情報保護學會論文誌
第17卷 第3號, 2007. 6

RCS 웜 시뮬레이션을 위한 Hybrid 모델링 방법

김정식†‡, 박진호, 조재익, 최경호, 임을규
한양대학교 정보통신대학

A Hybrid Modeling Method for RCS Worm Simulation

Jung Sik Kim†‡, Jin Ho Park, Jae Ik Cho, Kyoung Ho Choi, Eul Gyu Im
Hanyany University

요 약

인터넷에 대한 의존도가 증가하면서 인터넷 웜에 대한 연구의 필요성이 증가하게 되었다. 인터넷 웜을 연구하는 데 가장 많이 사용하는 방법 중의 하나는 시뮬레이션인데, 대규모 네트워크상에서 동작하는 웜을 시뮬레이션 하는 데에는 성능, 확장성 등의 문제가 발생한다. 이에 본 논문에서는 대규모 인터넷 웜, 특히 RCS(Random Constant Spreading) 특성을 갖는 웜을 시뮬레이션 할 때 발생하는 문제점을 줄여, 효율적인 시뮬레이션이 가능하도록 하는 hybrid 모델링 방법을 제안하였다. 본 논문에서 제안하는 hybrid 모델은 epidemic 모델과 유체 모델을 사용한 모델링 네트워크와 패킷 네트워크의 연동을 통하여 시뮬레이션을 수행하도록 하였으며, 이로 인하여 일반적인 모델링 기법의 장점인 빠른 수행 시간을 가짐과 동시에 패킷 네트워크를 이용하여 동적으로 인자값을 업데이트할 수 있게 되었다. 또한, 한 번의 시뮬레이션을 통해 모델링 네트워크로부터 거시적인 정보와 패킷 네트워크로부터 세부적인 정보를 모두 얻을 수 있다. 그리고 본 논문에서는 RCS 특성을 가지는 웜의 한 종류인 코드레드 웜에 대한 실험을 수행하여 hybrid 모델의 적합성을 보여주었다.

ABSTRACT

Internet becomes more and more popular, and most companies and institutes use web services for e-business and many other purposes. With the explosion of Internet, the occurrence of cyber terrorism has grown very rapidly. Simulation is one of the most widely used method to study internet worms. But, it is quite challenging to simulate very large-scale worm attacks because of various reasons. In this paper, we propose a hybrid modeling method for RCS(Random Constant Spreading) worm simulation. The proposed hybrid model simulates worm attacks by synchronizing modeling network and packet network. So, this model will be both detailed enough to generate realistic packet traffic, and efficient enough to model a worm spreading through the Internet. Moreover, our model have the capability of dynamic updates of the modeling parameters. Finally, we simulate the hybrid model with the CodeRed worm to show validity of our proposed model for RCS worm simulation.

Keywords : *RCS Worm Simulation, Hybrid Model*

Ⅰ. 서 론

인터넷이 널리 보급되면서, 이와 비례해서 인터넷 웜에 의한 피해도 증가하였다. 인터넷 웜은 1988년 모리스 웜을 시작으로 2001년 코드 레드 웜, 2003년 슬래머

접수일: 2006년 12월 6일; 채택일: 2007년 3월 15일
† 주저자, bisa1004@hanmail.net
‡ 교신저자, bisa1004@hanmail.net

웜 등이 출현하며 막대한 피해를 입혔다. 이 중 슬래머 웜의 경우는 전 세계적으로 최소 7만 5천대 이상의 호스트가 감염되었는데, 빠른 감염속도로 인해 대부분의 호스트가 웜 전파 시작 이후, 채 10분도 지나지 않아서 감염되어 더 큰 피해를 입혔다[1]. 이러한 피해를 최소화하기 위해 웜의 특성을 연구하여 여러 가지 방어책을 테스트하는 것이 요구되고 있다.

웜에 대한 연구는 한 호스트에서 이루어지는 웜의 개별적인 동작 방법을 연구하거나 네트워크상에서 이루어지는 전파 현상을 연구하는 것으로 나눌 수 있다. 웜의 개별적인 동작 방법에 대한 연구는 역어셈블을 이용한 분석 방법이 많이 사용되고 있고, 네트워크에서 전파 현상을 연구하는 방법으로는 시뮬레이션을 이용한 방법이 많이 사용되고 있다. 웜의 전파현상을 연구하기 위해서는 실제 네트워크에서 웜의 전파를 관찰하는 것이 가장 정확한 결과를 나타낸다. 하지만 웜은 대부분 수십만 호스트이상으로 구성되어있는 대규모 네트워크상에서 동작을 하기 때문에 실제 네트워크를 구성하여 웜을 연구하는 것은 인력, 공간, 비용의 문제로 현실적으로 불가능하기 때문에 대안적인 방법으로 시뮬레이션을 이용한 방법을 많이 사용하고 있다.

시뮬레이션을 이용하여 웜의 전파를 연구하게 될 경우, 공간, 비용적인 문제에 대한 제약이 상당수 사라지게 되어 다수 호스트로 구성된 네트워크를 구성하여 연구를 하는 것이 가능하게 된다. 이러한 이유로 시뮬레이션을 이용한 방법은 웜의 전파현상을 연구하는데 크게 도움이 되지만 여전히 문제점을 안고 있다. 상대적으로 작은 규모의 네트워크상에서 웜을 시뮬레이션 하는 경우에는 큰 문제가 발생하지 않지만, 수십만 호스트 이상으로 구성된 네트워크에서 시뮬레이션을 수행할 경우에는 수행 시간이 급격하게 증가하는 문제가 발생한다. 시뮬레이션 네트워크를 구성하는 호스트 수가 증가하게 되면 이에 비례하여 호스트별로 발생하는 이벤트도 증가하게 된다. 따라서 각 호스트별 이벤트의 발생 및 처리를 각각 처리할 경우, 전체적으로 처리해야 하는 계산량은 기하급수적으로 증가하게 된다.

