theft attack graphs from suspicious subgraphs, which help us to identify information theft attack. Finally, we evaluate DITA-NCG by using real world network flows and CDRs , and the result shows that DITA-NCG can effectively and globally detect information theft attack in mobile network.

Keywords: Information theft · CDR · Network flow · Node communication graph

1 Introduction

Smartphones, as the most popular mobile devices, provide a convenient communication way for users and always save amounts of users' information. With the explosive growth of mobile communications, users' privacy security issues are becoming more prominent. The rapid increase of app usage makes smartphones as prime targets for attackers to steal users' information.

As it is reported that most of information theft attacks use short message service (SMS) to spread and use mobile network to send back users' information. Swift Cleaner [23] reported by Trend Micro receives SMS commands to execute

ICCS Camera Ready Version 2022
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-08751-6_25

remote command, steal information, send short messages, etc. FakeSpy [15] is an information stealer delivered by SMS, which steals text messages, account information, contacts, and call records.

Due to the heavy use of SMS transmission in information theft events, call detail record (CDR) data may be an important factor to identify the attacks. Researchers have done some analyses with CDRs. For example, there is a survey [6] that focuses on the analysis of massive CDR datasets and mainly studies the structure of social networks and human mobility. Some CDR researches also focus on the users' privacy security, but most of them only care phone calls [18] or mobility patterns [25, 11]. We consider that we can use CDRs to find out some relationships among these communication nodes, which can help us to identify information theft attack graphs.

In order to protect users from information theft, researchers have done a lot of works to analyse malwares. Static analysis methods like signature-based detection [12] and content-based detection [5] always need some priori knowledge before analyse. Dynamic analysis methods like behavior-based detection [13] usually root the system or make a sand box for detection. There are also some works that use mobile network traffic to analyse users' behavior patterns or apps' features. But few of them focus on detecting information theft. Besides, these works are based on the local perspective of terminals, and it is hard to find all the victims and covert attackers in an information theft event.

In this paper, we propose DITA-NCG, which is a model globally identifying mobile network information theft attacks based on node communication graph (NCG). We use real data to verify the accuracy and validity of our detection model. The result of experiment shows that the detection model can effectively identify information theft attack graphs. The main contributions of this paper are summarized as follows:

- We construct suspicious subgraphs of NCG based on information entropy. By using CDR vectors to achieve social subgraph division, we calculate SMS content length information entropy for each subgraph. The lower value of information entropy means a subgraph is more suspicious.
- We identify information theft attack graphs based on network flow vectors. We employ Convolutional Neural Networks (CNN) to detect network flows of nodes, which are selected from a suspicious subgraph, and then use Support Vector Machine (SVM) to determine whether a new graph is an information theft attack graph or not.
- We evaluate our model by using real world CDRs (200,522 short message CDRs) and network flows (37,384 information theft network flows and 61,635 benign network flows). The result shows that the detection model achieves a 90.55% accuracy, a 92.02% precision and a 94.25% recall.

The rest of this paper is organized as follows: Section 2 highlights the related works. The detailed content of our method is described in Section 3. We present the performance of detecting information theft attack graphs in Section 4. In Section 5, we draw conclusions of this paper and discuss several limitations of our work.

ICCS Camera Ready Version 2022
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-08751-6_25

2 Related Work

Some researchers have investigated the malware identification and private information tracking using network traffic. Taylor et al. [21] implemented Appscanner, which could achieve automatic fingerprinting and real-time identification of Android apps from their encrypted network traffic. Conti et al. [10] designed a system that can identify the specific action when a user was performing on mobile apps, by analyzing the statistical features of Android encrypted traffic. Ren et al. [19] proposed Recon to reveal and control personal identifiable information leaks in mobile network traffic, in which the key/value pairs were used for detection. Wang et al. [22] proposed a malware detection method to identify malware by using the text semantics of network traffic. Alam et al. [4] proposed DroidDomTree to detect malware, which mined the dominance tree of API calls to find similar patterns in Android applications.

There are also some works about CDR data research. Zang et al. [25] proposed an approach to infer, from 30 billion call records, the "top N" locations for each user and correlate this information with publicly-available side information such as census data. Bogomolov et al. [7] presented an approach to predict crime in a geographic space from multiple data sources, particularly in mobile phone and demographic data. Sultan et al. [20] proposed a framework for classifying mobile traffic patterns, which was based on the spatiotemporal analysis of CDR data. Zhang et al. [26] presented a large-scale characterization of fake base station spam ecosystem, which investigated how fraudulent messages are constructed to trap users.

