Journal of The Korea Institute of Information Security & Cryptology ISSN 1598-3986(Print)
VOL.23, NO.6, Dec. 2013 ISSN 2288-2715(Online)
http://dx.doi.org/10.13089/JKIISC.2013.23.6.1169

사이버 보안 연구 프레임워크로써의 Emulab 활용 동향 연구*

이 만 희,[1†] 석 우 진[2‡]
[1]한남대학교, [2]한국과학기술정보연구원

Research on the Trend of Utilizing Emulab as Cyber Security Research Framework*

Man-hee Lee,[1†] Woo-jin Seok[2‡]
[1]Hannam University, [2]KISTI

요 약

Emulab은 미국 Utah 대학에서 개발된 연구 프레임워크로써 연구자들에게 필요한 시스템을 원하는 네트워크 구조로 연결하여 주는 On-demand 서비스를 그 핵심으로 하고 있다. 특히 Emulab은 시뮬레이션 또는 가상화 기술을 사용하지 않고 실제 시스템에 운영체제를 구동시킴으로써 연구의 검증환경을 실제 환경에 가깝게 하는 것이 특징이다. 이로 인해 Emulab은 사이버 보안 분야뿐만 아니라 네트워크 분야 등에서도 다양하게 활용되고 있었지만 국내에는 활용된 바가 거의 없었다. 최근 한국과학기술정보연구원(KISTI)에서 소규모 Emulab을 자체적으로 구축·운영함에 따라 국내에서도 Emulab을 활용한 연구가 활성화 될 것으로 기대된다. 이를 기점으로 본 논문은 Emulab을 국내에 소개하고, 사이버 보안 연구 프레임워크로써의 Emulab 활용 동향을 소개한다.

ABSTRACT

Emulab is a research framework developed by Utah university, proving on-demend research environment service so that researchers can set up and use the environment at anytime. The main advantage of Emulab over other research methodologies like simulation or virtualization is to use real systems and networks using real operating systems, making the research environment much similar to the real world. Even though Emulab has been actively used in many areas such as security and network, there has been little use in Korea research community. As KISTI recently constructed a small Emulab, it is expected that many researchers and educators would make use of the Emulab. In this study, we introduce Emulab to Korea research community and give an overview of utilization trend of Emulab as a cyber security research framework.

Keywords: Cyber security, Emulab, Research framework

I. 서 론

악의적인 해커에 의해 날로 정교해지고 파괴력이 막강해지는 사이버공격을 원천적으로 봉쇄할 수 없으므로 이 사이버공격을 보다 효과적으로 탐지하고 대응하는 것이 매우 중요하다. 일반적으로 관제라고 알려진 사이버공격의 대응 단계는 크게 수집·탐지, 분석, 대응으로 이루어진다. 수집·탐지단계에서는 네트워크 및 시스템 모니터링을 통하여 이상 징후를 탐지한다. 분석단계에서는 탐지된 트래픽 및 시스템에 대한 상세 분석이 이루어지고 심각한 사이버공격인지, 그렇다면 어떻게 대응할 것인지에 대한 분석이 이루어진다. 대응단계에서는 분석된 결과를 기반으로 해당 트래픽을

접수일(2013년 9월 3일), 게재확정일(2013년 10월 15일)
* 본 논문은 2013년 한남대학교 교비연구비에 의해 수행되었음.
† 주저자, manheelee@hnu.kr
‡ 교신저자, wjseok@kisti.re.kr(Corresponding author)

차단하고 관련된 시스템에 대한 보호 및 후속조치가 이루어진다. 사이버공격의 효과적인 대응을 위해서는 첫 번째 두 단계인 네트워크 및 시스템의 이상 징후 자동 탐지 및 분석 기술이 고도화 되어야 한다. 또한 사이버 공격에 대한 방어 연구를 위해서는 필수적으로 사이버 공격 자체에 대한 연구도 같이 수행되어야 함은 자명하다.

