An Analysis of the Internet Parasite: A Biological Analog in the Digital World

Mohamed Faacy Farook, McNair Scholar, Penn State University

Faculty Research Advisor
Dr. Patrick McDaniel
Assistant Professor Computer Science and Engineering
College of Engineering
Penn State University

Graduate Assistant
Kevin Butler
Ph.D. Candidate, Computer Science and Engineering
Penn State University

Abstract

The Internet is rife with malware: worms and viruses are rampant. However, a new form of malware, the Internet parasite, could have an even more devastating effect on host systems worldwide. Like their biological counterparts, Internet parasites evolve through mutation and create new attack vectors, propagating silently through their victims. In this study, we attempt to simulate parasitic behavior by extending our "parasim" simulator to more realistically model parasitic propagation across real-world topologies.

Introduction

Computer worms pose a major threat to internet, and it is a general belief that understanding their means of propagation will help to devise efficient control strategies (Leveille 2003). Kevin Butler and Patrick McDaniel (2005) asserted that worm attacks based on mutation and covert propagation are likely to be ultimately more damaging and long lasting, which was supported by parasitic behavior in natural systems. They proposed a new form of computer worm called the "Internet Parasite." Like its biological counterpart, survival of the parasitic worm depends on mutation. While residing in a machine undetected, it dynamically discovers new invasive techniques and covertly propagates across the network. In most cases, the mutations and the attack vectors will fail, but the few successful ones will result in spectacularly successful growth and spread throughout the network. In order to create counter measures for a Parasitic Worm, it is essential to model and understand its behavior. This study extends the previous study by simulating parasitic behavior in real world network topologies at the sub-network level. It is hypothesized that the propagation of the Internet parasite will be slower across real-world network topologies than in a fully connected network but will still exhibit similar behavior as in a fully connected network.

Methodology

The "parasim" simulator was originally written by Kevin Butler in Java programming language. The simulation parameters were the number of hosts in the network, probability of infection Pi, the probability of inoculation Pn, and the probability of mutation Pm and the number of initial hosts infected. These probability distributions are exponential. The "parasim" simulator assumed a fully meshed topology of 500 hosts, where each host is connected to every other host. Also, the hosts in the network topology are assumed to be homogeneous, in the sense that an infected host is equally like to infect any of the other susceptible hosts. Therefore during the simulation of parasitic worm propagation in a fully meshed network, a particular strain of an infected host attempted to infect any other randomly selected host in the network.

A network topology can be thought as a graph. Graphs are composed of a set of nodes (vertices) connected by a set of edges. Vertices connected to a given node (each through a different edge) are called the 'neighbors' of that node. The number of neighbors of a given node is called its 'degree'. In a graph that represents a network topology, each node represents a host in the network and each edge in between two nodes represents the network connection between the hosts. The degree of the node represents the number of network connections to the particular host under consideration. The simplest possible graph is the fully-connected graph: each node is connected to every other node. It has been argued that fully-connected graphs do not offer a realistic account of computer networks (Kephart & White 1991). Users tend to communicate with a subset of users, not with everyone in the network. Therefore, the pattern of connections is not really fully-connected. (Leveille 2003)

In order to simulate parasitic worm propagation in a real world network topology, the "parasim" simulator had to account for the network characteristics such as the actual heterogeneous connections between hosts, connection bandwidth, direction of network data flow, and hierarchical properties of the network. The real world network topology was simplified with the following constrains:

- Hosts and network characteristics are homogeneous
- An infected host is able to infect only a directly connected host
- Network connections are bi-directional (i.e. the edges in the graph are undirected)

The first step was to generate an Internet network topology and to read it into the "parasim" simulator. Boston University Representative Internet Topology Generator, (BRITE) written by Alberto Medina, Anukool Lakhina, Ibrahim Matta, and John Byers, was chosen as it is was able to generate Power law topology in which our study is interested. Furthermore it is available in java source code format which is the language in which the "parasim" simulator is written.

BRITE supports generation of two-level hierarchical topologies and Top-down is one such approach. In a Top-down hierarchical topology, BRITE generates first an Autonomous Systems (AS)-level topology and then for each node in the Autonomous Systems (AS)-level topology BRITE generates a router-level topology. (Medina et al 2001)

Figure 1 shows the generated topology. A two level top-down hierarchical topology was generated using the Waxman model with default parameters. The topology generated contains 20 Autonomous Systems (AS) level nodes each with 25 router level nodes connected. Thus the network consists of a total of 500 hosts in total. A Java program was written to convert the BRITE output file format (.brite) to an edge file format (.edg) which can be inputted into the "parasim" simulator.