Through summarizing the existing research works as above, we know that although there are some studies focusing on detecting malwares, it is lack of using network flows to detect attack events. And most of CDR data researches focus on human mobility or data analysis. It is lack of using CDRs to identifying information theft attacks.

3 Modeling and Methodology

We propose DITA-NCG to globally detect information theft attacks. Fig. 1 shows the process of detecting information theft attack graphs. And we will introduce our detection model in two phases: data pre-process and attack detection.

3.1 Data Pre-process

First of all, the prototype data need to be reconstructed and filtered. The CDR data generate CDR vectors, and the network traffic data generate network flow vectors.

CDR Vector Generation. A CDR is a data record produced by a telephone communications or other telecommunication transactions. CDR contains various attributes [14, 16, 17], such as: originating telephone number, terminating telephone number, call duration, International Mobile Subscriber Identity

ICCS Camera Ready Version 2022
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-08751-6_25

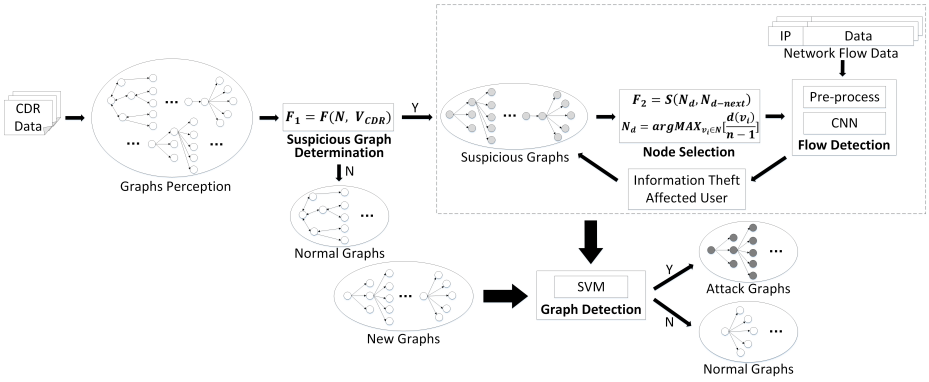

Fig. 1: The Model of Detecting Information Theft Attack Graphs.

(IMSI) number, International Mobile station Equipment Identity (IMEI) number, disposition or the results of the call, call type, etc. In actual modern practice, CDRs are much more detailed.

In information theft events, attackers always use SMS to achieve command and control. And these messages always have the same content and format. Since the information theft apps spread and command through SMS, we take the data redundancy elimination for CDRs. Firstly, we reserve the originating and terminating phone numbers since the phone number is essential. Secondly, we get the start-time, end-time and duration from the time stamp. Thirdly, we remain the telecommunication type which can determine whether the communication is "send" or "received". At last, the SMS content' length is retained, which is an important characteristic for detecting malicious communication nodes, and we will explain it in Section 4. The other data segments like IMSI, IMEI, location data, etc., are discarded. In order to protect users' privacy, phone numbers and SMS contents are masked and desensitized.

We extract some features and statistics to generate the CDR vector, which contains: originating number, terminating number, send time, content length, average content length and time interval for the node, out-degree and incoming-degree of originating and terminating number.

Flow Vector Generation. Information theft attackers usually use apps to steal users' personal information through network traffic, so we want to analyse network flows to detect information theft. From our previous research works [8, 9], we recognized that: (1) Different flows have different lengths (packet number) in a solitary complete session. (2) The flows' content sizes are much different from each other. (3) The ratio of incoming traffic and outgoing traffic is distinct for different flows. Therefore, we decide to use the network flows to detect information theft attack.

In this phase, we restructure the traffic data and divide them into a time series of packets. The network flow is regarded as a fundamental entity, and we define it as a sequence of TCP/IP packets ordered by time in a single session. Because

ICCS Camera Ready Version 2022
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-08751-6_25

really valued data are just parts of network flows, flow filtering is necessary to further improve the effectiveness of extracted features. We discard worthless information by taking domain filtering, packet filtering and so on.