이러한 문제를 해결하기 위하여, 즉 시뮬레이션 시 발생하는 계산량을 줄이기 위하여 다양한 모델링 방법이 제안되었다. 모델링을 이용한 시뮬레이션이란, 네트워크 또는 네트워크에서 발생하는 현상을 수학적인 식으로 이루어진 모델을 통해 시뮬레이션을 수행하는 방법이다. 이러한 방법은 네트워크의 규모가 증가하여도 수학적인 계산을 통해 네트워크의 현상을 파악하기 때문에 일반적인 시뮬레이션과 같이 수행시간이 급격히 증가하는 현상이 나타나지 않는다. 하지만 모델링을 이용한 시뮬레이션에서는 모델링된 네트워크가 실제 네트워크를 단순화, 추상화시킨 네트워크이기 때문에 시뮬레이션 수행 시 실제 패킷의 교환 등의 세부 이벤트가 표현되지 않아서 자세한 네트워크 상태를 알아보기 어렵고, 실제 네트워크에서 발생할 수 있는 현상을 동적으로 적용하기 어렵다.

일반적인 시뮬레이션 방법과 모델링 시뮬레이션은 이렇게 장·단점을 가지고 있는데, 본 논문에서는 두 방법을 적절히 조화시킨 hybrid 모델을 이용한 시뮬레이션 방법을 제안한다. 본 논문에서 제안한 hybrid 모델링 방법은 다양한 웜 중에서 특히 RCS(Random Constant Spreading)[13] 특성을 갖는 웜의 전파현상을 시뮬레이션 하는데 효과적인 모델링 방법이다. 또한 본 논문에서는 CodeRed V2 웜의 시뮬레이션을 통하여 제안한 모델링 방법이 실제 웜의 전파를 잘 표현할 수 있는지 실험하였다.

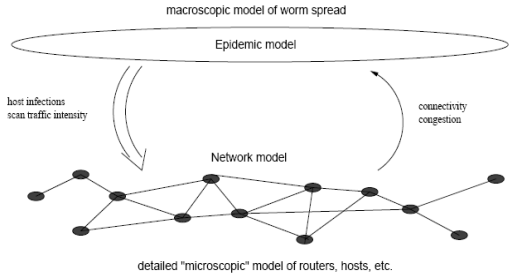
본 논문의 구성은 다음과 같다. Ⅱ장에서는 대규모 네트워크 시뮬레이션에 사용된 기존의 모델링 방법을 정리하고 기존 방법의 문제에 대해 정의하였다. Ⅲ장에서는 웜 시뮬레이션을 위한 hybrid 모델링 방법을 제안하였고, Ⅳ장에서는 제안한 모델을 SSFNet을 이용하여 구현하고 시뮬레이션을 수행하였다. 마지막으로 Ⅴ장에서 향후 연구방향을 제시하고, 결론을 맺었다.

Ⅱ. 관련연구

2.1. 기존 연구

기존 연구에서는 여러 모델을 이용하여 모델링 네트워크 시뮬레이션을 수행하는 방법에 대해 연구를 진행하였다. 이 장에서는 모델링 방법 중 대규모 네트워크 시뮬레이션에 많이 사용하는 유체 모델, Epidemic 모델 등을 이용한 모델링 방법에 대해 알아보도록 한다.

Misra 등은 TCP 트래픽 모델링을 위해 유체 모델을 기반으로 한 동적 유체 흐름 모델(Dynamic Fluid Flow Model)을 제안하였다[2]. 그리고 Liu 등은 이 모델링 방법을 수정한 모델링 방법을 제안하였는데 이로 인해 네트워크 내부의 트래픽 전달을 좀 더 정확하게 표현할 수 있게 되었다[3]. Liu 등이 제안한 수정된 유체 모델을 살펴보면, 트래픽 흐름에 대하여 동일한 경로와 전송 지



〔그림 1〕 Epidemic 모델을 사용한 시뮬레이션 시 네트워크 구성도

연 값을 갖는 클래스들을 정의하고, 각 노드에서의 출발 비율과 도착 비율을 계산하여 전체 네트워크의 흐름을 표현하였다. 이러한 유체 모델을 이용한 모델링 방법의 주요 목적은 백본 네트워크에 흐르는 대규모 트래픽을 효과적으로 표현하기 위한 것이다. 하지만 이러한 모델링 방법을 이용하여 패킷레벨의 트래픽 변화를 효과적으로 표현하는 데는 한계가 있다. Liu 등에 의한 수정된 유체 모델링 방법에서는 패킷 트래픽의 흐름을 미리 정의하고 이러한 패킷 트래픽과 대규모 백본 트래픽을 함께 고려하여 모델링에서 사용되는 인자 값을 구한 후에 이를 이용하여 모델링을 수행하였다. 이러한 수정된 방법에서도 패킷 트래픽의 변화에 따른 전체 트래픽의 변화는 표현하기 어려운 실정이다.

Epidemic 모델을 이용한 모델링 방법은 원래 생물학 분야에서 생물학적 바이러스 등의 감염을 모델링할 때 많이 사용되어온 모델링 방법이었다. 이 모델링 방법에서는 감염가능 객체 수, 감염된 객체 수 및 치료된 객체 수를 계산해 주는 모델로, 웜의 전파를 연구하는데 효과적인 모델이다.[4][5] 이 모델에서는 전체 호스트의 개수를 N개라고 가정할 때, 다음 수식을 통해 네트워크의 상태를 파악하게 된다.

$$\begin{aligned} \frac{ds(t)}{dt} &= -\beta s(t)i(t) \\ \frac{di(t)}{dt} &= \beta s(t)i(t) - \gamma i(t) \\ \frac{dr(t)}{dt} &= \gamma i(t) \end{aligned} \qquad (1)$$

단, $s(t) + i(t) + r(t) = N, \forall t \geq 0$

수식 (1)에서 상수 β는 감염 파라미터 (감염 비율), Υ는 삭제 파라미터, *s(t)*는 *t* 시점에서의 감염가능(susceptible) 호스트 수를 나타내고, *i(t)*는 *t* 시점에서의 감염된(infected) 호스트 수를, *r(t)*는 *t* 시점에서의 치료된(removed) 호스트 수를 나타낸다. 이 모델은 네트워크 호스트의 수가 충분히 커서 확률적 시스템의 변화는 충분히 예측할 수 있고, 호스트 사이의 상호작용이 거의 일정하다고 가정하였다.

Epidemic 모델을 이용하여 시뮬레이션을 수행할 경우, 〔그림 1〕과 같이 네트워크를 광범위 모델(Macroscopic model)과 극소범위 모델(Microscopic model)로 나누어 시뮬레이션을 수행하게 된다. 광범위 모델은 호스트의 감염을 나타내고, 극소범위 모델은 백본 네트워크의 연결을 표현한다. 즉, 백본 네트워크의 연결여부 및 속도에 따라서 어느 서브네트워크가 어느 정도의 속도로 감염되는지가 결정되게 된다. 하지만 이 시뮬레이션에서는 광범위 모델에 중점을 두었기 때문에 극소범위 모델의 정보교환은 대부분 시뮬레이션을 수행하기 전에 인자값이 결정되는 정적인 정보의 교환이 이루어지게 된다.

