情報保護學會論文誌
第17卷 第5號, 2007. 10

AAWP와 LAAWP를 확장한 웜 전파 모델링 기법 연구*

전 영 태[1†], 문 종 섭[1‡], 서 정 택[2]
[1]고려대학교, [2]ETRI 부설 연구소

A Study of Worm Propagation Modeling extended AAWP, LAAWP Modeling*

Young-Tae Jun[1†], Jong-Sub Moon[1‡], Jung-Taek Seo[2]
[1]Korea University, [2]The Attacked Institute of ETRI

요 약

웜에 의한 사이버 위협이 증가함에 따라 웜의 전파 특성을 분석하기 위한 웜 전파 모델링 기법들이 연구되고 있다. 대표적인 예로 수학적 모델링 기법인 Epidemic, AAWP(Analytical Active Worm Propagation Modeling), 및 LAAWP(Local AAWP) 등의 모델링 기법들이 제시되었다. 하지만, 이들 기존 모델링 기법들은 대부분 IPv4 전체 네트워크를 대상으로 하는 랜덤 스캐닝 기법에 대해서만 모델링이 가능하며, 웜에 대한 인간의 대응활동인 보안패치 및 백신프로그램 업데이트 등의 행위를 표현하는데 한계점을 가지고 있다. 이에 본 논문에서는 AAWP와 LAAWP 모델링 기법들의 수식과 파라미터를 확장하는 모델로 ALAAWP(Advanced LAAWP Modeling)를 제안한다. 제안하는 모델은 웜 모델링에 있어 네트워크 및 스캐닝 기법 표현에 유연성을 가지며, 다양한 파라미터의 추가를 통하여 웜의 전파에 의한 피해정도 및 방어대책의 적절성 검증에 효과적으로 이용이 가능하다.

ABSTRACT

Numerous types of models have been developed in recent years in response to the cyber threat posed by worms in order to analyze their propagation and predict their spread. Some of the most important ones involve mathematical modeling techniques such as Epidemic, AAWP (Analytical Active Worm Propagation Modeling) and LAAWP (Local AAWP). However, most models have several inherent limitations. For instance, they target worms that employ random scanning in the entire IPv4 network and fail to consider the effects of countermeasures, making it difficult to analyze the extent of damage done by them and the effects of countermeasures in a specific network. This paper extends the equations and parameters of AAWP and LAAWP and suggests ALAAWP (Advanced LAAWP), a new worm simulation technique that rectifies the drawbacks of existing models.

Keywords : *Worm, Worm Modeling, AAWP, LAAWP, ALAAWP*[1)]

접수일: 2007년 5월 7일; 채택일: 2007년 7월 2일
* 본 연구는 정보통신부 및 정보통신연구진흥원의 대학 IT연구센터 지원사업의 연구결과로 수행되었음(IITA-200 6-(C1090-0603-0025))
† 주저자, jbacteria@korea.ac.kr
‡ 교신저자, jsmoon@korea.ac.kr

I. 서 론

최근 출현하는 웜은 로컬 스캐닝 기법 및 랜덤 스캐닝 기법 등 다양한 전파기법을 이용하여 빠른 속도로 전파되는 특성을 보이고 있다. 이러한 웜의 전파 특성을

표현하기 위하여 Epidemic, AAWP 및 LAAWP 등의 수학적 모델링 기법을 이용한 웜 전파 모델링 기법들에 대한 연구가 이루어지고 있다. 이들 연구는 실제 네트워크에서 발생 가능한 웜 전파 특성에 대하여 빠르게 분석하고, 예측을 통한 대응방안 마련에 활용이 가능하다.

하지만, 기존의 웜 모델링 기법들은 대부분 IPv4 전체 네트워크망(2^{32}개 호스트)에서 랜덤 스캐닝으로 전파되는 웜을 대상으로 하며, 보안패치 및 백신프로그램의 업데이트 등 인간의 대응활동에 대한 표현에 한계점을 가지고 있으며, 특정 내부 네트워크에서 전파되는 웜의 전파특성을 모델링 하는데 한계가 있다.[1][2][3][17]

따라서, 본 논문에서는 이러한 단점을 보완하고자 AAWP 및 LAAWP 모델링 기법의 수식과 파라미터를 확장하는 새로운 웜 전파 모델링 기법인 ALAAWP Advanced LAAWP Modeling) 모델링 기법을 제안한다.

이를 위해 Ⅱ장에서는 웜 스캐닝 기법에 대해 살펴보고, Ⅲ장에서는 웜 전파 예측 모델과 관련한 기존의 연구에 대해 살펴본다. Ⅳ장에서는 본 논문에서 제안하는 ALAAWP 모델링 기법에 대해 설명하고, Ⅴ장에서는 실험 및 검증을 통하여 제안하는 기법과 기존의 기법들을 비교 분석한다. 마지막으로 Ⅵ장에서는 결론 및 향후 연구방향에 대하여 기술한다.

Ⅱ. 웜의 스캐닝 기법

특정 내부 네트워크에서 웜이 전파될 때, 웜의 입장에서 보면 감염대상은 내부 네트워크뿐만 아니라 IPv4 전체 네트워크망(2^{32}개 호스트)을 감염대상으로 하며 감염속도를 높이기 위해 랜덤 스캐닝, 로컬 스캐닝, 토폴로지컬 스캐닝 등 다양한 스캐닝 기법을 사용한다.[4][5][6] 랜덤 스캐닝은 CodeRed와 Slammer 웜이 사용하는 기법으로 감염 대상 호스트 IP를 랜덤하게 선택하는 방식이며 로컬라이징 스캐닝은 CodeRed1 v2와 Nimda 웜이 사용하는 기법으로 로컬 주소를 우선적으로 스캔하여 감염 대상 호스트 IP를 선택한다. 마지막으로 토폴로지컬 스캐닝은 Morris 웜이 사용하는 기법인데 새로운 감염 대상 호스트에 저장되어 있는 IP 정보를 이용하여 전파된다.

이러한 웜의 다양한 스캐닝 기법에 대한 연구를 위해 스탠포드대학에서는 웜 초기 전파단계에서 전파 속도를 높이기 위한 'hitlist'개념을 고안하였다.[7] hitlist는 취약한 호스트들과 전파 이전에 이미 감염되어 있는 감염 대상 호스트들로 구성되어 있는데, 대표적인 예로 flash 웜이 있다. flash 웜은 웜 제작자가 미리 준비해 놓은 취약한 호스트들의 리스트를 대상으로 전파를 시도하며 실제 알려진 사례는 없지만 모든 스캔에 대해 취약한 호스트들을 선택할 수 있어 전파 속도가 가장 빠를 것으로 추측된다.[8]

그러면 웜이 이러한 스캐닝 기법을 사용할 때 어느 정도의 전파 효율성이 있는지 살펴보도록 하겠다. 예로서 랜덤 스캐닝 기법을 사용하는 CodeRed 웜의 전파 효율을 계산해보면, CodeRed 웜이 2^{32}개의 전체 IPv4 주소 공간상에서 360,000대의 취약한 호스트를 감염시킨다고 가정할 때, 자신의 전파를 위해 취약한 호스트를 선택할 확률은 $\frac{360000}{2^{32}} = 8.38 \times 10^{-5}$ (1)이 된다. 이것으로 보아 CodeRed 웜은 사실 취약하지 않은 호스트를 스캐닝하기 위해 상당한 시간을 낭비하고 있으며, 수식 (1)의 확률 값을 높일 수 있는 효율적인 스캐닝 기법을 사용하는 웜일수록 전파속도가 빠를 수 있다는 것을 알 수 있다.

