INTERNATIONAL JOURNAL OF RESEARCH IN COMPUTER APPLICATIONS AND ROBOTICS
ISSN 2320-7345

MALWARE DETECTION IN DTN BASED ON ITS BEHAVIOR

R.P.Kaaviya Priya[(1)],G.Bhavani[(2)],

[(1)]PG Student-CSE, [(2)] Assistant professor-CSE,
Krishnasamy College of Engineering and Technology, Cuddalore.
[(1)]kavy0690@gmail.com,
[(2)]bhavaniamir@gmail.com

Abstract

The Behavioral characterization of malware is an effective alternative to pattern matching in detecting malware. The viable communication with mobile consumer electronics equipped with short range communication technologies such as Bluetooth, Wi-Fi Direct is DTN. There exists a general behavior characterization of proximity malware based on Naive Bayesian model, It was identified with two unique challenges for extending Bayesian malware detection to DTNs (Delay Tolerant Networks) . so we propose a simple and effective method "look ahead", to address the challenges with two extensions to look ahead, dogmatic filtering, and adaptive look ahead, they address the challenge of "malicious nodes sharing false evidence." Real mobile network traces are used to verify the effectiveness.

Keywords: Adaptive Look ahead, Bayesian filtering, Dogmatic filtering, Proximity malware.

1. Introduction

The widespread adoption of the mobile devices, coupled with strong economic incentives, induces a class of malware that specifically targets DTNs. We call this class of malware proximity malware. An early example of proximity malware is the Symbian based *Cabir* worm[1] A later example is the iOS-based *Ikee* worm, which exploited the default SSH password on jail-broken[2] iPhones to propagate through IP-based Wi-Fi connections[3]. Previous researches [4],[5] quantify the threat of proximity malware attack in NFC and Wi-Fi direct[6][7]. Proximity malware based on the DTN model brings unique security challenges that are not present and also malware propagation cannot be detected by the cellular carrier in the traditional model. In this paper, we consider a general behavioural characterization of proximity malware. Behavioural characterization, in terms of system call and program flow has been previously proposed as an effective alternative to pattern matching for malware detection [8], [9].Malware-infected node behaviours are observed by others during their multiple opportunistic encounters: Individual observations may be imperfect, but abnormal behaviours of infected nodes are identifiable in the long-run. The imperfection of a single, local observation was previously in the context of distributed IDS against slowly propagating worms [10].Instead of assuming a sophisticated malware containment capability, such as patching or self-healing [11],[12] we consider a simple "cut-off" strategy. Our focus is on how individual nodes shall make such cut-off decisions against potentially malware-infected nodes,

based on direct and indirect observations. In the context of DTNs, we face a dilemma when trying to detect proximity malware: Hypersensitivity leads to false positives, while hyposensitivity leads to false negatives. Our solution, *look ahead*, reflects individual node's intrinsic risk inclinations against malware infection, to balance between these two extremes. Essentially, we extend the naive Bayesian model, which has been applied in filtering email spams [13], [14], [15], detecting botnets [16], and designing IDSs [10], [17] and address two DTN specific, malware-related, problems on observation with individual nodes:

1. Insufficient evidence versus evidence collection risk: In DTNs, evidence (such as Bluetooth connection or SSH session requests) is collected only when nodes come into contact. But contacting malware-infected nodes carries the risk of being infected. Thus, nodes must make decisions (such as whether to cut off other nodes and, if yes, when) online based on potentially insufficient evidence.

2. Filtering false evidence in sequential and distributed manner: Sharing evidence among opportunistic acquaintances helps alleviating the afore mentioned insufficient evidence problem; however, false evidence shared by malicious nodes (the liars) may negate the benefits of sharing. In DTNs, nodes must decide whether to accept received evidence sequentially. This paper includes the following contributions:

1. A general behavioural characterization of proximity malware, which captures the imperfect nature in detecting proximity malware.
2. With a simple cut-off malware containment strategy, we formulate the malware detection process as a distributed decision problem.
3. Also the benefits of sharing assessments among nodes, and address challenges derived from the DTN model: liars (i.e., bad-mouthing and false praising malicious nodes) and defectors (i.e., good nodes that have turned rogue due to malware infections).
4. Two alternative techniques are proposed, *dogmatic filtering and adaptive look ahead*, which naturally extend look ahead to consolidate evidence provided by others, while containing the negative effect of false evidence. A nice property of the proposed evidence consolidation methods is that the results will not worsen even if liars are the majority in the neighbourhood. Real contact traces are used to verify the effectiveness of the methods.