이외에도 패킷 단위 네트워크와 모델링 네트워크가 혼재되어 동작하는 여러 기법이 제안되었는데, Global Mobile Information System Simulator (GloMoSim)[10]은 유체 모델을 이용하는 컴포넌트와 패킷 단위의 시뮬레이션을 수행하는 모델로 나뉘어져 있으며, 유체 모델을 통하여 전송지연과 패킷 손실을 계산하여 트래픽의 흐름을 표현하고, 패킷 단위의 모델에서는 유체 모델의 트래픽을 고려하여 개별 패킷의 전송 지연과 손실율을 계산하게 된다. 이 모델링 방법에서는 유체 모델에서 표현되는 트래픽 량이 패킷 단위의 모델을 이용하여 표현되는 트래픽 량보다 월등히 많으므로, 두 모델사이의 상호작용은 무시된다. 이와 유사한 연구로서 Hybrid Discrete-Continuous Flow Network Simulator (HDCF-NS)[11]와 Kiddle[12] 등이 제안한 혼성 기법이 있다. 또한, Kalyan S. Perumalla[9]등은 실제 네트워크와 가상 네트워크가 함께 동작하는 high-fidelity 모델링 방법을 제안하였다. 이 방법들은 실제 네트워크를 구성했다는 점에서 시뮬레이션의 유연성을 크게 향상시켰다고 할 수 있으나, 실제 네트워크와 가상 네트워크 사이의 트래픽 교환 표현 방법, 실제 네트워크의 확장성 부족 등의 한계로 인하여 이러한 방법의 활용이 제한되고 있다.

2.2. 문제 정의

웜의 전파 현상을 연구하는 방법으로 실제 호스트를

구성하여 네트워크를 구축한 다음, 이 네트워크상에서 웜을 전파시키는 것이 가장 정확한 결과를 나타낸다는 것은 쉽게 알 수 있다. 하지만 일반적으로 웜의 전파 현상을 연구하기 위해서는 수만 호스트 이상으로 구성된 네트워크를 구축해야 하는데, 네트워크를 구성하는데 드는 비용, 인력, 시간, 공간을 생각하면 이러한 방법은 현실적으로 불가능한 방법이라는 것을 알 수 있다. 이에 대한 대안으로 시뮬레이션 방법이 널리 사용되고 있다. 네트워크 시뮬레이션에 사용하는 방법은 크게 다음 두 가지로 분류할 수 있다. 첫째, 패킷 또는 노드 수준에서 트래픽에 대한 시뮬레이션을 수행하는 방법으로, 네트워크를 이루고 있는 호스트와 라우터를 가상으로 구성하여 네트워크 트래픽을 시뮬레이션하는 방법이고, 둘째, 해당 네트워크의 특성을 분석하여 수학적 모델을 이용하여 시뮬레이션을 수행하는 방법이다. 어떤 방법을 선택하여 시뮬레이션을 수행하는가는 연구 목적에 따라 다르지만 두 방법 모두 장단점이 존재한다.

실제 네트워크와 비슷하게 가상의 시뮬레이션 네트워크를 구성하여 시뮬레이션을 수행하면, 네트워크의 현상을 패킷 단위의 정보까지 파악할 수 있다. 또, 시뮬레이션 네트워크를 구성해 두면 다른 시뮬레이션 시나리오도 약간의 조작만으로 쉽게 수행해 볼 수 있다는 장점이 있다. 하지만 전체 네트워크를 모두 구성하여 시뮬레이션을 수행하는 방법은 인터넷 웜의 시뮬레이션과 같이 대규모 네트워크에 대한 시뮬레이션을 수행해야 할 경우에는 적합하지 않다. 왜냐 하면 시뮬레이션의 수행시간과 노드 수가 증가함에 따라 전체적인 계산 량이 기하급수적으로 증가하기 때문이다. 또한, 웜의 전파에 대한 시뮬레이션은 네트워크에서 발생하는 현상을 거시적으로 관찰하는 경우가 많기 때문에, 세부적인 정보를 파악하는 것은 큰 이점이 되지 못한다.

따라서 이러한 문제를 해결하기 위하여 모델링을 이용한 네트워크 시뮬레이션에서는 실제 네트워크를 수학적 모델에 맞추어 모델링하여 수학적인 계산을 통해 시뮬레이션을 수행하는 방법을 취한다. 모델링 네트워크 시뮬레이션 방법을 사용할 경우 시뮬레이션 수행 속도를 크게 단축시키게 되는데, 이는 네트워크에서 발생하는 이벤트를 모두 처리하지 않고, 실제 네트워크에서 발생하는 현상을 특정 수식 또는 간단한 연산을 통해 계산하기 때문에 네트워크의 규모가 증가하더라도 처리해야 하는 연산의 수는 크게 증가하지 않기 때문이다. 이러한 모델링 방법을 통하여 전체적인 연산 시간을 크게 단축시킬 수 있으나, 모델링은 통한 추상화, 단순화 과정을 통하여 많은 정보가 축약되거나 상실되게 된다. 이러한 모델링을 이용한 시뮬레이션 방법은 거시적인 현상을 관찰하면서도 일부 경우에 대하여 상세한 정보가 필요한 경우에 적용하기 어려운 문제점이 있다.

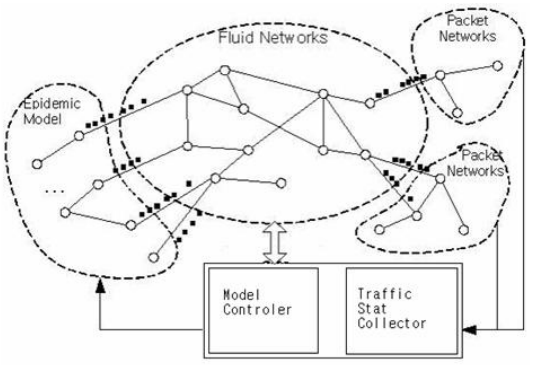
새로운 웜이 발생하여 전파될 때, 대규모 네트워크상에서 어떻게 생성·소멸되는지에 대한 시뮬레이션을 수행하기 위해서는 모델링을 이용한 시뮬레이션 방법과 패킷 단위의 시뮬레이션이 모두 필요하다. 또한 새로운 웜의 특성이 명확하지 않아서 다양한 인자를 적용하여 시뮬레이션하려 할 때, 시뮬레이션 인자값, 모델링에 사용되는 인자값 및 패킷 단위의 시뮬레이션에 사용되는 인자값을 동적으로 변화시키면서 이러한 변화에 따른 웜의 현상을 분석하는 것이 필요하다.

