# On Computer Viral Infection and the Effect of Immunization

Chenxi Wang        John C. Knight        Matthew C. Elder
Department of Computer Science
University of Virginia

## Abstract

*Viruses remain a significant threat to modern networked computer systems. Despite the best efforts of those who develop anti-virus systems, new viruses and new types of virus that are not dealt with by existing protection schemes appear regularly. In addition, the rate at which a virus can spread has risen dramatically with the increase in connectivity. Defenses against infections by known viruses rely at present on immunization yet, for a variety of reasons, immunization is often only effective on a subset of the nodes in a network and many nodes remain unprotected. Little is known about either the way in which a viral infection proceeds in general or the way that immunization affects the infection process. In this paper we present the results of a simulation study of the way in which virus infections propagate through certain types of network and of the effect that partial immunization has on the infection. The key result is that relatively low levels of immunization can slow an infection significantly.*

## 1. Introduction

In recent years, computer viruses, a type of deliberate fault, have increased dramatically in number, and they have also begun to appear in new and more complex forms [9], [10], [11]. As a result, the task of detection and prevention of viruses has become increasingly difficult. Compounding an already difficult problem is the increased connectivity of modern computer systems. This exacerbates the problem because viruses and worms can now use networks as a new medium for propagation. They can sweep quickly through thousands of hosts, an effect that is far more damaging than what would occur in a more traditional, stand-alone computing environment.

Traditional anti-virus techniques focus typically on detection of the static signatures of viruses. While these techniques are somewhat effective in their own right, they do not address the dynamic nature of a virus infection within the context of the underlying system. In a computer network, a virus can propagate through the network quickly, and it might infect and damage many, perhaps all, machines before the severity of the situation is recognized.

A valuable mechanism for tolerating this type of deliberate fault would be to detect the presence of an infection in a network at an early stage and to have the network react to the attack in real time to mitigate the damage. A number of challenges exist in developing such a scheme. First, a thorough understanding of the network-wide characteristics of viral infections is needed. If such characteristics were known, mechanisms might be developed to detect an on-going, wide-area infection. Perhaps of greater importance is the prospect of developing defense mechanisms that would operate in real time and on a network-wide basis. Clearly the effect of factors such as the rate and pattern of infection, the underlying network topology, and stochastic variations in the network must be well understood before a comprehensive view of infection could be developed. Second, techniques for acquiring a global perspective of the infection and real-time controls of the network are essential for thwarting viral infections. This implies the need for real-time, reliable network monitoring and management, a topic of much ongoing research [1], [3], [6].

In this paper, we report on a study of network viral infection. The study was conducted using simulation and examined several key characteristics of infection, including the rate of infection through the network and the rate at which individual nodes are re-infected during an attack. As a key part of the study, we have examined the impact of immunization on infection, a difficult practical problem in network management. Clearly, immunization can protect a system from the effects of a known virus, but in a large network it is essentially impossible to be sure that all the nodes are properly immunized. This raises the question of what the effect might be of immunization that is only effective on cer-

tain nodes. Dealing with this question might also allow insight into the conscious use of selective immunization where the task of immunizing a complete network is infeasible.

The paper is organized as follows. In section 2, we review the basic concepts of viruses and some of the current anti-virus work. Section 3 discusses the factors influencing computer viral infection and the factors used in this study. We discuss the limitations of analytic modeling in Section 4, then present the design and framework for our experiments in section 5. In sections 6 and 7, we present our experimental results and explore the issue of immunization, both random and selective. We discuss the open issues in virus research and summarize the paper in section 8.

## 2. Related Work

Viruses and worms are self-replicating programs that sometimes have the goal of damaging their hosts and arranging for copies of themselves to propagate to new hosts [2]. For simplicity we use the term virus throughout the rest of this paper to mean an infectious agent that can infect computers to which it has access.

Viruses and worms have been studied extensively by both the research and the application communities. Cohen's work in the 1980's formed the theoretical basis for the field [2]. In the ensuing decade, many significant scientific and technological advances have been made in the battle against computer viruses.

The majority of the current anti-virus techniques employ static scanning methods in which programs are scanned in search of a sequence of instructions known as the virus *signature*. Each time a new virus is discovered, its signature is added to the database of virus signatures. In response to this approach, virus writers have developed more complex and innovative ways to write viruses that are capable of evading simple scanning (e.g. polymorphic viruses). Producers of virus protection systems have countered with new scanning methods to cope with the latest viruses. This co-evolution, eloquently described by Nachenberg [9], summarizes the last fifteen years of an arms race between virus writers and the anti-virus industry.

The work on virus protection has produced many useful tools and technologies. However, the approaches are limited to the individual properties of the virus, such as the signature it carries, the types of programs it might infect, and so on. It is not surprising that each time a new virus appears, the anti-virus industry finds itself scrambling to produce yet another defense mechanism. There is an evident lack of study of virus activities in the context of the underlying systems with regard to the many system attributes that might impact the viral infection—few

attempts have been made to investigate how fast viruses can spread, the patterns of infection, and how factors such as the network topology affect their prevalence, etc.

Kephart, Chess, and White of IBM conducted a study of viral infection based on epidemiology models [4], [5]. They constructed an analytical model in which they characterized viral infection in terms of birth rate (the rate at which machines are infected), death rate (the rate at which machines are cured), and the patterns of transmission of information between computers.

The IBM study was based on a model in which viruses were spread via activities mostly confined to local interactions. The authors indicate that at the time of the study this was one of the more prevalent interaction models where infection takes place when individuals share disks because "most individuals exchange most of their software with just a few others and never contact the majority of the world's population" [4]. Based on this model, they concluded that most virus activities were localized, and virus propagation rarely reached the exponential rate indicated by the classical epidemiological models. While their findings are sound and supported by strong empirical evidence, new patterns of interaction and changes in system connectivity suggest that it is necessary to reevaluate some of the assumptions and simplifications of the IBM model.