CAIDA(Cooperative Association for Internet Data Analysis) 조사 결과에 의하면, 최근에 출현한 웜 중 Code Red 웜은 13시간 만에, 그리고 Blaster 웜은 4 ~ 5시간 만에 전 세계 네트워크망을 감염시킬 수 있다고 한다. 또한 2003년 1월에 발생한 Slammer 웜의 경우, 3분 만에 전 세계 IP대역을 스캐닝 할 수 있는데 이는 8.5초마다 감염지역을 2배씩 확대, 10분 만에 전 세계 네트워크망을 장악할 수 있다는 의미이며 감염 속도가 엄청나게 빠르다는 것을 짐작할 수 있다.[9][10]

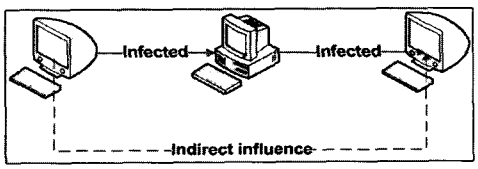
Ⅲ. 관련연구

3.1. Epidemic 모델링 기법

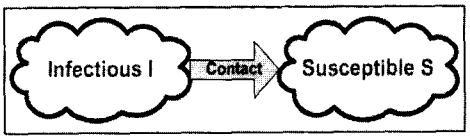
웜 모델링에서 초기의 수학적인 모델링 기법 연구는 바이러스에 대한 모델링 기법에서 시작되었는데, 데이터의 흐름 또는 정보의 흐름이 바이러스 전파에 매우 많은 영향을 미친다는 가정에서 이를 바이러스 모델링에 적용하였다. Fred Cohen등은 [그림 1]과 같이 각 호스트 간의 데이터 흐름, 또는 통신 메커니즘에 의해 바이러스가 전파된다는 가정을 하였다.[11] 따라서 데이터 흐름, 통신 메커니즘을 제한한 시스템은 잠재적으로 안

전한 시스템이라고 주장 하였으며, 이 연구를 시작으로 Gleissner 등은 이를 확장하여 다중 사용자 시스템에서 바이러스 전파 모델을 제시 하였고, 바이러스는 기하급수적인 전파를 보인다는 것을 증명하였다.[12]

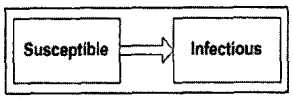
Epidemic 모델링 기법은 [그림 2]와 같이 시스템 내부의 어떤 호스트가 다른 호스트들과 접속(접속수$\propto I \times S$)할 수 있는 가능성이 모두 같다고 가정한 수학적 웜 전파 모델링이다.[13] 여기서부터 출발한 Simple Epidemic Model 기법은 [그림 3]처럼 모든 상태를 감염된 상태(Infectious)와 취약한 상태(Susceptible) 두 가지로 표현하는 수학적 모델링으로서 SI(Susceptible-Infectious) 모델링이라고도 하며 기준시간, 취약한 호스트 수, 네트워크 내의 총 호스트 수 등을 파라미터로 사용한다. 하지만 이 모델링 기법은 한번 감염된 호스트가 영원히 감염된 상태로 남게 되어 취약점이 제거된 상태(removed)가 표현되지 않는다는 단점이 있다. General Epidemic Model(Kermack-McKendrick Epidemic Model)은 [그림 4]처럼 이러한 SI 모델링 기법의 단점을 보완, 취약점이 제거된 상태(Removed)를 표현할 수 있는 모델링 기법으로서 SIR(Susceptible-Infectious-Removed)이라고 불린다. 하지만, 이 모델링 기법은 [그림 5]와 같이 취약한 호스트가 웜에 감염되기 전에 취약점이 조치된 상태의 표현이 불가능하다.[13]



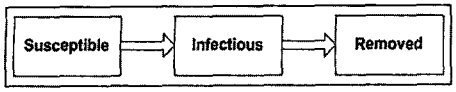
[그림 1] 초기 웜 모델링 개념



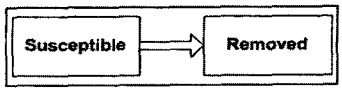
[그림 2] Epidemic Modeling 개념



[그림 3] SI Model



[그림 4] General Epidemic Model



[그림 5] 웜 감염 전, 취약점 조치 상태

3.2. AAWP(Analytical Active Worm Propagation) 모델링 기법

Epidemic 모델링 기법은 컴퓨터가 완전히 감염되지 않은 상태에서도 다른 컴퓨터를 감염시키며 보안패치 등 환경적인 요소를 고려하지 않아 부정확한 단점이 있는데 Zesheng Chen, Lixin Gao, Kevin Kwiat는 이것을 보완하고 랜덤스캐닝 기법으로 전파되는 웜을 적용할 수 있는 모델링 기법인 AAWP를 제안하였다.[14] [표 1]은 AAWP에서 사용되는 파라미터와 그에 해당하는 설명이다.

[표 1] AAWP 모델링 파라미터

파라미터	표기법	설명
취약한 호스트 수	N	취약한 전체 호스트 수
hitlist 크기	h	웜 전파 이전 감염되어 있는 호스트 수
스캔률	s	감염된 호스트에 의해 특정시간 동안 스캔되는 평균 호스트 수
death rate	d	보안패치가 아닌 다른 방법으로 호스트의 취약점이 제거되는 비율
패치율	p	감염된 호스트 또는 취약한 호스트의 취약점이 제거되는 비율

$$I_{t+1} = I_t + (N - I_t)[1 - (1 - \frac{1}{\Omega})^{sI_t}] \quad (2)$$

위 수식 (2)의 I_t는 AAWP에서 특정 시간 $t(t \geq 0)$동안 감염된 호스트 숫자를 계산하는 수식으로서, N은 IPv4 전체 네트워크망에 존재하는 취약한 호스트의 총 수, s는 웜의 스캔률, Ω는 스캔 대상 IP 주소 공간의 크기(2^{32})를 표현하며 t=0일 때 즉, I_0는 hitlist를 나타낸다. 여기서 백신프로그램, 개인방화벽 설치 등 보안패치가 아닌 다른 방법으로 호스트의 취약점이 제거되는 비율 d(death rate)와 보안패치 설치비율 p(patching rate)가 적용될 때, 수식 (2)는

$$I_{t+1} = (1-d-p)I_t + [(1-p)^s N - I_t][1-(1-\frac{1}{2^{32}})^{sI_t}] \quad (3)$$

이 된다. 수식 (3)은 더 이상 취약한 호스트가 존재하지 않거나 웜이 감염 호스트를 더 이상 늘릴 수 없을 때까지 반복적으로 수행된다.

3.3. LAAWP 모델링

AAWP 모델링은 랜덤 스캐닝 기법으로 전파되는 웜에만 적용이 가능한 단점이 있으므로 LAAWP를 제안하여 로컬 스캐닝 기법으로 전파되는 웜에도 적용할 수 있도록 하였다.[14] 이를 위해 LAAWP 모델링에서는 랜덤 스캐닝 기법의 웜이 특정 IP 대역을 스캐닝하는 시간을 아래와 같이 3가지 방식으로 구분하였다.

- P0% 시간 동안 네 개의 옥텟에 대해 랜덤하게 선택
- P1% 시간 동안 첫 번째 옥텟은 고정되고 나머지 세 개의 옥텟에 대해 랜덤으로 선택
- P2% 시간 동안 첫 번째와 두 번째 옥텟은 고정되고 나머지 두 개의 옥텟에 대해 랜덤으로 선택

따라서, P0, P1, P2로 정해진 로컬 스캐닝 기법 웜의 스캔 시간을 LAAWP의 스캔률에 적용하기 위해 다음과 같이 세 가지 타입의 서브넷을 정의하여 모델링 대상 호스트 수를 AAWP의 IPv4 전체 호스트 개수인 2^{32} 개 대신 2^{16}개로 축소시켰다.

- 스페셜 서브넷(k_1) : hitlist 크기보다 큰 규모의 서브넷
- 2^8-1 서브넷(k_2) : 스페셜 서브넷과 첫 번째 옥텟이 같은 2^8-1 크기의 서브넷
- $2^{16}-2^8$ 서브넷(k_3) : $2^{16}-2^8$ 크기의 서브넷

또한 수식 (4), (5), (6)은 P0, P1, P2와 그에 해당하는 서브넷 타입 k_1, k_2, k_3에 해당하는 스캔률이며 수식 (7)은 특정 시간 t($t \geq 0$) 동안 감염된 호스트의 수를 계산하는 수식이다. 즉, LAAWP 모델링은 AAWP에서 일정하게 적용되던 스캔률 s 대신 각 서브넷 타입에 해당하는 스캔률 k_i(i=1, 2, 3)를 사용하여 로컬 스캐닝 기법 웜이 감염 대상 호스트 IP를 선택하는 확률을 스캔률에 반영할 수 있도록 하였다.

$$k_1 = p_2 s b_1 + p_1 s[b_1 + (2^8-1)b_2]/2^8 + p_0 s[b_1 + (2^8-1)b_2 + (2^{16}-2^8)b_3]/2^{16} \quad (4)$$

$$k_2 = p_2 s b_2 + p_1 s[b_1 + (2^8-1)b_2]/2^8 + p_0 s[b_1 + (2^8-1)b_2 + (2^{16}-2^8)b_3]/2^{16} \quad (5)$$

$$k_3 = p_2 s b_3 + p_1 s b_3 + p_0 s[b_1 + (2^8-1)b_2 + (2^{16}-2^8)b_3]/2^{16} \quad (6)$$

$$I_{t+1} = I_t + (\frac{N}{2^{16}} - I_t)[1-(1-\frac{1}{2^{16}})^{k_i}] \quad (7)$$

