Mobile Contagion: Simulation of Infection & Defense

Everett Anderson *Computer Science*
Kevin Eustice *Computer Science*
Shane Markstrum *Computer Science*
Mark Hansen *Statistics*
Peter Reiher *Computer Science*
University of California, Los Angeles

Abstract

For worms with known signatures, properly configured firewalls can prevent infection of a network from the outside. However, as several recent worms have shown, portable computers provide worms with an entry point into such networks, since these computers are connected behind the firewall. Once inside, the firewall provides no protection against the worm's further spread. Wireless networks are particularly dangerous in this regard, as the act of connection is often invisible, and improperly configured wireless networks will allow anyone within radio range to connect. In this paper, we use real data on a large-scale wireless deployment to analyze the speed with which a worm could spread if it used only this propagation vector. We discuss several possible solutions and provide analysis on how much protection those solutions would provide.

1. Introduction

The worms that achieved the largest infected populations have been designed to spread across arbitrary networks. In most cases, they travel from one wired node through the Internet to another wired node. Along the way, they are likely to either pass through or be rejected by a firewall that separates the target node from the rest of the Internet. Since worms typically make use of known vulnerabilities, rather than zero-day exploits, one method of controlling their spread is to install filtering rules in firewalls that look for either particular worm payloads or attempts to exploit known vulnerabilities. Substantial research is ongoing in developing methods to automatically detect and analyze worm behavior, with a goal of spreading signature information to firewalls, which can then stop further infections by filtering on the signature.

The Blaster and Sasser worm events, however, demonstrated that this approach is not sufficient. Sites whose firewalls were fully cognizant of the infection vector and well prepared to reject any such messages still found their internal networks infected by these worms.

Why? In general, not because of imperfections in the firewalls, but rather because mobile computing allows nodes to move from network to network without traversing any firewall. A mobile computer is plugged into an already infected or unprotected network, is infected with the worm, and is then disconnected. Later it is brought into a protected office environment, where it connects to the network behind the firewall. Its packets are never examined for the worm, since they don't go through the firewall, and it is thus free to infect the entire network. Many sites had serious problems with these worms for precisely this reason.

Further, many of the proposed solutions to detecting and handling worms are based on the presumption that the worm is both aggressive and noisy. By acting in characteristically different ways than the infected node would ordinarily behave, in its attempt to infect as many nodes as possible, as quickly as possible, the worm offers defenders an opportunity to detect its presence and take responsive measures. Some have postulated that if effective defenses of this type are built, they could be stymied by a worm that was slower, more careful, and less aggressive [1]. Propagating only on human movement is one simple way to achieve that effect.

These problems are unlikely to go away. In fact, they will get worse. With wired networks, there is an explicit step that a user takes to connect up to the protected network: he plugs in a cable. Perhaps users can be educated to regard this as an event with security significance, requiring them to perform extra actions, such as a virus scan.

With wireless networks, there is no such user-initiated event. Typically, wireless nodes automatically detect and join local wireless networks. The user does not necessarily even know it has happened. Wireless networks are becoming increasingly common, and wireless advocates are working diligently towards a world with nearly ubiquitous coverage and transparent mobility from network to network. In addition to the popularity of wireless PDAs and laptop computers, cell phones powerful enough to download and run Java code are widespread [2], increasing their utility and offering users, as well as malware authors, many new possibilities.

Existing research has modeled the spread of worms that rely on the standard method of using some IP transport protocol to move across arbitrary networks, but it has not yet provided any insight on how fast and how far a worm could spread by making use of mobile computers and wireless networks. This paper addresses that question. We use existing data of real users working in a campus-wide wireless environment over the course of several months to provide realistic data on mobility and connectivity patterns. We perform simulations based on this data to observe how a worm might propagate using only local wireless connections and human user mobility. Further, since various groups are already building defenses against this form of infection based on examining and quarantining dangerous machines as they enter networks behind the firewall, we analyze several strategies of this kind of defense.