In addition, the epidemiology model used in the IBM study is primarily concerned with the global aspects of the viral propagation. Details of individual infections, such as variations of infection experienced by different hosts during a virus attack, are, to a large extent, ignored. While tracing low-level details of individual infection is an intractable problem in any sizable population, we argue that a study of carefully selected low-level characteristics can be beneficial in that it might unveil information useful in establishing effective defense mechanisms—we will show such an example in the selective immunization study in later sections. This type of information cannot be discerned from study of the global behavior only, and simple analytic modeling is likely to overlook them.

Finally, we note that in the Serrano project at the University of California, San Diego, Marzullo and his colleagues are investigating fault tolerance in large networks including the effects of viral infection [8]. However, their work is specially tuned to the study of multicast protocols and the effect on self-propagating attacks. The applicability of the model is therefore limited.

## 3. Factors Influencing Network Infection

Many factors can influence the way a viral infection progresses, including those from the environment and those that are inherent to the infecting agent. The rate of spread of Melissa, for example, depended on how often

users read e-mail and what entries they had in their address books. The rate of reading e-mail corresponded to the "processing rate" that the virus could expect and the address book entries defined the topology of the network that the virus could infect.

Before proceeding with any analysis of infection, a precise and complete framework for that analysis needs to be established. The goal of this framework is to identify the factors that influence infection characteristics and enumerate the values that each factor can take. The factors that affect viral infection are in two areas: (1) the underlying target computer system, and (2) the infection process used by the virus.

## 3.1. The target system

We assume a target system consisting of a large network of heterogeneous nodes connected by some mechanism that is not necessarily a traditional network link. For example, a node-to-node connection for the Melissa virus required that the virus obtain a valid e-mail address for a remote machine and that a mail connection exist between the machines. A node that was merely connected by an Ethernet to an infected node, in this case, would not necessarily become infected.

The factors of interest pertaining to the target system are the following:

*System Topology.* The system topology defines the paths that a virus can follow when propagating. We note that this does not necessarily mean either a fully-interconnected topology or an infection path along every network link.

Our interest lies in the networks used to support critical infrastructure applications. Such applications employ private networks whose topologies are determined in large measure by the needs of the application [7]. This contrasts considerably with the fully connected, open nature of the Internet. In this study, we chose two network topologies—*hierarchical* and *clustered*. By hierarchical me mean a network with a tree like structure in which nodes are connected to parent and child nodes. This topology is typical of those found in the banking and financial networks. By clustered we mean a network in which nodes are organized in clusters that have high connectivity within clusters and low connectivity between clusters. This topology is typical of many transportation- and energy-control networks.

*Node Immunity.* The IBM study characterized nodes as being in one of two states—s*usceptible* and *infected*. Once an infected node is cured, it immediately enters the susceptible state again. We broaden the state space by bringing in the notion of immunity to represent the lack of susceptibility of a node to a particular virus. For example, a Unix host is immune to a Windows virus, and a node infected by a particular virus might not be susceptible to the same virus

at a later time because of changes in the environment such as patches, upgrades, the repair of a flaw that the virus exploited, and so on. Using this notion, a node can be in one of three states—*susceptible*, *infected,* and *immune.*

In the study reported here, for any given model we assume that nodes can be either permanently immune or either susceptible or infected. We further assume that once a node has been infected, i.e., changed from susceptible to infected, then it remains infected.

*Temporal Effects.* The temporal characteristics of the underlying system such as processing and communication delays are likely to have a significant effect on the propagation of viruses. A virus will have to compete with other processes for system resources, and so replication and propagation might take time that is both significant and variable.

We model the processing time required by a virus to complete the infection of a node as a constant value of one clock tick, and we assume transmission time from one node to another is instantaneous.

## 3.2. The infection process

By the infection process we mean the underlying algorithm that the virus uses to propagate itself. It should be noted that we are not concerned with some of the properties of the infecting agent such as the payload, i.e., whatever code or data it carries to permit it to inflict damage on the host. Factors of interest regarding the infection process are the following:

*Propagation Selection.* The spread of viruses from one node to others is determined by the propagation algorithm of the viral program. It is not necessarily optimal from the virus' point of view to infect everything that it can immediately.

In this study, we assume that the virus can choose to infect any subset of the nodes to which its host is connected, and that each copy of the virus makes independent decisions at each infection point. The decisions are assumed to be random and independent of past infection history.

*Multiple Infections.* An infected node need not be protected from subsequent reinfection by the same virus. If reinfection occurs, a single node might become host to multiple copies of the virus. In this study, we assume that a node can be infected multiple times and concurrently by multiple copies of the same virus.

*Stochastic Effects.* The infection process will be affected by non-determinism in the virus itself. A virus will have to make choices both to improve the chances of its infection being successful and to improve whatever disguises it chooses to use.

### 3.3. Characteristics studied

We studied three characteristics of network viral infection: total infection time; rate of propagation; and node reinfection count.

*Total infection time* is the time taken by the virus to infect the entire network. Knowing the details of the time for an infection to spread through an entire network is useful in preparing a response. For example, if the expected time were especially long it might be possible to continue normal operations for lengthy periods during an infection. This would make prompt detection of an infection less critical. Moreover, if total infection time were lengthened by certain network design factors, these could be deliberately introduced.

The generalization of total infection time is the *rate of propagation*. By rate of propagation we mean the rate at which nodes become infected over time during an attack expressed as the fraction of nodes infected at time $t$ (total infection time is the time at which 100% of the nodes are infected). Rate of propagation is indicative of the nature of the infection. In particular, if a pattern of rate changes were observed, it might be possible to use this as part of an infection signature. Similarly, changes in rate might show how timely a response to an attack needs to be deployed. If for a given virus and a certain type of network an attack is known to be slow at some point, the approach to treatment could exploit this relative "lull" in activity. It might also indicate that some fraction of the population could expect to be attacked in a relatively late stage (or early stage) of the infection so that they could be responsible (or not) for critical functions. Finally, if widespread immunization is to be attempted during an attack (assuming that a suitable "vaccine" could be synthesized in real time), the preferred approach to distribution of the vaccine would depend upon detailed knowledge of the way in which the infection progresses.

