ISSN: 0975-5662, July, 2015 www.ijrct.org

Suppression of Malware and Behavioral Detection in Delay Tolerant Networks

#M.Chandana[1], M.TechComputer Science &Engineering, E-mail: chandana22.m@gmail.com
#Dr.Siddhartha Ghosh[2],Professor And HOD,Department of CSE, E-mail: profsiddhartha@gmail.com
# Keshav Memorial Institute Of Technology(KMIT), Narayanguda, Hyderabad, Telangana, India.

Abstract: With the universal presence of short-range connectivity technologies (e.g., Bluetooth and, more recently, Wi-Fi Direct) in the consumer electronics market, the delay tolerant network (DTN) model is becoming a viable alternative to the traditional infrastructural model. In this paper, we address the proximity malware detection and containment problem with explicit consideration for the unique characteristics of DTNs. We formulate the malware detection process as a decision problem under a general behavioral malware characterization framework. We analyze the risk associated with the decision problem and design a simple yet effective malware containment strategy, look-ahead, which is distributed by nature and reflects an individual node's intrinsic trade-off between staying connected (with other nodes) and staying safe (from malware). Furthermore, we consider the benefits of sharing assessments among directly connected nodes and address the challenges derived from the DTN model to such sharing in the presence of liars (i.e., malicious nodes sharing false assessments) and defectors (i.e., good nodes that have turned malicious due to malware infection). Real mobile network traces are used to verify our analysis.

Index Terms: Malware Delay-tolerant networks, Proximity malware, and Behavioral malware characterization.

1. INTRODUCTION

Mobile consumer electronics permeate our lives. Laptop computers, PDAs, and more recently and prominently, smart-phones, are becoming indispensable tools for our academic, professional, and entertainment needs. These new devices are often equipped with a diverse set of non-infrastructural connectivity technologies, e.g., Infra-red, Bluetooth, and more recently, Wi-Fi Direct. With the universal presence of these short-range connectivity technologies, the communication paradigm, identified by the networking research community under the umbrella term Delay-tolerant Networks (DTNs), is becoming a viable alternative to the traditional infrastructural paradigm. Because of users' natural mobility, new information distribution applications, based on peer-to-peer contact opportunities instead of persistent connection channels among nodes, are considered to be the game changer for future network applications.

The popularity of new mobile devices (e.g., smart phones), the adoption of common platforms (e.g., Android), and the economic incentive to spread malware combined exacerbate the malware problem in DTNs. Malware is a piece of malicious code which disrupts the host node's functionality and duplicates and propagates itself to other nodes via contact opportunities. In the traditional infrastructural model, the carrier serves as a gatekeeper who can centrally monitor network abnormalities and inhibit malware propagation; moreover, the resource bottleneck for individual nodes naturally limits the impact of the malware. However, the central gatekeeper and natural limitations are absent in the DTN model. Proximity malware, which exploits the temporal dimension and distributed nature of DTNs in self-propagation, poses serious threats to users of new technologies and challenges to the networking and security research community. A common malware detection method currently in practice is pattern matching. More concretely, a sample of malware is first reported by an infected user. The sample is analyzed by security specialists, and a pattern which (hopefully) uniquely identifies the malware is extracted; the pattern can be either code or data, binary or textual. The pattern is then used for the detection of malware.

The analysis and extraction often involve extensive manual labor and expertise. The overhead, the lack of generality, and high false positive rate in one round of analysis make it unsuitable for promising DTN applications on smart devices. The quest for a better malware detection method comes to the very question of how to characterize proximity malware in DTNs. In this paper, we consider an approach to characterize proximity malware by the behaviors of an infected node observed by other nodes in multiple rounds. The individual observation can be imperfect for one round, but infected nodes' abnormal behavior will be distinguishable in the long-run. Methods like pattern matching can be used in one round of observation for the behavioral characterization of proximity malware.

Instead of assuming a sophisticated malware containment capability, such as patching or self-healing, we consider the simple capability of "cutting off communication". In other words, if a node suspects another node j of being infected with the malware, i may cease to connect with j in the future. We want to explore how far such a simple technique can take us. Our focus is on how individual nodes make such cut-off decisions based on direct and indirect observations.

Due to the temporal dimension and distributed nature of DTNs, the major challenge faced by the proximity malware behavioral detection and containment mechanism is a decision problem: when to cut-off? This challenge can be compared with a motivating example in real life. When a person smells something burning, he or she is facing with two choices. One is to call the fire emergency service immediately; the other is to collect more evidence and to make a more informed decision later. The first choice is associated with a high cost for a possible false fire alarm, while the second choice is associated with the risk of losing the early opportunity of containing the fire. We are facing a similar dilemma in the context of proximity malware in DTNs. Hyper-sensitivity leads to high false positive while hypo-sensitivity leads to high false negative. In this paper, we present a simple yet effective solution which reflects an individual node's intrinsic trade-off between staying connected with other nodes and staying safe from the malware. We also consider the benefits of sharing observations among nodes and address the challenges of liars and defectors derived from the DTN model.

