Modeling and Simulations of TCP MANET Worms

April 12, 2006

Abstract

Mobile Ad-hoc Networks (MANETs) are used for emergency situations like disaster-relief, military applications, emergency medical situations. These applications make MANETs attractive targets for cyber-attacks and make the development of counter-measures paramount.

The study of worm behavior is critical to the design of effective counter measures in (MANETs) environment. This paper studies the behavior of TCP based worms in MANETs. We develop an analytical model for the worm spread of TCP worms in the MANETs environment that accounts for payload-size, bandwidth-sharing, radio range, nodal density and several other parameters specific for MANET topologies. We also present numerical solutions for the model and verify the results using packet-level simulations.

The results show that the analytical model developed here matches the results of the packet-level simulation in most cases except when the probability of buffer overflow is high.

1 Introduction

Internet worms are programs that can replicate and propagate on the Internet by exploiting security flaws in some services. Once resident on a host computer, the worm implements a search strategy for the selection of future host targets. This involves an algorithm for choosing a host IP address from the total IP address range. Various target host selection algorithms have been discussed, analyzed and found within Internet worms. Once a target host is selected, the worm attempts to transfer its payload to the target. If successful, these newly infected hosts continue the infection process. This results in the well known exponential growth of infected hosts as reflected in the Standard Epidemic Model [1] [5].

Worms have demonstrated that they can cause serious damage to the economy. In 2001 the Code Red worm infected 360,000 hosts in 14 hours [12]. The direct costs of recovering from this epidemic (including subsequent strains of Code Red) have been estimated to be in excess of $2.6 billion [13]. In 2002 the Slammer worm infected 90% of its vulnerable hosts (75,000) in less than 10 minutes [11], the estimated loss was about $1 billion [21]. In 2003 the Blaster worm was estimated to have infected more than 500,000 systems worldwide and the cost to North American companies was $1.3 billion [21]. However, a more recent report showed that the number of infections was actually between 8 million and 16 million systems [10]. In 2004 the Witty worm had a malicious payload that targeted firewalls. Not only did it spread to additional hosts, but it also formatted a portion of the hard drive of the infected host [18].

According to [20] the severity of the worm threat has grown recently with (i) the increasing degree to which the Internet has become part of a nation’s critical infrastructure, and (ii) the recent, widely publicized introduction of very large, very rapidly spreading Internet worms, such that this technique is likely to be particularly current in the minds of attackers.

The effect of worms on a Mobile Ad-hoc Network (MANET) topology is much more serious, due to the resource constrained nature of such a network. According to the recent DARPA BAA Defense Against Cyber Attacks on MANETS [6], “One of the most severe cyber threats is expected to be worms with arbitrary payload that can infect and saturate MANET based networks on the order of seconds.”

A MANET is a self-configuring network of mobile nodes that act as routers and hosts, these nodes are connected by wireless links. The topology of MANETs is arbitrary and dynamic. MANETs can operate in a standalone fashion, or may be connected to the Internet. Because of the fact that MANETs require minimal configuration and are quickly and easily deployed, they are suitable for emergency situations like disaster-relief, military applications, emergency medical situations. These applications make MANETs attractive targets for cyber-attacks and make the development of counter-measures paramount. As such, we have embarked on a program to investigate worm propagation and mitigation in MANETs [3] [4].

Here we extend our prior analysis to the investigation of TCP-based worms; our previous work focused on UDP-based worms. Our investigations involve extensive, high fidelity simulations of TCP-based worms using the Georgia Tech Network Simulator (GTNetS) [17]. Further, we develop and investigate a simple, first model of the TCP worm propagation in MANETs. The model results are
compared against the simulation experiments. The simple model is an extension to the Standard Epidemic Model, which is

$$\frac{di(t)}{dt} = \beta i(t)[1 - i(t)]$$  \hspace{0.5cm} (1)$$

where $i(t)$ is the probability of nodal infection at time $t$, and $\beta$ is the rate at which a given infected node is successful in infecting other susceptible nodes.