The *node reinfection count* is the number of times a copy of the virus visits a given node irrespective of whether the node has already been infected. In many real-world scenarios (and in our model), viruses do not keep track of the hosts that they have infected and so attempt reinfection of already infected nodes. Knowing how many times this occurs allows decisions to be made about prevention and treatment. For example, the utility of merely disinfecting an infected node may or may not be effective, depending upon whether reinfection is likely to happen in a rapid succession. Similarly, the rate of reinfection will permit choices to be made about the speed with which immunization needs to become effective. Immunizing a node once it has been disinfected might be a lengthy process and its utility is a trade-off that is heavily influenced by reinfection rates. A final possibility once reinfection counts are known is to use the characteristics of reinfection that a single node experiences as part of a local signature of the attack.

## 4. The Limitations of Analytic Modeling

In principle, the characteristics of computer viral infections could be studied using analytic models, simulation, or a combination of both. Obviously, analytic models are desirable because they provide the most comprehensive means to study a problem. However, despite the success of previous work in analytic modeling, the combination of complex network topologies, sophisticated infection strategies, and the level of details that we wish to model makes the type of analytic modeling that has been reported in the literature intractable.

As an example of the difficulties that arise with attempts to build analytic models, consider the issue of modeling the probability of infection for a given node in the network. To make the analysis tractable, it has been assumed in some analytic models that this probability is the same for all nodes and that it is constant in time. In fact, neither of these assumptions holds.

The probability that a node becomes infected is not the same for every node because it is a function of the node's connectivity and of the infection characteristics of the viral program. Similarly, the probability that a node becomes infected is not fixed in time because, as more and more nodes become infected, the probability of an un-infected node becoming infected increases. The stochastic nature of both the network and the infection process is likely to render considerable variations in this probability for different nodes and in different instances of time. Any analytic model that fails to capture the variance in these parameters is likely to be in error.

There is no simplification that can be applied here that will allow a tractable model to be developed nor is it possible to seek a steady-state solution since by definition, no meaningful steady state exists. An approach that might be able to capture the complexity of the systems of interest and that might be feasible in this case is Markov analysis. We are pursuing such models in an ongoing study. The study reported here was undertaken with simulation.

## 5. Experimental Design

### 5.1. Simulation environment

Our experiments have been conducted using a special-purpose simulation environment that is capable of simulating thousands of computing nodes with any desired network communications topology and any viral infection

process. The network topology that is used in a simulation is read by the system from a file that contains a description of all the nodes in the network and all the inter-node connections.

The file that describes the desired network is synthesized from a high-level specification of the topology so as to permit rapid generation of different instances of the same type of topology and instances of different topologies. This permits handcrafting of detailed requirements or the creation of a specific network topology of interest.

A virtual time mechanism is implemented to keep track of network time during simulation. The system simulates infection decisions and transmission activity for each copy of the virus on each time tick and monitors the state of the infection as virtual time passes. Relevant data is recorded on each time tick and simulation stops when some prescribed state is reached, such as all nodes are infected.

## 5.2. System models

A 1,000-node instance for each of the two network topologies (hierarchic and clustered) was built for testing. For the hierarchic model, a single root node and a connectivity fan out of at most 20 from each node to its children was used. For the cluster model, 36 clusters with an average size of 27 nodes were sparsely connected.

Two viral infection models were analyzed: single fan out and multiple fan out. By *infection fan out* we mean the number of copies that a single copy of the virus can generate on nodes connected to the host upon which it is executing. In the single fan out infection model, a virus selects only one neighboring node to infect (i.e., the fan out of the virus is one). In this case, new infections occur one at a time for each copy of the virus and only one copy of the virus is ever replicating actively in the system.

The single fan out infection model represents the slowest rate at which an active virus could spread and so we refer to it as the *baseline* infection model. This model is perhaps simplistic but still a possible infection model in practice. For example, a virus trying to disguise its presence might very well implement an infection model much like this, endeavoring to propagate unobserved in the network at a slow speed so as to detonate a payload on all the nodes in the network at the same time. It appears that the resources consumed by large numbers of copies of a virus on a single node is often the first sign that an infection is underway, and so keeping this factor under control is an obvious strategy that a stealthy virus would employ.

In the multiple fan out infection model, a single copy of the virus is able to infect a random number of nodes connected to the infected host (i.e., the fan out of the virus is greater than one). The random number of new infections is chosen by the virus to be between one and a specified

bound. The bound for any particular infection is set between two and the maximum fan out that occurs in the topology (a parameter that is specified in the experiment configuration).

A few assumptions and simplifications were made to ensure feasibility of the experiment. First, while multiple copies of the virus can exist concurrently on the same host, we assume that the number of viruses on a single host does not exceed 100. This is to ensure that the experiment proceeds at a reasonable speed, and we believe that 100 is a reasonable value given that more copies will bog down the host completely. Second, a single starting point was used to release the virus, and this starting point was randomly chosen in each trial. Finally, the non-determinism of the infection process was simulated by repeating simulations using different random sequences for virus decision making.

## 6. Networks Without Immunization

In this section we present measurements of viral infections in networks where none of the nodes was immunized. These experiments provide insight into the characteristics of infection and they also serve as the control sample for the immunization experiments presented in the later sections.