2. Worm Models

As detailed in [1] and [3], an Internet worm such as Code Red can be modeled by applying the Susceptible-Infected (SI) epidemic formula, deriving an equation for the number of infected users over time based on an average contact rate. Moore et al went on to define this rate for Internet worms as the product of scanning frequency and the probability of finding a vulnerable host in the search space (2^{32} when scanning the entire IP address range), but a worm written specifically with behavior for wireless environments would take a different approach to infecting other hosts.

The intensive scanning performed by normal Internet worms would quickly give away their presence on a bandwidth-constrained device. In order to avoid detection, a worm on a mobile host would try to limit scanning of the wide area network and focus on peers with the assumption that eventually those peers would travel behind security boundaries and allow it to reach more hosts. If the worm has multiple attack vectors, it might switch to a more aggressive one suitable for a wired environment once it found an unprotected or particularly well-connected point.

For our theoretical worm, the average contact rate becomes a function of variables such as the number of access points users visit, their session lengths, the number of users per access point or subnet, and whether the worm can monitor the channel to discover new targets. Many of these factors lead back to mobility, and users currently seem to use the network in highly variable ways – large differences may exist between users or even in the same user's behavior over time. As wireless networks become more widely used and reliable models of mobility are discovered, a formula for the contact rate of our hypothetical worm should emerge. Work has already begun on extracting mobility models from traces gathered by Dartmouth College. [4]

In our work, we use a simple model of worm activity. The worm only propagates across wireless networks. Whenever a worm successfully associates with a new subnet, it attempts to infect any other nodes already associated with that subnet. The access points themselves cannot be infected. In our analysis, we assume that all other devices in the simulation are susceptible to the exploit the worm uses to attack new machines, and that once a machine is infected, it continues to try to spread the infection until the machine is disinfected and patched.

In our simulations, we always assume that only a single user in the system starts off with the infection, received either by contact with a hypothetical host outside the scope of the simulation or through some other attack vector. This assumption may not be realistic, as it would be likely that a successful worm might enter a population of mobile wireless users through several hosts. We plan to examine the effects of multiple infection points in future experiments.

Similarly, there are many other variants of worm behavior that could be studied this way, including a worm that sometimes propagates stealthily across wireless networks and sometimes propagates rapidly when well connected. A comprehensive analysis of these variants is beyond the scope of this paper, but several will be addressed in future work.

3. Dartmouth Trace Description and Interpretation

Performing our simulations required a source of data describing how users move between different access points. Rather than use a synthetic model of user mobility, we relied on extensive traces of real behavior.

Kotz and Essien presented a detailed study of traces for Dartmouth College's 802.11b campus network for the Fall 2001 semester in [5], and have continued to collect and make available large amounts of data online. The trace archive now contains full system logs from early in 2001 through the present with few interruptions, making it the largest public source available for studying real user behavior. These logs contain records of many events of interest in the wireless networks they studied, some of which are precisely the data we needed for our simulations.

The Dartmouth campus environment is especially useful for user mobility research since it spans 200 acres and over 160 buildings, and is saturated with 802.11b connectivity throughout. All students are required to have computers, and 70% of students purchasing computers from the campus store in 2001 bought laptops, all of which had 802.11b adapters.

3.1. Dartmouth System Logs

The system logs available from the Dartmouth archive were altered by Kotz et al to add timestamps, sanitize client MAC addresses, and change the names of access points to a format that identifies the class of building (academic, residential, etc).

We interpreted the entries closely following the methods in [5], including adjusting deauthentications due to user inactivity and removing users when access points rebooted.

3.2. Filtering

The system logs of the trace, covering September 23, 2003 through December 10, 2003 (Fall 2003 semester), were filtered as follows. We discarded entries with corrupted data, those with MAC addresses not in the trace format, and messages that do not represent one of the above user events.

Since the logs do not provide information about the organization of the network, we chose to form subnets out of the set of access points in each building. Network cards can freely switch access points if they find a stronger signal, but reassociating with an AP out of the subnet would interrupt the user's experience. Grouping our access points this way makes an association with a new subnet more closely correspond to a change in location. Using building scope is also practical since the outer walls would affect signal strength.