Except for regular features, we also select some new features such as values of Dynamic Time Warping (DTW). DTW is an algorithm used to measure the similarity between two temporal sequences. It is recursively defined as:

$$min\_dtw = MIN(DTW(i-1,j), DTW(i-1,j-1), DTW(i,j-1)) \quad (1)$$

$$DTW(i,j) = local\_distance(i,j) + min\_dtw \quad (2)$$

We use $DTW(s,t)$ to compare the "shapes" of network flow s and t. To reduce the computation burden of calculating DTW, a leader flow is elected for each app, and this can be expressed as:

$$arg\ MIN_{f_l \in F} \left(\sum_{i=1}^{n} DTW(f_l, f_i) \right) \quad (3)$$

The flow f_l represents the leader flow of F, and f_i represents other flows.

As a result, a network flow vector contains: the values of DTW, duration, and number of packets, flow length, average packet length, average packet interval for different flow type (incoming, outgoing and complete).

3.2 Attack Detection

Information theft attack graph detection can help us globally find all the relative nodes in information theft events. Fig. 2 illustrates a case of information theft attack. The attacker uses SMS to lure users to download malwares, which always contains a malicious URL. After receiving the malicious SMS, the user may not be controlled to resend the mal-SMS to all of his/her contacts. Because the usually contacted people have a higher degree of credibility, the infected user will be forced to send the mal-SMS to his/her frequent contacts. However, if a user does not read the message or download the suspicious app, he/she will be safe in this attack. When a user trusts the source of SMS and click the URL, he/she will access a server controlled by the attacker and automatically download the malwares. Then his/her smartphone will be infected and send the same mal-SMS to others.

In this phase, our model achieves the goal of detecting attack graphs of information theft. After the pre-process, we use the CDR vectors to generate NCG, which is based on the phone numbers of SMS. In graphs perception, the model gets a lot of communication subgraphs. Then the detecting model uses the similarity matching function to identify suspicious subgraphs and normal subgraphs. The function is defined as:

$$F_1 = F(N, V_{CDR}) \quad (4)$$

N represents the node set of a divided subgraph, and V_{CDR} represents the set of CDR vectors of N. Function $F()$ calculates the information entropy of the

ICCS Camera Ready Version 2022
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-08751-6_25

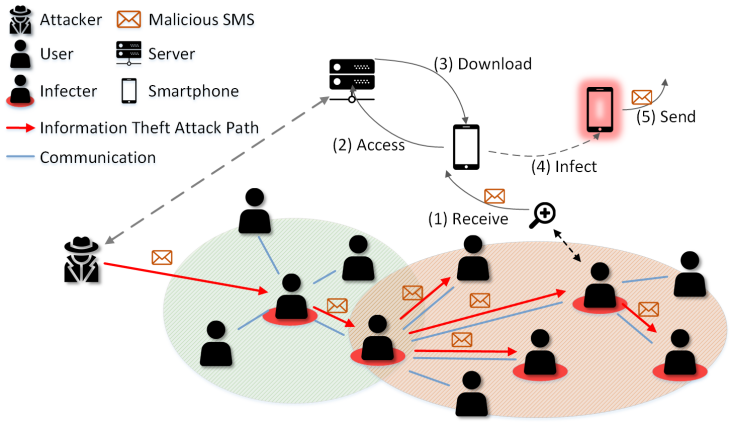

Fig. 2: The case of information theft attack.

communication subgraph and use the result to detect suspicious subgraphs. The information entropy of the subgraph is defined as:

$$H(M) = -\sum_{m \in M} p(m) \log p(m) \tag{5}$$

M represents the node's SMS content length set, and m represents a separate SMS content length. $p(m)$ indicates the probability of m occurrence in M. We use 2 as the base of the logarithm. $H(M)$ is the information entropy of SMS content length of the subgraph, and its value gets large, the SMS content lengths are volitile, and the information gets more confusing and complex. If the information entropy value gets lower, the subgraph will be more suspicious.