### 6.1. Hierarchic topology

Figure 1 shows the distribution obtained from 1,000 runs of the simulation of the total infection time for a hierarchic network topology with the single fan out infection model. The most important thing to note in this experiment is the tremendous variation that exists in the time to infect the entire network. The fastest total infection time was 31,986 clock ticks and the longest was 160,943 clock ticks, a ratio of over 5 to 1. Note that this is solely the effect of stochastic variation resulting from randomness in the infection process—the infection model and all other parameters were the same in every simulation trial.

This variation is a result of the sparse connectivity of the hierarchic topology and the low infection probability of the baseline infection model (fan out of one). Depending on the infection process (e.g., which path the infection follows, etc.), a virus might spend much of its time infecting and re-infecting a small part of the network, and a considerable amount of time could elapse before it manages to venture out to other parts of the network.

Figure 2 shows the rate of propagation averaged across the 1,000 runs of the baseline infection model. Note that the number of infected nodes quickly rose to 80% of the total population, and the infection growth leveled off after that (infecting the remaining 20% of the network took up
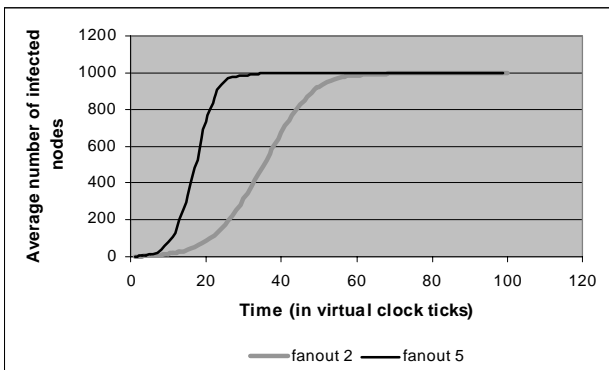
**mean = 70743, variance = 386070837**
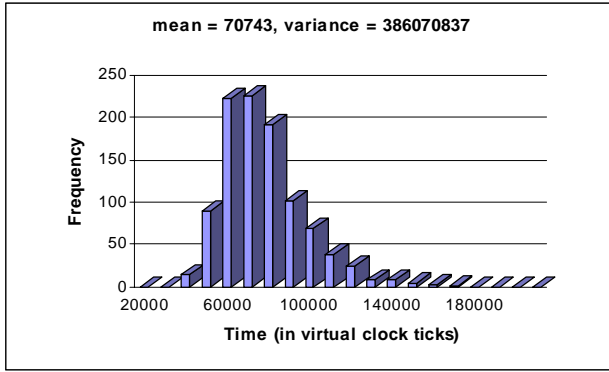
**Figure 1: Distribution of total infection time**



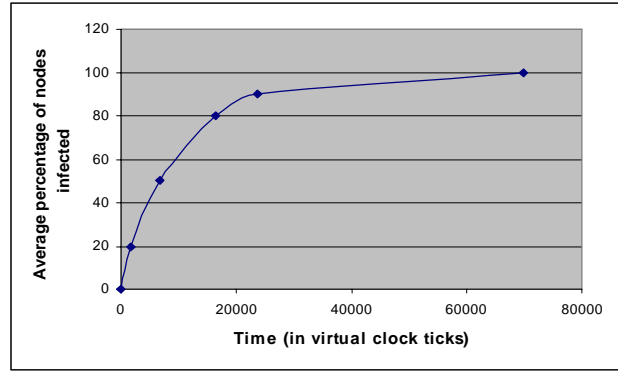**Figure 2: Average rate of propagation (hierarchic)**

the bulk of the total infection time). What this means is that, in the late stage of the infection, the virus spent much of its time revisiting nodes that had already been infected. This result is consistent with the IBM study. It further confirmed that treatment of a viral propagation in its early stage is both important and advantageous in the prevention of further propagation. However, it also implies that a considerable fraction of the nodes in a hierarchic network subject to this type of infection remain uninfected for long periods of time and this might be exploited to allow some forms of service to be maintained.

Much more rapid propagations were observed when larger fan outs were specified. Figure 3 shows the average rate of propagation over the 1,000 runs for infection fan outs of 2 and 5. Note that the difference in the rate of propagation between the higher fan outs and that of the baseline case expands several orders of magnitude. In addition, compared with the baseline study, there is significantly less variation in the higher fan out experiments. For example, the trials with a fan out of 2 produced a total infection time distribution with a mean of 46.9 clock ticks and a variance of 7.79. Less variance was observed with a fan out of 5, which produced a distribution with a mean of 23.6 clock ticks and a variance of 5.19.

A larger fan out value corresponds to a higher infection

probability. Results of these experiments showed that when infection probability is high, the sensitivity of the infection dynamics to stochastic variation decreases.

In order to better understand what might be happening during a viral infection, we measured the number of reinfections that each node experienced during an infection. Figure 4 shows the average number of reinfections for each of the 1,000 nodes in the baseline study. Clearly, some population of the nodes were attacked much more heavily than others. For example, node 302 was attacked 398 times on average, while node 525, 526, and 527 were attacked 33 times only. Further investigation showed little variation in the number of reinfections experienced by the same node in different trials. That is, the nodes that are attacked often in one simulation run are likely to be the most often attacked in other simulation runs, provided that the infection model remained the same.