3.4. 기존 모델링 기법의 한계점

Epidemic, AAWP 및 LAAWP 모델링 모두 웜의 전파 과정을 알아볼 수 있는 수학적인 모델링 기법이다. 각 모델링 기법의 한계점을 알아보면 첫째, Epidemic 모델링의 경우 기본 가정을 시스템 내부의 특정 호스트는 다른 호스트들과 접속할 수 있는 가능성이 모두 같은 것으로 보며 감염된 하나의 호스트가 다른 호스트를 감염시키는 접속을 '접속수$\propto I \times S$'로 표현한다.[13] 따라서 감염된 하나의 호스트에 의해 접속이 성공하면 즉시 감염이 된 것이며 웜이 전송되는 시간 등은 고려되지 않는다. 또한 보안패치 및 백신프로그램 최신버전 적용 등 인간에 의한 대응책이 고려되지 않았다. 둘째, AAWP 모델링은 IPv4 전체 호스트인 2^{32} 개의 호스트를 모델링 대상으로 하며 랜덤 스캐닝 기법으로 전파되는 웜에 대해 모델링이 가능하다.[14] 따라서, IPv4 내부의 특정 서브넷에서 전파되는 웜을 모델링할 수 없으며 로컬 스캐닝 기법의 웜도 모델링 할 수 없다. 마지막으로 LAAWP 모델링은 AAWP의 이러한 단점을 보완, 특정 서브넷을 기반으로 모델링을 할 수 있도록 하였지만 2^{16} 크기의 서브넷에만 제한적으로 적용이 가능하므로 2^{16}이 아닌 다른 크기의 서브넷은 모델링할 수 없는 단점이 있다. 또한 보안패치 및 백신프로그램 최신버전 적용 등 인간에 의한 대응책이 고려되지 않아 정확한 모델링 결과를 얻을 수 없다.

Ⅳ. 제안기법하는 ALAAWP(Advanced LAAWP) 모델링 기법

앞의 Ⅲ에서 언급한 것처럼 Epidemic, AAWP, LAAWP 모델링 기법에는 여러 가지 한계점이 있는데 이러한 한계점을 보완하고자 본 논문에서는 AAWP,

LAAWP 모델링의 수식과 파라미터를 확장한 웜 모델링 기법인 ALAAWP(Advanced LAAWP Modeling) 모델링 기법을 제안한다.

4.1. ALAAWP와 AAWP, LAAWP 파라미터 비교

ALAAWP 모델링 기법에서 로컬 스캐닝 기법 웜을 적용하기 위해 LAAWP 모델링과 유사한 스캔시간과 서브넷 타입을 정의하였다. [표 2]는 본 모델링 기법에서 사용하는 로컬 스캐닝 기법 웜이 감염 대상 호스트 IP를 스캔하는 방식이다. 로컬 스캐닝 웜은 먼저 P2시간 동안 첫 번째와 두 번째 옥텟을 고정하고 나머지 두 개의 옥텟에 대해 랜덤으로 IP를 생성하고 P1 시간 동안 첫 번째 옥텟은 고정한 상태에서 나머지 세 개의 옥텟에 대해 IP를 생성하며 스캐닝하게 된다는 의미이다.

Nimda 웜은 P0 = 0.5%, P1 = 0.25%, P2 = 0.25% 이므로 2^{16} 크기의 서브넷을 0.5 시간만큼 스캐닝하며 2^{24} 크기의 서브넷을 0.25 시간 동안 스캐닝하고, 2^{32}크기의 서브넷을 0.25 시간 동안 스캐닝하게 된다.[14] 이에 따라, ALAAWP은 [표 3]과 같이 이러한 웜의 스캐닝 기법을 이용하여 서브넷 타입을 3가지 종류로 구분하였다. 참고로 [표 2]에서 P2 + P1 + P0 = 1이 되며, 만약 본 모델링 기법에 랜덤 스캐닝 기법의 웜을 적용하려면 P0 = 1, P1=P2=0를 사용한다. 만약 CIDR을 사용하는 네트워크에서 특정 주소 블록을 중심으로 스캐닝하는 웜일 경우에도 모델링에 적용할 서브넷의 크기를 계산하여 [표 3]의 서브넷 타입(type1,type2,type3)을 사용할 수 있다.

AAWP 모델링 기법은 취약한 호스트가 취약하지 않은 호스트로 변경되는 수식을 $(1-p)^{i}N-I_t$ (p:호스트의 취약점이 제거되는 비율, N:취약한 전체 호스트 수, I_t:AAWP에서 특정 시간 t($t \geq 0$)동안 감염된 호스트 수)로 정의하여 호스트의 취약점이 제거되는 파라미터를 p 하나로 일괄 적용하였다. 하지만 실제 인터넷 환경에서 호스트의 취약점이 제거되는 방법은 보안패치 설치, 백신프로그램 및 개인방화벽 최신버전 설치로 세분화 될 수 있으며 네트워크 내부의 각 호스트들에 의해 설치되는 비율 역시 각각 다르므로 보다 정확한 모델링을 위해서 호스트의 취약점이 제거되는 방법을 구체화 시킬 필요가 있다. 또한, 각 호스트들이 보안패치, 백신프로그램, 개인방화벽 최신 업데이트 버전을 설치하는

〔표 2〕 로컬 스캐닝 기법 웜, 감염 대상 호스트 IP 생성 방법

스캔 시간	스캔 대상 IP (x : 랜덤생성, a : 고정)
P2 시간(%)	a.a.x.x (x = 2^{16} 비트)
P1 시간(%)	a.x.x.x (x = 2^{24} 비트)
P0 시간(%)	x.x.x.x (x = 2^{32} 비트)

〔표 3〕 서브넷 타입

서브넷 타입	IP 생성방법
type1	서브넷크기 $\leq 2^{16}$
type2	2^{16} <서브넷크기 $\leq 2^{24}$
type3	2^{24} <서브넷크기 $\leq 2^{32}$

비율이 아무리 높다 하더라도, O/S 제작업체, 백신업체 등에서 최신 업데이트 버전을 제공하는 시간이 지연된다면 설치비율이라는 것은 아무 의미가 없게 되므로 최신 업데이트 버전 제공시간 역시 파라미터로 고려할 필요가 있다.

[표 4]는 ALAAWP 모델링 기법에서 사용하는 파라미터를 정리한 것이다. AAWP에서 사용되는 파라미터와 비교해 볼 때, (*)는 새로 추가된 파라미터이며 (**)는 확장된 파라미터이고 나머지는 AAWP와 동일한 파라미터이다. 본 모델링에서 '스캔률'은 웜에 감염된 호스트가 취약한 한 개의 호스트를 감염시키는데 소요되는 시간을 의미한다.

ALAAWP 모델링 기법에서는 각 서브넷 타입에 해당되는 스캔률과 초기 감염 호스트 수, 내부 취약한 호스트수를 별도의 파라미터로 사용하여 모델링의 정확도를 높인다.

4.2. ALAAWP 모델링

ALAAWP 모델링에서 웜은 동시에 많은 컴퓨터들을 스캔하고 이미 감염된 호스트는 재감염시킬 수 없으며 모든 호스트는 백신프로그램과 개인방화벽을 설치하였으며 보안업데이트가 가능한 것으로 간주한다. 만약 이것을 무시하고자 한다면 보안패치 등 해당 파라미터를 0으로 설정하며 랜덤 스캐닝 웜의 경우 P1=1을 적용하여 사용한다. 그리고, 본 논문에서는 Weaver가 제안한 hitlist 개념을 도입하여 웜 전파 이전에 hitlist는 이미 감염된 상태로 보고 웜 감염에 필요한 시간은 1 time unit으로 가정한다.[8] 즉, 감염된 호스트에 의한 한 번의