2. Model

Consider a DTN consisting of n nodes. The neighbours of a node are the nodes it has (opportunistic) contact opportunities with. Proximity malware is a malicious program that disrupts the host node's normal function and has a chance of duplicating itself to other nodes during (opportunistic) contact opportunities between nodes in the DTN. When duplication occurs, the other node is infected with the malware. In our model, we assume that each node is capable of assessing the other party for suspicious actions after each encounter, resulting in a binary assessment. For example, a node can assess a Bluetooth connection or an SSH session for potential Cabir or Ikee infection. The watchdog components in previous works on malicious behavior detection in MANETs [18] and distributed reputation systems [19], [20] are other examples A node is either evil or good, based on if it is or is not infected by the malware. The suspicious action assessment is assumed to be an imperfect but functional indicator of malware infections: It may occasionally assess an evil node's actions as "non-suspicious" or a good node's actions as "suspicious," but most suspicious actions are correctly attributed to evil nodes. The functional assumption characterizes a malware infected node by the assessments of its neighbours. If node i has N (pair-wise) encounters with its neighbours and s_N of them are assessed as suspicious by the neighbours, its suspiciousness S_i is defined as

$$S_i = \lim_{N\to\infty} \frac{SN}{N}$$

$S_i \in [0, 1]$. A number $L_e \in (0,1)$ is chosen as the line between good and evil. Le depends on the quality of a particular suspicious-action assessment and, if the assessment is a functional discriminant feature of the malware and the probabilistic distribution of the suspiciousness of both good and evil nodes are known, L_e can be chosen as the (Bayesian) decision boundary, which minimizes classification errors[21]. Node i is good if $S_i \leq L_e$, or evil if $S_i > L_e$: We draw a fine line between good and evil, and judge a node by its deeds. Instead of assuming a sophisticated malware coping mechanism, such as patching or self-healing, we consider a simple and widely

applicable malware containment strategy: Based on past assessments, a node i decides whether to refuse future connections ("cut Off") with a neighbour j.

3. Design

A. Household Watch

Consider the case in which i bases the cut-off decision against j only on i's own assessments on j. Since only direct assessments are involved, we call this model household watch Let A = (a_1, a_2, . . . a_A) be the assessment sequence (a_i is either 0 for "non-suspicious" or 1 for "suspicious") in chronological order, i.e., a_1 is the oldest assessment, and a_A is the newest one. Baye's theorem tells us

$$P(Sj \mid A) \propto P(A \mid Sj)\ X\ P(Sj)$$

Where $P(S_j)$ encodes our prior belief on j's suspiciousness S_j; $P(A \mid S_j)$ is the likelihood of observing the assessment sequence A given Sj; $P(S_j|A)$ is the posterior probability, representing the plausibility of j having a suspiciousness of S_j given the observed assessment sequence A. Since the evidence $P(A)$ does not involve S_j and serves as a normalization factor in the computation, we omit it and write the quantitative relationship in the less cluttered proportional form. The structure of the behavioral malware characterization model (specifically, a single threshold L_e is used to distinguish the nature of a node) gives rise to a subtlety concerning i's prejudice against j in the distribution approach. The online supplemental material, if i makes no presumption on j's suspiciousness, when no assessment has been made yet. This leads to a discussion on whether such prejudices are warranted. The choice of Le depends on the assessment mechanism itself and, as mentioned previously, if the probabilistic distributions of suspiciousness of both good and evil nodes are known, can be determined by minimizing Bayesian decision errors. If $L_e > 0.5$, the assessment mechanism is biased toward false positive (good nodes actions being assessed as suspicious); if $L_e < 0.5$, the assessment mechanism is biased toward false negative (evil nodes' actions being assessed as non suspicious). However, before any assessment is made, i has no clue about the true nature of j.