This result is significant because it points out that, for a given topology and a given infection model, some nodes in the network are more prone to being attacked than others. It is not difficult to imagine that these nodes occupy critical locations where a viral infection must revisit in order to propagate to different parts of the system. Note that characteristics such as the reinfection rate cannot be easily captured using simple analytic models, as it is the function of
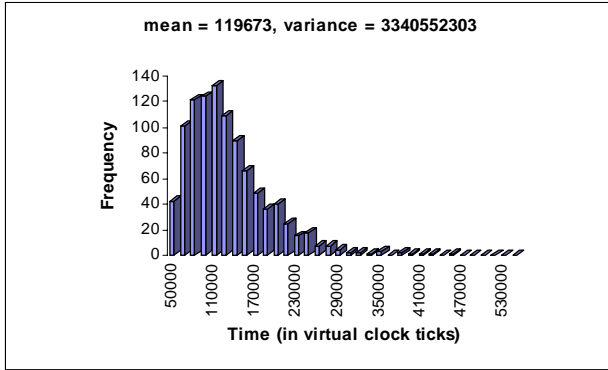


**Figure 3: Average rate of infection(hierarchic)**
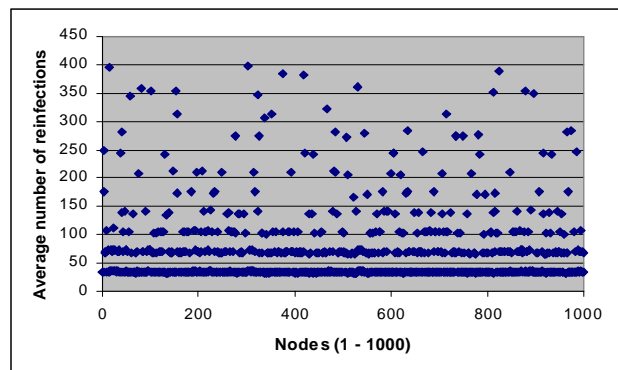


**Figure 4: Average reinfection counts**

**Figure 5: Distribution of total infection time**



**Figure 6: Average rate of propagation (fanout 1)**

many factors including time, the infection model, and the underlying topology.

## 6.2. Cluster topology

A similar set of experiments were conducted for the cluster network topology. Figure 5 shows the distribution of the total infection time for 1,000 runs of the baseline infection model. As with the hierarchic model, a great deal of variation exists in time to infect the entire network. Figure 6 shows the rate of propagation across the 1,000 runs with the baseline infection model. The initial infection growth in the cluster network is greater than that of the hierarchic case. However, infecting the last 10% of the network in the cluster case took a significantly larger amount of time than in the hierarchic network.

Figure 7 shows the rate of propagation for infection fan outs of 2 and 5. As with the hierarchic case, a much more rapid propagation resulted from higher values of fan outs.

Figure 8 shows the average number of reinfections for each of the 1,000 nodes in the cluster network across the 1,000 runs in the baseline case. The level of variation we observed from the hierarchic model remains. Note once again how the number of reinfections varied across nodes, i.e., a certain population of the network was attacked much
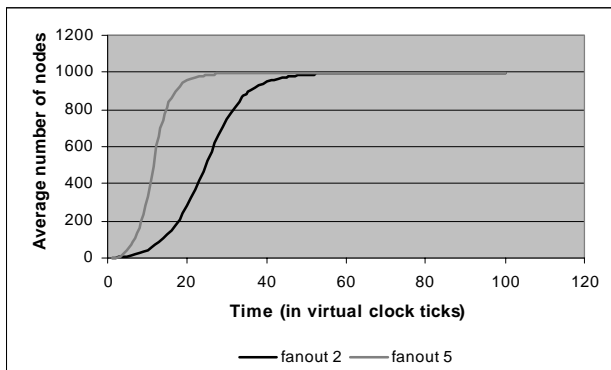
more often during an infection.

## 7. The Effect of Immunization

The result of our analysis showed that viral infection can propagate at an alarming speed in systems where dynamic detection and remedy are not present. Furthermore, the infection characteristics experienced by individual nodes vary significantly, and that these variations might be inherent to the propagation process. It begs the question of whether one could exploit these individual variations in the design of defense mechanisms, and so in this section we explore the effect of immunity.

Immunization in the computational realm is the ability to prevent a viral program from executing and replicating further to other hosts. There are many reasons a node might be immune to a virus. For example, a host running Unix is immune to Windows-based viruses, or a node can become immunized against a particular virus if the ways that the virus exploits the underlying host are disabled.

It is not our intent to investigate the ways in which immunization can be achieved. Rather, assuming that immunization techniques exist, our goal is to examine what the most effective strategy is for immunization. Clearly, it is often not feasible to immunize the entire network. A
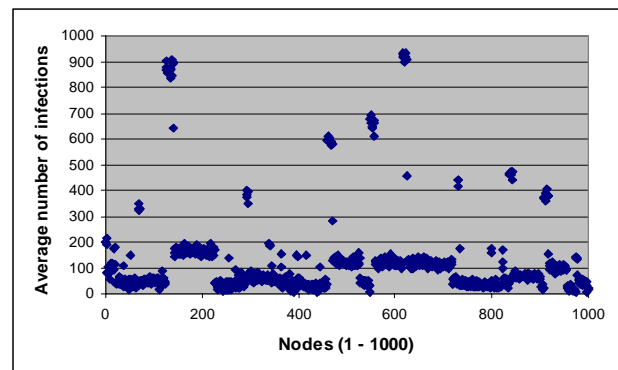


**Figure 7: Average rate of propagation (cluster)**



**Figure 8: Average reinfection counts**
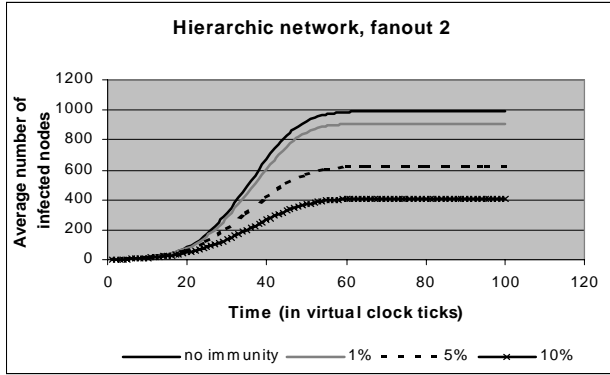
**Figure 9: Average rate of propagation**



**Figure 10: Average rate of propagation**

more realistic approach would be to immunize a subset of the population, and so choosing the appropriate size and membership of that subset becomes an important question.

