논문 2010-47CI-5-9

스캐닝 트래픽의 프로파일링을 통한 인터넷 웜 확산 모델링 기법

(An Approach for Worm Propagation Modeling using Scanning Traffic Profiling)

손 태 식*, 구 본 현**

(Taeshik Shon and Bonhyun Koo)

요 약

최근에는 일반적인 웜의 확산 특성을 분석하여 이를 이용해 웜으로 의심되는 트래픽을 사전에 차단하는 방법이 활발히 연구되고 있다. 그러나 아직 구체적인 모델링 방법에 대한 명확한 기준이 없으며, 급변하는 웜의 특성을 반영한 사전 탐지가 쉽지 않은 실정이다. 이에 본 논문에서는 현재 제시되어 있는 웜의 탐지 모델들을 분석하였고, 웜의 확산 과정에 있어서 새로운 감염대상을 찾기 위한 스캐닝 방법이 가장 주요한 요소임을 확인하였다. 이를 바탕으로, Lovgate, Blaster, Sasser 3가지 웜의 확산 과정시 스캐닝 방법을 분석하였고, 분석한 내용을 바탕으로 각각의 프로파일을 구축하였다. 구축된 스캐닝 프로파일 기법을 적용한 웜의 탐지 모델을 시뮬레이션 함으로써 실제 웜의 확산 과정을 예측해 본다. 또한 제안 기법의 검증을 위해 실제 테스트 베드를 구축하고 제안 모델을 적용하여 탐지 가능성을 확인하였다.

Abstract

Recently, the early detection and prevention of worm research is mainly studying based on the analysis of generalized worm propagation property. However, it is not easy to do Worm early detection with its attributes because the modeling method for Worm propagation is vague and not specified yet. Worm scanning method is exceedingly effect to Worm propagation process. This paper describes a modeling method and its simulations to estimate various worm growth patterns and their corresponding propagation algorithms. It also tests and varies the impact of various improvements, starting from a trivial simulation of worm propagation and the underlying network infrastructure. It attempts to determine the theoretical maximum propagation speed of worms and how it can be achieved. Moreover, we present the feasibility of the proposed model based on real testbed for verification.

Keywords: Internet Worm, Propagation Modeling, Scanning, Network Security

I. 서 론

인터넷이 널리 확산됨과 함께 악의적인 코드에 의한 공격 또한 다양한 형태로 나타나고 있다. 특히 2003년 1월 25일에 국내에서 발생한 인터넷 대란 사태와 2009년 7월 7일 우리나라 및 미국을 대상으로 한 인터넷 웜 기반의 DDoS 사태는 그 위력을 잘 보여 준 예라 할 수 있겠다[1]. 네트워크 전체를 짧은 시간 안에 마비시키는 악성코드를 흔히 웜(Worm)이라 일컫는데 이는 자동화된 방법으로 매우 빠른 속도로 자신을 복제하여 새로운 감염대상을 찾아 끊임없이 퍼져가므로 초고속 인터넷이 잘 발달한 나라일수록 그 피해규모가 더 커질 수 있다. 그 동안 웜의 전파를 차단하기 위해 다양한 노력이 기울여져 왔으며 이러한 방법으로는 트래픽 분석, 허니팟/다크 네트워크, 시그너쳐 적용 등이 있다[2~4]. 시그너쳐 기법의 경우 일단 웜이 출현하게 되면 웜 자체의 특성

* 정회원, ** 정회원-교신저자, 삼성전자 DMC R&D 센터
(DMC R&D Center, Samsung Electronics)
접수일자: 2010년3월7일, 수정완료일: 2010년8월31일

을 분석하여 이를 이용해 웜을 추출하고 삭제할 수 있다. 그러나 이 방법은 이미 웜의 피해를 입은 후에 치료하는 것이고 또한 끊임없이 새로운 웜이 생겨나고 있으므로 그 효과가 제한적일 수밖에 없다. 최근에는 일반적인 웜의 전파 특성을 분석하여 이를 이용해 웜으로 의심되는 트래픽을 사전에 차단하는 트래픽 기반 분석 방법이 활발히 연구되고 있다. 그러나 아직 정확한 모델링이 되지 못했고 따라서 웜에 대한 사전 방어 역시 쉽지 않은 실정이다. 허니팟은 악성 또는 웜/바이러스 등으로 예상되는 패킷을 의도적으로 유인하는 방법이며, 다크 네트워크 기법은 존재하지 주소를 목적지로 하는 패킷들을 모니터링 하는 기법이다. 이러한 허니팟은 웜이나 바이러스의 동작에 대해서 매우 상세한 행위들을 파악할 수 있는 장점이 있는 반면, 그 설정이 복잡하고 다크 네트워크는 수동적인 수집기법이지만 추후 분석을 통해 다양한 정보를 얻어 낼 수 있다. 한편으로 국내외에서 웜에 대한 다양한 확산 시뮬레이션에 대한 연구가 진행되어지고 있다[4~5]. 하지만 현실과 동일한 실험망 구축을 통해 웜의 확산을 관찰하는 것은 시간적·경제적인 비용으로 인해 불가능하다고 할 수 있다.