For a simple, UDP-based flash worm, $\beta$ is generally set to the product of: a) the inverse of the UDP packet transmission time out the communication interface of the infected host, times b) the probability that the packet is addressed to a susceptible host, times c) the probability that the UDP packet is not lost in transit due to network congestion. The first term is simply proportional to the communications line speed divided by the size of the single UDP packet. The second term is usually taken as the ratio of the susceptible host population divided by the entire address space, assuming a worm which implements a random address search strategy. Zou et. al. [2], have suggested that the last term be approximated by $[1 - \frac{\eta}{N(t)}]^n$ where $\eta$ is a fitting parameter. This term estimates the probability of packet receipt success at the susceptible host under conditions of packet losses within the network.

For a TCP-based worm, $\beta$ must assume a somewhat different form for two reasons: a) the rate at which an infected node can transmit the worm to other nodes is related to the number of simultaneous TCP connections divided by the mean time for a TCP connection to transmit the worm payload, and b) network congestion does not decrease the probability of receipt of the payload at the susceptible host, but instead congestion slows the time to transmit the payload due to bandwidth competition. Bandwidth competition can slow the TCP transmit times due to increased bandwidth sharing and increased packet losses. Therefore, we investigate a TCP model which assumes a lossless network, i.e., we assume the nodal buffers are large enough to hold all packets in transit across the network. The model we investigate accounts for bandwidth competition through accounting for bandwidth sharing under high loads. Future work will investigate the impact of packet loss on the spreading rate of TCP-based worms.

The rest of the paper is organized as follows. Section 2 gives a brief overview of related work. Section 3 describes the worm model that we are using. Section 4 describes the simulation experiments that were conducted. Section 4.1 provides the results and their discussion. Finally the conclusions of the paper are provided in the end.

## 2 Related Work

Several studies were carried out to analyze and model the propagation of computer worms in digital communication networks. In [22] the authors present the different kinds of worms depending on their scanning strategies, worm carrier mechanism, possible payload and plausible attackers who would employ such a worm. In [20] the authors provide an extensive investigation into the mechanisms of worm propagation and their performance. They also provide some improvements for the worms and the effort needed to mitigate worm propagation throughout the Internet.

Zou et. al. [2], studied the Code Red worm outbreak and provided an analysis of its propagation by accounting for two factors: one is the dynamic countermeasures taken by ISPs and users; the other is the slowed down worm infection rate due to the rampant propagation of the worm that caused congestion and troubles to some routers. They derived a general Internet worm model called the two-factor worm model.

The most relevant work to this paper is [3] and [4]. In [3] the authors investigated the impact of communications and mobility effects on worm propagation mechanisms in MANETs, where they found that network delays and channel congestion had a large impact on the worm propagation behavior and they also provided a set of relatively simple analytical models that reproduced these communication. In [4] the authors discussed the effect of mitigation techniques on the worm spread in MANETs and provided analytical models and simulation experiments to validate their findings. The authors represented the mitigation technologies as having a constant detection rate represented by a lifetime parameter of the worm, after which the worm dies and stops infection of other hosts. Also their study was limited to UDP worms. In this paper we extend their simulation models to be used in GTNetS and to include TCP style worms in MANETs as well.

## 3 The TCP Worm Propagation Model

We first investigate the performance of TCP worm propagation through a relatively straightforward propagation model. We will then present our simulation studies. We assume that the nodal buffers are large enough that packet discards rarely happen. Hence, the TCP transmission times decrease as the probability of infection increase solely due to the increased bandwidth sharing between transmitting infected nodes. In future studies we will relax this assumption and investigate the impact of packet losses on the TCP worm propagation performance.