사이버 공격 및 방어 연구의 일반적인 순서는 먼저 기존 공격 및 방어 기술의 분석을 통해 추후 가능한 공격·방어 방법에 대한 기초적인 아이디어를 제안하고 이를 구현 및 검증하는 것이다. 사이버 보안 분야 연구가 기타 분야 연구와 다른 점이 있다면 검증 방법론이 체계적이기 어렵다는 것이다. 예를 들면, 네트워크 분야에서는 널리 알려진 QualNet, NS-2, OPNET 등 검증받은 네트워크 시뮬레이션 도구가 많이 있다. 하드웨어 분야에서도 VCS, ModelSim 등 VHDL, Verilog 등으로 구현된 아이디어는 거의 실제와 똑같이 시뮬레이션 되고, FPGA의 발달로 소규모 로직인 경우 실제 칩으로 구현하여 검증해 볼 수도 있다. 하지만 사이버 보안기술 연구는 네트워크와 컴퓨터 시스템이 동시에 이용되는 경우가 많으므로 네트워크 시뮬레이션 또는 시스템 시뮬레이션만으로 온전한 검증 환경을 구축할 수가 없다. 이러한 어려움으로 인해 시스템의 동작은 추상화하고 주로 네트워크 시뮬레이션을 이용하거나, 시스템의 동작을 추상화하기 어려운 경우에는 실험실에 몇 대의 PC와 스위치 및 라우터를 이용하여 단순화된 소규모 테스트베드를 구축하여 검증하는 방법을 사용해왔다.

사이버 보안기술 연구에서 네트워크 시뮬레이터를 사용하는 방법은 악성코드가 실제 시스템에서 수행되며 발생하는 상호 작용은 무시하게 된다. 하지만 Floyd와 Paxon이 지적한 바와 같이 인터넷을 시뮬레이션 한다는 것도 매우 어려운 일인데[1], 악성코드의 시스템 동작은 시뮬레이션하지 않은 채 네트워크 부분만 시뮬레이션 한다는 것은 매우 제한적일 수밖에 없다. 다행히 최근에는 가상화 기술이 급속히 보급되어 VMWare, QEMU, VirtualBox등을 활용하여 악성코드의 시스템 상에서의 동작을 연구하는데 많이 용이해졌다. 하지만 이러한 가상화 환경을 탐지하여 악성코드가 작동하지 않는 경우도 있고[2], 수십 대의 컴퓨터에서 동시에 수행되는 악성코드의 집단행동 등은 여전히 분석하기 어렵다. 그래서 많은 사이버 보안 연구자들이 선택하는 방법이 소규모 테스트베드를 실제로 꾸미는 것이다. 이 방법은 재정만 허락한다면 비교적 손쉽게 구축할 수 있기 때문에 많이 선호하고 있다. 하지만 이런 테스트베드는 외부에서 원격 콘트롤이 어려운 경우가 많고, 한번에 한가지 테스트 환경을 꾸미기 때문에 공동 사용이 어렵다. 따라서 각 연구 주제에 따라 하나씩의 테스트베드를 운영해야 하므로 다양한 보안 연구를 수행하기 위해서 많은 테스트베드를 구축·운영해야 하므로 재정 문제뿐만 아니라 전력 및 공간 사용 등의 문제가 발생한다.

이러한 문제를 해결하기 위해 제안된 시스템이 Emulab이라고 할 수 있다. Utah 대학에서 제안 및 구현된 Emulab 기술은 수십~수백대의 클러스터 PC와 고성능 네트워크 및 스위치로 구성된 시스템으로써, 연구자들이 연구에 필요한 PC와 해당 PC들의 네트워크 토폴로지를 GUI(Graphic User Interface)를 이용하여 직접 그린 후, 필요한 운영체제를 선택하면, 실제 PC가 할당되면서 연구자가 그린 네트워크 토폴로지로 VLAN(Virtual Local Area Network) 기술을 이용하여 실제 네트워크로 연결해 주는 기술 및 서비스이다[3]. 하나의 실험에 사용되고 있는 PC에는 연구자가 선택한 운영체제가 실제 운영되며 루트권한을 가진 연구자는 해당 PC의 환경을 마음대로 바꿀 수 있다. 이 실험을 위해 사용되고 있는 PC들과 네트워크 환경은 언제든지 임시 저장소로 저장되었다가 다시 복구되는 Swap-In, Swap-Out 기능까지 구현되어 있으므로 많은 연구자들이 한 Emulab을 시간차를 두고 공유할 수 있다.