그러므로 본 논문에서는 기존 웜 확산 시뮬레이션에 대한 문제점을 해결하기 위해 실제 웜의 행동과정을 파악에 가장 효과적인 웜의 스캐닝 기법을 분석하고 이 분석에 따른 웜 스캐닝 트래픽 기반 프로파일을 통한 확산 탐지 모델을 제안한다. 또한 제안된 이론적인 프로파일을 바탕으로 본 논문에서는 먼저 웜의 전파과정을 물리적인 테스트베드에서 수행해본 후 현재 제시되어 있는 웜의 모델을 시뮬레이션 하여 제안 모델의 타당성을 검증한다.

본 논문의 구성은 다음과 같다. Ⅱ장에서는 관련 연구에 대한 소개, Ⅲ장에서는 본 논문에서 제안하는 모델을 설명한다. Ⅳ장에서는 제안 모델을 검증하기 위한 테스트 베드 구축 및 실험 방법/결과 등을 설명하며 Ⅴ장에서 본 논문의 결론 및 향후 연구 방향을 제시한다.

Ⅱ. 관련 연구

1. 네트워크 패킷 기반 웜 시뮬레이션 연구

NWS(Network Worm Simulation System)는 Bruce Ediger에 의해 만들어진 네트워크 웜 시뮬레이션 시스템으로서 웜들의 시뮬레이션을 통해 그들의 전파과정에 따른 감염된 호스트들의 효과를 시뮬레이션하기 위한 프레임워크를 제공한다[6]. NWS는 Perl로 작성되었고 감염된 호스트가 실행되어졌을 때, 사용자 정의 Perl 코드를 통해 피해에 대해 보다 사실적인 묘사의 설정이 가능한 점이다. 하지만, NWS 시스템에서는 라우터와 같은 네트워크 장비를 고려하지 않음으로써 웜/바이러스가 노드를 획득하기 위해 한 노드에서 다른 노드로 쉽게 통과할 수 있다. SSF.App.Worm[7]은 네트워크 패킷 기반 시뮬레이터 중 확장 가능한 시뮬레이션 프레임워크 네트워크 모델을 제공하며, CodeRed와 SQL Slammer와 같은 인터넷상의 특정 웜들에 대한 전파를 모델링이 가능한 것으로 알려져 있다. SSF.App.Worm은 취약한 호스트들의 전체 개체 수, 초기 감염된 호스트들의 수, 감염으로부터 호스트들이 복구됨을 지정할 수 있는 이동 함수와 같은 시뮬레이션 상에 설정할 수 있는 많은 구성 가능한 파라미터들이 있지만, SSFNet을 이용하여 네트워크 공격과 이에 따른 네트워크의 행동을 시뮬레이션하기에는 아직 보완되어야 할 사항이 있다. 예를 들면, 패킷 캡쳐 라이브러리에 해당하는 클래스가 없어서 현실적으로 공격용 프로그램을 작성하기 어렵다. 또한 SSFNet에 보안용 시스템을 모델링 하는 구성요소를 추가하여 방화벽과 같은 보안시스템이 갖추어진 네트워크를 시뮬레이션 할 수 있어야 한다.

2. 수학적 역학 모델 기반 기존 연구

Epidemic 모델을 이용한 모델링 방법은 원래 생물학 분야에서 생물학적 바이러스 등의 감염을 모델링 할 때 사용되어온 모델링 방법이다. SI Epidemic Model은 한 번 감염되면 영구히 감염 상태로 존재한다[8]. 즉, 호스트의 상태 변화는 "Susceptible → Infectious"만 존재한다. 호스트는 웜에 감염될 수 있는 취약한 상태이거나, 감염되어 웜을 전염시킬 수 있는 상태중 하나의 상태를 가진다. 호스트의 모집단 크기는 일정하게 고정되어 있으며, 시간 변화에 따른 감염 호스트 수의 변화를 나타내는 감염 공식은 미분 방정식으로 표현된다. 즉 감염 호스트 수의 변화는 감염률, 감염 호스트 수, 취약 호스트 수 등에 비례한다. I(t)는 t 시간 때, 감염 호스트의 수를 말하며, S(t)는 t 시간 때, 취약 호스트의 수, β는 웜의 감염률이며, 확산 공식은 아래와 같이 표현한다.

$$\frac{dI(t)}{dt} = \beta I(t)[N - I(t)]$$

감염 노드의 비율은 다음과 같이 표현하였다.