Assume that the nodes in the Mobile Ad-Hoc Network (MANET) and uniformly distributed throughout the grid. Define:

Then, assuming perfect sharing of channel bandwidth during the transmission of a TCP worm payload, the channel bandwidth is partitioned equally to each of the neigh-
bandwidth of the radio channel (Bps)
\(d\) = nodal density (nodes/sq.meter)
\(r\) = radio range of the channel (meters)
\(n(t)\) = mean number of infected neighbors of a given infected node in the MANET
\(\alpha\) = (zero load) TCP throughput in proportion to the channel bandwidth

\[ b = \text{bandwidth of the radio channel (Bps)} \]
\[ d = \text{nodal density (nodes/sq.meter)} \]
\[ r = \text{radio range of the channel (meters)} \]
\[ n(t) = \text{mean number of infected neighbors of a given infected node in the MANET} \]
\[ \alpha = \text{(zero load) TCP throughput in proportion to the channel bandwidth} \]

boring TCP transmissions. Thus, the portion of the channel bandwidth available to each TCP transmission is

\[ \bar{b} = \frac{b}{n(t) + 1} \]  \quad (2)

where \(n(t)\) is the mean number of infected neighbors at time \(t\). Assuming uniform placement of nodes in the MANET, we write

\[ \bar{b} = \frac{b}{(\pi dr^2 i(t) + 1)} \]  \quad (3)

where \(i(t)\) is the probability of nodal infection at time \(t\). So, we consider each infected node seeing a time varying available bandwidth due to increasing probability of infection over time. The Standard Epidemic Model becomes

\[ \frac{di(t)}{dt} = \beta(t)i(t)[1 - i(t)] \]  \quad (4)

where \(\beta\) is now time dependent.

For a single threaded TCP operation, let \(\tau_{tcp}\) be the average TCP throughput for the transmission of the worm payload. Then, the time to transmit the TCP worm payload is

\[ t_{tcp} = \frac{P_w}{\tau_{tcp}} \]  \quad (5)

where \(P_w\) is the payload size of the worm. From the above, modified Standard Epidemic Model, \(\beta(t)\) is

\[ \beta(t) = \frac{1}{t_{tcp}(t)} = \frac{\tau_{tcp}(t)}{P_w} \]  \quad (6)

and

\[ \tau_{tcp}(t) = \alpha\bar{b}(t) \]  \quad (7)

where \(\alpha\) is the proportion of the available channel bandwidth that a TCP connection is able to obtain under zero load situations [7] [14] and [16]. Hence, \(\alpha\) is a function of the mean number of hops across the MANET, the packet size, etc. But, for now, we assume that \(\alpha\) is a constant with respect to time. Inserting these expressions into the above Epidemic Model yields

\[ \frac{di(t)}{dt} = \left[ \frac{\alpha b}{P_w(1 + \pi dr^2 i(t))} \right] i(t)[1 - i(t)] \]  \quad (8)

This expression is comparable to the expression for the UDP-based Flash worm proposed by Zou, et. al., [2]

\[ \frac{di(t)}{dt} = \beta(1 - i(t))\beta i(t)[1 - i(t)] \]  \quad (9)

These equations are different because they consider the different effects of network congestion. Our TCP worm model is predicting a diminishing TCP throughput due to channel sharing at higher infection probabilities. While Zou’s model is predicting a decreasing probability of end-to-end packet delivery success due to buffer overflow under increased network competition at higher infection probabilities.

Note, our TCP model makes the following assumptions:

- A single threaded TCP operation.
- A lossless TCP throughput model based upon channel sharing and large buffers.
- A throughput model for the time to transmit the TCP worm payload, which assumes a large payload in relation to the TCP segment size in the network.
- Sufficient nodal density and radio transmission range to maintain connectivity across the MANET cluster.
- No back-ground traffic.

We plan to investigate improvements to our model relaxing these assumptions in future studies.