2012년 4월 국내 최초로 Emulab이 구축되어 시범 운영 중에 있다[4]. 본 논문은 이를 기점으로 Emulab을 국내 보안 연구자들에게 소개하고, Emulab을 이용한 국외 사이버 보안기술 연구 동향을 소개함으로써 국내 사이버 보안기술 연구에 새로운 방향을 제시하고자 한다. 본 논문의 구성은 다음과 같다. 2장에서는 Emulab 시스템 구성, Utah 대학과 KISTI Emulab 및 DETER 프로젝트를 소개한다. 3장에서는 Emulab을 이용한 사이버 보안기술 연구 동향에 대해서 설명하고, 4장에서는 Emulab을 이용한 DDoS 공격 사례 연구를 통해 Emulab의 효용성을 알아본 후, 5장에서 결론을 맺는다.

II. Emulab 소개

2.1 개발 목적 및 소개

네트워크와 분산 시스템 연구에 있어서 유효적절한

검증도구는 매우 중요하다. 필요에 따라 다양한 검증도구가 사용될 수 있는데, 검증도구가 얼마나 사용하기 쉬운가에 대한 사용성, 검증도구를 검증 전 또는 검증 중에 얼마나 제어가 가능한가에 대한 제어성, 마지막으로 검증도구가 얼마나 실제 상황과 일치하는가에 대한 실제성에 따라 적절한 검증도구를 선택하게 된다. NS-2이나 OPNET와 같은 시뮬레이션 기반 검증도구는 사용성과 제어성이 높아, 재실험 및 제어가 언제든지 필요한 연구에서 주로 사용한다. 하지만 시스템에서 운영되는 운영체제의 동작 및 인터럽트 등으로 인한 오버헤드 등 시스템 내부의 역학 관계가 전혀 반영되지 않으므로 실제성 부분에 근본적인 문제점을 내포하고 있다. 한편 실제 시스템 및 네트워크를 사용하는 방법은 실제성은 높은 반면에 사용성 및 제어성이 매우 낮아 검증에 필요한 환경 구성 및 재실험 등이 어렵다. 이런 문제를 해결할 수 있는 검증방법론이 에뮬레이션 방법론이라고 할 수 있다.

Emulab에서 취하고 있는 에뮬레이션 접근법은 실제 PC, 운영체제, 및 응용 어플리케이션, 실제 네트워크를 사용하여 실제성을 높이고 동시에 언제든지 제어가 가능한 프레임워크를 구축하여 제어성을 확보하는 한편, 원격 사용자가 손쉽게 시스템 및 네트워크 환경을 구축하고 이용할 수 있는 인터페이스를 제공함으로써 사용성을 높이고자 하였다. 이를 위해 Emulab은 수십~수백대의 클러스터 PC와 고성능 네트워크 및 스위치로 시스템을 구성한다.

Fig.1.은 Emulab 사용을 위한 절차를 설명한다. 연구자들은 온라인으로 Emulab 사용 승인을 신청하면 Emulab 관리자가 이를 승인하면 Emulab 자원을 사용할 수 있게 된다. 이후 연구자들이 Emulab 웹페이지에 접근하여 실험환경 생성을 요청한다.

실험환경 생성은 실험을 위해 필요한 PC와 해당 PC들의 네트워크 토폴로지를 GUI를 이용하여 직접 그리면 NS-2 시뮬레이터에서 입력으로 받을 수 있는 스크립트가 만들어진다. Fig.2.는 7개의 노드와 2개의 스위치로 이루어진 실험 네트워크를 구성하는 방법을 보여준다.