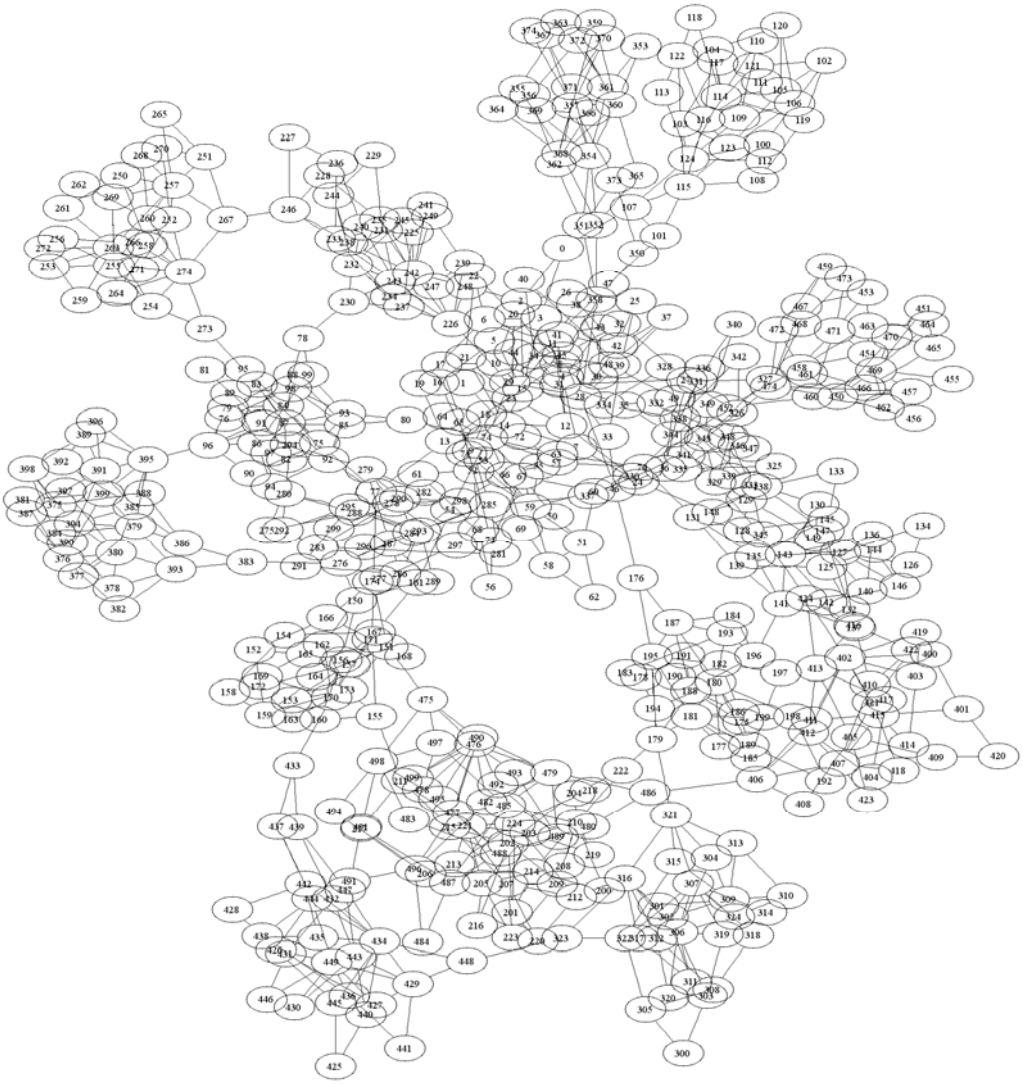


Figure 1: The graph of the Internet topology generated by BRITE simulator with 20 Autonomous Systems (AS) level nodes each with 25 router level nodes connected to give a total of 500 hosts in the network.

The "parasim" simulator was modified to read in a network topology in an edge (.edg) file format where each host in the network is given a unique identifier and the connections between hosts are output as a pair of unique host identifiers. The host connections are assumed to be undirected, so that any connected host can send and

receive data across the network and hence it is equally susceptible for infection by an Internet parasite. The topology was read into the "parasim" simulator and was stored as an internal data structure consisting of an array of an adjacency lists. Then the simulator was modified to attempt infection on a randomly selected directly connected host.

Also, a further modification to the simulator was the selection of the hosts for initial infections. In the original "parasim" simulator which models a fully meshed network the first 'a0' (simulation input parameter for number of initially infected hosts) hosts were selected. In a fully meshed network topology any host can be selected as initially infected because the network connections are identical respect to any other host. But in a hierarchical topology the network connections may differ for each host. Therefore the selection of the hosts can influence the spread of the parasite depending on the number of network connections of that particular host. A more realistic approximation to the initial infections was to model clustered infections. First, a host was chosen at random and then infected. Then all the directly connected hosts of the infected hosts were infected. The process is repeated for the newly infected hosts until the total number of infected hosts in the network reaches 'a0'. Then the time counter was started and the simulation was run.

There were additional constraints that needed to be added to the simulator to run with the available memory in the system. The following constraints were added to the simulator:

- the maximum number of parasite mutation strains in the network
- the maximum number of infection attempts for a given time period, and
- the total number of time periods simulated

It was noted that the constraint for the maximum number of infection attempts for a given time period was not a hard limit, i.e. the infections could go above the limit until the current host has completed the infection attempts with all the parasites it was infected with. Addition of these new constraints were necessary for the simulator to run to completion but it also created side effects with the realistic nature of the simulation. For example, when the maximum number of infection attempts for a given time period was reached, the simulator continues infection attempts with the current host until it completes and then goes to the next time period, by-passing the infection attempts of any remaining hosts attempting infection. This may cause a bias by not allowing newly infected hosts to propagate infections properly. To overcome this issue, newly infected hosts in the current time period were given preference to attempt infections in the next time period because they were added to the beginning of the list of hosts to attempt infection for the next time period. Also, the newly infected hosts in the current time period could attempt infections only at beginning of the next time period.

Our study focused on the propagation dynamics at the beginning of the Internet parasite infection for a given network topology. Therefore, code was added to stop the simulation when all the hosts in the network were compromised by the parasite. At this point the output values were copied till the end of the simulation time periods. The consequence of doing this is that the simulator does not simulate the dynamics of the network after the network has been completely compromised. But during the test runs it was observed that the number of infected hosts oscillates because of the infections and inoculations of the hosts.

Results

The simulation was run on both a fully meshed network (FM) and a top down hierarchical network (TP) for a range of infection probabilities, namely 0.035, 0.040, 0.045, 0.050, and 0.055 while all other simulation parameters were held constant. The simulation parameters for the "parasim" simulator were

- number of hosts in the network - 500
- coefficient of infection - 0.035, 0.040, 0.045, 0.050 or 0.055
- coefficient of mutation - 0.05
- coefficient of inoculation - 0.04
- number of hosts initially infected - 25
- type of initial infection - clustered
- maximum number of time periods - 1000
- maximum number of mutation strains in the system - 1,00,000
- maximum infection attempts per round - 2,00,000

Data was collected from 100 trials with 1000 rounds in each trial. The average number of infected hosts were calculated and plotted for comparison. Figure 2 shows the results. For clarity of the figures only 600 time periods are shown. Figures 3, 4, 5, 6 & 7 are drawn for easier comparison of the "parasim" simulation results for a fully meshed network (FM) and a top-down hierarchical network (TP) generated by BRITE simulator for coefficients of infection of 0.035, 0.040, 0.045, 0.050 and 0.055 respectively. Figure 8 shows the percentage of trials in which the parasite was able to compromise the whole network for the range of infection probabilities. Figure 9 tabulates the important results such as the maximum average number of infections, time period for which the maximum values are reached and the percentage of trials where the parasite is able to compromise the whole network.

