INTERNATIONAL RESEARCH JOURNAL IN ADVANCED ENGINEERING AND TECHNOLOGY (IRJAET) E - ISSN: 2454-4752 P - ISSN : 2454-4744
VOL 4 ISSUE 1 (2018) PAGES 2720 - 2723
RECEIVED : 15.01.2018 PUBLISHED : 13.01.2018
Feb 13, 2018

MALWARE DETECTION OF GAME WITH MOBILE DEVICE USING CLOUD TECHNOLOGY

[1]S. Deepa, ME Scholar, Department Of Computer Science And Engineering, Idhaya Engineering College for women, Chinna Salem.

[2]P. Mohanavalli, Assistant Professor, Department Of Computer Science And Engineering, Idhaya Engineering College for women, Chinna Salem.

ABSTRACT

As accurate malware detection on Mobile devices requires fast process of a large number of application traces, cloud-based malware detection can utilize the data sharing and powerful computational resources of security servers to improve the detection performance. In this project, I Investigate the cloud-based malware detection game, in which Mobile devices offload their application traces to security servers via base stations or access points in dynamic networks. I derive the Nash equilibrium (NE) of the static malware detection game and present the existence condition of the NE, showing how Mobile devices share their application traces at the security server to improve the detection accuracy, and compete for the limited radio bandwidth, the computational and Communication resources of the server. I design a malware detection scheme with Q-learning for a Mobile device to derive the optimal offloading rate without knowing the trace generation and the radio bandwidth model of other Mobile devices. The detection Performance is further improved with the Dyna architecture, in which a Mobile device learns from the hypothetical experience to increase its convergence rate. We also design a post-decision state learning-based scheme that utilizes the known radio channel model to accelerate the reinforcement learning process in the malware detection. Simulation results show that the proposed schemes improve the detection accuracy, reduce the detection delay and increase the utility of a Mobile device in the dynamic malware detection game, compared with the benchmark strategy.

1. INTRODUCTION:

Malwares such as viruses, worms, Trojans, and spy tools seriously threaten Mobile devices such as smartphones and tablets with the privacy leakage, economic loss, power depletion, and network performance degradation. Malware detection systems as proposed in explore the features of the runtime behavior of thousands of applications (apps) on Mobile devices and involve logging data at each application execution. For instance, lines of log data have to be scanned to detect malware on an Android smartphone according to Norton Mobile Security. The application traces have to be scanned in real time based on the latest malware signature files downloaded from the security database to avoid the privacy loss due to the zero-day vulnerability. However, that is not always applicable for a Mobile device with the limited battery life, computation resources and network bandwidth. The design of the search technique

has the following implications: first, the worms can spread much faster, since they do not have to probe for susceptible hosts and second, the rate of failed connections is less. Thus, rapid proliferation of malware can pose a serious security threat to the functioning of mobile networks. Understanding the factors affecting the malware spread can help facilitate network designs that are resilient to attacks, ensuring protection of the networking infrastructure.

This project addresses this issue and develops an analytic framework for modeling the spread of malware in mobile networks while accounting for the architectural, topological, and user related factors. It have proposed analytical models for the temporal evolution of information in the network. The focus of these works is on transfer of regular files and they do not apply to malware that spread actively. However, these models ignore node dynamics such as online-offline transitions and are applicable only to Bit Torrent networks. We develop the model in two stages: first, we quantify the average number of peers within TTL hops from any given peer and in the second stage incorporate the neighborhood information into the final model for malware spread.

2. RELATED WORK

In a mobile cloud offloading system mobile users such as smartphones or tablets offload security related data to a security server in a cloud via servered access points (APs) or base stations (BSs), in order to improve their malware detection speeds by utilizing the computational resources of the server, such as its CPUs, disks and memories. In a dynamic mobile cloud offloading game, with time variant network environment, it is very challenging for each smartphone to accurately estimate its radio bandwidth and the actions of the other mobile users that offload to the same security server. Hence, reinforcement learning methods, such as Q-learning, can be employed by smartphones to achieve their optimal offloading strategies ultimately.

3. LITERATURE SURVEY

Marco V. Barbera comes with In our proposed, when offloading decision takes place, input and code are likely to be already on the cloud. CDroid makes mobile cloud offloading more practical enabling offloading of lightweight jobs and communication intensive apps. Our experiments with real users in everyday life show excellent results in terms of energy savings and user experience. For this reason we build CDroid (Cloud-anDroid), an offloading-aware system transparent to the user, and distribute it to real-users to assess its performance. We design CDroid so that it enables offloading of communication-intensive apps, and increases the gain of offloading of computation-intensive apps. In CDroid every device comes with its own cloud counterpart, seen as just another resource of the real device, only a 3G/LTE Advanced connection away from it. So, it brings the device and the cloud closer, towards fully integrating.