### 3.1 ANALYTIC RESULTS

In this section we investigate the analytic solution to our simple TCP worm model given in Eq.(8). Following the method of factoring used to solve the Standard Epidemic Model of Eq.(1), we rewrite Eq.(8) as

\[ \frac{di(t)}{dt} = \gamma i(t) \left[ \frac{1 - i(t)}{1 + \theta i(t)} \right] \]  \quad (10)

where \(\gamma = \frac{\alpha b}{P_w}\) and \(\theta = \pi dr^2\). We factor this expression into

\[ (1 + \theta i(t)) \left[ \frac{1}{i(t)} + \frac{1}{1 - i(t)} \right] di(t) = \gamma dt \]  \quad (11)

Integrating both side of this equation and rearranging terms yields

\[ \frac{i(t)}{i(\alpha)} \left[ \frac{1 - i(\alpha)}{1 - i(t)} \right]^{1+\theta} = e^{-\gamma t} \]  \quad (12)

or

\[ \frac{i(t)}{[1 - i(t)]^{1+\theta}} = g(i_0)e^{-\gamma t} \]  \quad (13)

where

\[ g(i_0) = \left[ \frac{i_0}{1 - i_0} \right]^{1+\theta} \]  \quad (14)

Here \(i_0 = i(t = 0)\). This is as far as we can get in writing the explicit solution to our TCP model. For general values of \(\theta\), we cannot solve this expression for \(i(t)\). However,
when $\theta = 1, 2$ or $3$ this expression represents a quadratic, cubic or quartic expression in $i(t)$, respectively. For other values of $\theta$, i.e., $\theta$ real or $\theta > 3$, no known explicit solutions exist.

For $\theta = 0$, Eq.(13) reduces to

$$i(t) = \frac{g(i_0)e^{\gamma t}}{[1 + g(i_0)e^{\gamma t}]}$$  \hspace{1cm} (15)

This is the well known solution to the Standard Epidemic Model. For $\theta = 1$, Eq.(13) reduces to

$$i(t) = \frac{1 + 2g(i_0)e^{\gamma t} \pm \sqrt{1 + 4g(i_0)e^{\gamma t}}}{2g(i_0)e^{\gamma t}}$$ \hspace{1cm} (16)

The upper solution, obtained by choosing the upper plus sign, is a non-physical solution resulting in the probability of infection exceeding unity. The lower, i.e., minus sign, solution is the physical solution. In comparison with the previous $\theta = 0$ solution, this expression demonstrates the slowing effects of bandwidth competition in the network. Solutions are also possible for $\theta = 2$ and $3$, but due to space limitations we do not discuss these here.

We can determine the asymptotic behavior of $i(t)$ as $t \to \infty$ from Eq.(13). The right hand side of Eq.(13) clearly approaches infinity as $t \to \infty$. This implies that the left hand side of Eq.(13) also approaches infinity, which can only happen if $i(t \to \infty) = 1$. Also, by writing $i(t) \approx 1 - \epsilon$ where for large $t$, $\epsilon << 1$, we can perform an expansion of Eq.(13) in terms of $\epsilon$. This yields the follow expression for $i(t \to \infty)$.

$$\lim_{t \to \infty} i(t) \approx 1 - g^{-1/(1+\theta)}e^{-\gamma t/(1+\theta)} + ...$$ \hspace{1cm} (17)

Clearly, the larger $\theta$ is, the slower is the convergence of $i(t)$ toward unity, reflecting that fact that greater bandwidth competition is slowing the propagation of the TCP-based worm.

Finally, we can analyze Eq.(13) in the context of determining the time for the infection to reach 50% of the population, $t_{1/2}$. Substituting $i(t_{1/2}) = 1/2$ into Eq.(13), we get

$$t_{1/2} = \gamma^{-1}(\theta n 2 - ln g)$$ \hspace{1cm} (18)

where $\gamma = \alpha b / P_w$ which is the mean time for a host to transmit the worm payload under conditions of no bandwidth competition. In time units of $\gamma$, we expand out the above expression to yield

$$t'_{1/2} = \pi d r^2 \ln 2 - \ln \left( \frac{i_0}{1 - i_0} \right)$$ \hspace{1cm} (19)