While [5] performed a more sophisticated session analysis, we assume that as long as the user stays within one subnet, the session continues until a disassociate or deauthenticate message appears from the AP with which the user was last connected. Thus, the user's machine remains addressable at the same network location and any connections with other computers continue even if the network card reassociates with a stronger AP signal within the building.

Our resulting data set had 6630 unique MACs which we assume correspond to unique users, 165 subnets (buildings), 3,746,005 entries, 2,590,365 unique timestamps at the granularity of seconds, 1,164,458 Associates, 1,753,640 Reassociates, 380,159 Deauthenticates, 446,501 Disassociates, and 175 Reboots.

3.3. General Analysis

Since user mobility and sharing of access points are the main drivers behind the spread of our envisioned worm, it is helpful to look at related features in the trace.

Our median user visited just over 9 buildings, had a session length of 16.21 minutes, initiated 82 sessions during the simulation, and shared a subnet with 892 other unique users. Time spent online varied widely, with the median user active 582 hours, and including 14% who used the network less than 10 hours.

The subnets in the trace had a median of 17 maximum simultaneous users, received a median of 7,073 visits, and saw a median of 228 unique users. These numbers provide hints as to how likely an infected user would be able to come in contact with others, and thus contribute to forming the average contact rate. Interestingly, even when only simulating at the access point level so that each AP was its own subnet and movements to other APs broke connections, the infection results were equally severe.

Obviously, this trace and our use of it represent only one model of mobility and access point sharing. However, the trace is real, fairly recent, and representative of a reasonably large class of users whose mobile computing experience is largely limited to one large area. These experiments could obviously be run with other mobility models based on different assumptions or traces, likely yielding different results. Such investigations must be deferred to future work, however.

4. Worm Simulation

After filtering the trace, we ran a custom simulator written in Java to determine how worms would behave in this environment. For each of the 6630 users in the trace, we performed a simulation of what would happen if that user began the semester infected with a idealized worm that required 30 seconds to scan and infect the other machines on the subnet, and was capable of infecting any uninfected target, but not the APs. Even if the worm couldn't switch its host's network adapter to promiscuous mode to monitor new connections, since the maximum number of simultaneous users at any subnet in the simulation was 334 (at a library), the subnets would probably be small enough to scan quickly for susceptible hosts.

Figure 1 shows the resulting maximum, median, and lower quartile curves of the number of infected hosts over time. Please note that the x-axis has been normalized such that 0 is the first appearance of the initial infecting user. This allows us to compare infection rates across initial infectors.

The normalized curves show that the idealized worm exhibits similar infection behavior to the SI model, with a rapid jump in victims followed by a smoothing out as the worm finds fewer and fewer new targets. [1] discussed the reaction time of a containment system necessary to limit an infection to a given proportion of users. For the median curve of Figure 1, such a system would need to react within 1 hour and 50 minutes to contain the infection to 10% of the population and within 34 hours to contain it within 50%.

This is considerably better news for network administrators than the "Warhol worm" described in [3] that could infect all vulnerable hosts on the Internet in just minutes, and reflects the contagion approach of our hypothetical wireless worm.

Since mobility patterns and different patterns of use of the network clearly affect how likely this form of worm is to spread widely, we also present data on how many users would ultimately be infected given different users who were the original point of the infection. We found that 86 of the 6630 users in the semester were poor infectors, incapable of infecting any other systems due to low and/or short contact with peers. For the following results, we removed the simulations where one of these users was chosen as the initial infector.

As Figure 2 shows, even with just a single initial infector, most users are capable of causing near total contamination of the network. 83% of the users were able to infect over 90% of their peers. Except for the 86 users who had almost no contact with anyone on the network, almost all of the users were able to infect two thirds of the overall population. Clearly, while this method of spreading a worm is not incredibly fast, it can be extremely effective.