A bias in the assessment mechanism should not affect the i's neutrality on j's nature before the first assessment is made. Thus, we stipulate that the comparison between $P_g(A)$ and $P_e(A)$ should be made only when $A \neq \emptyset$.Alternatively, in the maximizer approach, i uses the suspiciousness distribution's maximizer (see (4)) when making the cut-off decision against j. The justification for the maximizer approach is that the suspicious distribution's maximizer is the single most probable estimation of j's suspiciousness given the evidence. The maximizer approach precludes the prejudice problem, because the maximizer is undefined when $A=\emptyset$. Similar to the distribution approach, i compares evidence that is both favourable and unfavourable to j. Evidence A is favourable to j if $s_A/|A| \leq L_e$ and is unfavourable to j if $s_A/|A| > L_e$. The maximizer approach significantly reduces the computation cost, in comparison with the distribution approach, while partially discarding information contained in the suspiciousness distribution derivable from the evidence collected so far. Whichever approach is taken, the cut-off decision problem has an asymmetric structure in the sense that cutting j off will immediately terminate the decision process (i.e., i will cease connecting with j; no further evidence will be collected), while the opposite decision will not. Thus, we only need to consider the decision problem when i consider cutting j off due to unfavourable evidence against j. The cut-off decision is made based on the risk estimation of such a decision.

The key insight is that i shall estimate the cut-off decision's risk by looking ahead. More specifically, given the current assessment sequence A=(a_1, . . . , a_A), the next assessment a_{A+1} (which has not been taken yet) might be either 0 (non suspicious) or 1 (suspicious If the evidence is still unfavourable toward j, we say that i's decision of cutting j off is one-step-ahead robust. If the cut-off decision is one-step ahead robust, i is certain that exposing itself to the potential danger of infection by collecting one further assessment on j will not change the outlook that j is evil. Similarly, i can look multiple steps ahead. In fact, the number of steps i is willing to look ahead is a parameter of the decision process rather than a result of it. This parameter shows i's willingness to be exposed to a higher infection risk in exchange for a higher certainty about the nature of j and a lower risk of cutting off a good neighbour; in other words, it reflects i's intrinsic risk inclination against malware infection.

Definition 1 (Look-Ahead λ):
The look-ahead _ is the number of steps i is willing to look ahead before making a cut-off decision. We can make a similar decision-robustness definition for look-ahead λ.
Definition 2 (λ-Robustness):
At a particular point in i's cut-off decision process against j (with assessment sequence A = (a_1, . . . , a_A)), i's decision of cutting j off is said to beλ-step-ahead robust, or simply λ-robust, if 1) the current evidence A is unfavourable toward j; and 2) even if the next λ assessments (a_{A+1},. . . ,$a_{A+}\lambda$)) all turn out to be non suspicious (i.e., 0), the evidence against j is still unfavourable. Given the look-ahead λ, the proposed malware containment strategy is to cut j off if the cut-off decision is λ-robust, and not to cut j off otherwise. In Section 2 of the online supplemental material, we discuss how to adapt the look-ahead λ to individual nodes' intrinsic risk inclinations against the malware.

TABLE 1
Data Set Statistics

	nodes	entries	time span	avg. interval
Haggle	41	112, 295	15 days	12 secs
MIT reality	96	114, 046	490 days	371 secs

TABLE 2
Neighbor Nature and Cut-Off Decision Combination

	...gets cut off.	...stays connected.
An evil neighbor...	True positive.	False negative.
A good neighbor...	False positive.	True negative.