연구자는 비슷한 네트워크 토폴로지가 필요한 경우, 같은 그림을 반복해서 그릴 필요 없이 기존 스크립트를 이용하는 것도 가능하다. Emulab은 실제 PC를 할당한 후, 주어진 스크립트를 기반으로 VLAN 기술을 이용하여 실제 네트워크로 연결한다. 각 PC는 새로운 IP와 hostname을 동적으로 할당받고 서로 통신이 가능하도록 자동으로 환경이 구성된다. 해당 실험에서 사용될 운영체제는 사용가능한 운영체제 중에 하나를 선택하면 해당 운영체제의 이미지가 할당된 PC로 전송되어 해당 운영체제로의 부팅이 되면서 실험환경이 할당이 완료되고 이를 Swap-in이라고 한다. 하나의 Emulab에서 하드웨어 자원이 허락하는 한, 다수의 실험이 동시에 진행될 수 있도록 노드 및 네트워크의 자원 관리가 자동화 되어 있다. 즉, 하나의 실험에서 사용되고 있는 노드와 네트워크는 동시에 수행되는 다른 실험의 구성과 운영에 독립적이다. 그러므로 여러 연구자들이 동시에 다른 실험에 영향을 거의 받지 않고 Emulab을 사용할 수 있다.

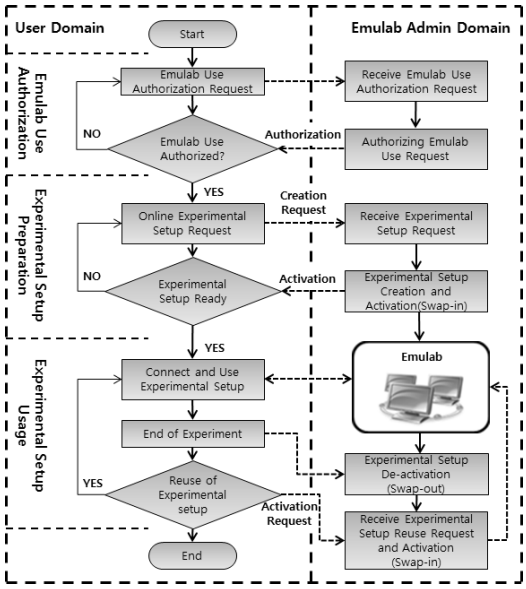

Fig.1. Emulab Usage Model

사용자는 실험이 끝나면 할당받은 실험환경을 반납하고 이 요청을 받은 Emulab은 실험환경을 저장한 후 할당된 자원을 회수(Swap-Out)한다. 저장된 실험환경은 사용자 요청에 따라 언제든지 다시 Swap-In 할 수 있다. 또한 실험이 너무 길어져 자원

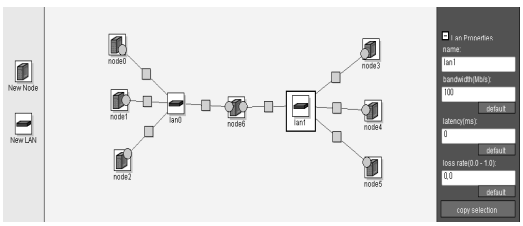

Fig.2. Emulab Experiment Setup

이 공평하게 사용되지 않을 수 있는 문제점을 제거하기 위해, 실험은 자동으로 Swap-Out 될 수 있다. 기본적으로 2시간 동안 Idle한 상태가 되면 Swap-Out이 되고 실험 환경 설정 변경을 통해 더 짧거나 길게 할 수도 있다. Idle 상태 판단은 네트워크 트래픽, 콘솔 작업 여부, CPU 사용량, 리부트와 같은 활동 등이 없으면 Idle 상태로 간주된다. 이 기능을 통해 많은 연구자들이 하나의 Emulab 시스템을 시간차를 두고 공유하는 동시에 Emulab의 사용성과 제어성을 높일 수 있다.