〔표 4〕 ALAAWP 모델링 파라미터

파 라 미 터	표 시	설 명
type1 스캔 시간(%)	p2	type1에 대한 스캔 시간
type2 스캔 시간(%)	p1	type2에 대한 스캔 시간
type3 스캔 시간(%)	p0	type3에 대한 스캔 시간
감염 시간	inf_time	한 개의 취약한 호스트를 감염시키는데 소요되는 시간
(*)서브넷 크기	subnet size	모델링 대상 서브넷의 크기
(*)type1 내부 취약한 호스트 수	type1 vul num	type1 내부에 존재하는 취약한 호스트의 총수
(*)type2 내부 취약한 호스트 수	type2 vul num	type2 내부에 존재하는 취약한 호스트의 총수
(*)type3 내부 취약한 호스트 수	type3 vul num	type3 내부에 존재하는 취약한 호스트의 총수
(*)type1의 스캔률	k1	type1에 감염된 호스트에 의해 발생하는 스캔률
(*)type2의 스캔률	k2	type2에 감염된 호스트에 의해 발생하는 스캔률
(*)type3의 스캔률	k3	type3에 감염된 호스트에 의해 발생하는 스캔률
서브넷 내부 전체 취약 호스트 수	N	서브넷 내부 전체 취약 호스트 수
(*)type1 hitlist 수	AS1 h1	type1 초기 감염 호스트 수
(*)type2 hitlist 수	AS2 h2	type2 초기 감염 호스트 수
(*)type3 hitlist 수	AS3 h3	type3 초기 감염 호스트 수
(**)백신업데이트 발표 시기	av time	시뮬레이션 시작 후,백신업체가 백신업데이트를 발표하는데 걸리는 시간
(**)백신업데이트 설치 비율	av rate	해당 PC에 백신업데이트가 설치될 비율
(**)보안패치 발표 시기	patch time	시뮬레이션 시작 후, 해당 OS에 대한 보안패치 발표되는데 걸리는 시간
(**)보안패치 설치 비율	patch rate	해당 PC에 보안패치가 설치될 비율
(**)개인방화벽 룰 발표 시기	firewall time	시뮬레이션 시작 후, 해당 웜을 방어할 수 있는 방화벽룰이 발표되는데 걸리는 시간
(**)개인방화벽 룰 설치 비율	firewall rate	해당 PC에 방화벽 룰이 업데이트될 비율
시뮬레이션 관찰 시간	time unit	전체 시뮬레이션 관찰을 위한 시간 단위

스캔이 특정 호스트로 접근하였을 때 이 호스트가 취약하거나 취약하지 않거나, 이미 감염된 상태이거나 사용되지 않는 IP일지라도, 웜이 한 개의 호스트와 통신을 끝내기 위한 시간은 1 time unit이 소요된다. 이 가정은 비록 현실성이 없을 수도 있지만 모델링을 간단화 시킬 수 있으며, 모델링 결과에 큰 영향을 미치지 않는다.

그러면, 로컬 스캐닝 기법을 사용하는 웜이 총 호스트 수 p개인 A 네트워크 내부에서 한 개의 호스트를 감염시킬 확률을 알아보도록 한다. 먼저 웜이 IPv4 네트워크망의 전체 호스트 개수인 2^{32}개 중 특정 네트워크를 선택할 확률은 $\frac{p}{2^{32}}$이다. 그 후, p개의 호스트가 존재하는 네트워크에서 한 개의 호스트가 선택할 확률은 $\frac{1}{p}$ 이다. 따라서 특정 네트워크 내부에 있는 한 개의 호스트가 선택될 확률은 $\frac{p}{2^{32}} \times \frac{1}{p} = \frac{1}{2^{32}}$ 이 된다. 즉, 어떤 크기, 어떤 IP 대역, 어떤 크기의 네트워크 내부에서 웜이 전파된다 하더라도 특정 네트워크 내부에서 한 개의 호스트가 선택될 확률은 결국 $\frac{1}{2^{32}}$이 된다.

time unit i($i \geq 0$)에 대해 x_i와 y_i를 감염된 호스트를 포함한 전체 취약한 호스트 수라고 하고 특정 네트워크 내부의 hitlist 수를 h라고 하자. 웜 전파 이전(i=0)에, $x_0 = N$ 이고 $y_0 = h$이다. 여기서 q_i를 time unit i ($i \geq 0$)동안 새로 감염된 호스트 수라고 할 때, t=1은 $(x_i - y_i)$개의 취약한 호스트가 아직 하나도 감염되지 않은 상태이며 새로 한 개의 호스트가 감염될 수 있는 확률은 $\frac{(x_i - y_i)}{2^{32}}$이며 이것은$(x_i - y_i)[1-(1-\frac{1}{2^{32}})^t]$ (8)로 표현할 수 있다. 수식 (8)은 t=w+1 스캔에 대해 w+1 스캔과 w 스캔 두 부분으로 나눌 수 있다. 여기에는 감염에 성공하여 새로운 감염 호스트가 늘어나거나 그 반대의 경우, 두 가지 가능성이 존재한다. 변수 S=1을 마지막 스캔이 취약한 호스트를 접근했을 때로 지정하고 S=0은 그렇지 않을 경우로 지정, 다음에 감염될 수 있는 호스트 수(Q)가 아래의 식을 만족함을 알 수 있다.

$$Q[q_{i+1}/t=w+1] =$$
$$(Q[q_{i+1}/t=w]+1)P(s=1)+(Q[q_{i+1}/t=w])P(s=0)$$
$$= (Q[q_{i+1}/t=w]+1)\frac{x_i - y_i - Q[q_{i+1}/t=w]}{2^{32}} +$$
$$(Q[q_{i+1}/t=w])(1-\frac{x_i - y_i - Q[q_{i+1}/t=w]}{2^{32}}) \quad (9)$$

수식 (9)에서 $Q[q_{i+1}/t=w]$를 A라고 하자.

$$(A+1)\frac{x_i - y_i - A}{2^{32}} + A(1-\frac{x_i - y_i - A}{2^{32}}) =$$
$$(\frac{x_i - y_i - A}{2^{32}})A + \frac{x_i - y_i - A}{2^{32}} + A - (\frac{x_i - y_i - A}{2^{32}})A$$
$$= A(\frac{x_i - y_i - A}{2^{32}} + 1 - \frac{x_i - y_i - A}{2^{32}}) + \frac{x_i - y_i - A}{2^{32}}$$
$$= A + \frac{x_i - y_i}{2^{32}} - \frac{A}{2^{32}}$$가 된다.

따라서, $\frac{x_i - y_i}{2^{32}} + (1-\frac{1}{2^{32}})Q[q_{i+1}/t=w] =$

$(x_i - y_i)[1-(1-\frac{1}{2^{32}})^{w+1}]$이 되므로 t=w+1도 역시 동일한 수식이 적용되는 것을 알 수 있다. k를 특정 네트워크 내부에서 웜에 의해 발생하는 스캔률(k=1,2,3)이라고 할 때,

$Q[q_{i+1}/k = scanrate] = (x_i - y_i)[1-(1-\frac{1}{2^{32}})^k]$이고

이것은 다음 time unit에 $(x_i - y_i)[1-(1-\frac{1}{2^{32}})^k]$개의 새로운 감염 호스트가 생긴다는 의미이다. 여기서 백신업데이트, 보안패치, 개인방화벽이 각 호스트에 적용되는 설치비율을 고려하면 다음 time unit에는 $avrate \times y_i + patchrate \times y_i + firewallrate \times y_i$개의 감염된 호스트가 생성된다. 이에 따라, 취약 호스트의 총 수는 $(1-avrate-patchrate-firewallrate)x_i$개로 감소하게 되며 다음 time unit에 감염된 호스트의 총 수는

$$y_{i+1} = y_i + (x_i - y_i)[1-(1-\frac{1}{2^{32}})^k] - (avrate + patchrate + firewallrate)y_i$$

개가 된다. 이와 동시에

$x_{i+1} = (1-avrate-patchrate-firewallrate)x_i$이므로

$x_i = (1-avrate-patchrate-firewallrate)^i$,

$x_o = (1-avrate-patchrate-firewallrate)^i N$이 된다.

즉, $i \geq 0$과 $y_0 = h$일 때, 수식(10)이 성립한다.

$$y_{i+1} = (1-avrate-patchrate-firewallrate)^i \quad (10)$$
$$x + [(1-avrate-patchrate-firewallrate)^i N - x_i][1-(1-\frac{1}{2^{32}})^{ki}]$$

[그림 6], [그림 7], [그림 8]은 [표 3]에서 정의한 서브넷 타입을 그림으로 표현한 것인데 이것을 이용하여 각 서브넷에 해당하는 스캔률(ki)을 구해보도록 한다. [그림 6], [그림 7], [그림 8]에서 모델링 대상 네트워크 크기를 s로 정의하고 모델링 대상 네트워크를 Subnet으로 표기하였다. 앞에서도 언급했듯이, 웜이 특정 서브넷 내부에서 전파될 때 감염대상은 대상 내부 네트워크뿐만 아니라 IPv4 전체 네트워크 망(2^{32}개 호스트)을 감염 대상으로 한다.