B. Neighbourhood Watch

Besides using i's own assessments, i may incorporate other neighbours' assessments in the cut-off decision against j. This extension to the evidence collection process is inspired by the real-life neighbourhood (crime) watch program, which encourages residents to report suspicious criminal activities in their neighbourhood. Similarly, i shares assessments on j with its neighbours, and receive their assessments on j in return. In the neighbourhood-watch model, the malicious nodes that are able to transmit malware (we will see next that there may be malicious nodes whose objective is other than transmitting malware) are assumed to be consistent over space and time. These are common assumptions in distributed trust management systems which incorporate neighbouring nodes' opinions in estimating a local trust value. By being consistent over space, we mean that evil nodes suspicious actions are observable to all their neighbours, rather than only a few. If this is not the case, the evidence provided by neighbours, even if truthful, will contradict local evidence and, hence, cause confusions:

Nodes shall discard received evidence and fall back to the household watch model. By being consistent over time, we mean that evil nodes cannot play strategies to fool the assessment mechanism. This is equivalent to the functional assumption in characterizing the nature of nodes by suspiciousness. The case in which the evil nodes can circumvent the suspiciousness characterization (such as by first accumulating good assessments, and then launch an attack through a short burst of concentrated suspicious actions) calls for game-theoretic analysis and design, and is beyond the scope of this paper. Instead, we propose a behavioural characterization of proximity malware; further game theoretic analysis and design could base on this foundation.

C. Challenges

Two cases complicate the neighbourhood watch model: liars and defectors. Liars are those evil nodes who confuse other nodes by sharing false assessments. A false assessment is either a false praise or a false accusation. False praises understate evil nodes' suspiciousness, while false accusations exaggerate good nodes' suspiciousness. Furthermore, a liar can fake assessments on nodes that it has never met with. To hide their true nature, liars may do no evil other than lying, and, therefore, have low suspiciousness. Defectors are those nodes that change their nature due to malware infections. They start out as good nodes and faithfully share assessments with their neighbours; however, due to malware infections, they become evil. Their behaviours after the infection are under the control of the malware. These complications call for evidence consolidation. Two extremal, but naive, evidence-consolidation strategies are 1) to trust no one and 2) to trust everyone. The former degenerates to the household-watch model with the twist of the defectors (defectors change their nature and hence their behavioral pattern); the latter leads to confusions among good nodes.

D. Evidence analysis and Evidence aging

For a pair of neighbouring nodes i and j, let N_i and N_j be the neighbours of i and j, respectively. At each encounter, i shares with j its assessments on the neighbour set Ni – {j}, and j shares with i its assessments on the neighbour set N_j –{i}.Since the cut-off decision only needs to be made against a neighbour, i only considers the assessments of its own neighbours $N_i \cap (N_j - \{i\})$ from the evidence provided by j without superimposed trust relationships among the nodes in the model, i and j only share their own assessments, instead of forwarding the ones provided by their neighbours

The presence of defectors breaks the assumption when we characterize a node's nature by suspiciousness. A defector starts as a good node but turns evil due to malware infections; the assessments collected before the defector's change of nature, even truthful, are misleading. To alleviate the problem of outdated assessments, old assessments are discarded in a process called evidence aging. Each assessment is associated with a timestamp. Only assessments with timestamps less than a specific aging window T_E from now are included in the cut-off decision. To see that the aging window T_E alleviates the defector problem, consider a node that is infected at time T. Without evidence aging, all evidence before T mounts to testify that the node is good; if the amount of this prior evidence is large, it may take a long time for its neighbours to find out about the change in its nature. In comparison, with evidence aging, at time T+ T_E, all prior evidence expires and only those assessments after the infection are considered, which collectively testify against the node. However, in practice, the choice of the aging window T_E depends on the context. While a small T_E may speed up the detection of defectors by reducing the impact of stale information, TE must be large enough to accommodate enough assessments to make a sound cut-off decision. If T_E is too small, a node will not have enough assessments to make an λ-robust cut-off decision.