After obtaining the suspicious subgraphs, DITA-NCG will choose some nodes of a suspicious subgraph to further detect whether the graph is normal or malicious. The node selection function is used to select communication nodes for detection, and it is defined as:

$$F_2 = S(N_d, N_{d-next}) \tag{6}$$

$$N_d = \arg MAX_{v_i \in N}[d(v_i)/(n-1)] \tag{7}$$

N_d represents the node that has the max degree centrality in the graph, and N_{d-next} is the next hop communicating nodes of N_d. $d()$ is the function to calculate the node's degree. For the selected nodes, DITA-NCG detects each node's network flows to identify whether it is truly under information theft attack or not.

In network flow detection, we use CNN for detecting. As one of the most popular deep learning algorithms, CNN overcomes the difficulties in feature extraction and is good at extracting local features. And in our previous work [9], we have already verified the effectiveness of CNN. In this paper, we use the network flow vectors as input. And then we set 4 filters with the kernel size of 3. At last, we use the "sigmoid" function to squash the single-unit output layer. After

ICCS Camera Ready Version 2022
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-08751-6_25

training, we take the number 0 as the label of normal flows, and the number 1 as the label of information theft flows. From the result of flow detection, we can judge whether a suspicious subgraph is an information theft attack graph or not.

Due to the time limit of detection, we train an SVM classifier to detect all of the information theft attack graphs by using the classified graph sets. In machine learning, SVM is a supervised learning model with associated learning algorithms. When the new graphs are added to the training samples, the model will retrain and update the classifier. The process of our model's detection is described in Algorithm 1.

Algorithm 1 The algorithm of detecting information theft attack graphs

Input:
Node vector data: $N = \{N_0, N_1, ..., N_n\}$.
Output:
Classified graphs: $G = \{G_{normal}, G_{attack}\}$.
Pre-processing and normalization N;
Establish the relationships of N to generate the communication graphs: G_{all};
Suspicious graph determination: $G_{suspicious}$;
while true **do**
  if $G_{suspicious}$!= NULL **then**
    Select and remove one subgraph $graph_s$ from $G_{suspicious}$;
    Random select nodes form $graph_s$;
    Network flow detection with CNN;
    if label == 1 **then**
      Classify the graph into G_{attack};
    else
      Classify the graph into G_{normal};
    end if
  else
    All the subgraphs of $G_{suspicious}$ is classified;
    break;
  end if
end while
Use SVM to detect new graphs with G;

4 Experiment and Evaluation

A server computer (Intel Core i7-4790 3.60GHz with 8GB DDR3 RAM) is used as a control center, which is running Windows 7 with two network cards. It is emulated as a router to receive and save network traffic data. Some smartphones are used to generate traffic data, like Galaxy Note 4 (SM-N9100) and Xiaomi 4 (MI 4LTE) both running the Android 6.0.1 operation system. And a computer is used to run several virtual devices to simulate traffic generation. These virtual devices run with different versions of system. A sever is used to save and analyse data with a GPU GeForce GTX Titan X.

ICCS Camera Ready Version 2022
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-08751-6_25

4.1 Experimental Data

We collect 113 information theft app samples provided by CNCERT/CC (National Computer Network Emergency Response Technical Team/Coordination Center of China). But we find that most of them can not run properly. Because these apps appear a long time, their C&C servers can not be connected, and they can not steal information by their program logic. And due to the update of operation system version, some of them can not be installed correctly. Therefore, we extract the information theft modules from these apps and recode them into ITM-capsule. Also we design a simulated C&C server for remote control. ITM-capsule steals the users' information such as phone identification information, contacts list, call history, SMS, etc. As shown in Fig. 3, ITM-capsule runs in the Android system to communicate with the server and transfer users' information. The process of information theft is: (1) ITM-capsule collects phone basic information(i.e. phone number, IMEI, MAC, etc.), and sends it to the C&C server. (2) The sever checks for presence of the phone's information. If there is no information about this phone, the database will create a new ID. (3) The server returns the ID to the phone. (4) ITM-capsule adds the ID to all the next transferred contents to ensure the information unity and sends them to the sever. (5) The sever saves the received information into the database. At last, when the whole transmission procedure ends, i.e., the sever has not received new data for a long time, it will send a stop command to stop ITM-capsule's work.