So, for a given initial condition, $t'_{1/2}$ increases linearly with the mean number of neighbors within radio range, e.g., $d r v^2$.

\[\text{Figure 1: The Simple TCP Model results (top) for various worm payload sizes compared with simulation results (bottom).}\]

\[\text{3.2 NUMERICAL RESULTS}\]

Even though a general, explicit solution to Eq.(8) is not known, it is certainly easy to numerically integrate the expression for $i(t)$. Therefore, we wrote a PERL script to numerically integrate Eq.(8). We integrated the expression for $i(t)$ for a number of cases, based upon the parameter set given in Table 1. These correspond to a set of simulation studies discussed in the next section. For our purposes, Eq.(8) contains a fitting parameter, i.e., $\alpha$, which is interpreted as the proportion of available channel bandwidth that a TCP connection is able to obtain under zero load situations. It is well known that TCP throughput decreases exponentially as a function of the number of hops in the wireless network due to self interference, see, e.g., [16]. In fact in typical situations, it is not unusual to find that $\alpha < 0.1$ is common. We choose $\alpha$ in order to fit the simulation results for the smaller payload cases.

Figure 1 shows the results of the simple TCP worm model for various TCP worm payload sizes, ranging from a low of 400 bytes to a high of four million bytes. For each run, the other parameters for the model, as identified in Table 1 remained fixed. The fitting parameter was set as $\alpha = 0.025$ to generate these numerical results. It can be
seen that the simple model does a good job in qualitatively representing the simulation results. However, for larger TCP-based worm payloads, the model results are not as accurate. We suspect the reason for this is that the probability of buffer overflow in the MANET increases as we increase the size of the worm payload. Since our simple TCP model assumes infinite buffers in the MANET, the impact of packet loss is not captured. We will investigate the impact of packet losses on TCP-worm propagation in future work.

4 Simulation

We present the simulation experiments conducted using Georgia Tech Network Simulator (GTNetS). GTNetS has an application that models the spread of a computer worm. The worm is designed as an application that exists on all susceptible nodes, which is listening on a specific port for incoming packets. When the worm application receives the infectious packet it is activated and starts choosing targets to send infectious packets to them.

There are different models for worms as discussed in [19], the models include a number of parameters that specify the behavior of the worm:

- **Transport protocol**: The underlying transport protocol used by the worm, which can be either UDP or TCP. UDP worms do not wait for any acknowledgment from the target, while the TCP worm requires a three-way handshake (SYN/SYN-ACK/ACK) before it can send its payload.

- **Infection length**: The size of the exploitation data that the worm needs to send to a host in order to infect it.

- **Infection port**: The transport layer port that exhibits the security vulnerability that is to be exploited.

- **Target vector**: The algorithm used by the worm to determine the IP address of a new victim. This can be either uniform, local preference or sequential scanning.

- **Scan rate**: The rate at which UDP worms send infection packets.

- **Scan range**: The range of addresses the worm chooses from.

- **Connections**: The number of simultaneous connection attempts used by TCP worms.

The MANET nodes are arranged in a rectangular grid, where they are uniformly placed across the grid. For our experiments the nodes are not moving. The effect of mobility on our simulation results is a subject of future work.

Table 1 define our baseline MANET TCP worm simulation model parameters.

Figure 2 shows the results of the TCP worm model and simulations after varying the transmission rate of the wireless channel, ranging from 0.1 Mbps to 2.0 Mbps. It is clear from the figures that as the transmission rate decreases the worm spread flattens due to saturation of the network and the infection growth tends to become linear. The model still does a good job in qualitatively representing the simulation results except for the low transmission rate cases. We again suspect that this is due to increased buffer overflow at low transmission ranges.

4.1 Results

In our experiments we create the MANET topology and set the simulation parameters according to Table 1 baseline case and then start the worm infection in the initial population and measure the spread against time for varying one parameter at a time and compare the average of the results of 30 runs with the output of the simple TCP worm model described in section 3.