표 1. RCS 모델상의 시뮬레이션 변수
Table 1. Simulation Parameters on RCS models

변수	설명
N	인터넷에 연결 되어 있는 취약점을 가진 총 호스트 수
K	웜 발생 시 감염된 호스트가 시간당 접촉(스캔) 할 수 있는 취약점을 가지는 호스트 수
$a(t)$	t에 의해 접촉 되는 취약한 호스트의 비율
$N * a(t)$	시간당 K 비율로 스캔 하는 각 호스트들의 의해 감염된 호스트의 수
$K*(1-a(t))$	감염되지 않은 호스트의 수
N	t 시간 동안 스캔에 의해 접촉 되는 호스트 수

$$a(t)=\frac{I(t)}{N}, \frac{da(t)}{dt}=Ka(t)[1-a(t)] where K=\beta N$$

RCS(Random Constant Spread)모델은 Staniford 등에 의해 CodeRed 웜에 의해 생성된 데이터를 사용하여 개발 하였다[9]. RCS 모델은 CR2를 기반으로 모델링 되었으며, 웜이 효과적인 랜덤 생성기를 가지고 있다는 가정 하에 실험이 수행되었다. 이 모델은 시스템의 패치 유무, 전원의 작동여부, 네트워크 접속 여부대한 영향은 고려하지 않은 모델이다. 또한 방화벽이나 VPN과 같은 방어 장비에 대해서 고려하지 않았으며, 인터넷 토폴로지는 undirected complete graph로 가정 하였다. RCS 모델에서는 표 1과 같이 시뮬레이션 모델 변수를 정의하고 있으며, 아래와 같이 표현될 수 있다.

$$N=(Na)\times(1-a)dt, \quad Nda=(Na)\times K(1-a)dt$$

$$\frac{da}{dt}=Ka(1-a), \qquad a=\frac{e^{K(t-T)}}{1+e^{K(t-T)}}$$

Kill Signal Model은 The KW Directed Graph SIS Model 모델을 기반으로 실제 생활에서 고려되어야 할 파라미터를 적용한 모델이다[10]. KW 모델을 만들었던 Kephart 와 White는 KW 모델이 실제 환경에서의 바이러스 전파를 정확히 나타내지 못하는 것을 확인하고, 이 모델의 정확도를 높이기 위하여 원인에 해당 하는 부분을 모델에 적용하였다. KW 모델은 바이러스 전파 시 다른 감염 시스템들과의 상관관계를 전혀 고려하지 않고 바이러스는 독립적으로 치료 된다고 가정하였지만, 예를 들어 'A가 바이러스의 감염된 PC의 이상 징후를 확인 하여 B, C, D등의 친구에게 바이러스 체크 및 백신 업데이트 등의 여러 가지 바이러스에 대한 정보를 전달 할 수 있는 경우'에는 적용 되지 않는다.

III. 제안하는 웜 확산 모델링 기법

3.1. 제안하는 스캐닝 프로파일링 기반 모델

제안 기법은 네트워크에서 웜 바이너리 코드 실행을 통한 전파 과정 및 발생 트래픽들의 모니터링을 통해 웜의 프로파일을 구축하고, 이를 바탕으로 웜의 확산 과정 시뮬레이션 하여 전파 과정을 예측하고 탐지에 필요한 정보를 제공 한다. 이러한 스캐닝 프로파일링 기반 제안 모델의 시뮬레이션 과정은 이 후 제시될 이론적 정의 기준을 바탕으로 수행되며, 발생하는 웜에 의한 감염률과 발생 트래픽의 예측이 가능한 특징을 가진다. 따라서 이를 통해 이상 탐지 기반의 침입탐지 시스템 등의 Threshold 기준 방안에 대해 확립할 수 있는 웜의 확산 모델을 제시하고자 한다. 제안 모델에서의 네트워크를 통한 웜 바이너리 코드의 실행 및 전파 과정은 VI장에서 구축한 테스트 베드를 통하여 재현 되었으며 이때 구축한 웜 패킷들의 특성을 아래 표 2와 같은 몇 가지 기준에 따라 분류하여 추출하였다. 추출된 데이터를 바탕으로 각 웜들의 순간 감염률, 평균 감염률 및 웜 트래픽과 정상 트래픽을 구분하는 Threshhold를 정의 할 수 있다.