Julinda Stefa comes with I implement CloneDoc, a real-time collaboration system for Smartphone users working simultaneously on the same document. I use CloneDoc as a paradigmatic p2p-like application that makes use of heavy crypto primitives and communication among peers. Our experiments

show that by making use of C2C with the CloneDoc system we enable energy saving to the real devices by offloading computation of heavy tasks to the cloud. Simultaneously, by offloading also communication, the C2C platform allows for cellular bandwidth savings. So, we demonstrate that C2C makes possible to run a whole new class of distributed applications on mobile devices.

4. PROPOSED SYSTEM

In this project addresses this issue and develops an analytic framework for modeling the spread of malware in MOBILE networks while accounting for the architectural, topological, and user related factors. We also model the impact of malware control strategies like node quarantine.

We have proposed analytical models for the temporal evolution of information in the network. The focus of these works is on transfer of regular files and they do not apply to malware that spread actively. In addition, they are specialized to Bit Torrent like networks and cannot be extended for MOBILE networks such as Gnutella or KaZaa.

In the authors use hyper cubes as the graph model for MOBILE networks and derive a limiting condition on the spectral radius of the adjacency graph, for a virus/worm to be prevalent in the network. The models do not account for the fact that once a peer is infected, any susceptible peer within a TTL hop radius becomes a likely candidate for a virus attack.

5. RESULT ANALYSIS

In the first simulation, both users have constant transmission bandwidths over time, with $1/6 \leq b1 \leq 1/2$ and $b2 = 1/4$. The performance of the NE of a static mobile cloud offloading game, as theoretic results, is similar to the stable performance of a dynamic game, as shown in Fig. 2. In addition, a user increases its offloading rate and thus obtains a higher utility, under a higher transmission bandwidth. On the other hand, the other mobile user has a performance degradation if competing a smartphone with a higher bandwidth. In the second simulation, we have evaluated the performance of a dynamic mobile cloud offloading game, averaged over all realizations of random and time variant bandwidths, with $bk\ 1 \in \{1/6, 1/5, 1/4\}$ and $bk\ 2 \in \{1/8, 1/7, 1/6\}$, $k = 1, 2, 3, \cdots$, i.e., user 1 usually has no worse transmission condition. As shown in Fig. 3, both mobile users quickly achieve their optimal offloading strategies. For instance, the average utility of user 1 increases from 1.5 to more than 2.5 after 1000 time slots from the start of the game. In addition, user 1 with a generally higher bandwidth relies more on the cloud and achieves a higher utility. Finally, the instantaneous utilities of two users in the dynamic offloading game are shown in Fig. 4, in which both $bk\ 1$ and $bk\ 2$ randomly change every 100 time slots, with $bk\ 1 \in \{1/6, 1/5, 1/4\}$ and $bk\ 2 \in \{1/8, 1/7, 1/6\}$, $\forall k$. It is shown that both users achieve their optimal offloading strategies with stable and high utilities.

CONCLUSION:

In this Project , I have formulated a cloud-based malware detection game and derived the NE of the game, showing how a mobile device chooses its offloading rate to make a tradeoff between the transmission cost and the detection performance. A Q-learning based malware detection strategy has been proposed for the dynamic game in time-variant radio networks, and the performance is further improved by applying the Dyna architecture and the known radio channel model. Simulation results show that the proposed Q-learning based malware detection improves the detection accuracy by 40%, reduces the detection delay by 15%, and increases the utility of the mobile devices by 47%, compared with the benchmark strategy with 100 mobile devices. The PDS-based malware detection has the highest detection accuracy and utility, while the Dyna-Q scheme has the fastest response.

REFERENCE

[1] L. Xie, X. Zhang, J. P. Seifert, and S. Zhu, "pbMDS: A behavior based malware detection system for cell phone devices," in *Proc.*ACM Conf. Wireless Network Security, 2010, pp. 37 – 48.

[2] B. Amos, H. Turner, and J. White, "Applying machine learning classifiers to dynamic android malware detection at scale," in Proc.IEEE Int'l Wireless Commun and Mobile Computing Conf. *(IWCMC),*2013, pp. 1666–1671.

[3] T. Isohara, K. Takemori, and A. Kubota, "Kernel-based behavior analysis for android malware detection, " in Proc. IEEE Int'l Conf. Computational Intelligence and Security *(CIS)*, 2011, pp. 1011 – 1015.

[4] B. Kang, K. Han, B. Kang, and E. Im, "Malware categorization using dynamic mnemonic frequency analysis with redundancy filtering," Digital Investigation, vol. 11, no. 4, pp. 323–335, 2014.

[5] S. Zonouz, A. Houmansadr, R. Berthier, N. Borisov, and W. Sanders, "Secloud: A cloud-based comprehensive and lightweight security solution for smartphones," Computers & Security, vol. 37, pp. 215–227, 2013.

[6] G. Portokalidis, P. Homburg, K. Anagnostakis, and H. Bos, "Paranoid android: Versatile protection for smartphones," in Proc. IEEE Computer Security Applications Conf., 2010, pp. 347 – 356.

[7] A. Houmansadr, S. A. Zonouz, and R. Berthier, "A cloud-based intrusion detection and response system for mobile phones," in *Proc. IEEE* Int'l Conf. Dependable Systems and Networks Workshops, 2011, pp. 31 – 32.