E. Evidence Consolidation

We propose two alternative methods, dogmatic filtering and adaptive look ahead, for consolidating evidence provided by other nodes, while containing the negative impact of liars. For exposition, we consider a scenario in which node i uses the assessments within the evidence aging window [T-T_E, T] provided by i's neighbours (other than one of the neighbours, say, j) in making the cut-off decision against j. The implications are as follows:

1. Given enough assessments, honest nodes are likely to obtain a close estimation of a node's suspiciousness (suppose they have not cut the node off yet), even if they only use their own assessments.
2. The liars have to share a significant amount of false evidence to sway the public's opinion on a node's suspiciousness.
3. The most susceptible victims of liars are the nodes that have little evidence.

Dogmatic filtering. Dogmatic filtering is based on the observation that one's own assessments are truthful and, therefore, can be used to bootstrap the evidence consolidation process. A node shall only accept evidence that will not sway its current opinion too much. We call this observation the dogmatic principle. Dogmatic filtering significantly contains the impact of liars on i while still allowing a change of certainty (on j's nature) comparable to its own.

Adaptive look ahead. Adaptive look ahead takes a different approach toward evidence consolidation. Instead of deciding whether to use the evidence provided by others directly in the cut-off decision, adaptive look-ahead indirectly uses the evidence by adapting the steps to look ahead to the diversity of opinion.

4. Simulation

A. Data Sets

We verify our design with two real mobile network traces: Haggle [22] and MIT reality [23]. The raw data sets are rich in information, some of which is irrelevant to our study, for example, call logs and cell tower IDs in MIT reality. Therefore, we remove the irrelevant fields and retain the node IDs and time-stamps for each pair-wise node encounter. Since the Haggle data set has only 22,459 entries spanning over three days, we repeat

it another four times to make it into a data set with 112,295 entries spanning over 15 days, and thus make it comparable to the MIT reality data set in quantity.

B. Setup

Without loss of generality, we choose Le =0.5 to be the line between good and evil. For each data set, we randomly pick 10 percent of the nodes to be the evil nodes and assign them with suspiciousness greater than 0.5, the rest of the nodes are good nodes and are assigned suspiciousness less than 0.5. For a particular pair wise encounter, a uniform random number is generated for each node; a node receives a "suspicious" assessment (by the other node) if the random number is greater than its suspiciousness and receives a "non suspicious" assessment otherwise. Thus, each assessment is binary, while the frequency of "suspicious" assessments for a particular node reflects its suspiciousness in the long term.

C. Performance Metric

The performance comparison is based on two metrics: detection rate and false positive rate. The categories of the "neighbour's nature" and "cut-off decision" combinations are shown in Table 2. For each combination, we sum up all the decisions made by good nodes (evil nodes' cut-off decisions are irrelevant) and obtain four counts: TP (true positives), FN (false negatives), TN (true negatives), and FP (false positives). The detection rate DR is defined as

$$DR = \frac{TP}{TP+FN} \text{ X } 100\%$$

And the false positive rate FPR is defined as

$$FPR = \frac{FP}{FP+TN} \text{ X } 100\%$$

5. Related Work

Proximity malware and mitigation schemes. Su et al. [24] collected Bluetooth traces and demonstrated that malware could effectively propagate via Bluetooth with simulations. Yan et al. [25] developed a Bluetooth malware model. Bose and Shin [26] showed that Bluetooth can enhance malware propagation rate over SMS/MMS. Cheng et al. [27] analyzed malware propagation through proximity channels in social networks. Akritidis et al. [4] quantified the threat of proximity malware in wide-area wireless networks. Li et al. [28] discussed optimal malware signature distribution in heterogeneous, resource-constrained mobile networks. In traditional, non-DTN, networks, Kolbitsch et al. [8] and Bayer et al. [9] proposed to detect malware with learned behavioral model, in terms of system call and program flow. We extend the Naive Bayesian model, which has been applied in filtering email spams [13], [14], [15], detecting botnets [16], and designing IDSs [10], [17], and address DTN-specific, malware-related, problems. In the context of detecting slowly propagating Internet worm, Dash et al. presented a distributed IDS architecture of local/global detector that resembles the neighbourhood-watch model, with the assumption of attested/honest evidence, i.e., without liars [10]. Mobile network models and traces. In mobile networks, one cost-effective way to route packets is via the short-range channels of intermittently connected smart phones [29], [30], [31]. While early work in mobile networks used a variety of simplistic random i.i.d. models, such as random waypoint, recent findings [32] show that these models may not be realistic. Moreover, many recent studies [33], based on real mobile traces, revealed that a node's mobility shows certain social network properties. Two real mobile network traces were used in our study. Reputation and trust in networking systems. In the neighbourhood watch model, suspiciousness, defined in (1), can be seen as nodes' reputation; to cut a node off is to decide that the node is not trustworthy. Thus, our work can be viewed from the perspective of reputation/trust systems. Our work differs from previous trust management work in addressing two DTN specific, malware-related, trust management problems:1) insufficient evidence versus evidence collection risk and 2) sequential and distributed online evidence filtering.