[그림 6]의 type1의 경우 모델링 대상 네트워크(서브넷)가 type1 내부에 속하므로 type1에 감염된 호스트에 의해 스캔률 k1이 발생한다. 이 경우, 웜의 스캔은 P2 시간동안 type1 네트워크를 스캐닝하는 동시에 P1과 P0를 통해 type2와 type3 네트워크를 대상으로도 스캔을 시도한다.

따라서 k1은 type1 내부 모델링 대상 네트워크의 초기 감염 호스트 수 AS1 h1에 의해 기본적으로 P2시간만큼 type1에 대해 스캐닝을 시도한다. 이와 동시에 P1과 P0 시간 동안 type2와 type3에 대해 스캐닝을 하는데 type1은 type2와 type3에 속해 있으므로 P1과 P0 스캔 역시 스캔률 K1에 영향을 미친다. K1은 수식 (11)과

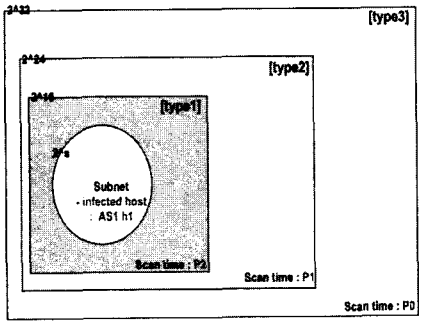
같이 계산된다.

$$k_1 = P2 \times scanrate \times AS1\,h1 + P1 \times scanrate \times \frac{AS1\,h1}{2^{24}} + P0 \times scanrate \times \frac{AS1\,h1}{2^{32}} \quad (11)$$

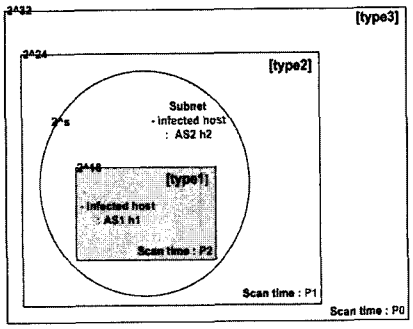
[그림 7]의 type2의 경우 모델링 대상 네트워크가 type1보다는 크고 type2 보다는 작다. 따라서 type1 내부에 속하는 감염된 호스트에 의해 스캔률 k1이 발생하고 type2 내부에 속하는 감염된 호스트에 의해 스캔률 k2가 발생한다. 즉, AS1 h1에 의해 P2시간 동안 type1에 대해 스캔을 하며 P1과 P0 시간동안 각각 type2와 type3에 대해 스캔을 한다. 하지만 P1과 P0의 경우 취약한 호스트 AS1 h1과 AS2 h2에 동시에 영향을 미치므로 감염대상 호스트는 각각

$\frac{[AS1\,h1 + (2^s - 2^{16})AS2\,h2]}{2^{24}}$과

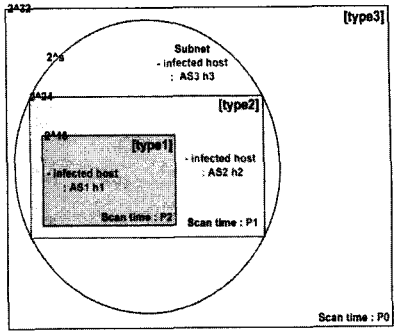
$\frac{[AS1\,h1 + (2^s - 2^{16})AS2\,h2]}{2^{32}}$로 계산된다. 따라서 k1과 k2는 수식 (12), (13)과 같다.



〔그림 6〕 type1



〔그림 7〕 type2



〔그림 8〕 type3

$$k1 = P2 \times scanrate \times AS1\,h1 + P1 \times scanrate \times \frac{[AS1\,h1 + (2^s - 2^{16})AS2\,h2]}{2^{24}} + P0 \times scanrate \times \frac{[AS1\,h1 + (2^s - 2^{16})AS2\,h2]}{2^{32}} \quad (12)$$

$$k2 = P2 \times scanrate \times AS2\,h2 + P1 \times scanrate \times \frac{[AS1\,h1 + (2^s - 2^{16})AS2\,h2]}{2^{24}} + P0 \times scanrate \times \frac{[AS1\,h1 + (2^s - 2^{16})AS2\,h2]}{2^{32}} \quad (13)$$

[그림 8]의 type3도 type2와 유사한 방식으로 계산되며 스캔률 k1, k2, k3이 모두 발생한다. 하지만 이 경우 취약한 호스트는 AS1 h1, AS2 h2, AS3 h3 세 가지가 있는데 P0 스캔시에 AS1 h1, AS2 h2, AS3 h3에 동시에 영향을 미치게 된다. type3의 스캔률 k1, k2, k3은 아래 수식 (14), (15), (16)과 같다.

$$k1 = P2 \times scanrate \times AS1\,h1 + P1 \times scanrate \times \frac{[AS1\,h1 + (2^{24} - 2^{16})AS2\,h2]}{2^{24}} + P0 \times scanrate \times \frac{[AS1\,h1 + (2^{24} - 2^{16})AS2\,h2 + (2^s - 2^{24})AS3\,h3]}{2^{32}} \quad (14)$$

$$k2 = P2 \times scanrate \times AS2\,h2 + P1 \times scanrate \times \frac{[AS1\,h1 + (2^{24} - 2^{16})AS2\,h2]}{2^{24}} + P0 \times scanrate \times \frac{[AS1\,h1 + (2^{24} - 2^{16})AS2\,h2 + (2^s - 2^{24})AS3\,h3]}{2^{32}} \quad (15)$$

$$k3 = P2 \times scanrate \times AS3\,h3 + P1 \times scanrate \times AS3\,h3 + P0 \times scanrate \times \frac{[AS1\,h1 + (2^{24} - 2^{16})AS2\,h2 + (2^s - 2^{24})AS3\,h3]}{2^{32}} \quad (16)$$

위에서 구한 스캔률 k1, k2, k3에 의해 AS 내부의 새로 감염되는 호스트 수는 보안패치, 백신프로그램, 개인 방화벽의 최신업데이트 발표 시간 및 이러한 것들이 호

스트에 실제 적용되는 설치비율에 영향을 받게 된다. time unit j에서 AS 내부의 새로 감염되는 호스트 수는 아래 1) ~ 8)과 같이 계산된다.

1) 백신프로그램, 보안패치, 개인방화벽 최신업데이트 발표 전

$$vi = ASihi + (N - ASihi) \times (1 - (1 - \frac{1}{2^{32}})^{ki})$$

2) 백신프로그램, 보안패치, 개인방화벽 최신업데이트 발표 후

$$vi = (1 - avrate - patchrate - firewallrate) \times ASihi + ((1 - avrate - patchrate - firewallrate)^{j} \times N) - ASihi) \times (1 - (1 - \frac{1}{2^{32}})^{ki})$$

3) 백신프로그램 최신업데이트 발표 이후 (패치, 개인방화벽 최신업데이트 발표 전)

$$vi = (1 - avrate) \times ASihi + ((1 - avrate)^{j} \times N) - ASihi) \times (1 - (1 - \frac{1}{2^{32}})^{ki})$$

4) 백신프로그램, 보안패치 최신업데이트 발표 이후 (개인방화벽 최신업데이트 발표 전)

$$vi = (1 - avrate - patchrate) \times ASihi + ((1 - avrate - patchrate)^{j} \times N) - ASihi) \times (1 - (1 - \frac{1}{2^{32}})^{ki})$$

5) 백신프로그램, 개인방화벽 최신업데이트 발표 이후 (보안패치 최신업데이트 발표 전)

$$vi = (1 - avrate - firewallrate) \times ASihi + ((1 - avrate - firewallrate)^{j} \times N) - ASihi) \times (1 - (1 - \frac{1}{2^{32}})^{ki})$$

6) 보안패치 최신업데이트 발표 이후 (백신프로그램, 개인방화벽 최신업데이트 발표 전)

$$vi = (1 - patchrate) \times ASihi + ((1 - patchrate)^{j} \times N) - ASihi) \times (1 - (1 - \frac{1}{2^{32}})^{ki})$$

7) 보안패치, 개인방화벽 최신업데이트 발표 후 (백신프로그램 최신업데이트 발표 전)

$$vi = (1 - patchrate - firewallrate) \times ASihi + ((1 - patchrate - firewallrate)^{j} \times N) - ASihi) \times (1 - (1 - \frac{1}{2^{32}})^{ki})$$

8) 개인방화벽 최신업데이트 발표 후 (백신프로그램, 보안패치 최신업데이트 발표 전)

$$vi = (1 - firewallrate) \times ASihi + ((1 - firewallrate)^{j} \times N) - ASihi) \times (1 - (1 - \frac{1}{2^{32}})^{ki})$$

ALAAWP 모델링 기법은 먼저 서브넷 타입별 스캔률을 계산한 후, 이것을 이용하여 모델링 대상 서브넷 내부의 새로 감염되는 호스트 수를 계산하는 방식으로 진행하며, 이 프로세스는 더 이상 취약 호스트가 남지 않거나 웜이 감염 호스트를 더 이상 늘릴 수 없을 때까지 반복된다. 향후 새로운 파라미터가 추가될 경우, 1) ~ 8)과 유사한 방식으로

$$vi = (1 - \sum_{1}^{n} \text{파라미터}) \times ASihi + ((1 - \sum_{1}^{n} \text{파라미터})^{j} \times N) - ASihi) \times (1 - (1 - \frac{1}{2^{32}})^{ki})$$

를 계산식으로 적용 수 있다.