2.2 Emulab 프로젝트 현황

Emulab 개발 및 소프트웨어 제공은 Utah 대학의 Flux 그룹에 의해 진행된다. Flux 그룹은 하나의 Emulab 시스템을 구축 운영 중에 있으며 이 외에도 전 세계에 37개의 Emulab이 네트워크, 분산시스템, 및 보안 연구 또는 교육 목적으로 사용되고 있다[5]. Utah 대학에 운영중인 Emulab은 2013년 8월 현재 약 600여대 테스트 노드와 7대의 각종 서버, 13대의 고성능 스위치(Cisco 6500 시리즈, HP 5400zl, Arista 7504)와 50여개의 파워 콘트롤러 및 25개의 서버랙으로 구성되어 있다.

Utah Emulab을 사용하기 위한 준비단계는 다음과 같다. 먼저 연구책임자가 새로운 연구 프로젝트를 시작하기 위한 요청을 한다. 교육 및 연구 기관과 상용 기업을 포함하여 모든 기관에 대해 사용 제한이 없으므로 누구든 신청이 가능하고 연구 목적에 제한은 없으나 Emulab을 사용할 만한 적합한 이유가 있어야 한다. 다만 프로젝트 리더는 여러 실험을 생성하고 해당 실험에 참여할 멤버를 생성하게 되므로 학생과 같이 책임을 지기 힘든 위치에 있는 사람에게는 프로젝트 리더를 허용하지 않고 연구 책임자를 수행할 수 있는 사회적 지위가 있는 사람에게만 프로젝트 생성 및 리더를 허용하고 있다. 대부분 몇 시간 혹은 며칠 내에 승인/거부 이메일이 오게 되며 승인이 되면 실험을 위한 테스트베드 생성 및 실험에 참여할 멤버 생성 및 승인 가능하다.

국내 연구자들의 연구 및 실험 환경 고도화를 위한 공용 서비스로 구축된 KISTI Emulab은 지속적으로 관련 도구들을 업데이트하면서 테스트 사용자들을 대상으로 시범 서비스 중이다. 현재 42개의 테스트 노드와 노드 제어용 서버 및 파일 서버를 포함한 서버 4대, 그리고 고성능 스위치 3대(Cisco 4507, 3500, 2700)와 파워 콘트롤러 5개로 구축되어 있다.

2.3 DETER 프로젝트 현황

2003년에 시작된 DETER (The Cyber DEfense Technology Experimental Research)는 연구 프로젝트인 동시에 사이버 연구를 위한 테스트랩인 DeterLab의 운영을 담당하는 주체이다[6]. 초기 미 과학재단과 미 국토안보부의 연구 프로젝트로 시작한 DETER는 현재까지 DARPA와 DoD등으로부터 지원을 받고 있으며 HP, Cisco, Juniper 등으로부터 장비 지원을 받고 있다.

DETER의 가장 큰 목적은 미국 국토방위와 주요 인프라 보호를 위한 사이버 보안기술 연구의 확장성 및 능력을 증진시키기 위해 사이버 보안기술 실험환경 고도화를 통하여 보다 효과적이고 혁신적인 사이버 보안기술을 개발하는데 있다. DETER 프로젝트를 통하여 나온 연구 결과물은 DeterLab에 적용되어 사이버 보안기술을 연구하는 연구자들이 직접 사용하게 하고 그 피드백을 이용함으로써 지속적으로 선진화된 DeterLab을 운영한다.

시스템 구성으로 보면 DeterLab는 Emulab을 DETER 프로젝트 취지에 맞게 적절한 기술을 추가한 것이므로 DeterLab도 Emulab 사이트 중에 하나로 계수된다. 하지만 DETER는 Emulab에서 다루지 못한 보안 이슈들을 하나씩 밝혀내고 보강하면서 보안 분야 연구에 적합하도록 DeterLab을 특화시켰다. DETER가 주목한 두 가지 위협모델은 내부에서 발생하는 위협과 외부에서 발생하는 위협으로 나뉜다. 내부에서 발생하는 위협 모델은 테스트 중인 악성코드가 테스트베드 자체를 공격하여 테스트베드 운영 권한을 탈취하는 경우와 악성코드의 트래픽 또는 악성코드 자체가 외부 인터넷으로 유출되는 경우이다. 또한 사이버보안 연구자간에 정보를 파괴 또는 탈취하는 문제도 발생할 수 있다. 외부에서 발생하는 위협으로는 테스트베드 외부에 존재하는 해커가 DeterLab 안으로 침입할 수 있는 위협이다.