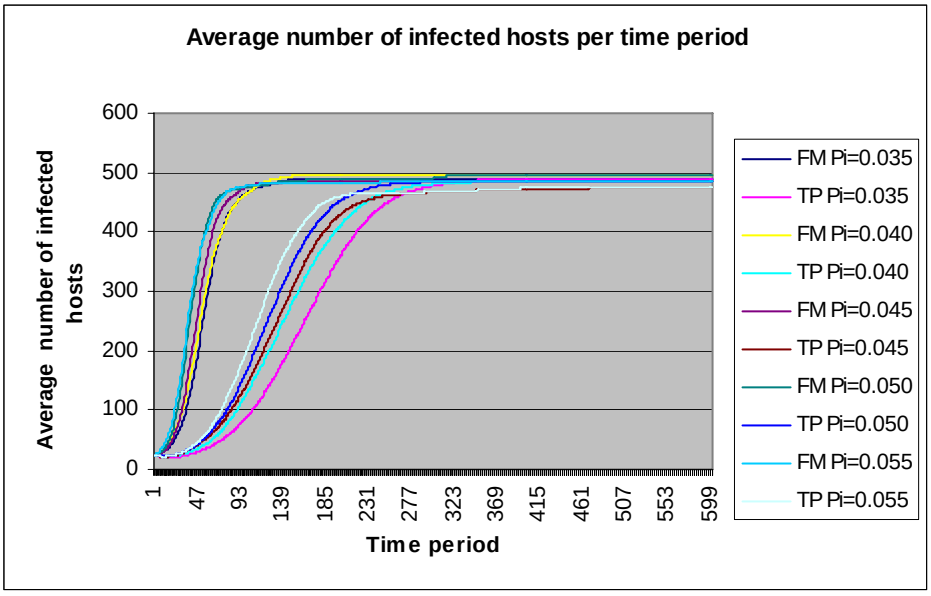


Figure 2: The average number of infected hosts from 100 trials for coefficients of infection 0.035, 0.040, 0.045, 0.50, and 0.055 in fully connected network (FM) and a top down hierarchical network (TP)

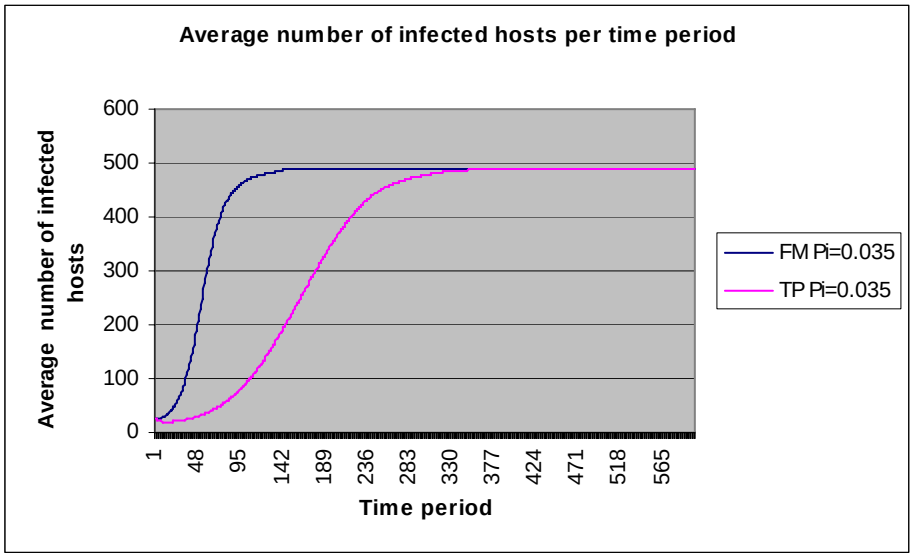


Figure 3: The average number of infected hosts from 100 trials for a coefficient of infection 0.035 in a fully meshed network (FM) and a top-down hierarchical network (TP)