2. RELATED WORK

Consider a DTN consisting of n nodes. The neighbors of a node are the nodes it has contact opportunities with. Each node keeps a log, chronologically recording the neighbors it had contact with, and uses this log to estimate its contact pattern with them. A proximity malware is a piece of malicious code which disrupts the host node's functionality and has a chance of duplicating itself to other nodes during inter-nodal communication; when duplication occurs, we say the other node is infected by the malware. Suppose each node is capable of assessing the other party for suspicious actions after each encounter, resulting in a binary assessment of either suspicious or non-suspicious. An example of a suspicious action is sending a self-signed program which modifies system configurations.

By the *functional* assumption on the suspicious-action assessment, we characterize a node's nature by its frequency of suspicious actions. More specifically, if node i has N (pair-wise) encounters with its neighbors and sN of them are assessed as suspicious by the other party, its *suspiciousness* Si is defined as:

$$S_i = \lim_{N\to\infty} \frac{s_N}{N}; \qquad (1)$$

We assume the existence of such a limit and therefore have Si ∈ [0, 1]. A number Le ∈ (0, 1) is chosen2 as the line between good and evil. Node i is deemed well if Si ≤ Le or evil if Si > Le. In other words, we draw a fine line between good and evil, and judge a node by its deeds.

At any particular time, we say a node's nature is either evil or good based on if it is or is not infected by the malware. We assume that the suspicious-action assessment is an imperfect, but functional indicator of malware infection: it may assess an evil node's actions as "non-suspicious" (or a good node's actions as "suspicious") at times, but most suspicious actions are correctly attributed to evil nodes.

Before proceeding to further discussions, we make explicit the assumptions in the neighborhood-watch model: an evil node's behavior is consistent and non-targeted.

- **Consistency.** This rephrases the functional assumption in characterizing a node's nature by the suspiciousness (Equation 1). Only those nodes with suspiciousness higher than the threshold Le are capable of transmitting the malware. In other words, a node cannot do the evil (transmitting the malware) and pretend to be good (maintaining a low suspiciousness).

- **Non-targetedness.** An evil node j's suspicious actions should be observable to all of its neighbors rather than a specific few. Otherwise, if j targets at i, i's other neighbors' opinions, even faithful ones, only confuse i. This assumption is vital to all evidence collecting methods which incorporate neighbors' observations.

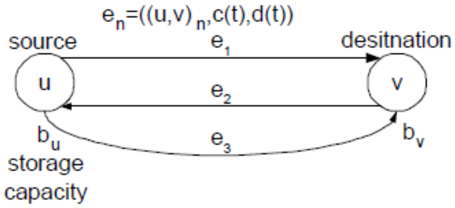

Figure 1: Edges in a DTN graph.

Proximity malware based on the DTN model brings unique security challenges that are not present in the infrastructure model. In the infrastructure model, the cellular carrier centrally monitors networks for abnormalities; moreover, the resource scarcity of individual nodes limits the rate of malware propagation. The Delay Tolerant Networks (DTNs) are especially useful in providing mission critical services including emergency scenarios and battlefield applications. However, DTNs are vulnerable to wormhole attacks, in which a malicious node records the packets at one location and tunnels them to another colluding node, which replays them locally into the network. Nodes in disruption tolerant networks (DTNs) usually exhibit repetitive motions. Several recently proposed DTN routing algorithms have utilized the DTNs' cyclic properties for predicting future forwarding. Opportunistic data forwarding can be abused by an adversary by injecting spurious packets in order to waste the resources of the network. Security and privacy are critical for DTNs.

DTN NETWORK MODEL:

The DTN graph is a directed multi-graph, in which more than one edge (also called link) may exist between a pair of nodes (see Figure 1). The reason for using a multigraph is straightforward: it may be possible to select between two distinct (physical) connection types to move data between the same pair of nodes. Furthermore, the link capacities (and to a lesser extent, propagation delay) are time-dependent (capacity is zero at times when the link is unavailable). Thus, the set of edges in the graph must capture both time-varying capacity and propagation delay as well as multiple parallel edges. A simple example of an edge captured by this description involves a ground station and a LEO satellite rising, passing directly overhead, and setting at the opposite horizon. As it rises, its channel capacity will generally increase until it is directly overhead and will decrease for the remaining time of the pass. This is because noise is minimal when the satellite is directly overhead but increases at lower elevations.

Another example would be a bus (carrying a wireless access point) passing by a village. The throughput of the wireless link would depend upon the distance of the bus from the village. When no communication is possible, the edge is assigned zero capacity. While DTN applications are expected to be tolerant of delay, this does not mean that they would not beneath from decreased delay. Furthermore, we believe this metric is an appropriate measure to use in exploring the differential evaluation of several routing algorithms in an application-independent manner. Minimizing delay lowers the time messages spend in the network, reducing contention for resources (in a qualitative sense). Therefore, lowering delay indirectly improves the probability of message delivery. This is validated by our simulation results.