제안하는 스캐닝 프로파일 모델 기반 시뮬레이션 기법의 장점은 다음과 같다. 첫째, 기존 시뮬레이터기반 웜 모델링의 문제점은 시뮬레이션 대상을 일반적 인터넷 환경 전체를 가정한 환경이므로 시뮬레이션의 결과가 추상적이라는 것에 있다[11]. 따라서 이를 위한 대안으로 수행하고자 하는 대상을 직접 설정함으로 인해 이러한 문제점을 해결할 수 있다. 즉, 실험 환경에 대한 직접적인 네트워크 설정을 통해 실험하고자 하는 대상 네트워크에서의 웜이 전파되고 확산 과정에 대한 결과가 확인 가능하다. 둘째, 네트워크 트래픽에 대한 구현상의 설정을 통해 웜에 의해 발생되어지는 네트워크 트래픽을 확인 할 수 있으며, 이에 따른 라우터상의 Threshold를 정의 할 수 있다. 제안하는 모델 상에서 추후 웜의 확산 과정을 시뮬레이션하기 위해 다음의 두 가지 전제 사항을 가진다. 첫째로 인터넷상의 연결된 통신이 가능한 모든 노드들을 하나의 개체로 가정하여 확산 모델링을 수행하며, 둘째로 일반적인 웜으로 인한 감염 발생 시 사용자에 의한 치료나 제거 등의 행위는 고려하지 않는다. 이와 같이 2가지 가정 하에 웜의 확산 모델에 대한 이론적인 시뮬레이션 모델을 제안한다. 제안 기법에 대한 시뮬레이션 변수는 표 4에서 설명되

표 2. 시뮬레이션 변수 정의
Table 2. The definition table for simulation variable.

변수	설명
TR_{worm}	웜에 의해 발생된 네트워크상의 전체 트래픽량
TH_{worm}	웜에 의해 감염된 하나의 호스트가 발생하는 트래픽량
IT_{worm}	네트워크상의 웜에 의해 발생하는 전체 트래픽량
NT	네트워크상의 발생하는 전체 정상 트래픽량
T	전체 시뮬레이션 시간
D	네트워크상의 지연시간(Delay Time)
I	웜에 감염된 호스트의 수

며, 제안 기법은 SI Epidemic Model의 경우 감염 진행이 지수적(exponential)으로 증가하지만[8], 웜에 의해 작동하는 스캐닝 함수를 적용하면, 웜의 취약 호스트 발견 시에만 새로운 트래픽의 증가가 발생하는 점을 고려해야 한다는 점을 보강하였다.

시뮬레이션 과정동안 웜에 의해 감염되어진 호스트의 수는 다음과 같이 나타낼 수 있다.

$$\cdot \frac{dI}{dt} = \lim_{\Delta t \to 0} \frac{\Delta I}{\Delta t} = \lim_{\Delta t \to 0} \frac{I(\Delta t) - I(0)}{\Delta t} \quad (1)$$

[평균 감염률]
시뮬레이션 과정동안의 평균 감염률은 다음과 같다.

$$\cdot \frac{\Delta I(t)}{\Delta t} = \frac{I(\Delta t) - I(0)}{\Delta t} \quad (2)$$

[순간 감염률]
웜 전파 과정동안의 시간 t 상에서의 순간 감염률은 다음과 같다.

$$\cdot \lim_{\Delta t \to 0} \frac{\Delta I}{\Delta t} = \lim_{\Delta t \to 0} \frac{I(t + \Delta t) - I(t)}{\Delta t} \quad (3)$$

시뮬레이션 과정을 거쳐 발생한 전체 웜 트래픽에 대한 계산 량은 아래와 같다.

$$\cdot TR_{worm} = \sum_{i=1}^{n} TR_{worm_i} \times \frac{\Delta t}{D} \quad (4)$$

이를 통해 감염된 호스트들의 전체 트래픽과 정상적인 Normal 트래픽을 (5), (6)과 같이 유도한다.

$$\cdot \ _{worm} = \sum_{i=1}^{n} TH_{worm_i} \times \frac{\Delta t}{D} \quad (5)$$

$$\cdot NT_{worm} = TR_{worm} - _{worm} \quad (6)$$

이러한 웜이 발생 시키는 트래픽에 대한 라우터 상의 Threshold는 Static Traffic Threshold 와 Dynamic Traffic Threshold의 두 가지로 정의 할 수 있다. Static Traffic Threshold를 이용한 탐지 알고리즘은 다음과 같이 이용되어 질 수 있다.

$$\cdot \left(\frac{TR_{Worm}}{Threshold} \geq 0 \right) - \text{Worm Traffic} \quad (7)$$

$$\cdot \left(\frac{TR_{Worm}}{Threshold} < 0 \right) - \text{Normal Traffic} \quad (8)$$

Dynamic Traffic Threshold는 네트워크의 트래픽에 따라 Threshold를 가변적으로 변경할 수 있다. 예를 들어, 시간에 따라 네트워크상의 평균 트래픽이 낮아질 경우 Dynamic Traffic Threshold 값을 하향 조절 할 수 있다.