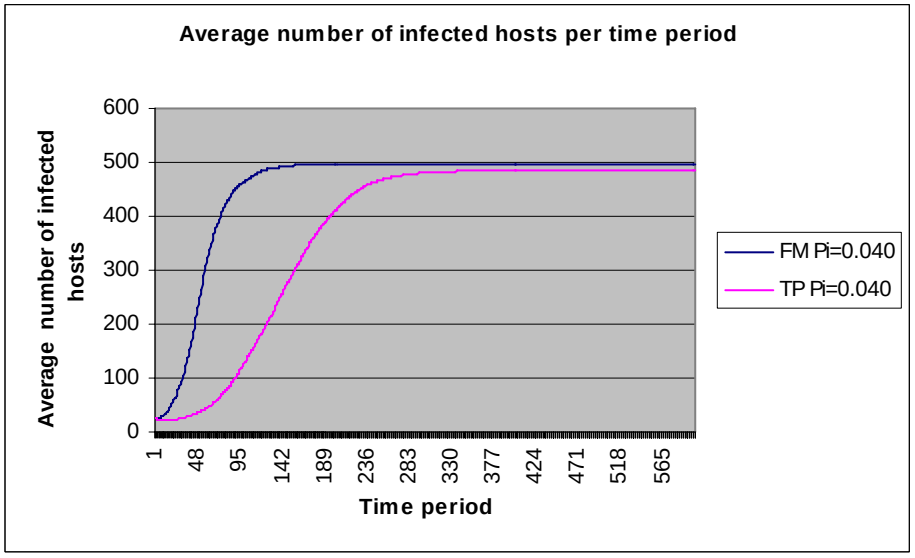
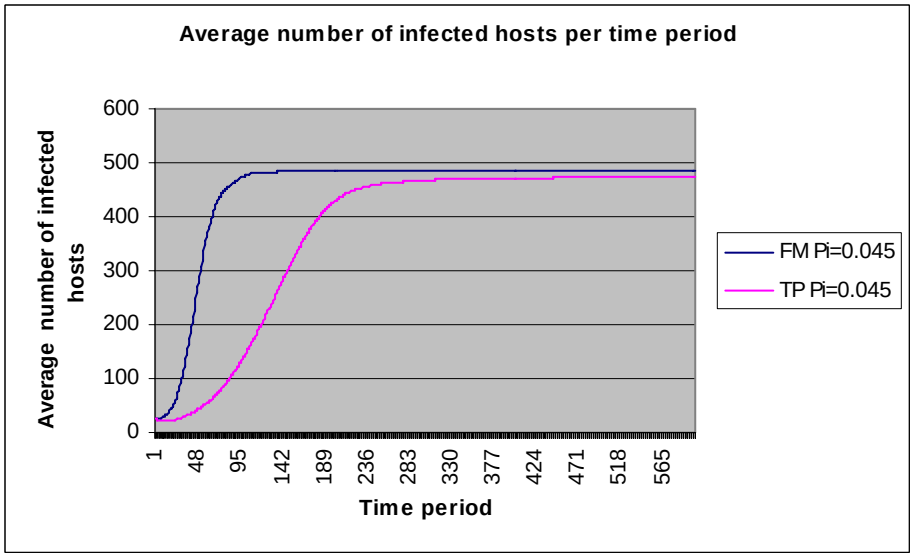
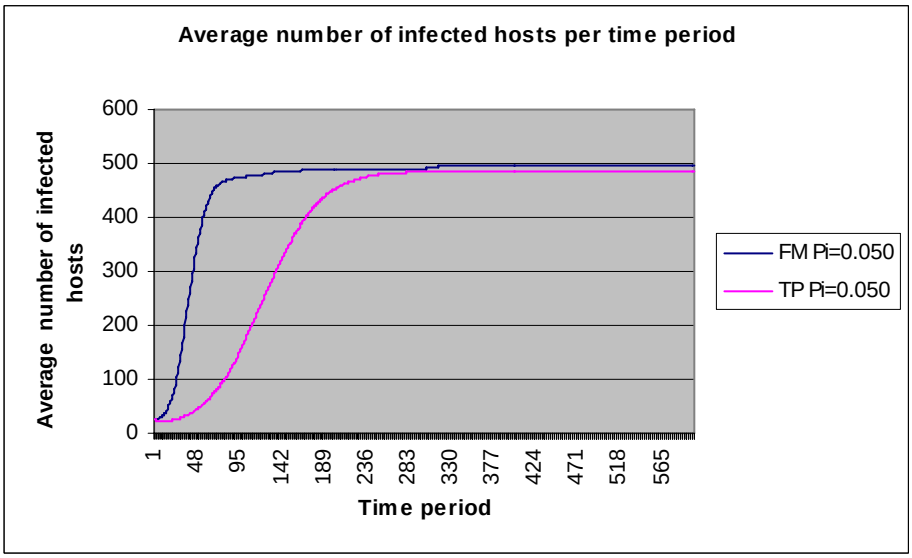
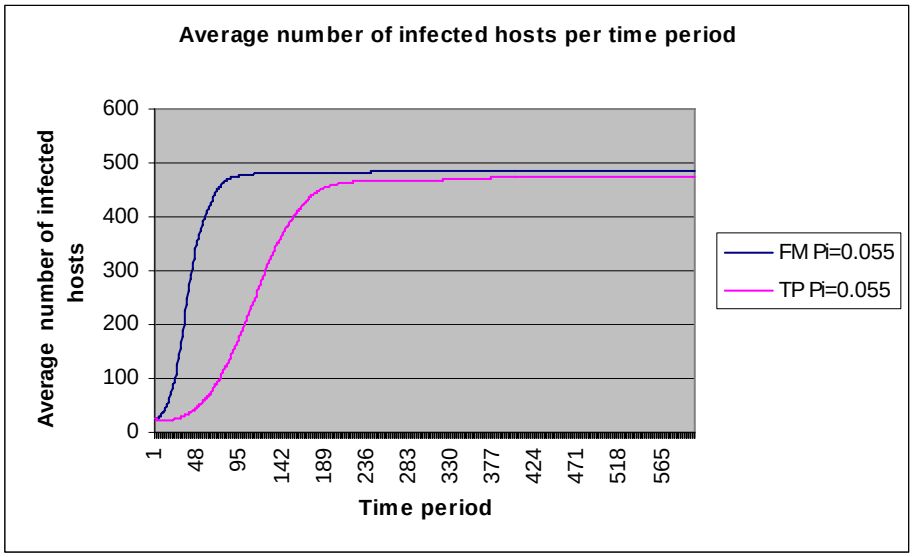


Figure 4: The average number of infected hosts from 100 trials for a coefficient of infection 0.040 in a fully meshed network (FM) and a top down hierarchical network (TP)







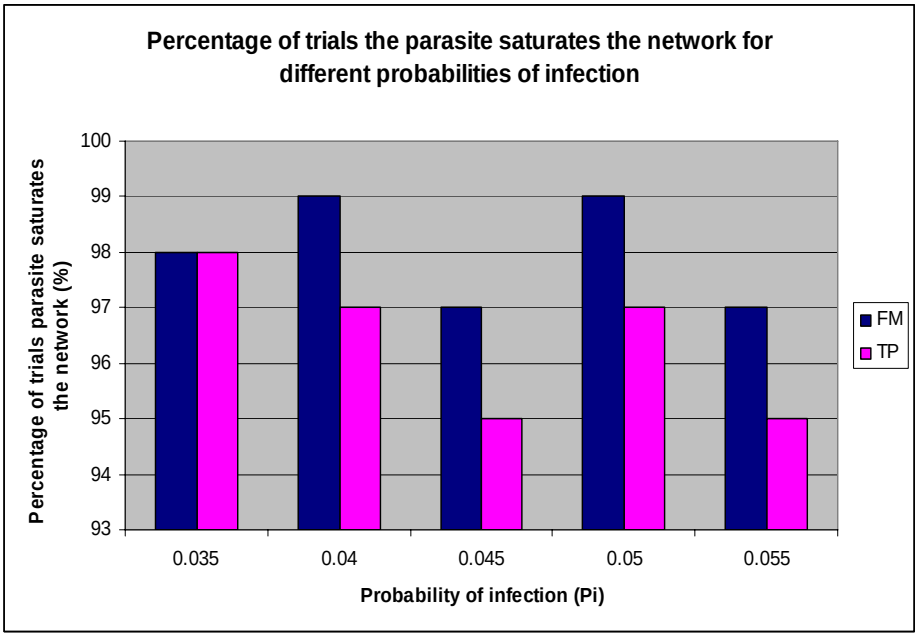


Figure 8: The percentage of trials in which the parasite is able to compromise the whole network for a range of coefficients of infection in a fully meshed network (FM) and a top down hierarchical network (TP)