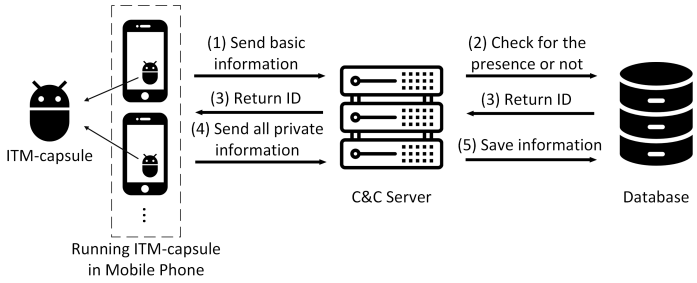

Fig. 3: The information theft process of ITM-capsule.

We also download lots of information theft apps from VirusShare [2], which is a repository of malware samples accessible for security researchers, and this set contains 293 samples. Furthermore, we collect 1,012 apps from YingYongBao, which is a popular third party app market of Tencent [1]. As we know that not all the collected apps are benign, we upload these apps to VirusTotal [3] to ensure the credibility of the training set. As a result, we get 997 benign apps. Using our own designed network traffic collection platform, the information theft apps and ITM-capsule generate 7.8 GB traffic data, and the benign apps generate 6.5 GB traffic data. Then we obtain 37,384 and 61,635 network flows respectively from information theft and benign traffic data. To ensure the independence of capture process, we tag all the flows for different devices and times. As a result, there are 11,752 clusters of network flows tagged.

ICCS Camera Ready Version 2022
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-08751-6_25

We get a set of CDRs through an internet service provider. It should be noted that the CDR data is desensitized and masked. We promise that all the users' private information is protected and we do not use SMS contents in the experiment, so there is no ethical issues. The CDR dataset is 5.2 GB, and it contains 3 days' 986,752 records. After discarding records of phone call and base station, etc., we extract 200,522 SMS CDRs, which belong to 10,329 phone numbers. And there are 1,846 phone numbers tagged with suspicious label. The format of the processed data is shown in the table 1. "OPN" means originating phone number, and "TPN" means terminating phone number.

Table 1: The Format of CDR Dataset

No.	type	OPN	TPN	content_len	time_stamp
1	7	68d81229d5ec9c...	b04b1fa4a5b3c0...	2	1512954871
2	7	b04b1fa4a5b3c0...	68d81229d5ec9c...	12	1513214984
3	8	b58765baf88359...	f9195b5d19f7b7...	66	1513046068
4	8	903eabe1b70d2b...	f9195b5d19f7b7...	26	1513130349
5	7	c2c9376a350ee9...	b8f02594b747d3...	17	1513146196
...	...	...	...	...	...

We have analysed two cases of information theft apps, Cckun and xxShenQi [24]. Combining with CDR dataset, Fig. 4 shows the comparison of the SMS content length of different node type examples. In this figure, "ad_1" and "ad_2" represent two advertisement node types; "sus_1" and "sus_2" represent two suspicious node types. The figure illustrates that SMS content length is an important feature for classification. If the node's SMS content lengths are convergent, its information entropy value will be low, and this node will be more suspicious. From the examples of Fig. 4, we can realise that suspicious nodes' SMS content length is similar to malicious one's. How to distinguish advertisement nodes and malicious nodes is really a question. To solve this problem, we use network flows to determine whether a node is malicious or not.

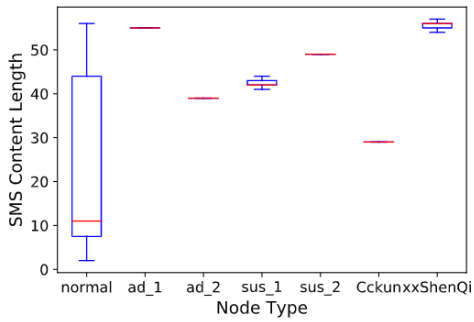

Fig. 4: The comparison of the SMS content length.

Based on the real proportion of these actual information theft events, we match the network flow vectors and CDR vectors by labels. From the collected data (11,752 clusters of tagged network flows and 10,329 phone numbers of

ICCS Camera Ready Version 2022
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-08751-6_25

CDRs), we select 10,000 as the number of nodes. To ensure the randomness and generality of our research, we randomly select 10,000 phone numbers' CDRs and devices' network flows to make a node vector set. The set contains 10,000 nodes and 79,637 communication paths, and 1,722 infected nodes and 18,915 information theft propagation paths. There are also 1,247 advertisement communication paths, which add some extraneous factors for simulating reality scenarios.