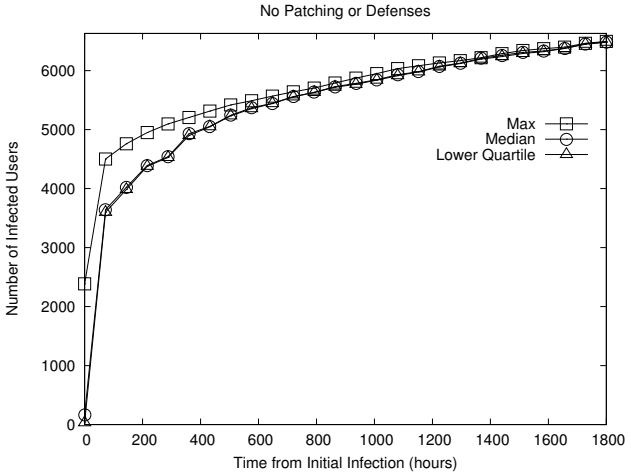

Figure 1. Plot of max, median, and lower quartile summaries of infection over time when there are no defense mechanisms.

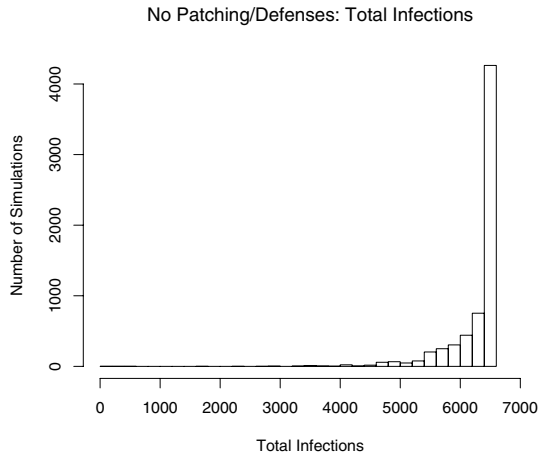

Figure 2. Histogram of total number of users infected across simulations of the 6544 initial infectors.

5. Defense Systems

How can a worm that uses this type of propagation be stopped? We modeled a variety of types of defense systems to see how well they contain the threat, performing tens of thousands of simulations per type. For all simulations, we assume that users run virus-scanning software and disinfect before installing patches.

One important note is that due to the fact that we are replaying recorded trace log data in hopes of having an accurate user mobility model, we are ignoring the possibility that the users would change their network usage when infected or when interacting with one of the defense systems. For instance, if a defense that denied access to infected users were only partially deployed, infected users might decide to walk to a location that didn't have it after being denied access at locations that do.

We simulated random 25% deployments of the defense models across the subnets, as well as the combination of users patching their own systems with the infrastructure solutions. We evaluate the results based on four metrics: percent of simulations where the worm never infected more than 1% of the users, the number of total infections that occurred, the number of infected users who remain infected by the end of the semester, and the general shape of the infection over time plots.

The following are descriptions of the models and their individual results, with comparison and analysis in section 6. Figures 3, 4, and 5 compare data for the several models.

5.1. Unpatched User Shunning (US)

These systems are capable of detecting that a user is vulnerable and send him a warning message that if he doesn't install the patch within a certain period of time, he will be shunned from using the network, presumably until

he fixes his system and reports to an administrator to request removal from the shun list. This model allows infections on its subnet and communication between its users. We used the following parameters: The users were given 5 days to patch, 95% would patch some time within that period, and all the users who became shunned would patch within 3 days. We also assumed that the shun list is global – all the installations of this system are coordinated by a campus-wide administrator.

For 25% deployment, in about 2.5% of the simulations, the US defense model contained the infection to less than 1% of the users. In the median case, it still allowed a total 89% of the users to become infected. Figure 4 shows that it flattens the infected population, keeping up the rate of disinfection with the worm, until eventually all the users who are going to travel to one of the US systems do so and the infectable population is exhausted.

5.2. Infected/Unpatched User Shunning (IUS)

This model is the same is the Unpatched User Shunning case except it also prevents infections attempted at defense locations, warns infected users, and then adds them to the global shun list. An IUS system might actively analyze traffic looking for signatures or anomalies and be able to cut off an infected system before it contaminates others. We assumed that all users warned that they are infected will disinfect and patch within 24 hours.

IUS performed much better in containing the threat to less than 1% of the users, doing so in almost 20% of the simulations. It allowed 65% of users to become infected in the median case, and flattened the infected population to about 25%.