Figure 2 shows the results of the TCP worm model and simulations after varying the transmission rate of the wireless channel, ranging from 0.1 Mbps to 2.0 Mbps. It is clear from the figures that as the transmission rate decreases the worm spread flattens due to saturation of the network and the infection growth tends to become linear. The model still does a good job in qualitatively representing the simulation results except for the low transmission rate cases. We again suspect that this is due to increased buffer overflow at low transmission ranges.

Figure 3 shows the results of the TCP worm model and simulations after varying the initial infected population size, ranging from 1 to 20 nodes. The model does a good job in qualitatively representing the simulation results. The results show as expected that with increasing the initial population, the worm spread rate is increased.
Parameter Description | Range       | Base Case |
----------------------|-------------|-----------|
Number of hosts       | 50-150      | 50        |
Initial population size | 1-20       | 1         |
Transport layer protocol | TCP       | TCP      |
Simultaneous connections | 1          | 1         |
Range of vulnerable Addresses | 50 - 500  | 50        |
Transmission Rate (Mbps) | 0.1 - 2.0  | 2.0       |
Time delay(μ seconds) | 2           | 2         |
Transmission range (m) | 100 - 500  | 250       |
Area of topology (m²) | 1000        | 1000      |
Routing protocol       | DNVR, DSR, AODV | AODV |
Payload size (kbytes)  | 0.4 - 4000  | 4         |
Simulation time (seconds) | 200        | 200       |

Table 1: Parameter definition for simulation experiments

The increase is not significant because this is a low-scan
ing worm (only one simultaneous connection).

Figure 4 shows the results of the TCP worm model and simulations after varying the radio range of the nodes, ranging from 100 meters to 500 meters. It is clear from the simulation figures that for low radio ranges some of the nodes cannot communicate with each other and therefore the infection rate is very low. As the radio range increases to 250 meters the infection spread improves, afterwards the increase in radio range has a negative effect on the worm spread due to more bandwidth competition between nodes. The TCP worm model captures the effect of bandwidth competition but not the fact that at low radio ranges there will be effectively no communication.

Figure 5 shows the results of the TCP worm model and simulations after varying the number of nodes (nodal density). The simulation results show the effect of increased contention with increasing nodal density, which results in increase in packet drop rate. The TCP worm model does not match these results due to the fact that packet drops are ignored in our assumptions for this model. We will investigate the effect of packet drops on our model in future work.

Figure 6 shows the results of the simulations after varying the routing protocol used. Three different routing protocols were used in these experiments; DNVR [9], DSR [8], and AODV [15]. It is clear from the figures that for our experiments there is no significant difference in the performance of these routing protocols.

5 Conclusion

We have presented a study of TCP worm propagation in MANETs. We investigated the impact of payload size, channel bandwidth, initial infection probabilities, radio range and routing protocols on the effectiveness of the worm propagation. Previous studies have proposed analytic models of UDP-based worm propagation in MANETs. Here, we develop a simple, analytic model of TCP-based worm propagation in MANETs. The models compare reasonably well to our simulation results. However, at large payload sizes and high probabilities of worm infection, the agreement between model and simulation results get worse. We suspect this is due to a high probability of packet losses in the MANET; an effect that our simple TCP model does not currently take into account. This is work for our future investigations.

We believe that our studies will aid in the design of efficient counter measures for worm attacks in MANETs.

All these factors need to be addressed when designing an efficient mitigation technique. Further studies of worm propagation and mitigation in the challenging networking environment afforded by MANETs is required. In future studies we hope to further quantify the behavior of TCP-based worms and to investigate the efficiency of more specific worm mitigation technologies.

References


Figure 3: The Simple TCP Model results (top) for various initial population size with simulation results (bottom).

Figure 4: The Simple TCP Model results (top) for various radio ranges with simulation results (bottom).


Figure 5: The Simple TCP Model results (top) for various number of nodes with simulation results (bottom).

Figure 6: The simulation results for varying the routing protocol