Important Results					
Coefficient of infection (Pi)	**0.035**	**0.040**	**0.045**	**0.050**	**0.055**
	Time period at which the maximum average number of infections occurs first				
Fully Meshed (FM)	189	172	158	326	261
Top-down (TP)	423	399	533	337	445
	Maximum average number of infected hosts in the network				
Fully Meshed (FM)	490	495	485.02	495	485.01
Top-down (TP)	490	485	475	485	474.74
	Percentage of trials in which the parasite saturates the network				
Fully Meshed (FM)	98	99	97	99	97
Top-down (TP)	98	97	95	97	95

Figure 9: The maximum average number of infections in a given network topology and the time period of first occurrence with the percentage of trials converging to saturation from data collected through 100 trials of the "parasim" simulator for a range of infection probabilities.

We observe (Figure 9) that in each case, the maximum average number of infections for Internet topology (TP) is less or equal to a fully connected topology (FM) and it takes a longer time to reach the maximum number of infections in the Internet topology (TP) than in a fully connected topology (FM). Furthermore, the percentage of trials the parasite is able to compromise the whole network is less for the Internet topology (TP) than in a fully connected topology (FM).

Conclusions and Recommendations

The simulation results show that in majority of cases the network became completely infected with the parasite and in a few others the number of infected hosts converged to zero. The extinction of the parasite is explained intuitively by the fact that a statistical fluctuation might wipe out the parasites before it propagates to sufficient hosts to become established (Kephart & White 1991).

The graphs of average number of infections per time period shows that rate of propagation of a parasite (as given by the gradient/slope of respective graphs) is slower in the simplified Internet topology (TP) as hypothesized. This observation can be explained by the fact that the degree of each node in the TP network is less than that of a FM network. Therefore as the number of hosts connected to a particular host decreases, the number of ways a particular host can be infected by others will be less. Another explanation is that when a particular host at a parent node of a tree is inoculated against a particular parasite strain, then it prevents the parasite from propagating to the child nodes

and beyond. It is also observed that the average rate of propagation for the top down hierarchical topology increases and comes closer to that of the fully meshed topology when the coefficient of infection is increased. However, the Internet parasite is able to ultimately infect more hosts on average in a fully connected network. The latter observation is intuitive because, given the vast number of rapidly infected hosts in a fully-meshed topology and the high number of connected hosts (N-1 connections per host where N is the total number of hosts in the network); the uninfected hosts will be found faster and infected in many ways. These observations support the conjecture that the results obtained for a fully connected network topology by Butler et al (2005), provide an upper bound to the behavior of an Internet parasite.

Further study is necessary to optimize the simulator for efficient memory usage. Also it is necessary to rerun simulations on a variety of different network topologies including network topologies represented as directed graphs, incorporating network characteristics such as link bandwidth, and possibly mathematical analysis.

Acknowledgements

I would like to thank my Faculty Research Adviser, Dr. Patrick McDaniel, and my Research Mentor Kevin Butler for providing the opportunity to work with their research in the Systems and Internet Infrastructure Security (SIIS) laboratory in the Computer Science and Engineering Department at The Pennsylvania State University. Also, I would like to extend my gratitude to The Ronald E. McNair Scholars Program at Penn State.

Appendix – Related Work

Internet Parasite

Kevin Butler and Patrick McDaniel (2005) asserted that worm attacks based on mutation and covert propagation are likely to be ultimately more damaging and long lasting, which was supported by parasitic behavior in natural systems. They proposed a new form of computer worms called the "Internet parasites." Like its biological counterpart, survival of the parasitic worm depended on mutation. While residing in a machine undetected, it dynamically discovered new invasive techniques and secretly propagated across the network. They provided empirical results based on their "parasim" simulator that modeled real propagation of the Internet parasite. The "parasim" simulator modeled a fully meshed network of 500 nodes and the simulation parameters were the probability of infection P_i, the probability of inoculation P_n, and the probability of mutation P_m. They assumed that the probabilities of infection, mutation and inoculation were exponentially distributed and examined literature in epidemiology and parasitology to determine a numerical basis for the values of P_i (=1/0.0056), P_n (=1/0.04) and P_m (=a value randomly selected from the distributions). They found that small changes in the rates of infection, mutation, and inoculation can have dramatic changes on whether a parasite will die out or eventually fully propagate to every host in the network. Their key observation was that in

most cases, the mutations failed, but the few successful ones resulted in spectacularly successful growth and spread throughout the network. They pointed out that for a sufficiently high rate of mutation, even a well-defended network will eventually succumb. They stated further that the effectiveness of a parasite's infection vector and its resistance to host inoculation were major factors in determining whether the network would collapse. They concluded that Internet parasites have the potential to mimic their biological counterparts and spread throughout the virtual world to form an unwelcome relationship with machines and their users. They recommended further detailed analysis and simulation of Internet parasite in real-world network topologies for future work.

Computer Worms – Definitions, Analysis, & Defenses

Weaver, et al, (2003) defined a computer worm as "a program that self-propagates across a network exploiting security or policy flaws in widely-used services." They noted that, the line between worms and viruses was not all that sharp. However, they distinguished between worms and viruses by the fact that viruses cannot self-propagate and require some sort of user action. They described a preliminary classification of computer worms based on worm's target discovery and selection strategies, carrier mechanisms, activation, possible payloads and attackers who would employ a worm. They concluded that the carrier, activation, and payload are independent of each other, and describes the worm itself. They recommended developing more robust defenses by focusing on preventing worms that use one or more of the techniques described by them. They also pointed out the importance of understanding not only the technology used but also the motivations of those that launch the attacks because "worms are ultimately written by humans, and sometimes the easiest way to defend against a worm is to remove the motivation for writing a worm in the first place."