5.3. Active Disinfect/Patch (Active)

In an active defense model similar to [6], users who connect to the network are quarantined until they prove that they are up to date with the latest patches and are uninfected. Infected users are disinfected and unpatched users are provided the patch. In our simulations, we assumed 10 seconds are required to disinfect and patch an infected user. Infected users who leave the network before the time is up are not disinfected. Thus, in most cases an infected machine will be disinfected if it ever visits a location where the defense is deployed.

Active reduced the infected population by the end of the simulation to about the same level as IUS; however it was able to contain the threat to less than 1% of the population in almost 27% of the cases and reduced the number of users ever infected to 51% in the median case. This suggests that it did its work faster and was more likely to prevent the outbreak in the first place. Nonetheless, even a strong and aggressive defense of this kind can frequently permit a high degree of infection if its deployment is limited to merely 25% of all systems. Clearly, higher degrees of deployment will be required to achieve greater protection, rather than merely more sophisticated defenses at a limited number of locations.

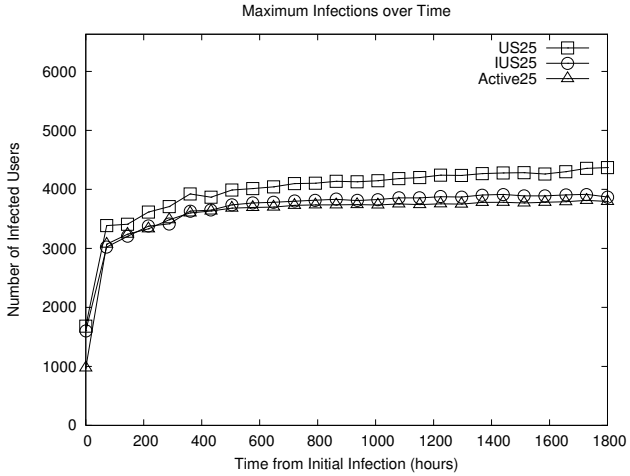

Figure 3. Maximum infection over time curves for US, IUS, and Active at 25% deployment.

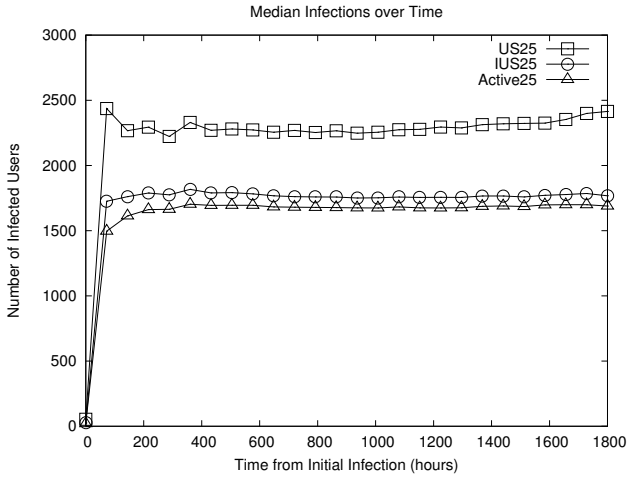

Figure 4. Median infection over time curves for US, IUS, and Active at 25% deployment.

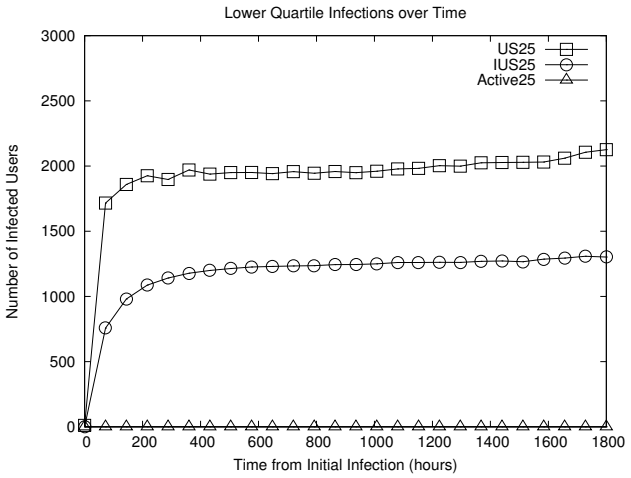

Figure 5. Lower quartile infection over time curves for US, IUS, and Active at 25% deployment.