Ⅲ. 제안하는 Hybrid 모델

앞에서 본 바와 같이, 대규모 네트워크의 시뮬레이션을 수행하기 위해서 수 만개의 호스트를 모두 시뮬레이션 네트워크로 구성하는 것은 수행 시간의 측면에서 매우 비효율적이다. 반대로 모델링 네트워크를 사용하여 시뮬레이션을 수행할 경우에는 네트워크를 낮은 수준까지 관찰하기 힘들고, 네트워크에서 발생하는 현상을 동적으로 반영하기 힘들다. 이러한 문제들을 해결하기 위해, 이 장에서는 두 방법을 혼합한 형태의 hybrid 모델링 방법을 이용한 시뮬레이션 방법을 제안한다.

Hybrid 모델링 방법은 대규모 네트워크상에서 주로 RCS 특성을 갖는 인터넷 웜에 대한 시뮬레이션을 수행하는 것을 목적으로 한다. Hybrid 모델링 방법에서는 〔그림 2〕에서 보는 바와 같이 크게 모델링 네트워크와 패킷 네트워크로 구성된다. 모델링 네트워크는 유체 모델과 epidemic 모델을 이용하여 구성되는데, epidemic 모델을 이용하여 호스트의 감염을 주로 표현하고, 유체 모델을 이용하여 이러한 호스트들을 연결하는 백본 네트워크의 트래픽이 표현된다. 이러한 두 모델이 서로 연동하여 하나의 모델처럼 동작하게 된다.

이러한 모델링 네트워크에 패킷 네트워크라고 부르는 소규모 네트워크가 연결이 되어있는데, 패킷 네트워크는 네트워크에서 예측하지 못한 이벤트를 발생시켜주는 역할을 하며, 낮은 수준의 정보를 관찰할 수 있도록 해준다. 마지막으로 모델링 네트워크, 패킷 네트워크가 자연스럽게 연동되도록 해주는 모델 제어기 및 정보 수



〔그림 2〕 Hybrid 모델의 구조

집기가 존재한다.

3.1. 모델링 네트워크

Hybrid 모델에서는 유체 모델과 epidemic 모델을 연동시킨 모델링 네트워크를 사용하게 된다. Epidemic 모델은 인터넷 웜의 전파 현상을 시뮬레이션 하는데 효과적이고, 유체 모델은 대규모 네트워크 트래픽 계산에 효과적이다. 따라서 hybrid 모델에서는 인터넷 웜을 시뮬레이션 하기 위해 epidemic 모델을 사용하게 되고, epidemic 모델의 단점인 정적인 백그라운드 트래픽 생성 문제를 해결하기 위해 유체 모델을 연동시키게 된다. 유체 모델과 epidemic 모델은 상호 보완을 하며 동작하게 되는데, 다음은 두 모델의 연동 방법에 대한 설명이다.

- Epidemic 모델 : 유체 모델은 모델링 네트워크에 입력된 트래픽을 기초로 네트워크의 출력 트래픽과 발생하는 트래픽 손실률을 계산해 주게 된다. Epidemic 모델은 이 중 트래픽 손실률을 사용하여 모델링 인자 값을 업데이트한다. Epidemic 모델에는 웜의 감염되는 비율을 나타내는 변수 β가 존재하는데 모델링 네트워크에서 트래픽의 손실률이 높아지게 되면 웜 패킷의 손실도 그만큼 증가하게 된다. 즉, 웜 패킷이 전송될 확률이 그만큼 낮아지게 된다. 따라서 유체 모델의 수행 결과로, 모델링 네트워크에서 트래픽의 손실이 발생하게 되면 그 비율을 β값에 적용시켜 웜의 전파 속도를 감소시켜 주게 된다. 다음은 손실률을 이용하여 β값을 계산하는 수식이다.

$$\beta = \beta_0 \times (1 - \text{손실률}) \quad (2)$$

여기서 β_0는 손실률이 적용되지 않은 초기의 β값이다.

- 유체 모델 : Epidemic 모델은 총 호스트의 수, 감염된 호스트의 수, 치료된 호스트의 수를 가지고 있다. 이 중 감염된 호스트의 수와 총 호스트의 수를 이용하면 감염된 호스트의 비율을 구할 수 있는데, 유체 모델에는 이 값을 이용하여 모델링 인자 값을 수정한다. 모델링 네트워크에서 많은 수의 호스트가 웜에 감염된다면 그 네트워크는 감염된 호스트의 수와 비례하는 양의 웜 패킷을 발생시킨다. 감염된 호스트의 비율이 증가하면 모델링 네트워크에서 발생하는 트래픽의 양도 증가한다는 것을 의미한다. 따라서 감염된 호스트의 비율이 유체 모델의 입력 트래픽에 영향을 주게 되는데, 다음은 유체 모델의 입력 트래픽을 구하는 수식이다.

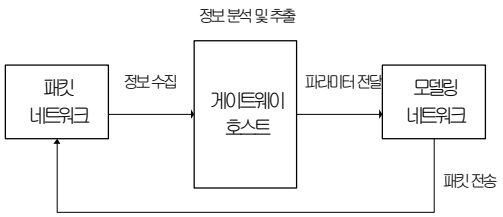
$$\text{Inbound traffic} = \text{Network size} \times \text{Infected rate} \times k \quad (3)$$

여기서 k는 웜의 감염에 따른 호스트별 패킷 생성 속도이다.

3.2. 패킷 네트워크

패킷 네트워크는 모델링 네트워크와 연결되어 시뮬레이션을 수행하는 네트워크로 모델링 네트워크에서 표현하지 못하는 패킷 단위의 정보를 가지고 있으며, 그 밖에 모델링 네트워크에서는 발생하기 힘든 현상도 표현할 수 있다. 패킷 네트워크는 일반적인 소규모 시뮬레이션 네트워크와 같이 구성되는데, 웜을 시뮬레이션하는데 그 목적이 있으므로 웜의 감염 및 치료현상이 표현 가능하여야 한다.