IV. 실험 및 결과

4.1. 실험 환경

제안 기법의 검증에 사용될 웜 스캐닝을 보다 현실적인 특성을 반영하여 생성해 내기 위해 실제 테스트 베드를 구축하며, 웜 바이너리코드를 실행하여 실제 웜의 전파과정을 검증한다. 실험환경 구성은 시스코 라우터 2524, 2502와 Windows를 사용하는 Pentium-IV 2.7G PC 2대를 이용하였으며, 각 PC에는 가상 호스트를 2개씩 설정하여 총 6대의 호스트를 연결하여 네트워크를 구축하였다. 실제로 그림 1과 같이 외부 네트워크로부터 물리적인 경로 차단하여 실험에서 발생하는 패킷들

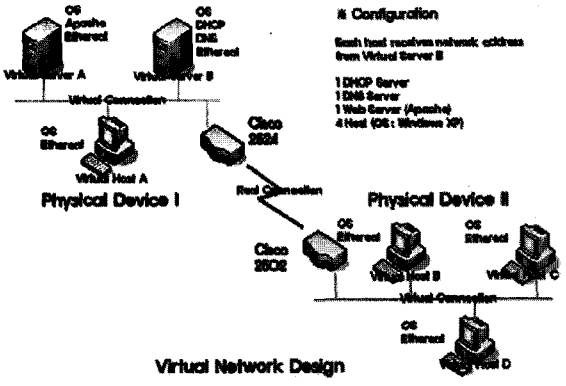

그림 1. 가상 네트워크 실험 장비 구성도
Fig. 1. Virtual Network Design and Deployment.

표 3. 가상 실험 네트워크 구성내용
Table 3. Network configuration for virtual experiment network.

실험 장비	설정 내용	
Cisco 2502	Ethernet0	200.200.100.1/24
	Serial0	200.200.200.1/24
	Routing algorithm	RIP
	RIP network	200.200.100.0, 200.200.150.0 200.200.200.0
	Serial clock rate	56000
Host A	VMware 1	200.200.100.10/24
	VMware 2	200.200.100.20/24
Host B	VMware 3	200.200.150.10/24
	VMware 4	200.200.150.20/24

은 라우터와 각 호스트들에 의해서만 생성됨에 따라 웜 트래픽에 대한 신뢰성을 확보하였다. 라우터의 라우팅 알고리즘은 RIP로 설정하였고, A, B class address를 사용할 경우 기본 서브넷 마스크가 설정 되는 것을 방지하기 위해 C class 주소를 사용하였으며 세부내용은 표 3과 같다. 웜 트래픽은 라우터의 Cisco IOS(IGS-I-L Ver. 1.0)에서 지원하는 Debug 기능을 사용하였으며, 호스트에서는 Ethereal 을 이용하여 각각 수집하였다. 수집한 트래픽 데이터는 Protocol Type, Source/ Destination Address, Response Type등 이다.

4.2 실험용 웜 데이터 수집 및 구성

먼저 실험 및 검증용 데이터를 수집하기 위하여 Lovgate, Blaster, Sasser의 3가지 웜 바이너리 코드를 실행하여 각각 웜들이 알려진 특성대로 동작하는지부터 앞서 서술한 실험환경에서 테스트하였다. Sasser 웜의 경우는 새로 감염시킬 호스트를 찾기 위해 스캐닝 어드레스를 서로 랜덤하게 생성하는 결과를 확인하였고, Lovgate와 Blaster의 경우 Symantec의 분석결과[12]와 동일하게 Destination 주소를 순차적으로 증가시키며, 호스트를 찾는 것을 확인하였다. 그림 2와 3은 Lovgate와 Sasser 웜 실험 결과를 분석한 윈도우의 화면으로,

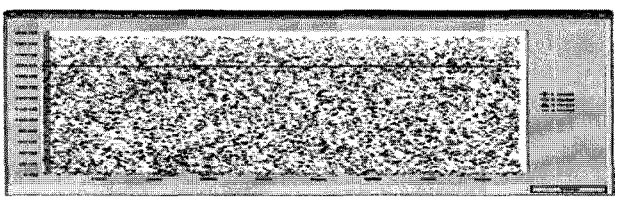

그림 2. Sasser Worm의 스캐닝 트래픽 분포
Fig. 2. The Scanning Traffic Distribution of Sasser Worm.

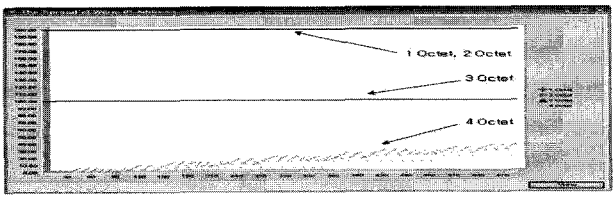

그림 3. Lovgate Worm의 스캐닝 트래픽 분포
Fig. 3. The Scanning Traffic Distribution of Lovgate Worm.