3. IMPLEMENTATION

Proximity malware and existing prevention schemes. A number of studies demonstrate the severe threat of proximity malware propagation. Su et al. collected Bluetooth scanner traces and used simulations to show that malware can effectively propagate via Bluetooth. Yan et al. developed a Bluetooth malware model. Bose and Shin showed that malware that uses both SMS/MMS and Bluetooth can propagate faster than by messaging alone. Rather than assuming a sophisticated malware containment capability, such as patching or self-healing in previous works, we base our design on quarantine and develop a decision mechanism using direct and indirect observations to deal with proximity malware. Packet forwarding in mobile networks. In mobile networks, one cost-effective way to route packets is via shortrange communication capabilities of intermittently connected smart phones. While early work in mobile networks used a variety of simplistic random id. models, such as random waypoint, recent findings show that these models may not be realistic. Moreover, many recent studies, based on real mobile traces, revealed that nodes' mobility showed certain social network properties. We use two real mobile network traces in our study.

Trust evaluation schemes. We base our design on the observation that trust evaluations can link past experiences with future predictions. Various frameworks have been designed to model trust relationships. Three schools of thoughts emerge from studies. The first one uses a central authority, which by convention is called the trusted third party. In the second school, one global trust value is drawn and published for each node, based on other nodes' opinions of it. EigenTrust is an example. The last school includes the trust management systems that allow each node to have its own view of other nodes. Unlike these works, we evaluate trustworthiness on pieces of evidence rather than on individual nodes; this allows us to promptly cope with changing nature of nodes with minimum overhead.

Protecting a victim (host or network) from malicious traffic is a hard problem that requires the coordination of several complementary components, including nontechnical and technical solutions. The implementing malware detection and access control rules are very tedious because the network has so many vulnerabilities and security issues. The proposed system introduces a new protocol which is named as COMPACT (Combinatorial Optimal Malware Proclamation and Content Tracking). The decentralized approach provides effective rule matching and verification process in the network while data transmission. Access Control List has also applied in order to maintain black and white list of users and nodes for effective data restriction. The importance of the COMPACT protocol is facilitating a solution against filter selection problem.

CONCLUSION

In this paper, we address the proximity malware detection and containment problem with explicit consideration for the characteristics of DTNs. Rather than relying on a particular malware detection technique (e.g., viral pattern matching), we propose a general behavioral characterization of malware infection, which allows for functional but imperfect assessments on malware presence. Under this framework, we formulate the malware detection process as a decision problem, analyze the risk associated with the decision problem, and design a simple yet effective malware containment strategy, lookahead, which is distributed by nature and reflects an individual node's intrinsic trade-off between staying connected (with other nodes) and staying safe (from malware). Furthermore, we consider the benefits of sharing assessments among directly connected nodes and address the challenges derived from the DTN model in the presence of liars (i.e., malicious nodes sharing false assessments) and defectors (i.e., good nodes that have turned malicious due to malware infection). Real mobile network traces are used to verify our analysis. In prospect, the proposed behavioral malware characterization and the presented malware detection and containment method provide clearer understanding on the prevention of proximity malware in DTNs and serve as a foundation for future work along this line.

REFERENCES:

[1] NFC Forum. about NFC, http://goo.gl/zSJqb, 2013.

[2] Wi-Fi Alliance. Wi-Fi Direct, http://goo.gl/fZuyE. 2013.

[3] CPLEX: Linear Programming Solver. http://www.ilog.com/.

[4] DTN Research Group. http://www.dtnrg.org/.

[5] K. Fall. A Delay-Tolerant Network Architecture for Challenged Internets. In ACM SIGCOMM, Aug. 2003.

[6] L. R. Ford and D. R. Fulkerson. Flows in Networks. Princeton University Press, 1962.

[7] V. Erramilli, M. Crovella, A. Chaintreau, and C. Diot, "Delegation forwarding," in Proc. of MobiHoc. ACM, 2008.

[8] W. Hsu, T. Spyropoulos, K. Psounis, and A. Helmy, "Modeling time-variant user mobility in wireless mobile networks," in Proc. of INFOCOM. IEEE, 2007.

[9] E. Daly and M. Haahr, "Social network analysis for routing in disconnected delay-tolerant manets," in Proc. of MobiHoc. ACM, 2007.

[10] N. Djukic, M. Piorkowski, and M. Grossglauser, "Island hopping: Efficient mobility-assisted

forwarding in partitioned networks," in Proc. of SECON. IEEE, 2006.

[11] A. Josang, R. Ismail, and C. Boyd, "A survey of trust and reputation systems for online service provision," Decision Support Systems, vol. 43, no. 2, pp. 618–644, 2007.

[12] Trend Micro Inc. SYMBOS_CABIR.A., http://goo.gl/aHcES, 2004.

[13] http://goo.gl/iqk7, 2013.

[14] Trend Micro Inc. IOS_IKEE.A., http://goo.gl/z0j56, 2009.