패킷 네트워크에서 시뮬레이션을 통하여 수집되는 정보를 모델링 네트워크에 적용하여 모델링 네트워크를 동적으로 업데이트할 수 있게 된다. 하지만 정확한 결과를 얻기 위해서는 패킷 네트워크를 구성할 때, 발생하는 몇 가지 문제가 있는데, 패킷 네트워크와 모델링 네트워크의 시간 연동 문제가 대표적이다. 모델링 네트워크는 수학적인 계산을 통해 시간을 계산하고, 패킷 네트워크는 매 클럭마다 이벤트가 발생하기 때문에 모델링 네트



〔그림 3〕 게이트웨이 호스트의 위치

워크와 패킷 네트워크를 동시에 동작시킨다 하더라도 두 네트워크의 시간이 맞지 않게 된다. 그렇기 때문에 모델링 네트워크와 패킷 네트워크의 시뮬레이션을 관리하는 제어 모듈에서 시간 연동을 해 주어야 한다. 이 밖에도 특정 웜 전파의 구현, 패킷 네트워크의 유효 크기에 관한 문제 등이 발생할 수 있다.

3.3. 게이트웨이 호스트

패킷 네트워크와 모델링 네트워크를 연결하려면 네트워크의 정보를 수집하는 정보 수집기와 수집된 정보로부터 추출된 파라미터를 업데이트해주는 순서 등을 관리하는 모델 제어기가 필요하게 된다. Hybrid 모델에서는 '게이트웨이'라 부르는 하나의 호스트가 이 역할을 수행한다. 〔그림 3〕은 hybrid 모델에서 게이트웨이 호스트가 존재하는 위치를 보여준다. 게이트웨이 호스트는 패킷 네트워크와 모델링 네트워크가 연결되는 중간에 위치하여 패킷 네트워크로부터 정보를 수집, 가공하여 특정 파라미터를 추출하고, 이 파라미터를 모델링 네트워크에 전달해 주게 된다. 하지만 모델링 네트워크에서 패킷 네트워크로 패킷을 전송할 경우에는 게이트웨이 호스트를 거치지 않고 곧바로 전송한다.

게이트웨이 호스트를 매개로, 패킷 네트워크와 모델링 네트워크는 정보를 교환하게 된다. 대부분의 정보는 패킷 네트워크에서 모델링 네트워크로 전달되게 되는데, 패킷 네트워크에서 얻을 수 있는 정보는 다음과 같다.

- 감염 가능한 호스트의 수(# of susceptible hosts)
- 감염된 호스트의 수(# of infected hosts)
- 치료된 호스트의 수(# of removed hosts)
- 호스트 당 발생하는 트래픽

패킷 네트워크로부터 수집된 정보는 모델링 네트워크에서 사용하는 파리미터로 변환되게 되는데, 이 파라미터는 감염률과 치료율이다. 감염률은 감염된 호스트의 수를 이용하여 계산해 주게 되고, 치료율은 치료된 호스트의 수를 이용하여 계산한다. 이 파라미터는 패킷 네트워크로부터 실시간으로 수집되어, 모델링 네트워크에 동적으로 업데이트되게 된다.

Ⅳ. 실 험

본 논문에서 제안한 hybrid 모델을 이용하여 Code-red v2 웜에 대해 시뮬레이션을 수행해 보았다.

4.1. 시뮬레이션 환경

본 실험은 event-driven 방식의 네트워크 시뮬레이터인 SSFNet[6]을 사용하여 수행하였다. SSFNet은 대표적인 네트워크 시뮬레이터인 NS-2[7]에 비해 사용량이 적지만, 대규모 시뮬레이션 수행을 효과적으로 지원해 준다. SSFNet은 Java를 이용하여 구현되어 있으며, TCP/IP 및 네트워크 인터페이스 프로토콜까지 정의되어 있다.

실제 시뮬레이션을 수행한 컴퓨터 환경은 〔표 1〕과 같고, 시뮬레이션에서 사용된 네트워크의 초기 설정은 〔표 2〕와 같다. 웜의 전파에 많은 영향을 미치는 것으로 최초 감염 호스트의 수와 웜의 초기 감염률이 있다. 본 실험에서 최초 감염 호스트의 수, 감염 시작 시간은 네트워크의 구분 없이 1로 동일하고, epidemic 모델에서 사용되는 초기 감염률은 패킷 네트워크가 존재하지 않을 때는 1.6으로 code-red v2의 실제 감염률을 사용하였고, 패킷 네트워크가 존재할 때에는 패킷 네트워크로부터 감염률을 계산하기 때문에 초기 감염률을 0으로 설정하였다.

〔표 1〕 시뮬레이션 환경

CPU	Intel Pentium 3.0 GHz
RAM	1GB RAM
OS	Debian GNU/Linux
Simulator	SSFNet 2.0

〔표 2〕 초기 네트워크 설정

	패킷 네트워크	모델링 네트워크 (패킷 네트워크가 연결되지 않은 경우)	모델링 네트워크 (패킷 네트워크가 연결된 경우)
총 호스트의 수	200	360,000	360,000
최초 감염 호스트의 수	1	1	1
Code-red v2의 초기 감염률	-	1.6	0
감염 시작 시간 (초)	1	1	1