4.3. 기존 모델링 기법과 ALAAWP 모델링 비교

[표 5]는 본 논문에서 제안하는 ALAAWP 모델링을 기존 모델링 기법과 비교한 결과이다. ALAAWP 모델링 기법은 LAAWP와 같이 랜덤 스캐닝과 로컬 스캐닝 웜에 모두 적용할 수 있으며, AAWP 처럼 인간에 의한 대응책을 적용하여 웜의 전파 속도 예측과 동시에 방어책에 대한 효과를 확인해 볼 수 있다. ALAAWP가 다른 모델링 기법들과 구분되는 가장 큰 특징은 IPv4 네트워크망 내부의 어떤 크기의 내부 네트워크라도 융통성 있게 모델링이 가능하다는 것이다.

〔표 5〕 기존 모델링 기법과 ALAAWP 비교 분석

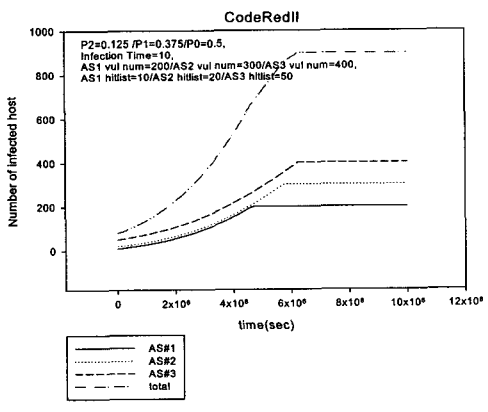
	Epidemic	AAWP	LAAWP	ALAAWP
적용 가능한 웜 스캐닝 기법	랜덤	랜덤	랜덤, 로컬	랜덤, 로컬
모델링 대상 네트워크 크기	2^{32}	2^{32}	2^{16}	2^{1} ~ 2^{32}
인간의 대응책 적용 여부	X	O	X	O

V. 실험 및 검증

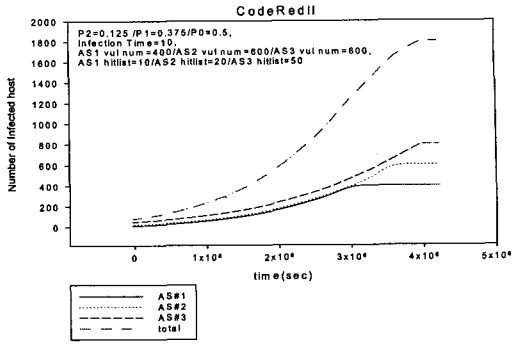
본 논문에서 제안하는 ALAAWP 모델링 기법의 검증을 위해 IPv4 기반의 실제 수집된 CodeRed1 v2 웜 전파 데이터와 ALAAWP type3로 CodeRed1 v2 웜을 모델링한 결과를 비교 분석해 보았다. 그리고 CodeRedII 웜을 type2 서브넷을 대상으로 모델링하여 서브넷별 hitlist와 감염대상 호스트 IP 생성 비율을 각각 변화시킬 경우 감염 호스트 수가 어떻게 변화하는지 살펴보았다. 마지막으로 type1 서브넷을 대상으로 서브넷 내부에서 전파되는 특정 웜을 모델링하여 보안패치, 백신업데이트, 개인방화벽 최신버전이 설치되었을 때 이에 따른 방어 효과를 예측해 보았다.

5.1. 실제 웜 전파 데이터와 비교분석

ALAAWP 모델링 기법의 검증을 위해 랜덤 스캐닝 기법으로 전파되는 CodeRed1 v2를 대상으로 실험을 한 결과, CAIDA에서 수집한 데이터와 거의 유사함을 알 수 있었다. CAIDA 수집 데이터에서는 CodeRed1 v2 웜 전파가 시작된 후, 약 16시간(57600초)이 지난 후부터 급격히 감염속도가 증가하기 시작하여 약 22시간(79200초)이 지난 후 취약한 호스트가 거의 모두 감염된 상태를 보여주고 있다. 반면, ALAAWP 모델링 결과를 보면 약 13시간(50000초) 후부터 급격히 감염속도가 증가하기 시작하여 약 18시간(65820초) 경에 취약한 호스트가 거의 모두 감염된 것으로 결과가 나왔다. 이 두 결과의 차이는 CAIDA 수집 데이터의 경우, 실제 인터넷 환경에서 발생할 수 있는 혼잡 및 네트워크 속



〔그림 9〕 CodeRed II 모델링(type2)



〔그림 10〕 CodeRed II 감염호스트 두 배 증가(type2)

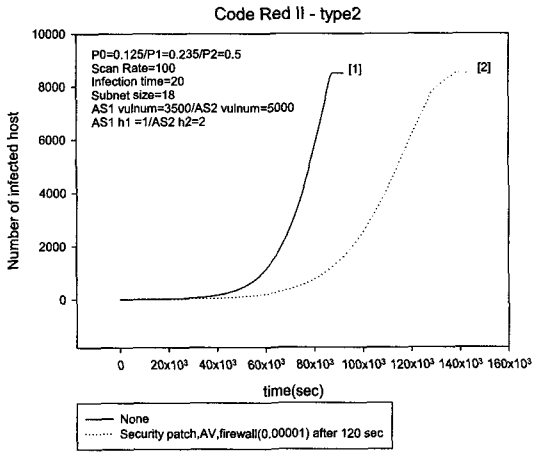
도 지연 문제, 감염호스트 PC의 전원이 꺼진 경우, 감염 사실 인지 후 네트워크와 분리 조치 등 여러 가지 상황이 존재하지만 ALAAWP 모델링에서는 단지 보안패치, 백신업데이트, 개인방화벽 최신버전 설치만을 고려하였으므로 CAIDA 수집 데이터에 비해 감염 속도가 빠른 결과가 나온 것으로 분석된다.

5.2. CodeRedII 대상 type3 모델링 결과

CodeRedII의 취약한 호스트 수를 서브넷 타입별로 각각 200, 300, 400대로 정하고 시뮬레이션을 수행한 결과, [그림 9]와 같이 총 900대의 취약한 호스트를 감염시키는데 총 6231920초가 걸리는 것으로 나왔다. 여기서 취약한 호스트 수를 서브넷 별로 각각 2배로 증가시켜 실험한 결과 [그림 10]과 같이 감염시간이 4210000초로 약 1.5배 빨라짐을 알 수 있다. 감염속도가 빨라지는 이유는 서브넷 내부에 취약한 호스트 수가 많을수록 웜이 취약한 호스트를 선택할 확률($\frac{\text{취약한 호스트 수}}{2^{32}}$)값이 높아지기 때문이다.