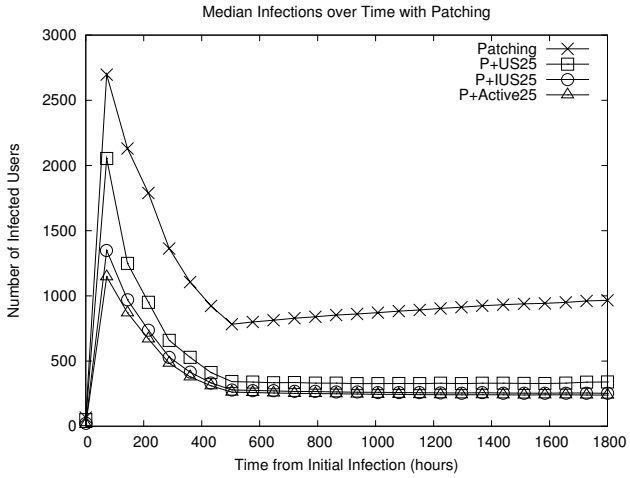

Figure 6. Median infection over time plots for the combination of Proactive Patching and the three defense systems at 25% deployment.

5.4. Proactive Patching

Since most popular consumer operating systems now support automatic warnings of new vulnerabilities and make it easy to install patches, we expect more users will keep their systems up to date. To examine the impact of this effect, we used the following schedule for patch installation starting at the beginning of the trace data: 50% of users patch some time within the first 7 days, 25% in the next 7 days, 10% in the third 7, and finally 15% will never patch on their own. Users were allowed to patch even if they were not on the wireless network, since we generally assume that they have other sources for network connectivity.

If users patch their own systems according to the schedule we devised, they can have a significant effect. However, the worm is still fast enough to infect a large percentage of the population. In less than 1% of the simulations, the worm was contained to 1% or less of the users, and in the median case, 63% of the users were infected, though only 15% were still infected by the end of the simulation due to our schedule. One of the curves on Figure 6 shows the effect of relying only on patching (with this pattern) to combat a worm.

5.5. Combining Proactive Patching and Infrastructure Defense

While the combination of proactive patching with the defense systems had significantly better results than any single system alone, the relative performance of US, IUS, and Active remained about the same. Due to space constraints, we will only show the infections over time graph of the median cases, and present a complete summary in section 6. Note that even the best combined defense still left around 250 users infected in the median case.

6. Comparison and Analysis

Tables 1, 2, and 3 summarize results for three of our metrics. The two most important results are for the percentage of simulations for which the defense contained the infection to less than 1% of the population and the total number of users infected. With a random 25% deployment, the best solution across all our metrics is the combination of proactive patching plus an active network defense system.

Table 1

Percentage of simulations in which the defense strategy contained the threat to 1%, 25%, and 50% of the total users.

Containment	1%	25%	50%
No Protection	0.02%	0.05%	0.17%
US25	2.56%	4.99%	21.07%
IUS25	19.67%	24.35%	39.07%
Active25	26.68%	31.92%	49.35%
Patching	0.93%	9.36%	20.19%
P + US25	3.54%	16.18%	26.80%
P + IUS25	20.56%	32.21%	56.45%
P + Active25	28.28%	41.47%	96.10%

Table 2

Summaries of the total number of users infected during each type of simulation.

	Max	3rd Qu.	Median	1st Qu.	Min
No Protect.	6541	6538	6506	6282	2
US25	6487	6335	5912	3644	1
IUS25	6295	5380	4292	1736	1
Active25	6034	4430	3353	1	1
Patching	4390	4241	4188	3722	1
P + US25	4263	4106	4016	3094	1
P + IUS25	4099	3528	3202	574	1
P + Active25	3841	2852	2230	1	1

Table 3

Summaries of the number of infected users who remained infected at the end of the simulation.