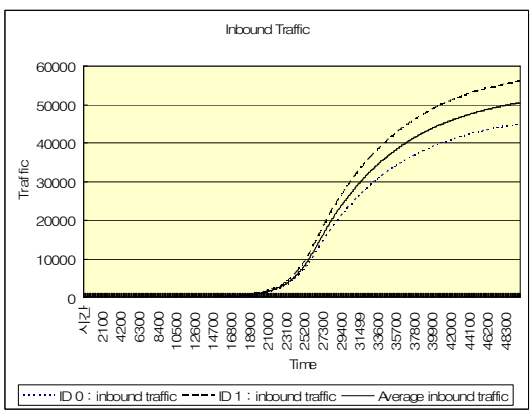
```
----------------------------------------------------------
t: 29820.0 infect Ratio : 0.45752072
id: 0 inbound traffic: 21960 outbound traffic: 12500 network size: 120000 drop rate: 0.43078327178955078
id: 1 inbound traffic: 27451 outbound traffic: 12500 network size: 150000 drop rate: 0.54462659358978271
t: 29820.0 Average drop rate : 0.48770493
t: 29820.0 previous beta : 6.3744493E-10, current beta : 6.3256844E-10
wormModel : 6.3256844E-10
t: 29820.0 Macroscopic model update, new t=29820.0 s, i=165928.1
----------------------------------------------------------
t: 29880.0 infect Ratio : 0.46091238
id: 0 inbound traffic: 22129 outbound traffic: 12500 network size: 120000 drop rate: 0.43495160341262817
id: 1 inbound traffic: 27654 outbound traffic: 12500 network size: 150000 drop rate: 0.54796948029899597
t: 29880.0 Average drop rate : 0.49146056
t: 29880.0 previous beta : 6.3256844E-10, current beta : 6.2804628E-10
wormModel : 6.2804628E-10
t: 29880.0 Macroscopic model update, new t=29880.0 s, i=167141.55
----------------------------------------------------------
t: 29940.002 infect Ratio : 0.46428186
id: 0 inbound traffic: 22285 outbound traffic: 12500 network size: 120000 drop rate: 0.43905943632125854
id: 1 inbound traffic: 27856 outbound traffic: 12500 network size: 150000 drop rate: 0.55124753713607788
```

〔그림 4〕 모델링 네트워크의 실험 결과 (일부)

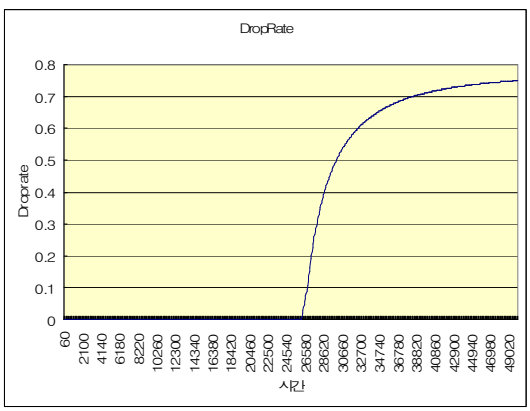
4.2. 모델링 네트워크

패킷 네트워크까지 포함된 완전한 hybrid 모델을 이용하여 코드레드 웜을 시뮬레이션하기 전에 모델링 네트워크가 정상적으로 연동되어 작동하는지에 대해 시뮬레이션을 수행해 보았다. 이 장에서 실험한 네트워크에는 패킷 네트워크가 추가되어 있지 않기 때문에 모든 시뮬레이션에서 사용한 웜의 초기 감염률은 1.6이다. 또, 웜이 감염되는 현상만을 실험하였으므로, 치료현상은 적용되지 않았다. 〔그림 4〕는 시뮬레이션을 수행을 때 실제 출력 화면의 일부이다. 이 시뮬레이션 결과를 정리하여 그래프로 그린 것이 〔그림 5〕부터 〔그림 8〕까지 이다.

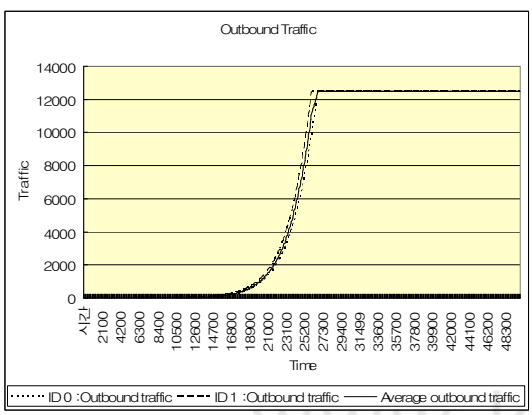
〔그림 5〕, 〔그림 6〕은 유체 모델의 결과로 시간의 변화에 따른 입력 트래픽과 출력 트래픽을 보여준다. 네트워크의 트래픽 처리량은 한계가 존재하므로 입력 트래픽이 일정 이상 증가하게 되면 출력 트래픽이 더 이상 증가량을 따라가지 못하고 〔그림 7〕에서와 같이 트래픽 손실이 발생하는 것을 볼 수 있다. 덧붙여, 〔그림 7〕의 트래픽 손실률 그래프는 〔그림 5〕, 〔그림 6〕의 평균



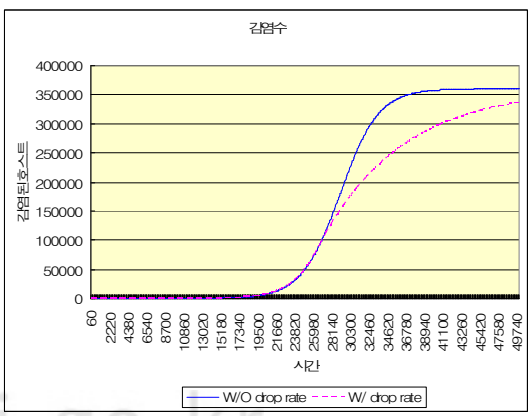
〔그림 5〕 유체 모델로의 입력 트래픽 량



〔그림 7〕 유체 모델의 트래픽 손실률



〔그림 6〕 유체 모델로부터의 출력 트래픽 량



〔그림 8〕 시간 당 감염 호스트의 수

입출력 그래프만을 이용해 계산되었기 때문에 1개의 그래프만 존재한다. 〔그림 8〕은 epidemic 모델의 결과(W/ Drop rate)인데, 비교를 위해 유체 모델이 적용되지 않았을 때(W/O Drop rate)의 결과와 같이 그려주었다. 그림에서 유체 모델을 적용해 주게 되면, 트래픽의 증가에 따라 감염률이 감소하는 것을 볼 수 있다.

4.3. 모델링 방법에 따른 실행 속도 비교

〔그림 9〕는 모델링 방법에 따른 시뮬레이션 실행속도 비교 결과를 나타내고 있다. 이 실험에서는 기존 epidemic 모델링 방법에서 제안한 바와 같이 epidemic 모델과 네트워크 모델(즉, 패킷 네트워크)을 이용하여 네트워크를 구성하고 실험했을 경우, 구성하는 네트워크 모델의 크기 변화에 따른 수행시간을 구하였다. 네트워크 모델은 앞서 언급한 바와 같이 백본 네트워크를 표현하기 위한 것으로 일반적으로는 소규모 네트워크로 구성되나, 시뮬레이션 하고자 하는 네트워크의 크기가 증가할 경우 네트워크 모델의 크기도 증가할 수 있다. 이러한 이유로 본 논문에서 제안하는 방식에서는 모델링 네트워크에서 epidemic 모델과 유체 모델을 이용하여 전체 네트워크를 표현할 수 있도록 하였다. 다시 말해서, 패킷 네트워크로 백본 네트워크를 표현하지 않고 유체 모델을 이용하여 백본 네트워크를 표현하도록 하였다. 〔그림 9〕와 〔표 3〕은 이러한 두 가지 구성 방법의 시뮬레이션 수행시간을 비교하고 있다.