## 7.1. Random immunity

The first set of immunization experiments we conducted was with random immunity, i.e., for each trial, a set of nodes was selected at random to be immune. The objective of experimenting with random immunity was to investigate the effect of immunization with respect to the size of the immunized population. In this case it is not important which nodes are immunized, but how many.

We performed experiments using the multiple fan out infection model on both the hierarchic and cluster topology, with 1%, 5%, and 10% of the population immunized. For each of the 1,000 runs, the simulation ran until all of the copies of the virus died (by propagating to immune nodes) or 100 virtual time ticks were reached. For the 1% immunity case, the immune nodes successfully killed off the virus in 19 out of 1000 runs. As the immunity level increased, the probability of epidemic decreased; in the 5% immunity case 147 of the 1000 runs resulted in elimination of the virus, and in the 10% immunity case 227 did not survive due to the virus' elimination.

For the simulation runs in which the virus survived and successfully propagated, we recorded the rate of propagation for each trial, and computed the average rate of propagation over those runs (981 for the 1% case, 853 for the 5% case, and 773 for the 10% case). Figure 9 shows the average rate of propagation for the various immunity levels in the hierarchic topology. As shown in Figure 9, there is little difference in the rate of propagation of 1% immunity and that of no immunity. A significant decrease in the rate of propagation occurred as the size of the immunized population rose to 5%, and more so with 10% of the population immunized.

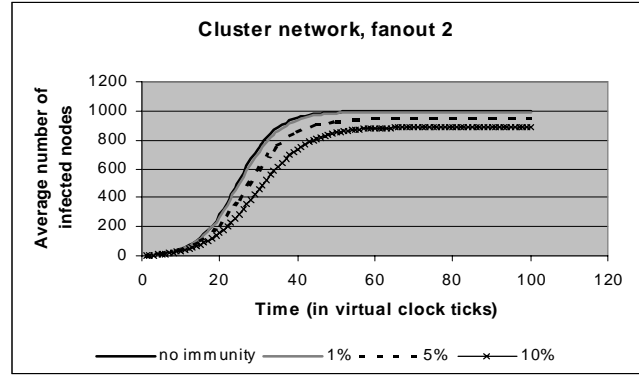Similar outcomes were observed with the cluster topology. With 1% of the population randomly immunized, the

virus was completely eliminated in only 20 runs, but with 5% immune that figure rose to 110 runs and with 10% immune it occurred 248 times. Figure 10 shows the average rate of propagation for the cluster topology with the various immunity levels in the remaining runs where the virus survived.

The result of these experiments is intuitive. One would expect a lower rate of infectious spread when more members of the population are immunized. The reason that immunization performs better in the hierarchic topology is also intuitive: in a hierarchic structure, there is only one path from one node to any other node. It is therefore possible to cut off an entire subtree of population if the root node of the subtree was immunized and the infection started from outside of that subtree. In the cluster topology, however, there may exist multiple paths between clusters, and similarly between pairs of nodes. Immunization could slow down the spread of infection, but not at the same rate or magnitude as in the hierarchic case.

In practice, random immunity models the scenario in which a large network consists of independently administered subdomains. Although the goal is to immunize all the nodes in the network, many remain vulnerable for various reasons—cost, defective installation, lack of awareness, and so on. In such cases, the nodes which are properly immunized are likely to be "randomly" distributed through the network. Knowing something of the effect of such incomplete immunization is therefore useful.

## 7.2. Selective immunity

A second set of immunization experiments was designed to investigate the effect of *selective* immunity. By selective immunity we mean that the set of immunized nodes is prescribed and they remain the same throughout different trials of the experiment. The objective of this experiment was to investigate how the dynamics of viral propagation were affected by the details of *which* nodes are immunized, in addition to how many are immunized.
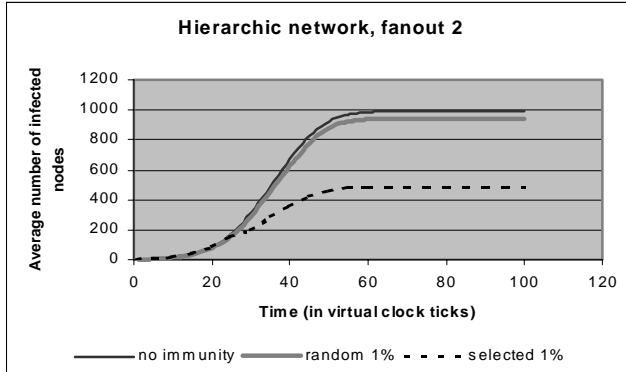
**Figure 11: 1% immunization**



**Figure 12: 1% immunization**

As seen in the results presented in Section 6, there exist nodes that tend to be much more heavily attacked than others during a viral infection. This suggests that the locations of these nodes bear more importance for viral propagation, and that a careful investigation of immunizing precisely these locations is a worthwhile exercise. In these experiments, we identified the population with the highest reinfection counts as indicated by the control study for each topology, and selected three sets, the nodes with the top 1%, 5%, and 10% rates of reinfection, as the targets of immunization.

**7.2.1. Hierarchic topology.** For the hierarchic topology, the immunized population was selected as the ones with the highest reinfection counts seen in the control study. This set also corresponds to the set of nodes with the most number of neighbors in the topology. We performed 1,000 runs with 1%, 5%, and 10% of the population immunized; for each run, the infection fan out parameter was set to at most 2. As in the random immunity case, each simulation run ended when all of the copies of the virus died or 100 virtual time ticks were reached. Out of the 1,000 runs with 1% of the population selectively immunized, the virus was completely eliminated in 105 cases (compared to 19 runs in the random immunity case). Figure 11 shows the average rate of propagation in the remaining 895 runs with 1% of the population selectively immunized, plotted with random immunity and no immunity for comparison. Selective immunity performed considerably better than 1% random immunity.