6. Concluding Remarks

Behavioral characterization of malware is an effective alternative to pattern matching in detecting malware, especially when dealing with polymorphic or obfuscated malware. Naive Bayesian model has been successfully applied in non-DTN settings. The general behavioural characterization of DTN-based proximity malware with look ahead proposed, along with dogmatic filtering and adaptive look ahead, to address two unique challenging in extending Bayesian filtering to DTNs: "insufficient evidence versus evidence collection risk" and "filtering false evidence sequentially and in a distributed manner." In prospect, extension of the behavioral characterization of proximity malware to account for strategic malware detection evasion with game theory is a challenging yet interesting future work.

REFERENCES

[1] Trend Micro Inc. SYMBOS_CABIR.A., http://goo.gl/aHcES, 2004.
[2] http://goo.gl/iqk7, 20 13.
[3] Trend Micro Inc. IOS_IKEE.A., http://goo.gl/z0j56, 2009.
[4] P. Akritidis, W. Chin, V. Lam, S. Sidiroglou, and K. Anagnostakis, "Proximity Breeds Danger: Emerging Threats in Metro-Area Wireless Networks," Proc. 16th USENIX Security Symp., 2007.
[5] A. Lee, "FBI Warns: New Malware Threat Targets Travelers, Infects via Hotel Wi-Fi," http://goo.gl/D8vNU, 2012.
[6] NFC Forum.about NFC, http://goo.gl/zSJqb, 2013.
[7] Wi-Fi Alliance. Wi-Fi Direct, http://goo.gl/fZuyE. 2013.
[8] C. Kolbitsch, P. Comparetti, C. Kruegel, E. Kirda, X. Zhou, and X. Wang, "Effective and Efficient Malware Detection at the End Host," Proc. 18th Conf. USENIX Security Symp.
[9] U. Bayer, P. Comparetti, C. Hlauschek, C. Kruegel, and E. Kirda, "Scalable, Behavior-Based Malware Clustering," Proc. 16th Ann. Network and Distributed System Security Symp. (NDSS), 2009.
[10] D. Dash, B. Kveton, J. Agosta, E. Schooler, J. Chandrashekar, A. Bachrach, and A. Newman, "When Gossip is Good: Distributed Probabilistic Inference for Detection of Slow Network Intrusions," Proc. 21st Nat'l Conf. Artificial Intelligence (AAAI), 2006.
[11] G. Zyba, G. Voelker, M. Liljenstam, A. Me´hes, and P. Johansson, "Defending Mobile Phones from Proximity Malware," Proc. IEEE
INFOCOM, 2009.
[12] F. Li, Y. Yang, and J. Wu, "CPMC: An Efficient Proximity Malware Coping Scheme in Smartphone-Based Mobile Networks," Proc.
IEEE INFOCOM, 2010.
[13] I. Androutsopoulos, J. Koutsias, K. Chandrinos, and C. Spyropoulos, "An Experimental Comparison of Naive Bayesian and Keyword-Based Anti-Spam Filtering with Personal E-Mail Messages," Proc. 23rd Ann. Int'l ACM SIGIR Conf. Research and Development in Information Retrieval (SIGIR), 2000.
[14] P. Graham, "Better Bayesian Filtering," http://goo.gl/AgHkB, 2013.
[15] J. Zdziarski, Ending Spam: Bayesian Content Filtering and the Art of Statistical Language Classification. No Starch Press, 2005.
[16] R. Villamarı´n-Salomo´n and J. Brustoloni, "Bayesian Bot Detection Based on DNS Traffic Similarity," Proc. ACMymp. Applied Computing (SAC), 2013.
[17] J. Agosta, C. Diuk-Wasser, J. Chandrashekar, and C. Livadas, "An Adaptive Anomaly Detector for Worm Detection," Proc. Second USENIX Workshop Tackling Computer Systems Problems with Machine Learning Techniques (SYSML), 2007.
[18] S. Marti et al., "Mitigating Routing Misbehavior in Mobile Ad Hoc Networks," Proc. ACM MobiCom, 2000.
[19] P. Michiardi and R. Molva, "Core: A Collaborative Reputation Mechanism to Enforce Node Cooperation in Mobile Ad Hoc
Networks," Proc. IFIP TC6/TC11 Sixth Joint Working Conf. Comm. and Multimedia Security, p. 107, 2002.