Seely (1998) provided a chronology for the outbreak of the Morris Worm and presented a detailed description of the internals of the worm, based on a C version produced by decompiling. The self-replicating program was released in the Internet in November 2, 1988 which spread across the U.S. in just a few hours, invading VAX and Sun-3 computers running versions of Berkeley UNIX, and used their resources to attack more computers like a chain reaction. He pointed out the importance of analyzing computer worms by stating that "The worm story was on the front page of the New York Times and other newspapers for days. … judging by the response, it has scared us. … but I will say that I think these issues have been ignored for much longer than was safe, and I feel that a better understanding of the crisis just past will help us cope better with the next one. Let's hope we're as lucky next time as we were this time." (Seeley 1988)

According to Staniford et al. (2002), computer worms can be exploited by attackers to rapidly gain control of vast numbers of Internet hosts to pose an immense risk to the overall security of the Internet. They derived the "Random Constant Spread" (RCS) model from empirical data of the spread of Code Red I in July, 2001. Then they discussed, developed and evaluated some possibly strong techniques: hit-list scanning to create a Warhol worm, permutation scanning to enable self-coordinated scanning, and the use of Internet-sized hit-lists to create a flash worm. They proposed the possible threat of a new class of surreptitious worms that spread more slowly but in a much harder to detect

contagion fashion. They also considered robust mechanisms by which attackers can control and update deployed worms, namely direct worm-to-worm communication and programmable updates. They concluded that given the magnitude of Internet-scale threats, it was critical for nations, concerned with cyber-warfare in particular, to attempt to mitigate the immense risk. They recommended creating a "Cyber-Center for Disease Control" (CDC) and identified six key roles of a CDC: identifying outbreaks, rapidly analyzing pathogens, fighting infections, anticipating new vectors, proactively devising detectors for new vectors, and resisting future threats.

Zou et al. (2002) identified two major factors that affect an Internet worm propagation based on the Code Red worm incident. According to them human countermeasures against worm spreading, like cleaning, patching, filtering or even disconnecting computers and networks would remove both susceptible hosts and infectious hosts from circulation and slowing down of worm infection rate due to worm's impact on Internet traffic and infrastructure. Accounting for these factors they presented the Two-factor worm model. They showed that Code Red did not infect almost all susceptible online computers at 19:00 UTC as concluded in by Staniford et al. from the RCS model. Instead, Two-factor worm model showed that Code Red infected roughly 60% of all susceptible online computers at that time. They acknowledged that "However, Internet worm models have their limitations. For example, the two-factor worm model as well as other worm models is only suitable for modeling a continuously spreading worm, or the continuously spreading period of a worm. They can't predict those arbitrary stopping or restarting events of a worm … we can only find such events through manual code analysis."

Zou et al (2003) looked at implementing automatic worm mitigation techniques such as dynamic quarantine on computer networks. Motivated by the methods used in epidemic disease control in real world, they presented a dynamic quarantine method based on the principle "assume guilty before proven innocent." In this method, a host is quarantined whenever its behavior looks suspicious by blocking traffic on its anomaly port and released from the quarantine after a short time, even if the host has not been inspected by security staff. They presented mathematical analysis of three worm propagation models under the dynamic quarantine method which showed that the dynamic quarantine reduced a worm's propagation speed and raised the worm's epidemic threshold which in turn reduced the chance for a worm to spread. Their simulation results verified the analysis and demonstrated the effectiveness of the dynamic quarantine defense.

Zhang et al (2004) presented a worm propagation model that effectively reduced a worm's propagation speed. It was based on the classical epidemic Kermack-Kermack model, and adopted dynamic quarantine strategy, dynamic infection rate and removal rate. Through simulation they verified the effectiveness of their model.

In order to understand how worms propagate and how different scanning strategies affect the dynamics of worm propagation, Zou et al (2006) systematically modeled and analyzed worm propagation under various scanning strategies, such as uniform scan, routing scan, hit-list scan, cooperative scan, local preference scan, sequential scan, divide-and-conquer scan, and target scan. They showed the underlying similarity and relationship between different worm scanning strategies. They provided an analytical

model for the Witty worm's destructive behavior and based on the simulation and analysis of Blaster worm propagation, they provided a guideline for building a better worm monitoring infrastructure.

Cheetancheri (1998) provided and discussed a simple worm model and the aspects involved in defending the Internet against a worm. He developed a life cycle model of worm defense, including prevention, prediction, detection and mitigation. . The models that were developed for each of these techniques were able to automatically respond to a worm outbreak. The "friends' model" and the "hierarchical model" were two mitigating models and "TrendCenter model" was a predicting model. He concluded that worms are dangerous to the Internet but there are ways to mitigate their impact. (Cheetancheri 1988)

Costa et al (2004) proposed, Vigilante, a new host centric approach for automatic worm containment that addressed the limitations of a network centric approach. According to them, worm control must be automatic because worms can spread faster than humans can respond. A network centric approach to automate worm control by analyzing network traffic to derive a packet classifier that blocks (or rate-limits) worm propagation, has the fundamental limitation that the analysis has no information about the application vulnerabilities exploited by worms. But Vigilante relied on collaborative worm detection at end hosts in the Internet and does not require mutual trust between hosts. The hosts detected worms by analyzing attempts to infect applications and broadcasted self-certifying alerts (SCAs) which were automatically generated machine-verifiable proofs of vulnerability. SCAs are independently and inexpensively verified by any host. Then the hosts used SCAs to generate filters or patches that prevented infection. Their preliminary results showed that Vigilante controlled fast spreading worms that exploited unknown vulnerabilities.