위 결과에서 알 수 있듯이 백본 네트워크를 직접 패킷 네트워크로 구성할 경우, 네트워크 크기 증가에 따라서 전체 시뮬레이션 수행시간이 증가하게 되지만, 유체 모델을 이용하여 표현할 경우에는 네트워크 크기 증가에 따른 영향을 크게 받지 않는다. 위 실험은 Intel Pentium 4 CPU(3.20 GHz)/1 GB RAM/RedHat Linux Enterprise verion 3 환경에서 SSFNet version 2.0으로 시뮬레이션을 수행한 결과이다.

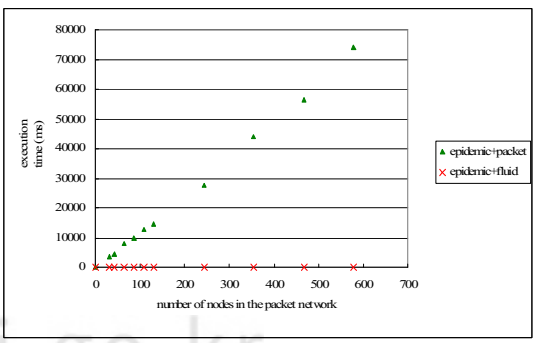
4.4. 코드레드 웜(Code-Red Worm V2)

코드레드 웜은 2001년 7월에 발생하여 전 세계적으로 큰 피해를 입힌 인터넷 웜이다. 감염 시작 14시간 안에 359,000 호스트 이상을 감염시켰는데, 최고 1분당 2,000 호스트 이상을 감염시켰다고 한다. 〔그림 10〕는 실제 코드레드 웜의 감염 그래프이다[8]. 이 그림은 모델링 네트워크를 시뮬레이션 한 결과인 〔그림 9〕과 매우 유사한 그래프를 그린다는 것을 볼 수 있는데, 두 그래프만을 비교하여 보았을 때는 모델링 네트워크만을 이용하여 시뮬레이션을 수행하여도 충분히 RCS 모델의 특성을 가지는 웜의 시뮬레이션을 수행할 수 있을 것으로 보인다. 하지만 모델링 네트워크는 초기 네트워크 설정이 변하지 않으면 시뮬레이션 중간에 네트워크 변화를 전혀 고려하지 않고, 항상 같은 결과를 출력해 준다. 따라서 모델링 네트워크만을 사용한 시뮬레이션은 여러 가지 네트워크 현상을 동적으로 반영하기 힘들게 된다.

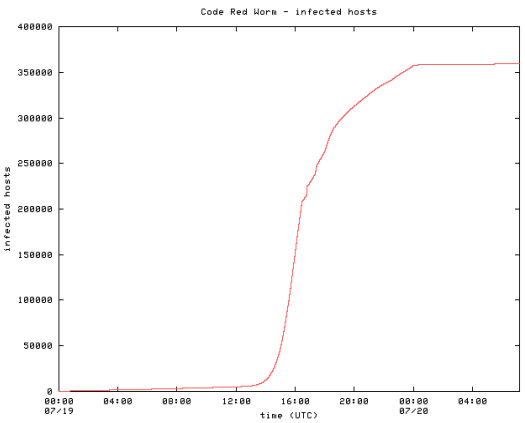
모델링 네트워크의 단점을 보완하기 위해서 패킷 네트워크를 추가한 모델이 본 논문에서 제안한 hybrid 모델이다. 〔그림 11〕은 hybrid 모델을 이용하여 코드레드 웜에 대한 시뮬레이션을 수행한 결과이다. 이 실험에서는 감염률만이 아닌 웜의 치료현상까지 구현한 결과를 나타내고 있다. Hybrid 모델은 모델링 네트워크에서 초

〔표 3〕 시뮬레이션 수행 시간 비교

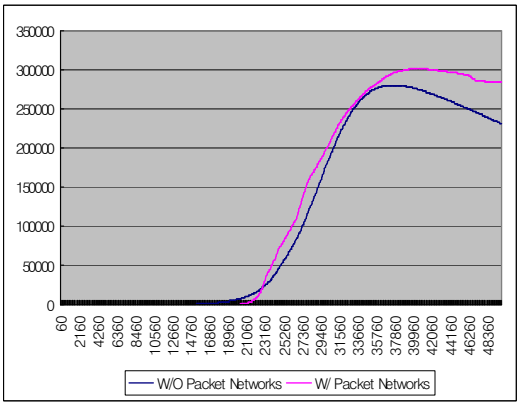
packet network 노드수	실행 시간 (ms)	
	epidemic model + packet network	epidemic model + fluid model
31	3,669	179
43	4,392	179
65	7,924	178
87	9,949	179
109	12,806	179
131	14,375	178
243	27,472	179
355	44,101	179
467	56,295	179
579	74,008	179



〔그림 9〕 시뮬레이션 수행 시간 비교



〔그림 10〕 코드레드 전파 그래프



〔그림 11〕 코드레드 웜 시뮬레이션 결과

기 감염률을 그대로 사용하는 것과는 달리 웜의 전파에서 사용되는 감염률을 패킷 네트워크로부터 얻어오게 된다. 그리고 결과를 보면 실제 패킷 네트워크를 구성하는 호스트는 200개에 불과하지만 이 호스트에서 추출한 감염률이 적용된 모델링 네트워크의 실험 결과는 기존 패킷 네트워크가 존재하지 않을 때와 매우 유사한 것을 알 수 있다. 또, 패킷 네트워크에서 정보를 추출하기 때문에 모델링 네트워크의 단점인 정적인 실험 결과가 아닌 여러 가지 네트워크 상황을 동적으로 변화시켜 가면서 시뮬레이션을 수행할 수 있는 장점이 있다.