4.2 Experimental Result and Analysis

Because it is almost impossible to deal with the network flows of millions of nodes in the tolerable time. We need to find out the suspicious nodes firstly. We can establish NCG with the generated CDR vectors, and our model can lock on suspicious graphs with extracted features. The next step is to identify whether these nodes are really in information theft events or just are advertisement nodes or some other third service nodes. It is practicable to detect information theft flows and identify node type by using the network flow vectors. After that, we can check the other nodes of the communication paths by victims' CDRs. Fig. 5 shows an example of detecting information theft attack graph. The big red spot is a attacker that sends malicious SMS (No.1 and 5 indicate the different propagation paths), and the small red spots are victims, who are infected but not controlled to resend malicious SMS. The yellow spots are also infected, moreover, they send malicious SMS to their contacts (like No.2, 3 and 4 indicating). The green spots are advertiser and their target users. With these confirmed spots, we can identify the information theft attack graphs.

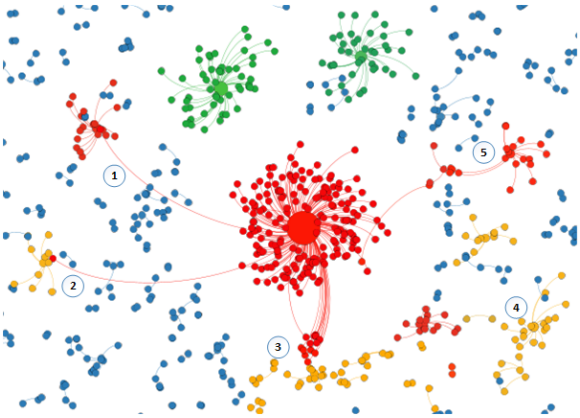

Fig. 5: An example of information theft attack graph detection.

In addition, we analyse the training set size requirement. Fig. 6 illustrates different values with different sizes of the training set. The values of recall, F-Measure, precision and false positive rate (FPR) are indicated with different lines. We can find out that when the training set size reaches 2,000, the values

ICCS Camera Ready Version 2022
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-08751-6_25

of metrics get stable and effective. Therefore, we set 2,000 as the size of the training set and 8,000 for the testing set.

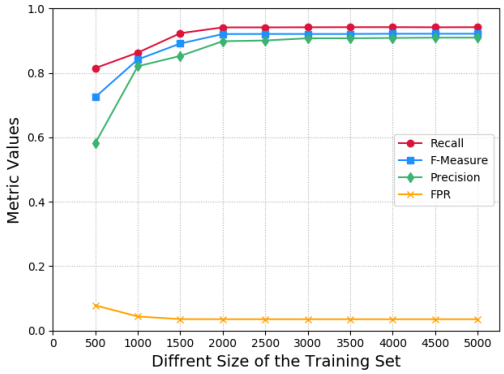

Fig. 6: The recall, F-Measure, precision and FPR values with different sizes of the training set.

Table 2 shows the results of accuracy, precision, recall, FPR and F-Measure. The detection rate for information theft attack graphs reaches 94.25% whereas the misjudgment rate for normal paths is only 3.53%. All the values demonstrate that our model is effective in identifying information theft attack graphs in mobile network.

Table 2: The Performance of DITA-NCG

	Accuracy	**Precision**	**Recall**	**FPR**	**F-Measure**
Value	0.9055	0.9102	0.9425	0.0353	0.9260

The chosen of SVM in our model is based on the comparison with other representative machine learning algorithms, such as Random Forest, Naive Bayes and AdaBoost. Fig. 7 illustrates that different algorithms have different performances. Particularly, SVM has a higher accuracy, precision, recall and F-Measure in this work, while its FPR is lower than others. Overall, the results show that SVM is better to solve the problems of detecting information theft attack graphs.

4.3 Comparison with Other Methods

Due to the differences of research perspective and experimental data, almost all the popular detection methods can not run with our experimental data. Therefore, we make a qualitative analysis to compare our model with others. Table 3 shows the comparison between our model and other methods, such as TaintDroid [13], AppScanner [21] and Recon [19]. D_1, D_2 and D_3 separately represents information theft detection, mobile network traffic detection and attack events detection. “No Root” and “No Sand-box” mean that there is no rooting system and no establishing a sand-box during detection. “Perspective” shows the detection method is based on the global or local perspective.