[20] S. Buchegger and J. Le Boudee, "Self-Policing Mobile Ad Hoc Networks by Reputation Systems," IEEE Comm. Magazine, vol. 43, no. 7, pp. 101-107, July 2005.
[21] R.O. Duda, P.E. Hart, and D.G. Stork, Pattern Classification, second ed. Wiley-Interscience, Nov. 2001.
[22] J. Scott, R. Gass, J. Crowcroft, P. Hui, C. Diot, and A. Chaintreau, "CRAWDAD Data Set Cambridge/Haggle (v. 2006-09-15)," http://goo.gl/RJrKN, Sept. 2006.
[23] N. Eagle and A. Pentland, "CRAWDAD Data Set MIT/Reality (v. 2005-07-01)," http://goo.gl/V3YKc, July 2005.
[24] J. Su, K. Chan, A. Miklas, K. Po, A. Akhavan, S. Saroiu, E. de Lara, and A. Goel, "A Preliminary Investigation of Worm Infections in a Bluetooth Environment," Proc. Fourth ACM Workshop Recurring Malcode (WORM), 2006.
[25] G. Yan, H. Flores, L. Cuellar, N. Hengartner, S. Eidenbenz, and V. Vu, "Bluetooth Worm Propagation: Mobility Pattern Matters!," Proc. Second ACM Symp. Information, Computer and Comm. Security (ASIACCS), 2007.
[26] A. Bose and K. Shin, "On Mobile Viruses Exploiting Messaging and Bluetooth Services," Proc. SecureComm and Workshop, 2006.
[27] S. Cheng, W. Ao, P. Chen, and K. Chen, "On Modeling Malware Propagation in Generalized Social Networks," IEEE Comm. Letters, vol. 15, no. 1, pp. 25-27, Jan. 2011.
[28] Y. Li, P. Hui, L. Su, D. Jin, and L. Zeng, "An Optimal Distributed Malware Defense System for Mobile Networks with Heterogeneous Devices," Proc. IEEE Eighth Ann. Comm. Soc. Conf. Sensor, Mesh and Ad Hoc Comm. and Networks (SECON), 2011.
[29] A. Vahdat and D. Becker, "Epidemic Routing for Partially- Connected Ad Hoc Networks," technical report, Duke Univ., 2002.
[30] J. Burgess, B. Gallagher, D. Jensen, and B. Levine, "MaxProp: Routing for Vehicle-Based Disruption-Tolerant Networks,"
[31] V. Erramilli, M. Crovella, A. Chaintreau, and C. Diot, "Delegation Forwarding," Proc. ACM MobiHoc, 2008..
[32] E. Daly and M. Haahr, "Social Network Analysis for Information Flow in Disconnected Delay-Tolerant MANETs," IEEE Trans.Mobile Computing, vol. 8, no. 5, pp. 606-621, May 2009.