	Max	3rd Qu.	Median	1st Qu.	Min
US25	4384	2622	2303	1980	0
IUS25	3881	2032	1643	1113	0
Active25	4008	1992	1567	0	0
Patching	1098	989	968	946	0
P + US25	708	385	337	288	0
P + IUS25	589	306	246	165	0
P + Active25	599	298	233	0	0

7. Future Work

It remains to be seen whether results from the Dartmouth traces are applicable to other large wireless networks, such as those at other universities or those from commercial providers such as T-Mobile. More traces must be gathered and analyzed to develop credible mobility models. Alternately, mobility models developed from traces that are not directly available could be used to drive the simulation, instead of actual traces. In addition, as the number of wireless access points increases and devices that use network resources become more ubiquitous, early models of user behavior may become irrelevant. Therefore, work on determining how changing models of user mobility and behavior affect worm propagation should continue.

The results shown here obviously suggest that a higher degree of defense employment than the 25% tested here is needed to really stop spread of this kind of worm. We must perform further experiments to determine the degree of deployment required to stop worms reliably for various forms of defense.

One approach to getting more value from the defense systems is to strategically place them on subnets based on heuristics such as the maximum number of simultaneous users, number of unique users seen, and number of visits. Initial results suggest that a good heuristic can improve defense by over 50% for the same percentage of deployment.

For a given wireless network trace, more work can be done to identify what aspects of user mobility are the greatest contributors to the contact rate parameter of the SI epidemic model, thus yielding a contact rate formula for the network. Another approach might be to look at the "infection potential" of users based on aspects of their mobility or their system configurations, and explore the ways that an attacker could most efficiently achieve a large infection base in the shortest amount of time through releasing the worm at targeted locations or hosts. A larger experiment could be conducted to take random users at random times as the initiators of the infection, allowing multiple infectors. This might provide a stronger guarantee of the effectiveness of the defense systems.

As mentioned earlier, studies of worms that combine different modes of infection depending on circumstances are needed. A simple example is a worm that spreads at high speed to random addresses when it has infected a site with a high speed wired link, but spreads more slowly when connected only to a wireless network. Another example is making use of two entirely different types of wireless connection, such as Bluetooth and 802.11, to spread. Other varying patterns are also possible and worthy of study.

Finally, the simulations run here generated a vast amount of data, only some of which was presented in this paper. Closer examination of this data might well reveal other interesting phenomena, such as the relationship of worm infection rates to real world events.

8. Related Work

After the seminal paper on worm models [3], others have looked at worm modeling. [7] describes their general-purpose simulator for worm propagation. It is unclear whether their model would capably support a wireless environment simulation. [8] uses measurements to show that active worms seem to propagate slightly slower than standard models indicates, but have a very similar, if more limited, overall behavior. [9] discusses the spread of the Slammer worm. They do not provide a model, but show that more damaging worms can similarly propagate quickly throughout the Internet. [10] uses a somewhat different quarantine model to study protection against standard worms in a wired environment.

In addition to Kotz et al, others have analyzed different networks for user mobility and performance, and have made their own trace data available from the Dartmouth trace archive. Tang and Baker have examined several levels of networks, from a single wireless LAN [11] to a seven-week trace of a large metropolitan Ricochet-based network called Metricom with nearly 25,000 users. [12]. Balazinska and Castro analyzed SNMP records from a corporate LAN in [13], and Balachandran, Voelker, Bahl, and Rangan looked at traces recorded at the ACM

SIGCOMM'01 conference [14]. These studies could be used to examine the worm discussed here in other environments.

Inspired by recent worm incidents, a number of parties have examined systems that quarantine and sometimes try to disinfect mobile computers brought into a network behind the normal firewall. The best known is Cisco's Network Admission Control [15]. InfoExpress, ZoneLabs, Sygate, Symantec, Network Associates, and other companies and research groups offer products with functionality of this kind, as well, differing in ways not relevant to this paper [16].