The reason for this difference is that the set of immunized nodes included the root nodes of two substantial subtrees. In effect, this partitioned the network into several large chunks and any virus outbreak from a single point is capable of infecting a single piece only, not the entire network. In this aspect, immunizing a low-level node (e.g. one that is a leaf or a near-leaf node) is not as effective as immunizing high-level nodes. As the size of the immunized population rises, the network becomes further frag-
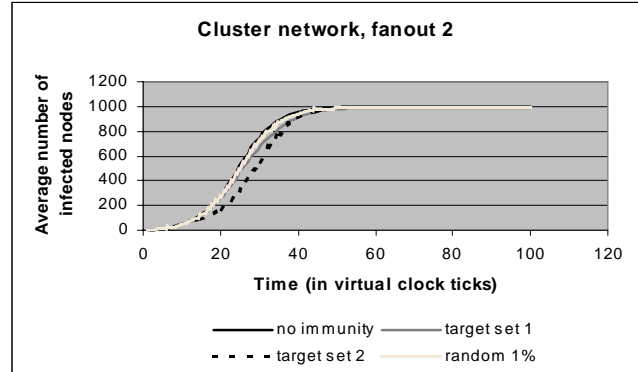
mented, and the probability of an epidemic developing diminishes. In the experiment where we immunized 5% of the population selectively, the virus survived in only 476 out of 1,000 trials (compared to 853 runs in the random 5% immunity experiment). Even more telling is the average rate of propagation in the remaining 524 runs: the virus only infected an average of 13 nodes even when it did survive. This further quantified the conclusion that in some network topologies (e.g., a hierarchic network), it is more important and cost-effective to concentrate on *who* should be immunized rather than the size of the immunized population.

**7.2.2. Cluster topology.** A cluster network has very different characteristics than a hierarchic network. For example, there is no longer a single path between any pair of nodes, and depending on the connectivity between clusters, such a topology can be easily transformed to represent either a random graph or a strongly connected network topology.

In the cluster network, however, the set of nodes with the highest reinfection counts in the cluster control study is not the same as those with the largest number of links. The reason is that in this cluster network, the top 20% of the nodes that were attacked the most often belong predominantly to a few clusters. Not surprisingly, these clusters have the most links to other clusters. If the cluster graph is transformed into another graph where each cluster is represented by a single node, it is then readily apparent that, in the second graph, these "cluster-nodes" with a greater number of links to other clusters will be attacked more often than others, and that nodes belong to those clusters are likely to have a higher number of reinfections.

We were interested in examining the effect of immunizing the set of nodes with the highest reinfection rate as well as those with the most important links. For convenience, we refer to the former as target set #1 and the latter as target set #2. Intuitively, inter-cluster links are more important than those that connect nodes in the same cluster. To take this factor into consideration when determining target
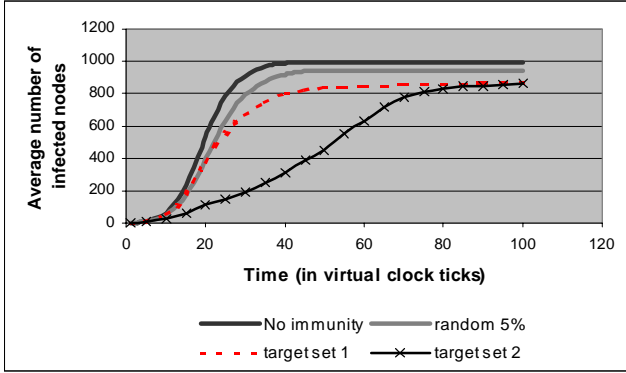
**Figure 13: 5% selective immunity (cluster)**



**Figure 14: 10% selective immunity (cluster)**

set #2, we used a weighting scheme in a definition of "connectivity"—an inter-cluster link is weighted ten times as much as an internal link.

Again, we selectively immunized 1%, 5%, and 10% of the total population and performed 1,000 trials, each until all copies of the virus were eliminated or virtual time 100 was reached. Table 1 shows the number of epidemics—that is, runs where the virus survived—that occurred for both target sets across the different levels selective immunization.
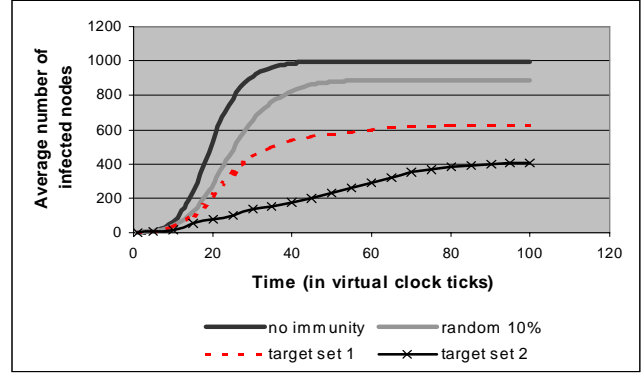
The data in Table 1 shows that target set #1 (chosen

| Immunized population | Random selection | Target set #1 | Target set #2 |
|:---:|:---:|:---:|:---:|
| 1% | 980 | 897 | 970 |
| 5% | 890 | 604 | 880 |
| 10% | 752 | 306 | 770 |

**Table 1: Number of epidemics in 1,000 runs**

based on reinfection count) consistently results in fewer epidemics than both the random immunity sets and target set #2 (chosen based on number of weighted links). In other words, the immune nodes in target set #1 were far more successful in killing off all of the copies of a viral propagation than the other two strategies. Random immunity and target set #2 produced roughly the same results over 1,000 runs, with target set #2 being more effective with the smaller immune populations.

In the runs in which the virus survived, the average rate of propagation was calculated. Figure 12 shows the average rate of propagation with 1% of the population immunized. Both target sets display a slightly better rate of propagation than the random immunization case, although target set #2 exhibits a slower rate than all others at earlier points in time.