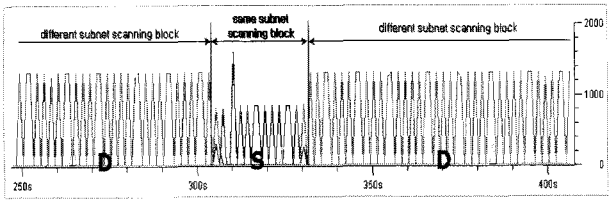

그림 4. 테스트베드상의 Blaster Worm 스캐닝 트래픽
Fig. 4. The Scanning Traffic Distribution of Blaster Worm on the test-bed.

4개의 옥텟 각각을 시각화 윈도우 프로그램을 통해 표현하였다. X축은 발생한 Packet의 수, Y축은 스캐닝 대상이 되는 Destination 주소의 범위이다. 그림 3의 Lovgate는 생성된 랜덤 IP 주소로부터 세 개의 옥텟은 고정된 값으로 마지막 IP의 옥텟 부분의 스캐닝 IP 주소가 점차 증가하는 것을 확인하였다.

Blaster 웜 바이너리 코드 실행을 통해 발생한 스캐닝 트래픽은 그림 4와 같다. 그림 4에서 X축은 발생 시간(sec), Y축은 패킷들의 bytes들을 나타내며, 그래프에서 각 선은 ARP, TCP, CDP 프로토콜을 보여주고 있다. 웜에 의한 트래픽이 발생하지 않는 Normal 트래픽에서는 라우터에 의한 CDP 프로토콜만 불규칙적으로 간혹 발생하였지만, 그림 4와 같은 Blaster 웜에 의한 스캐닝 트래픽들에서는, 테스트베드상의 서브넷과 동일한 스캐닝을 진행시에 ARP 브로드캐스트 패킷을 사용(S-구간)하였으며, 테스트베드 상에 존재하지 않는 다른 서브넷상의 스캐닝 진행시에는 TCP 프로토콜을 이용함(D-구간)을 확인할 수 있었다.

표 4는 실험상의 Normal 트래픽과 웜의 스캐닝 트래픽에 대한 실험 결과로, 제안 스캐닝 프로파일을 통한

표 4. Normal 트래픽과 Worm 스캐닝 트래픽
Table 4. The analysis of normal and worm traffic.

	Normal 트래픽	Worm 스캐닝 트래픽
초당 평균 발생 패킷 수	0.022 개	11.547 개
초당 평균 패킷 크기	258.000 bytes/sec	722.229 bytes/sec
각 패킷의 평균 크기	5.796 bytes	62.000 bytes

웜 모델링과 감염 트래픽의 탐지에 활용된다. 이러한 순차적 랜덤 스캐닝의 탐색 기법을 이용하는 Blaster 웜 및 유사 웜들에 대한 모델링을 통해, 프로파일을 생성하였으며, 이를 기준으로 검증 시뮬레이션을 수행하였다.

4.3. 실험 결과 및 분석

4.2에서 수행된 웜 테스트 데이터 구성을 통해 웜 스캐닝 패킷의 프로파일로 구성하여 이러한 프로파일에 대해 1 step 당 스캔 할 수 있는 순환 함수를 설정하여 실험을 수행하였다. 예를 들어 랜덤 생성 IP 주소가 실험 대상 네트워크(1.1.1.0) 범주에 포함이 되고 1.1.1.1인 경우로 생성되어지는 경우 스캐닝의 사이즈는 1.1.1.1~1.1.1.359까지 수행되게 되어 지며, 이 범주 안에 정상적인 호스트가 탐색되어질 경우 감염상태로 전환하고 새로운 감염 대상 호스트를 스캐닝하게 된다.

시만텍의 웜 분석 보고서[12]에 따르면 Blaster 웜의 경우 초기 감염 시 타켓을 검색하기 위해 생성시키는 랜덤 IP 주소의 확률은 웜이 감염된 호스트가 속한 서브넷일 경우가 40%, 로컬이 아닌 랜덤 IP 주소일 확률은 60%의 확률을 보인다고 분석하였다. 이에 따라 로컬 IP 주소와 랜덤 IP 주소의 확률을 다음과 같이 설정하고 시뮬레이터의 모델에 적용하였다.

로컬 IP Address일 확률
: P(Local Address | Worm Infection) = 0.4
랜덤 IP Address일 확률
: P(Random Address | Worm Infection) = 0.6

제안 기법은 그림 5와 같은 웜 감염 절차가 수행되는 동안 웜 전체 감염율, 호스트 평균 감염율, 순간 감염율 등을 분석하였다. 그림 5의 절차는 웜에 의해 감염된 호스트가 행동을 시작하면, 웜을 확산시키기 위해 새로운 호스트를 검색하는 랜덤 스캐닝 IP 주소를 생성 시킨다. 이 때 생성되어지는 초기 IP 주소가 감염된 호스트가 포함된 서브넷 주소 범위인 경우 순차적인 호스트 검색을 실시하게 된다. 반면, 호스트가 포함된 서브넷이 아니라면 확산 행동을 진행하지 못하고, 다음 시뮬레이션 시간(step) 까지 행동을 진행하지 않고 현재 step 이 종료된 후 새롭게 랜덤 스캐닝 IP 주소를 생성 시킨다.