Tang & Chen (2005) attempted to answer two questions, namely, "can a localized defense system detect new worms that were not seen before and capture the attack packets?" and "how to identify polymorphic worms from the normal background traffic?" They presented the design of a double-honey pot system, which was able to automatically detect new worms and isolate the attack traffic. They also introduced a position-aware distribution signature (PADS), which was capable of handling certain polymorphic worms, and proposed two algorithms based on Expectation-Maximization (EM) and Gibbs Sampling for efficient computation. Their experiments showed that the algorithms accurately separate new variants of the MS Blaster worm from the normal background traffic. (Tang & Chen 2005)

Ellis (2003) presented a general framework for reasoning about network worms and potency of worms within a specific network. A life cycle of a worm based on a survey of contemporary worms was discussed to build a relational model that associates worm parameters, environmental attributes, and the subsequent potency of the worm. The worm analytic framework captured the generalized mechanical process and the states a worm goes through while moving through a specific environment. According to the author, the Worm Coverage Transitive Closure (WCTC) which was a computation of a worm's final infection set given its parameters and operating environment was sufficient to describe a worm's potency with respect to a particular environment because based on current

defensive technology there are no defensive countermeasures that respond within the time scale of most worm conflicts. It was concluded that the framework can be used to evaluate worm potency and develop and validate defensive countermeasures and postures in both static and dynamic worm conflict. (Ellis 2003)

Epidemiological Studies & Network Topologies

Kephart & White (1998) studied the interaction between topology and computer epidemics by placing the susceptible-infected-susceptible (SIS) epidemiological model on a directed graph. The heterogeneous communication pattern between computer systems was represented by a directed graph. Directed edges from a particular node represented the set of systems that can be infected by a particular node. A rate of infection was associated with each edge and a rate of inoculation was associated with each node. They investigated the behavior of SIS model on the random graph, weak link, hierarchical and spatial models. They discovered that topology influences the ability of viruses to spread.

Zhou et al (2006) with the goal of understanding how the topological structures of networks affect the dynamics upon them, reviewed studies of epidemic dynamics on complex networks, including the description of classical epidemic models, the epidemic spread on small-world and scale-free networks, and network immunization. According to their findings many systems can be described as complex networks and Internet is one such example. Their study of topological structures of the networks used to model the interconnection systems has gone through three stages. The first stage used regular structure such as Euclidean lattices and hypercube networks, while the second stage (late 1950s) used random graphs. The regular networks have great clustering coefficient and long average distance, while the random networks have small clustering coefficient and short average distance. With the advancement of technology, it was later found that most real-life networks were neither completely regular nor completely random. The results of many empirical studies and statistical analysis indicated that the networks in various fields have some common characteristics such as the small-world effect and scale-free property. A small-world network has a small average distance and great clustering coefficient. Networks with power-law degree distribution are referred to as scale-free networks. Some definitions provided by them are quoted below for clarity:

- "In a network, the distance between two nodes is defined as the number of edges along the shortest path connecting them. The average distance L, then, is defined as the mean distance between two nodes, averaged over all pairs of nodes. The number of the edges incident from a node x is called the degree of x, denoted by k(x). Obviously, through the k(x) edges, there are k(x) nodes that are correlated with x; these are called the neighbor-set of x, and denoted by A(x). The clustering coefficient C(x) of node x is the ratio between the number of edges among A(x) and the total possible number, the clustering coefficient C of the whole network is the average of C(x) over all x."
- "Another important characteristic in real-life networks is the power-law degree distribution, that is $p(k) \sim k^{-\alpha}$, where k is the degree and p(k) is the probability density function for the degree distribution. α is called the power-law exponent, and usually between 2 and 3 in real-life networks. This power-law distribution

> falls off much more gradually than an exponential one, allowing for a few nodes of very large degree to exist."

Finally, they listed a few interesting problems for further investigation such as considering the role of network topology because "classical theory of infectious diseases does not care about the network topology," whether network structure affects the spreading velocity and ways to reduce it because "the spreading velocity is a very important measure especially in the outbreaks," and whether network characteristics such as community structures and the hierarchical properties affects the epidemic behaviors. (Zhou et al 2006).

Zou et al (2006) presented an Internet worm monitoring system. Their "trend detection" methodology to detect a worm at its early propagation stage by using Kalman filter estimation was based on the idea of "detecting the trend, not the burst" of monitored illegitimate traffic. They predicted overall vulnerable population size and estimated how many computers were really infected in the global Internet based on the biased monitored data for uniform-scan worms such as Code Red. They also showed that for monitoring a non-uniform scan worm, especially a sequential-scan worm such as Blaster, it is critical for the address space covered by the worm monitoring system to be distributed as possible. They recommended investigation of more detailed models to reflect a future worm's dynamics such as worm spread through a topology, or multiple vulnerability exploits, or meta-servers which may not follow the propagation models presented by them. (Zou et al 2006)

Leveille (2003) proposed Progressive Susceptible-Infected-Detected-Removed (PSIDR) epidemiological model for computer worm epidemics which incorporated aspects related to the availability of antivirus signatures, existence of direct immunization, and presence of a curing phase. Current response strategies as well as the effect of virus throttling was investigated and it was shown that slowing the progress of worms could significantly reduce costs especially in scale-free networks.

Bu and Towsley (2002) investigated the effectiveness of several power law topology generators for generating representative Internet topologies at the Autonomous System (AS) level. The Internet consists of a large collection of hosts interconnected by networks of links and routers. It is divided into thousands of administrative domains, each of which possesses one or several autonomous systems (ASs) and can be considered as either a graph of interconnected routers or a graph of interconnected ASs. They studied the AS-level Internet topology where nodes represented ASs and links represented the relationship of exchanging traffic between them. Even though the real AS-level Internet topology is unknown, it can be inferred from Border Gateway Protocol (BGP) routing tables because BGP is an inter-AS path-vector protocol. According to them, topology generators modeling Internet, fall into one of three classes, random graph generators, structural generators, and degree power law generators. The topology produced by power law topology generators resembles the AS-level Internet topology better than those produced by random graph generators or structural generators.
Their study made several key contributions which included: use of clustering coefficient and characteristic path length to distinguish power law topology generators from one another; a generalized linear preference model coupled with the incremental algorithm

generated topologies that more closely models the Internet; observed that the Internet exhibits the small world properties, and pointed out the advantage of working with the empirical complementary distribution rather than the node degree histogram for studying the node degree power law.

Epidemiological Studies with Analytical Solutions

Kyrychko & Blyuss (2005) derived and studied a time-delayed SIR model with a general incidence rate. The time delay represented temporary immunity period, i.e. time from recovery to becoming susceptible again. Both trivial and endemic equilibria were found, and their stability was investigated. Numerical simulations supported their analytical conclusions of the model.

Oli et al (2006) presented a framework for modeling the dynamics of infectious diseases in discrete time based on the well-founded theory of matrix population models. The modeling framework presented can be used to model any infectious disease of humans or wildlife with discrete disease states, irrespective of their numbers.