ICCS Camera Ready Version 2022
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-08751-6_25

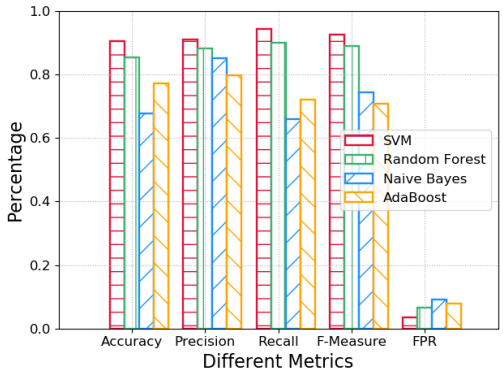

Fig. 7: The comparison between SVM and other machine learning algorithms.

Table 3: The Comparison with Other Methods

	D_1	D_2	D_3	*No Root*	*No Sand-box*	*Perspective*
DITA-NCG	√	√	√	√	√	global
TaintDroid	√	-	-	-	-	local
AppScanner	-	√	-	√	√	local
Recon	√	√	-	√	-	local

5 Conclusion and Discussion

Information theft attacks pose a significant threat to the security of smartphone users. In order to globally find out the victims and attackers of information theft events, we propose DITA-NCG. We use SMS information entropy to identify suspicious subgraphs, and use CNN to detect the network flows of the selected nodes in suspicious subgraphs. At last, we use SVM to detect information theft attack graphs from new ones. We also develop a simulated information theft app, ITM-capsule, to solve the problem of C&C server invalidation and system version incompatibility. In the experiment, we evaluate the performance of DITA-NCG, which shows that our method achieves a high rate in accuracy, precision and recall. And it should be noted that all personally identifiable information (PII) in the dataset used in our experiment have been anonymized. The result indicates that DITA-NCG is effective to detect mobile network information theft attack globally, and it has a great potential to be applied in real scenarios for internet service providers.

However, there are some limitations about the proposed model. We use network flows and CDRs to generate vectors for identification, but most of the data are collected after information theft attacks. Although we can detect the information theft, how to prevent and block the attacks is remained to be solved in the future. And we will continue the research to establish a blocking mechanism to perfect the detecting model.

ICCS Camera Ready Version 2022
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-08751-6_25

There are also some other ways to spread malwares, like uploading a fake app to third party markets, using phishing web sites, etc. These propagation ways may not spread rapidly or not have a bad influence, although they are really exist in the world. In the further research, we will comprehensively consider the information theft propagation.

The automation needs to be improved for DITA-NCG. And the model still needs to be ran with the data of real information theft events, although we have established a simulated communication node set based on some real samples.

Acknowledgement. This work is supported by the National Key Research and Development Program of China (Grant No.2019YFB1005201). We would also like to thank the reviewers for the thorough comments and helpful suggestions.

References

1. App market of yingyongbao. https://android.myapp.com/ (2021)
2. Virusshare. https://virusshare.com/ (2021)
3. Virustotal. https://www.virustotal.com/ (2021)
4. Alam, S., Alharbi, S.A., Yildirim, S.: Mining nested flow of dominant apis for detecting android malware. Computer Networks **167**, 107026 (2020)
5. Arzt, S., Rasthofer, S., Fritz, C., Bodden, E., Bartel, A., Klein, J., Le Traon, Y., Octeau, D., McDaniel, P.: Flowdroid: Precise context, flow, field, object-sensitive and lifecycle-aware taint analysis for android apps. Acm Sigplan Notices **49**(6), 259–269 (2014)
6. Blondel, V.D., Decuyper, A., Krings, G.: A survey of results on mobile phone datasets analysis. EPJ data science **4**(1), 10 (2015)
7. Bogomolov, A., Lepri, B., Staiano, J., Oliver, N., Pianesi, F., Pentland, A.: Once upon a crime: towards crime prediction from demographics and mobile data. In: Proceedings of the 16th international conference on multimodal interaction. pp. 427–434. ACM (2014)
8. Cheng, Z., Chen, X., Zhang, Y., Li, S., Sang, Y.: Detecting information theft based on mobile network flows for android users. In: 2017 International Conference on Networking, Architecture, and Storage (NAS). pp. 1–10. IEEE (2017)
9. Cheng, Z., Chen, X., Zhang, Y., Li, S., Xu, J.: Mui-defender: Cnn-driven, network flow-based information theft detection for mobile users. In: International Conference on Collaborative Computing: Networking, Applications and Worksharing. pp. 329–345. Springer (2018)
10. Conti, M., Mancini, L.V., Spolaor, R., Verde, N.V.: Analyzing android encrypted network traffic to identify user actions. IEEE Transactions on Information Forensics and Security **11**(1), 114–125 (2016)
11. De Montjoye, Y.A., Hidalgo, C.A., Verleysen, M., Blondel, V.D.: Unique in the crowd: The privacy bounds of human mobility. Scientific reports **3**, 1376 (2013)
12. Desnos, A., et al.: Androguard: Reverse engineering, malware and goodware analysis of android applications. URL code. google. com/p/androguard **153** (2013)
13. Enck, W., Gilbert, P., Chun, B., Cox, L.P., Jung, J., Mcdaniel, P., Sheth, A.: Taintdroid: an information flow tracking system for real-time privacy monitoring on smartphones. Communications of The ACM **57**(3), 99–106 (2014)