5.3. CodeRed II 대상 type2 서브넷 모델링 결과

[그림 11]은 CodeRed II가 전파되는 과정을 서브넷 type2를 대상으로 모델링한 결과이다. 실험시 임의로 P0=0.125/P1=0.375/P2=0.5, Scan Rate=100, Infection time=20, Subnet size=18, AS1 vul num=3500/AS2 vul num=5000, AS1 h1=1/AS2 h2=2로 고정한 상태로 모델링을 하였다. 여기서 보안패치와 백신업데이트, 개인방화벽 최신버전 설치 비율을 0.00001로 설정한 경우



〔그림 11〕 CodeRedII, type2 모델링 결과

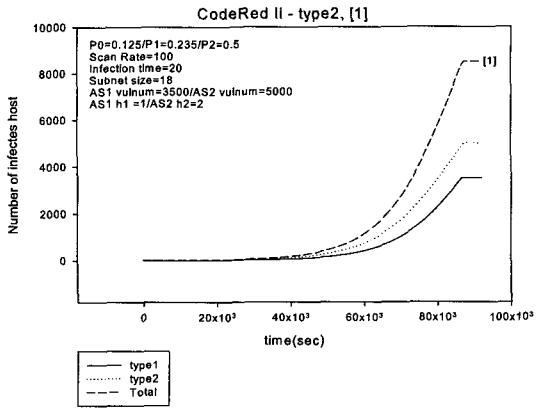
결과는 [그림 11]의 [2]와 같고, 보안패치, 백신업데이트, 개인방화벽 최신버전 설치를 적용하지 않은 경우는 [그림 11]의 [1]과 같은 결과가 나왔다. 즉, 보안패치와 백신업데이트, 개인방화벽 최신버전 설치가 적용되면 감염속도가 현저히 느려진다는 것을 알 수 있다. [그림 12]와 [그림 13]은 [그림 11]의 [1]과 [2] 결과를 서브넷 단위로 나타낸 그래프이다. P0와 P1 비율의 차이가 0.11로 근사하므로 type1과 type2의 감염호스트의 수는 거의 비슷한 비율로 증가하는 것을 볼 수 있다. 그리고 [그림 15]는 보안패치, 백신업데이트, 개인방화벽 최신버전 설치를 적용하지 않은 상태에서 모델링한 결과인데, P2=1인 임의의 웜이 감염 속도가 가장 빠르며 그 다음으로 Nimda, CodeRed II 순서임을 알 수 있다. type2를 대상으로 모델링한 결과이므로 type2 내부를 스캔하는 확률인 P1의 값이 높은 웜일수록 감염속도가 빠르다는 것을 알 수 있다.

〔표 6〕 모델링 대상 웜의 스캔 시간[15]

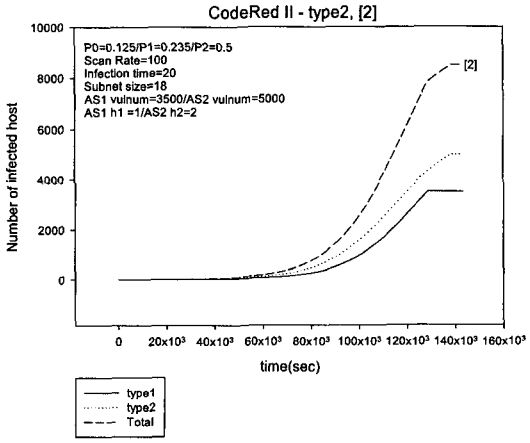
웜	P2	P1	P0
CodeRedII	0.125	0.375	0.5
Nimda	0.25	0.25	0.5
임의의 웜	1	0	0

5.4. CodeRedII 대상 type1 서브넷 모델링 결과

총 32개의 호스트 중 30개의 취약한 호스트가 존재하는 특정 서브넷 내부에서, 보안패치 등 여러 가지 상황에 따라 CodeRed II가 전파되는 과정을 모델링하기 위해 아래와 같이 세 가지 방법으로 실험을 하였다.



〔그림 12〕 〔그림 11〕 〔1〕의 서브넷 별 모델링 결과

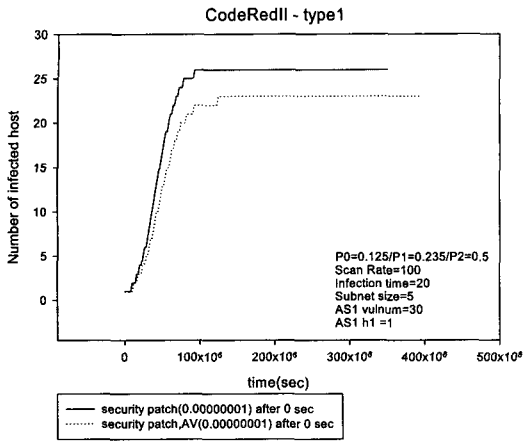


〔그림 13〕 〔그림 11〕 〔2〕의 서브넷 별 모델링 결과

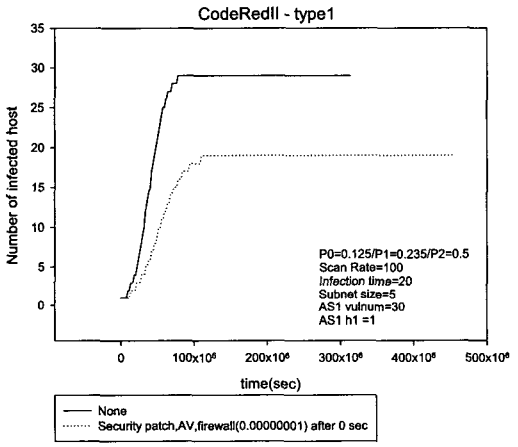
(1) 보안패치, 백신업데이트, 개인방화벽 최신버전을 0.00000001 설치비율로 적용한 상태와 아무것도 적용하지 않은 상태 비교 - [그림 16]

(2) 백신업데이트 및 개인방화벽은 최신버전 설치는 고려하지 않고 보안패치만 0.0000001 비율로 적용한 상태와 보안패치와 백신업데이트를 각각 0.0000001 비율로 적용한 상태 비교 - [그림 17]

(3) 백신업데이트가 0초 후 발표되었으며 0.00000001 설치비율 적용된 상태, 백신업데이트가 146000000초 후 발표되었으며 0.00000001 설치비율 적용된 상태, 백신업데이트가 146000000초 후 발표되었으며 0.0000001



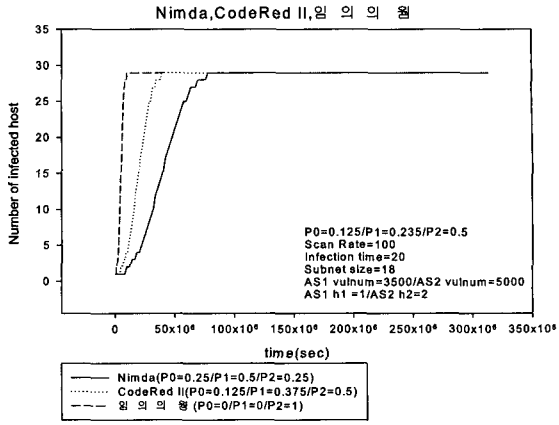
〔그림 14〕 CodeRedII 모델링 결과



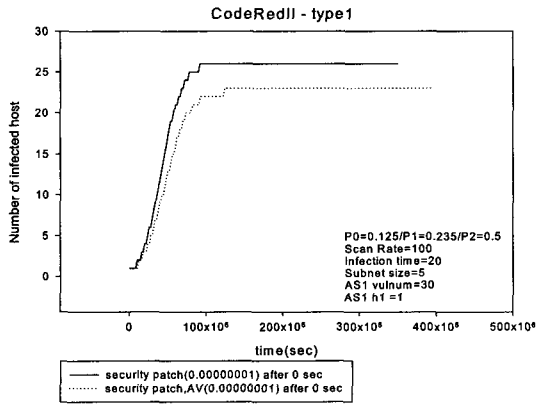
〔그림 16〕 CodeRed II-type1 모델링 결과 #1

설치비율 적용된 상태 비교 - [그림 18]

모델링 결과, [그림 16]에서 아무것도 적용하지 않은 상태에 비해 보안패치, 백신업데이트, 개인방화벽 최신버전을 설치한 경우 감염속도가 현저히 떨어지는 것을 확인할 수 있다. 이와 비슷하게 [그림 17]에서 보안패치만 일정 비율로 적용한 경우보다 보안패치와 백신프로그램 최신버전을 함께 적용한 경우의 감염속도가 떨어지는 현상을 볼 수 있다. 마지막으로 [그림 18]은 백신업데이트 최선버전 설치만을 고려하여 모델링한 결과인데 웜 전파와 동시에 백신업데이트가 적용된 경우는 취약한 호스트를 더 이상 감염시키지 못한 채 일정수준을 유지한다. 특정 시간이 지난 후 백신업데이트가 적용될 경우 취약한 호스트들을 모두 감염시킨 후 감염 호스트



〔그림 15〕 Nimda, CodeRedII, 임의의 웜 모델링 결과



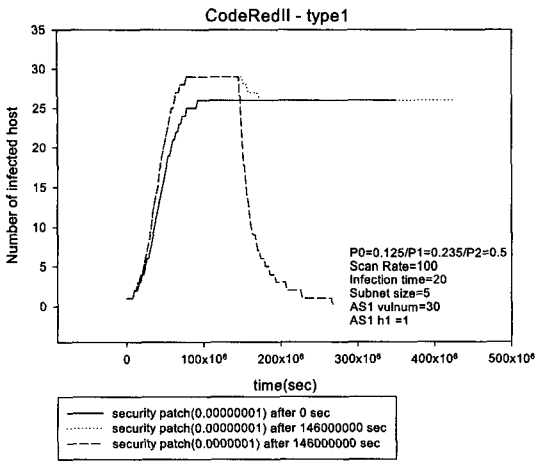
〔그림 17〕 CodeRed II-type1 모델링 결과 #2

수가 감소하며 백신업데이트 최신버전 설치 비율이 감염비율보다 높아질 경우 감염 호스트 수는 계속 감소하여 0이 됨을 알 수 있다.