Stollenwerk & Jansen (2003) formulated and analyzed a model for infectious diseases transmitted by asymptomatic carriers (Neisseria meningitidis in case of meningococcal disease) by extending the classic epidemic model of susceptible-infected-recovered system (SIR) for the harmless infective agent, acting as a background to a mutant strain Y which occasionally creates severely affected hosts X. The full system of SIRYX was described in the master equation framework. With limiting assumptions of a reduced YX-system with the SIR-system in stationary, they analytically showed convergence to power law scaling typical for critical states and the divergence of the variance of outbreaks near criticality. (Stollenwerk & Jansen 2003)

Gomes & Medley (2002) provided an overview of different models describing the dynamics of “n” distinct strains of infectious agents co-infecting a host population and compared them by using the same system of coordinates with a uniform notation. They organized the coupling structure of multiple strain system into an nxn matrix, termed as the “Cross-immunity matrix”. They pointed out that the general form of the Cross immunity matrix makes the thorough mathematical analysis very difficult and the generality of the result would make it practically inapplicable. They emphasized carefully by imposing symmetry constrains to deal with this issue. The models investigated included Anderson, Lin and Levin (ALL) model; Gupta, Ferguson and Anderson (GFA) model; May and Nowak (MN) model.

References

Kevin Butler, Patrick McDaniel: Understanding Mutable Internet Pathogens, or How I Learned to Stop Worrying and Love Parasitic Behavior. In: Proceedings of 1st International Conference on Information Systems Security (ICISS), Dec. 2005, Kolkata, India.

Kevin Butler. Parasim. A simulator that models Internet parasite propagation. Systems and Internet Infrastructure Security Laboratory (SIIS), The Pennsylvania State University, May 2005.

Alberto Medina, Anukool Lakhina, Ibrahim Matta, and John Byers: BRITE. A universal topology generator. QoS Networking Laboratory (QNL), Boston University, 2002.

Alberto Medina, Anukool Lakhina, Ibrahim Matta, and John Byers: BRITE: Universal Topology Generation from a User's Perspective. (User Manual) BU-CS-TR-2001-003. April 05, 2001.

Jasmine Leveille. Epidemic Spreading in Technological Networks. Information Infrastructure Laboratory HP Laboratories Bristol HPL-2002-287. Oct. 2003.

Jeffrey O. Kephart and Steve White. Directed graph epidemiological models of computer viruses. In Proceedings IEEE Symposium on Security and Privacy, 1991.

Nicholas Weaver, Vern Paxson, Stuart Staniford, Robert Cunningham: A Taxonomy of Computer Worms. In: WORM'03 - Proceedings of the 2003 ACM Workshop on Rapid Malcode, Washington, DC, USA (2003) 11-18.

Stuart Staniford, Vern Paxson, Nicholas Weaver: How to Own the Internet in Your Spare Time. In: Proceedings of the 11th USENIX Security Symposium, San Francisco, CA, USA (2002) 149 – 167

Cliff C Zou, Weibo Gong, Don Towsley. Code red worm propagation modeling and analysis. Proceedings of the 9th ACM conference on Computer and Communications Security, Washington, DC, USA (2002) 138 – 147.

Donn Seeley. A Tour of the Worm. Department of Computer Science,University of Utah. securitydigest.org (1988)

Senthilkumar G Cheetancheri. Modelling a Computer Worm Defense System. Master's thesis for B.E. Computer Science & Engineering. Coimbatore Institute of Technology, Coimbatore, India. (1998)

Tao Zhou, Zhongqian Fu, and Binghong Wang. Epidemic dynamics on complex networks. Progress in Natural Science, 16(5): 452-457 (2006)

Zou, C. C., Gong, W., Towsley, D., and Gao, L. The monitoring and early detection of internet worms. *IEEE/ACM Trans. Netw.* 13, 5 (Oct. 2005), 961-974.

Manuel Costa, Jon Crowcroft, Miguel Castro and Antony Rowstron. Can we contain Internet worms? Proceedings of the 3rd Workshop on Hot Topics in Networks 2004.

Tian Bu and Don Towsley. On Distinguishing between Internet Power Law Topology Generators. Twenty-First Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings IEEE. INFOCOM 2002.

Yong Tang, and Shigang Chen. Defending Against Internet Worms: A Signature-Based Approach. 24th Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings IEEE. INFOCOM 2005.

Yun-Kai Zhang, Fang-Wei Wang, Yu-Qing Zhang, and Jian-Feng Ma. Worm propagation modeling and analysis based on quarantine. In Proceedings of the 3rd international Conference on information Security (Shanghai, China, November 14 - 16, 2004). InfoSecu '04, vol. 85. ACM Press, New York, NY, 69-75.

Cliff C. Zou, Don Towsley, Weibo Gong. On the performance of Internet worm scanning strategies. Performance Evaluation 63 (2006) 700–723

Zou, C. C., Gong, W., and Towsley, D. Worm propagation modeling and analysis under dynamic quarantine defense. In Proceedings of the 2003 ACM Workshop on Rapid Malcode (Washington, DC, USA, October 27 - 27, 2003). WORM '03. ACM Press, New York, NY, 51-60. 2003.

Ellis, D. 2003. Worm anatomy and model. In *Proceedings of the 2003 ACM Workshop on Rapid Malcode* (Washington, DC, USA, October 27 - 27, 2003). WORM '03. ACM Press, New York, NY, 42-50.

Yuliya N. Kyrychko, Konstantin B. Blyuss. Global properties of a delayed SIR model with temporary immunity and nonlinear incidence rate. Nonlinear Analysis: RealWorld Applications 6 (2005) 495 – 507

Madan K. Oli, Meenakshi Venkataramana, Paul A. Kleinb, Lori D. Wendland, Mary B. Brown. Population dynamics of infectious diseases: A discrete time model. Ecological Modelling Article in press. (2006)

Nico Stollenwerk, Vincent A.A. Jansen. Meningitis, pathogenicity near criticality: the epidemiology of meningococcal disease as a model for accidental pathogens.
Journal of Theoretical Biology 222 (2003) 347–359

M. Gabriela M. Gomes, Graham F. Medley. Dynamics of multiple strains of infectious agents coupled by cross-immunity: A comparison of models. IMA Volumes in Mathematics and Its Applications. 2002. Vol. 126, 171-192.