내부위협을 줄이기 위해 DETER에서 가장 우선적으로 구현하는 개념은 봉쇄(Containment)이다. 봉쇄기능은 내부에서 실험중인 트래픽 또는 코드 자체가 외부로 나가는 것을 차단하는 것이다. 이를 위해 DeterLab은 일차적으로 동적으로 구성된 실험 네트워크에서 발생된 트래픽은 인터넷으로 라우팅 되지 않도록 하고 방화벽 여러 개를 구축하여 아웃바운드 트

래픽을 효과적으로 차단한다. 추가적 기능으로써 테스트 노드를 제어하기 위한 네트워크인 콘트롤 네트워크를 이용하여 테스트 트래픽이 유출되는 것을 막기 위해 콘트롤 네트워크를 지속적으로 모니터링하고 콘트롤 트래픽이 아닌 경우 경보를 발생한다. 다른 연구자간 데이터 탈취 또는 외부 해커에 의한 침입 후 자료 유출 등의 내부위협은 암호화 파일 시스템을 사용하여 제거한다. 테스트 노드에 이전 실험에서 사용되었던 자료가 다음 실험에 남아있지 않기 위해서 실험이 끝난 후, 테스트 노드의 디스크에 0으로 overwrite를 하여 모든 자료를 제거한다. 외부 위협을 줄이기 위해서는 일반적으로 알려져 있는 방안인 방화벽과 침입탐지도구 등을 구축하여 운영한다.

이상과 같이 DeterLab은 Emulab을 기반으로 일반 기능의 고도화 및 사이버 보안기술 연구를 위해 추가 보안기능이 적용된 테스트베드이다. 그러므로 일반적인 사이버 보안기술 연구를 위해서 Emulab을 그대로 사용하는 데는 큰 문제가 없으나, 고도로 민감한 악성코드의 테스트 및 고도의 보안이 필요한 테스트를 진행해야 할 경우에는 Emulab을 구축한 후, DeterLab과 같은 특화가 필요할 것이다.

III. Emulab을 이용한 사이버 보안기술 연구 동향

Emulab이 사용되고 있는 연구 분야는 무척 다양하지만, 본 장에서는 Emulab이 특히 사이버 보안기술 분야에서 어떻게 활용되었는지 살펴본다.

3.1 DDoS 연구

Emulab이 가장 많이 활용된 사이버 보안기술 연구 분야 중에 하나는 DDoS 연구 분야다. 수십에서 수백대의 테스트 노드와 네트워크를 원하는 대로 쉽게 구성할 수 있을 뿐 아니라, 다양한 DDoS 공격 툴을 실제 시스템에서 작동시킬 수 있으므로 DDoS 연구에 장점이 있다. 첫 번째로, Emulab은 DDoS 공격 도구 및 탐지 알고리즘 검증 테스트베드로 사용된다. 몇 가지 연구 내용을 소개하면 다음과 같다. Yu Chen 등은 다수의 네트워크 도메인에서 동시에 일어나는 DDoS 공격을 플로우 레벨에서 효과적으로 탐지하는 협업 탐지 방안을 제시하였다[7]. 이를 검증하기 위해 USC ISI Emulab에서 실험을 진행하였고, 4개의 AS 도메인 상의 32개 라우터를 Emulab에서 테스트

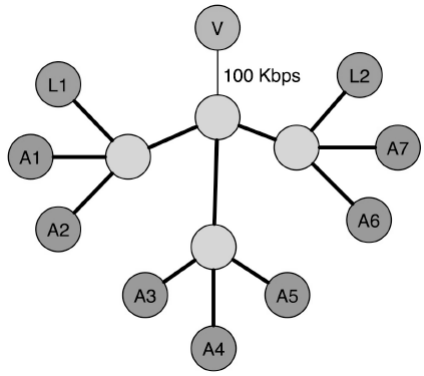

Fig.3. Experimental setup for ddos defense using client reputations[8]