Ⅳ. 결론 및 향후 연구

최근 많은 피해를 입히고 있는 웜에 대한 전파 모델링 기법에 대한 연구는 실제 네트워크에서 발생 가능한 웜 전파 특성에 대하여 빠르게 분석하고, 예측을 통한 대응방안을 마련하는데 매우 효과적이며 필요한 연구이다. 기존에 이에 대한 연구로 Epidemic, AAWP 및 LAAWP 등의 수학적 모델링 기법을 이용한 웜 전파 모델링 기법들의 연구가 이루어졌다.

하지만, 기존의 웜 전파 모델링 기법들은 대부분



〔그림 18〕 CodeRed II-type1 모델링 결과 #3

IPv4 전체 네트워크망(2^{32}개 호스트)에서 랜덤 스캐닝으로 전파되는 웜을 대상으로 하였으며, 보안패치 및 백신프로그램의 업데이트 등의 인간의 대응활동에 대한 표현에 한계점을 가지고 있으며, 특정 내부 네트워크에서 전파되는 웜의 전파특성을 모델링 하는데 한계가 있었다.

따라서, 본 논문에서는 이러한 단점을 보완하고자 AAWP 및 LAAWP 모델링 기법의 수식과 파라미터를 확장하는 새로운 웜 전파 모델링 기법인 ALAAWP (Advanced LAAWP Modeling) 모델링 기법을 제안하였다.

ALAAWP 모델링 기법은 웜이 사용하는 스캐닝 기법을 토대로 하여 IPv4 네트워크를 3가지 타입으로 분류하여 표현하므로 어떠한 크기의 서브넷도 모델링이 가능하다. 또한, 특정 서브넷 내부에서 전파되는 웜 전파속도의 표현은 물론 보안패치, 백신프로그램 업데이트, 개인방화벽 룰 업데이트 등의 인간에 의한 대응활동을 표현하고, 이들 방어대책에 대한 검증이 가능하다. ALAAWP 모델링 기법을 IPv4 전체 네트워크에 적용할 경우, 서브넷 타입별 감염 호스트 수와 그에 따른 전체 감염 호스트 수의 증가율을 함께 예측해볼 수 있다.

하지만, 본 논문에서 제안한 ALAAWP 모델링 기법은 서브넷과 서브넷 간을 연결시키는 네트워크 장비와 네트워크 속도 등을 고려하지 않았으며, 네트워크 혼잡 상태에 의해 웜의 스캔률이 감소되는 특성을 적용하지 않아, 실세계의 네트워크 환경을 완벽하게 반영하지는 못하고 있다. 향후에는 이러한 한계점의 극복을 위한 연구가 필요하다.

참고문헌

[1] http://www.trendmicro.com/kr/products/network/viruswall1200/evaluate/cases/kookminilbo.htm

[2] Cliff Changchun Zou, Weibo Gong, Don Towsley, "Worm Propagation Modeling and Analysis under Dynamic Quarantine Defense", Univ. Massachusetts Amherst, MA1

[3] Yong Huang, "Code-Red: a case study on the spread and victims of an Internet worm" 15-20, David Moore, Colleen Shannon, K Claffy CAIDA, San Diego Supercomputer Center, UCSD IMW 2002 Presented by: Yong Huang

[4] Hyundo Park, Heejo Lee, "Detection Unknown Worms Using Randomness Check", KOREA University

[5] Zesheng Chen, Chuanyi Ji, "Optimal worm-scanning method using vulnerable-host distributions"

[6] S. Staniford, V. Paxson, N. Weaver, "How to 0wn the Internet in your spare time," in Proc. of the 11th USENIX Security Symposium (Secu- rity'02), San Francisco, CA, Aug. 2002.

[7] N.Weaver, WarholWorms, "The Potential for Very Fast Internet Plagues", http://www.cs.berkeley.edu/~nweaver/warhol.html., august 15th, 2001

[8] Stuart Staniford, Vern Paxson, Nicholas Weaver, David Moore, "The Top Speed of Flash Worms", Workshop on Rapid Malcode, 2004

[9] http://www.caida.org/analysis/security/code-red/#crii

[10] http://www.caida.org/publications/papers/2003/sapphire/sapphire.html

[11] F.B Cohen, "A Formal Definition of Computer Worms and Some Related Results", Computers & Security, 7(11) (1992), pp. 641-

652, ISSN 0167-4048, 1992

[12] Dr. Winfried Gleissner, "A Mathematical Theory for the Spread of Computer Viruses", Computers & Security, 8, 1989, pp. 35-41, ISSN 0167-4048, February 1989

[13] Cliff Changchun Zou, Weibo Gong, Don Towsley "Code Red Worm Propagation Modeling and Analysis", Conference on Computer and Communications Security, 2002

[14] Zesheng Chen, Lixin Gao, Kevin Kwiat, "Modeling the Spread of Active Worms" pp. 1-11, ieee

[15] Brian D. Carrier, Sundararaman Jeyaraman, Sarah Sellke, "IMPACT OF NETWORK DESIGN ON WORM PROPAGATION", Center for Education and Research in Information Assurance and Security, Purdue University, West Lafayette, IN 47907-2086

[16] http://www.caida.org/analysis/security/code-red/#crii

[17] R. Russell and A. Machie, "Code Red II Worm," Tech. Rep., Incident Analysis, SecurityFocus, Aug. 2001.

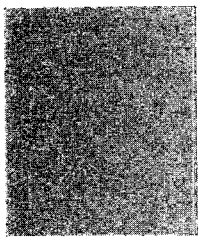
[18] A. Machie, J. Roculan, R. Russell, and M. V. Velzen, "Nimda Worm Analysis," Tech. Rep., Incident Analysis, SecurityFocus, Sept. 2001.

[19] CERT/CC, "CERT Advisory CA-2001-26 Nimda Worm," http://www.cert.org/advisories/CA-2001-26.html, Sept. 2001.

[20] D. Song, R. Malan, and R. Stone, "A Snapshot of Global Internet Worm Activity," Tech. Rep., Arbor Networks, Nov. 2001.

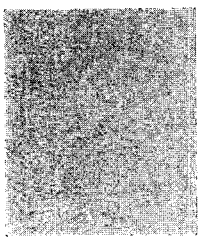
[21] J. O. Kephart and S. R. White, "Directed-graph Epidemiological Models of Computer Viruses," in Proc. of the 1991 IEEE Computer Society Symposium on Research in Security and Privacy, May 1991, pp. 343~359.

〈著者紹介〉



전 영 태 (Young-Tae Jun) 정회원
2002년 2월: 명지대학교 컴퓨터공학과 졸업
2003년 8월~현재: 고려대 정보보호대학원 석사과정
<관심분야> 컴퓨터 바이러스, 네트워크보안

문 종 섭 (Jong-Sub Moon) 정회원
1981년~1985년: 금성 통신 연구소 연구원
1991년: Illinois Institute of technology 졸업(전산학 박사)
1993년 ~ 현재: 고려대 전자 및 정보공학부 교수
<관심분야> 생체인식, 침입탐지, 운영체제



서 정 택 (Jung-Taek Seo) 종신회원
1999년 2월 : 국립충주대학교 컴퓨터공학과 졸업
2001년 2월: 아주대학교 컴퓨터공학과 공학석사 졸업
2006년 2월: 고려대학교 정보보호대학원 공학박사 졸업
2000년 11월~현재: ETRI 부설 연구소 선임연구원