앞서의 언급과 같이 제안 기법을 통한 시뮬레이션의 결과는 전체 감염률, 순간 감염률, 평균 감염률 세 가지 요소를 확인할 수 있다. 그림 6은 전체 감염률의 확산에 대해 Ⅲ장에서 제안한 모델링 기법을 적용하여, 실험된 결과이며, 네트워크상의 웜 확산 과정에 따른 호스트의 감염률을 시각화시킨 그래프이다. 이를 통해 시간에 따른 호스트의 감염률에 대해 보다 쉽게 이해할 수 있으며, 웜의 확산 결과 그래프는 CADIA등의 보안 업체와 Cliff Chang Zou 등의 실험[12~14]에서 제공하는 결과와 동일한 결과임을 알 수 있었으며 또한, 웜의 확산 단계를 초기 웜의 전파, 빠른 확산 그리고 늦은 확산 세 가지 과정으로 확인 할 수 있었다. 또한, 평균 및 순간 감염률 모델링을 통해 웜의 확산 초기에 빠르게 퍼져나가는 과정을 확인하고 시간의 흐름에 따라 서서히 1step당 감염되어지는 호스트의 평균 개수

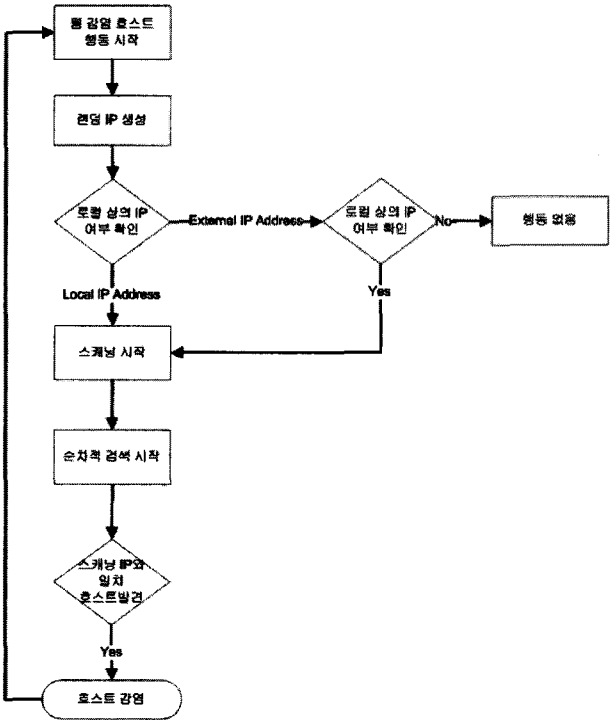

그림 5. 감염 진행 절차
Fig. 5. The procedure of infection processing.

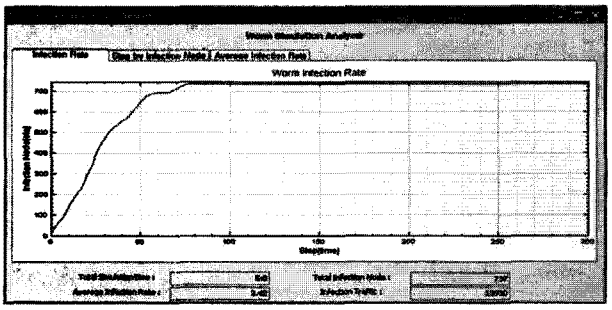

그림 6. 실험 네트워크상의 웜 전체 감염률
Fig. 6. The total infection rate of network worm.

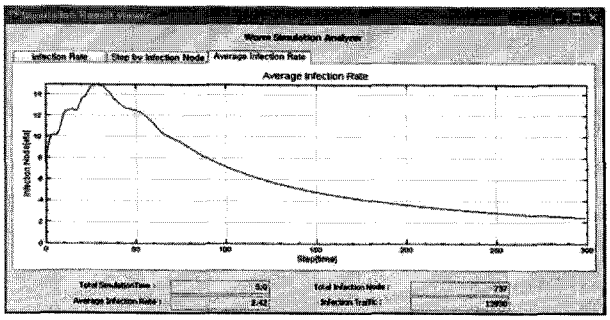

그림 7. 웜 확산에 따른 호스트 평균 감염률
Fig. 7. The host average infection rate by worm spread.