9. Conclusion

The increasing penetration of wireless networks and mobile computing are likely to make these technologies attractive to those who write worms. As shown by previous worms, mobility offers a back door for entry into otherwise protected networks, and wireless networks exacerbate the problem. To understand how to combat this propagation vector, we must first understand the characteristics of worms that use it. This paper is a first step towards such an understanding.

Our results, based on simulations using real trace data gathered at Dartmouth College from wireless networks supporting mobile computers, suggest that a worm using this attack vector would have a characteristic infection curve similar in shape to that of other worms, but would spread much more slowly However, given sufficient time, even a single infected user is highly likely to infect virtually the entire susceptible population.

The combination of users proactively patching their machines plus an active network defense system that prevents infection, disinfects the infected, and patches the unpatched can significantly reduce the reach of the worm we modeled. But with defense deployment limited to 25% of susceptible systems, worms can still frequently achieve high infection rates.

All of these results are based, of course, on a single set of mobility data. More study is required to generalize these results to the wider community, but the initial results are promising. Dartmouth's situation matches many other institutions that are installing widespread wireless networking, and their users are likely to behave in similar ways. Generally, more study is needed on the likely infection rates and patterns of worms using different infection strategies, and on the effects of deploying varying kinds of defenses to stop them.

10. References

[1] D. Moore, C. Shannon, G. M. Voelker, and S. Savage, "Internet Quarantine: Requirements for Containing Self-Propagating Code", *INFOCOM 2003*.

[2] Sun Microsystems, "Sun Strengthens Lead in Worldwide Mobile Data Services with Java," http://www.sun.com/smi/Press /sunflash/2003-10/sunflash.20031014.4.html, October 14, 2003.

[3] S. Staniford, V. Paxson, and N. Weaver, "How to 0wn the Internet in Your Spare Time," *Proceedings of the 11th USENIX Security Symposium*, August 2002.

[4] R. Jain, A. Shivaprasad, X. He, and D. Lelescu, "Towards an Empirical Model of User Mobility and Registration Patterns," poster, *Proceedings of MobiHoc 2004*.

[5] D. Kotz and K. Essien, "Analysis of a Campus-Wide Wireless Network, *Proceedings of MobiCom 2002*.

[6] K. Eustice, L. Kleinrock, S. Markstrum, G. Popek, V. Ramakrishna, P. Reiher, "Securing Wi-Fi Nomads: The Case for Quarantine, Examination, and Decontamination," *Proceedings of the New Security Paradigms Workshop (NSPW) 2003*.

[7] A. Wagner, T. Dubendorfer, B. Plattner, and R. Hiestand, "Experiences with Worm Propagation Simulations," *10th ACM CCS Workshop on Rapid Malcode (WORM '03)*, 2003.

[8] Z. Chen, L. Gao, and K. Kwiat, "Modeling the Spread of Active Worms," *INFOCOM 2003*.

[9] D. Moore, V. Paxson, S. Savage, C. Shannon, S. Staniford, and N. Weaver, "Inside the Slammer Worm," *IEEE Magazine of Security and Privacy*, August 2003.

[10] C. C. Zou, W. Gong, and D. Towsley, "Worm Propagation Modeling and Analysis under Dynamic Quarantine Defense," *10th ACM CCS Workshop on Rapid Malcode (WORM '03),* 2003.

[11] D. Tang and M. Baker, "Analysis of a Local-Area Wireless Network," *Proceedings of MobiCom 2000.*

[12] D. Tang and M. Baker, "Analysis of a Metropolitan-Area Wireless Network," *Proceedings of MobiCom 1999*.

[13] M. Balazinska, P. Castro, "Characterizing Mobility and Network Usage in a Corporate Wireless Local-Area Network," *Proceedings of MobiSys 2003*.

[14] A. Balachandran, G. M. Voelker, P. Bahl, and P. V. Rangan, "Characterizing User Behavior and Network Performance in a Public Wireless LAN," *SIGMETRICS 2002*.

[15] Cisco, "Cisco NAC: The Development of the Self-Defending Network," http://www.cisco.com/warp/public /cc/so/neso/sqso/csdni_wp.htm.

[16] J. Vijayan, "Extended Enforcement," *Computer World*, May 10, 2004.