ICCS Camera Ready Version 2022
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-08751-6_25

14. Horak, R.: Telecommunications and Data Communications Handbook. Wiley (2007), https://books.google.com/books?id=dO2wCCB7w9sC
15. N, B.: Fakespy - android information stealing malware attack to steal text messages, call records & contacts. https://gbhackers.com/fakespy/ (2019)
16. Petersen, J.: The Telecommunications Illustrated Dictionary. CRC Press advanced and emerging communications technologies series, CRC Press (2002), https://books.google.com/books?id=b2mMzS0hCkAC
17. Peterson, K.: Business Telecom Systems: A Guide to Choosing the Best Technologies and Services. Taylor & Francis (2000), https://books.google.com/books?id=W79R0niNU5wC
18. Ratti, C., Sobolevsky, S., Calabrese, F., Andris, C., Reades, J., Martino, M., Claxton, R., Strogatz, S.H.: Redrawing the map of great britain from a network of human interactions. PloS one **5**(12), e14248 (2010)
19. Ren, J., Rao, A., Lindorfer, M., Legout, A., Choffnes, D.: Recon: Revealing and controlling pii leaks in mobile network traffic. In: Proceedings of the 14th Annual International Conference on Mobile Systems, Applications, and Services. pp. 361–374. ACM (2016)
20. Sultan, K., Ali, H., Ahmad, A., Zhang, Z.: Call details record analysis: A spatiotemporal exploration toward mobile traffic classification and optimization. Information **10**(6), 192 (2019)
21. Taylor, V.F., Spolaor, R., Conti, M., Martinovic, I.: Appscanner: Automatic fingerprinting of smartphone apps from encrypted network traffic. In: 2016 IEEE European Symposium on Security and Privacy (EuroS&P). pp. 439–454. IEEE (2016)
22. Wang, S., Yan, Q., Chen, Z., Yang, B., Zhao, C., Conti, M.: Detecting android malware leveraging text semantics of network flows. IEEE Transactions on Information Forensics and Security **13**(5), 1096–1109 (2017)
23. Wu, L.: First kotlin-developed malicious app signs users up for premium sms services. http://t.cn/EMSyiof (2019)
24. Yun, X., Li, S., Zhang, Y.: Sms worm propagation over contact social networks: Modeling and validation. IEEE Transactions on Information Forensics and Security **10**(11), 2365–2380 (2015)
25. Zang, H., Bolot, J.: Anonymization of location data does not work: A large-scale measurement study. In: Proceedings of the 17th annual international conference on Mobile computing and networking. pp. 145–156. ACM (2011)
26. Zhang, Y., Liu, B., Lu, C., Li, Z., Duan, H., Hao, S., Liu, M., Liu, Y., Wang, D., Li, Q.: Lies in the air: Characterizing fake-base-station spam ecosystem in china. In: Proceedings of the 2020 ACM SIGSAC Conference on Computer and Communications Security. pp. 521–534 (2020)

ICCS Camera Ready Version 2022
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-08751-6_25