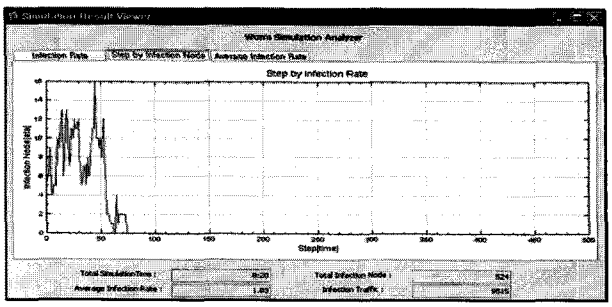

그림 8. 웜 확산에 따른 호스트 순간 감염률
Fig. 8. The instant infection rate by worm spread.

가 낮아지고 순간 감염률이 발생 초기 이후에 줄어듦을 알 수 있었다.

V. 결론 및 향후 연구 방향

웜의 스캐닝 기법은 웜의 확산 과정에 가장 크게 영향을 미치는 요소이다. 이에 이러한 웜의 스캐닝 기법을 모델링하고 이를 기반으로 웜의 확산 과정을 시뮬레이션 해봄으로써 발생되는 트래픽과 감염 노드의 발생 과정을 확인할 수 있었다. 또한 네트워크 트래픽상의 증가율에 대한 직접적인 확인이 가능했으며, 이를 바탕으로 실제 네트워크 환경상의 Threshold 정의 등을 구성하는 웜의 전파 모델링을 설계 할 수 있다. 실제 실험 결과에서는 실험 네트워크상의 웜 전체 감염률, 호스트 평균 감염률 그리고 호스트 순간 감염률에 대하여 실험 결과를 분석하여 적용 기법의 타당성을 검증하였다.

추후 연구를 통해 실제 웜의 트래픽을 차단할 수 있는 웜 차단 시스템의 구현 등이 필요하다.

참 고 문 헌

[1] Gorman, Siobhan and Ramstad, Evan. "Cyber Blitz Hits U.S., Korea." The Wall Street Journal 9 Jul 2009: A1.

[2] 김익균외 4명, 버퍼 오버플로우 웜 고속 필터링을 위한 네트워크 프로세서의 Bloom Filter 활용, 전자공학회논문지-TC 전자공학회논문지 제43권 TC편 제7호, 2006. 7, pp. 93 ~ 103

[3] 김성기외 3명, DDoS 공격에 대응하는 분산 네트워크 보안관리 기법, 전자공학회논문지-TC 전자공학회논문지 제43권 TC편 제7호, 2006. 7, pp. 72 ~ 83

[4] D. M. Kienzle, M. C. Elder, "Recent Worms: A Survey and Trends", WORM'03 of the ACM, October 2003.

[5] Harsha Talkad, "Survey of Worm Traffic Simuator: Course project for Security and Privacy in Computing", Csci, 8980-002, 2003.

[6] Simulating Network Worms, NWS by Bruce Ediger, http://www.users.qwest.net/~eballen1/nws

[7] SSF.App.Worm, A Network Worm Modeling Package, http://www.csdartmouth.edu/~mili/research/ssf/worm/index.html

[8] C. C. Zou, W. Gong, D. Towsly, "Worm Propagation Modeling and Analysis under Dynamic Quarantine Defense", WORM'03 of the ACM, October 2003.

[9] D. Moore, V. Paxson, S. Savage, C. Shannon, S. Staniford, N. Weaver, "Inside the Slammer Worm", IEEE Security and Privacy, 4(1), pp. 33-39, July-August 2003.

[10] Jeffrey O. Kephart and Steve R. White, "Measuring and Modeling Computer Virus Prevalence", Proceedings of the 1999 IEEE Computer Society Symposium on Research in Security and Privacy, pp.2-14, May 1993

[11] George F. Riley, "Simulating Internet Worms", 12th IEEE International Symposium MASCOT, pp. 268-274, October 2004.

[12] symantec, symantec technical report, http://securityresponse.symantec.com/avcenter/venc/data/w32.- blaster.worm.html

[13] CAIDA, CAIDA Technical Report, "The Spread of the Sapphire/Slammer Worm", http:///www.caida.org/outreach/papers/2003/sapphire/

[14] Cliff C. Zou, Don Towsley, and Weibo Gong, "On the Performance of Internet Worm Scanning Strategies", Journal of Performance Evaluation (extended from Umass ECE Technical Report TR-03-CSE-07, November 2003.

저 자 소 개

손 태 식(정회원)
2005년 아주대학교 정보 및 컴퓨터공학부 졸업
2002년 아주대학교 컴퓨터공학 석사
2005년 고려대학교 정보보호학 박사
2004년~2005년 Research Scholar, Univ. of Minnesota
2005년~현재 삼성전자 DMC 연구소 책임연구원
<주관심분야 : Wireless/Mobile Network Security, Wireless Sensor Network, Anomaly Detection>

구 본 현(정회원)-교신저자
2005년 동서대학교 정보통신공학과 학사
2007년 고려대학교 정보보호대학원 석사
2007년~현재 삼성전자 DMC 연구소 선임연구원
<주관심분야 : Mobile Security, Wireless Sensor Network, Visualization>