Ⅴ. 결 론

네트워크 연구에서 많이 사용하는 시뮬레이션 방법은 인터넷 웜과 같은 대규모 네트워크 시뮬레이션을 수행하려 할 때, 여러 가지 문제가 발생하게 된다. 이러한 문제를 해결하기 위하여 여러 가지 모델을 이용한 시뮬레이션 방법이 사용되고 있는데, 모델링에 따른 정보 손실 등의 문제가 야기되고 있다. 본 논문에서는 인터넷 웜 중 RCS 모델의 특성을 지니고 있는 웜의 시뮬레이션에 효과적으로 적용할 수 있는 hybrid 모델을 이용한 시뮬레이션 방법을 제안하였다.

Hybrid 모델은 대규모 네트워크 시뮬레이션의 문제인 시간 지연을 모델링 네트워크를 사용하여 해결하였고, 또한 모델링 네트워크와 패킷 네트워크를 연동시켜서 모델링 네트워크의 단점인 패킷 단위의 정보 표현이 가능하도록 하였다. 아울러 패킷 네트워크 시뮬레이션 결과 값이 모델링 네트워크의 모델링 인자 값에 동적으로 반영되게 하여 기존의 방법과는 달리 동적인 시뮬레이션이 가능하도록 하였다. 마지막으로 본 논문에서는 제안한 hybrid 모델을 기반으로 SSFNet을 사용하여 RCS 특성을 가지는 대표적인 웜인 코드레드 웜에 대한 시뮬레이션을 수행하여 hybrid 모델의 응용 가능성을 보여주었다.

향후 연구 방향으로는 모델링 네트워크와 패킷 네트워크가 교환하는 파라미터에 대한 연구를 진행하여 RCS 특성의 웜만이 아닌 다른 종류의 웜 시뮬레이션이 가능하도록 하고, 다양한 종류의 패킷 네트워크를 통해 네트워크의 현상을 정확하게 반영할 수 있는 방법에 대한 연구를 계속 진행할 계획이다.

참고문헌

〔1〕 David Moore, Vern Paxson, Stefan Savage, Colleen Shannon, Stuart Staniford, Nicholas Weaver. “The Spread of the Sapphire/Slammer Worm”. CAIDA Technical report, 2003.

〔2〕 Vishal Misra, Wei-Bo Gong, Don Towsley. “Fluid-based Analysis of a Network of AQM Routers Supporting TCP Flows with an Application to RED”. ACM SIGCOMM Computer Communication Review, Volume 30, Issue 4, pp. 151-160, 2000.

〔3〕 Yong Liu, Francesco Lo Presti, Vishal Misra. “Fluid Models and Solutions for Large-Scale IP Networks”. ACM SIGMETRICS Performance Evaluation Review, Volume 31, Issue 1, pp. 91-101, 2003.

〔4〕 D.J.Daley, J.Gani. “Epidemic modelling : an introduction”, Cambridge University Press, 1999

〔5〕 Michael Liljenstam, David M. Nicol, Vincent H. Berk, Robert S. Gray. “Simulating Realistic Network Worm Traffic for Worm Warning System Design and Testing”. In Proceedings of the ACM WORM 2003, pp.24-33.

〔6〕 SSFNet Web page. http://www.ssfnet.org/

〔7〕 NS-2 Web page. http://www.isi.edu/nsnam/ns/

〔8〕 David Moore, Colleen Shannon. “The Spread of the Code-Red Worm (CRvw)”. CAIDA Analysis page. www.caida.org/analysis/security/code-red/coderedv2_analysis.xml

〔9〕 Kalyan S. Perumalla, Srikanth Sundaragopalan. “High-Fidelity Modeling of Computer Network Worms”. In Proceedings of the 20th Annual Computer Security Applications Conference(ACSAC ’04), pp. 126-135, 2004.

〔10〕 Xiang Zeng, Rajive Bagrodia, Mario Gerla. “GloMoSim: A library for parallel simulation of large-scale wireless networks”. In Proceedings of the 12th Workshop on Parallel and Distributed Simulation, pp. 154-161, 1998.

〔11〕 Benjamin Melamed, Shuo Pan, Yorai Wardi. “Hybrid discrete-continuous fluid-flow simulation”. SPIE, Volume 4526, pp. 263-270, 2001.

〔12〕 Cameron Kiddle, Rob Simmonds, Carey Williamson, Brian Unger. “Hybrid packet/fluid flow network simulation”. In Proceedings of the Seventeenth Workshop on Parallel and Distributed Simulation (PADS’03), pp. 143, 2003.

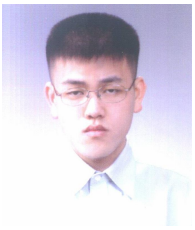
〔13〕 Stuart Staniford, Vern Paxson, Nicholas Weaver. “How to Own the Internet in Your Spare Time”. In Proceedings of the 11th USENIX Security Symposium, pp. 149-167, 2002.

〈著者紹介〉

김 정 식 (Jung Sik Kim) 학생회원
2006년 2월: 한양대학교 컴퓨터전공 학사
2006년 3월~현재: 한양대학교 전자컴퓨터통신공학 석사
<관심분야> 센서 네트워크, 시뮬레이션, 네트워크 보안

박 진 호 (Jin Ho Park) 학생회원
2006년 2월 : 용인대학교 컴퓨터정보처리학과 학사
2006년 3월~현재 : 한양대학교 정보통신공학과 석사
<관심분야> IPv6, 시스템 보안

조 재 익 (Che Ik Cho) 학생회원
2005년 8월 : 한양대학교 정보경영공학과 학사
2005년 9월~현재 : 한양대학교 정보통신공학과 석사
<관심분야> MANET, 네트워크 보안



최 경 호 (Kyoung Ho Choi) 학생회원
2006년 2월: 성결대학교 정보통신학과 학사
2006년 3월~현재: 한양대학교 전자컴퓨터통신공학 석사
<관심분야> 정보보호

임 을 규 (Eul Gyu Im) 종신회원
1992년 2월 : 서울대학교 컴퓨터공학과 학사
1994년 2월 : 서울대학교 컴퓨터공학과 석사
2002년 5월 : University of Southern California 컴퓨터과학 박사
2000년~2002년 : WiseNut Inc. Sr. SW Engineer
2002년~2005년 : 국가보안기술연구소 선임연구원
2005년~2007년 : 한양대학교 정보통신대학 전임강사
2007년~현재 : 한양대학교 정보통신대학 조교수
<관심분야> 유무선 네트워크 보안, 정보보안