Figures 13 and 14 show the average rate of propagation for epidemics with 5% and 10% of the population selectively immunized, respectively. In both cases selective immunization performed significantly better than random immunization. Target set #2 again demonstrated a slower rate of propagation at earlier points in time, although in the 10% selective immunization experiment it maintained the better rate of propagation throughout.

## 7.3. Analysis of immunization

These results show that by both measures of effectiveness—number of epidemics and average rate of propagation—selective immunization performs better than random immunization. Injecting immunity into a carefully selected set of nodes can yield a network in which more nodes are likely to survive, the spread of infection is likely to be much slower, and the possibility of epidemic is reduced.

Selective immunization in a hierarchic network is straightforward. The most effective strategy is to immunize a set of nodes with the most number of neighbors since these nodes often correspond to root nodes of sizeable subtrees. The effect of immunizing these nodes is equivalent to fragmenting the network into smaller subnets, and viral propagation is then confined to individual subnets. Our results also showed that a strategically placed 1% immunity in a hierarchic topology is sufficient hamper or even thwart many virus attacks.

Selecting the appropriate set of nodes to immunize in the cluster topology is more challenging. The two target sets we selected both yielded a slower infection rate than random immunity. However, there is a trade-off involved in the selection strategy's effectiveness. Selection based on the node reinfection rate (target set #1) consistently and significantly prevented more epidemics from occurring, at the cost of a higher rate of propagation in the case of epidemics. Selection based on the number of links, giving inter-cluster links more weight (target set #2), did a far better job in slowing the rate of propagation when epidemics

occurred, but was not as effective in stopping epidemics from occurring. In any case, compared to the hierarchic model, a larger immunized population is required in the cluster topology to achieve a similar effect—a 5% immunity produced, on average, a 46% reduction in the initial infection growth, in contrast to the 38% reduction created by a 1% immunity in the hierarchic case.

In general, it is an encouraging result that selective immunization outperformed random immunization, particularly in the absence of dynamic detection and defense mechanisms. Rather than focusing on the size of the immunized population, better results can be achieved by carefully selecting the individuals to be immunized, and then ensuring that those nodes are properly immunized.

What these simulations did not tell us is how to generalize the observations made here to other kinds of network topologies. Clearly, the underlying topology has a substantial impact on the immunization strategy. A comprehensive understanding of the role of topology and to what degree it impacts the immunization decision will be possible only when more comprehensive analytical models are developed to capture the essence of the underlying topology and the infection characteristics.

Nevertheless, studies such as this are helpful in many ways. First, this study produced results based on which some general statements can be made about the effect of immunity, both random and selective. Even in cases where the results are intuitive and produced little surprise, it is still beneficial to confirm intuitions with statistical results and to express them in numbers that can be easily compared and understood. Second, it provided a starting point where analytical modeling can use to instrument its perspectives and verify its assumptions.

## 8.  Conclusions

In this paper we have presented the results of a simulation study on the characteristics of viral propagation in computer networks. The study was carried out as part of an ongoing effort to identify characteristics of infection that might be used to detect and treat infections while they are underway. Two network topologies were considered, and the effect of selective immunity was investigated.

As an approach to defending against viral infections, immunization is well-understood in the classical epidemiology sense. In the computational realm, however, it has not been examined closely. We investigated immunization as a potential defense mechanism, and showed that in certain topologies, a relatively small number of strategically placed immune nodes can have a significant effect on viral propagation.

The results obtained in this study only cover a small range of the possible investigations that might be conducted. Some of the conclusions drawn are preliminary and much work still remains before a comprehensive understanding of viral propagation in large networks can be obtained.

## 9.  Acknowledgements

## 10. References

[1]     M. A. Bauer, R. B. Bunt, A. El Rayess, P. J. Finnigan, T. Kunz, H. L. Lutfiyya, A. D. Marshall, P. Martin, G. M. Oster, W. Powley, J. Rolia, D. Taylor, and M. Woodside, "Services Supporting Management of Distributed Applications and Systems", *IBM Systems Journal*, Vol. 36 No. 4, pp. 508-526, 1997.

[2]     F. Cohen, "Computer Viruses: Theory and Experiments", *Computers & security*, Vol. 6, pp. 22-35, February 1987.

[3]     M. A. Hiltunen and R. D. Schlichting, "Adaptive Distributed and Fault-Tolerant Systems", *International Journal of Computer Systems and Engineering*, Vol. 11 No. 5, pp. 125-133, 1995.

[4]     J. O. Kephat and S. R. White, "Directed-graph Epidemiological Models of Computer Viruses", *Proceedings of the 1991 IEEE Computer Society Symposium on Research in Security and Privacy*, pp. 343-359, 1991.

[5]     J. O. Kephart, S. R. White, and Chess, "Computers and Epidemiology", *IEEE Spectrum*, May 1993.

[6]     J. Knight, M. Elder, and X. Du, "Error Recovery in Critical Infrastructure Systems", *Proceedings: Computer Security, Dependability, and Assurance Workshops 1998*, IEEE Computer Society Press, Los Alamitos, CA, pp. 49-71, 1999.

[7]     J. Knight, M. Elder, J. Flinn, and P. Marx, "Summaries of Three Critical Infrastructure Applications", Technical Report CS-97-27, Department of Computer Science, University of Virginia, November 1997.

[8]     M-J Lin, A. Ricciardi, and K. Marzullo, "A New Model for Availability in the Face of Self-Propagating Attacks", *Proceedings of the New Security Paradigms Workshop*, Charlottesville, VA, September, 1998.

[9]     C. Nachenberg, "Computer Virus – Coevolution", *Communications of the ACM*, Vol. 40 No. 1, pp. 46-51, January 1997.

[10]    NCSA 1997 Computer Virus Prevalence Survey. *National Computer Security Association*, 1997.

[11]    ICSA 1998 Computer Virus Prevalence Survey. *ICSA*, 1998.