를 진행하였다. Maitreya Natu 등은 클라이언트 평판도 기반으로 DDoS를 효과적으로 탐지하고 차단하는 방안을 제시하였다[8]. 이를 검증하기 위해 Emulab을 사용하였고 하나의 피해자 노드 (V), 두 개의 정상 노드 (L1, L2), 그리고 7개의 공격 노드 (A1~A7)을 Fig.3.과 같이 구성하여 테스트 하였다.

Jean Mirkovic 등은 분산된 네트워크에서 DDoS 공격을 빠르게 탐지하고 대응하기 위한 분산 DDoS 방어 시스템인 DefCOM을 제안하였다[9]. 이를 검증하기 위해 Fig.4.와 같이 Emulab을 이용해 64개 테스트 노드와 22개 라우터를 구성하여 테스트 하였다.

마지막으로 Matt Beaumont-Gay는 Emulab을 이용하여 세 가지 SYN flooding 공격 탐지 알고리즘을 비교했다[10].

Emulab의 DDoS 연구에서의 또 다른 활용 분야는 DDoS 검증 방법론이다. 시뮬레이터나 소규모 네트워크에서 이루어진 DDoS 공격 및 탐지 도구 검증은 한계가 있음을 보이고, Emulab이 DDoS 연구를 위한 뛰어난 검증 도구임을 보이는 것이다. Jean

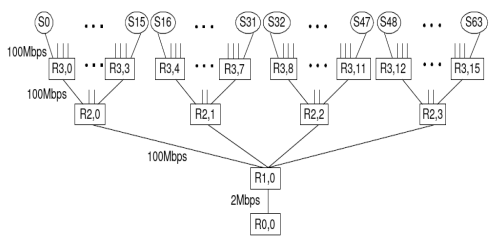

Fig.4. Experimental setup for ddos defense system, DefCOM[9]

Mirkovic는 기존 검증 방법론에 대한 단점을 밝히고 보다 나은 DDoS 검증을 위해 표준화된 벤치마크 프로그램을 개발하고 이의 성능을 측정하기 위한 측정 메트릭을 제안하고 이를 Emulab을 이용하여 검증하였다[11][12]. Jean Mirkovic는 또한 일반 연구자들이 DDoS 실험을 빠르게 진행할 수 있도록 DDoS 검증을 자동화하는 방안도 제안하였고 Emulab을 DDoS 공격의 파괴력을 측정하기 위한 도구로도 제안하였다[13][14]. 이외에도 [15][16][17] 등의 연구에서 더 나은 DDoS 검증 방법론 제시를 위해 모두 Emulab을 사용하고 있다.

3.2 Worm, Botnet 연구

Emulab이 왕성히 활용되고 있는 또 다른 분야는 웜 및 봇넷 관련 연구다. DDoS와 비슷하게 Emulab에서 실제 시스템과 네트워크에서 웜, 봇넷 등의 악성코드를 안전하게 테스트 해 볼 수 있어서 Emulab이 많이 활용되고 있다. 첫 번째로, Emulab은 웜 등의 악성코드를 실행하고 그 행위를 분석하는데 쓰여지고 있다. Cliffork Neuman 등은 DeterLab에서 자가 유포를 하는 악성코드를 테스트를 하는 방법을 설명하고 비슷한 테스트를 위한 유의사항을 제시했다[18]. Steve Hanna 등은 전파속도가 매우 빠른 하나의 flash worm을 구현하고 그 웜이 얼마나 빨리 전파될 수 있는지를 Emulab에 테스트 했으며 사용한 토폴로지는 Fig.5.와 같다[19]. L. Li 등은 같은 그룹에서 제안했던 KMSim 웜 모델을 확장하고 이를 이용하여 Witty와 Blaster 웜을 Emulab 상에서 테스트하였다[20].

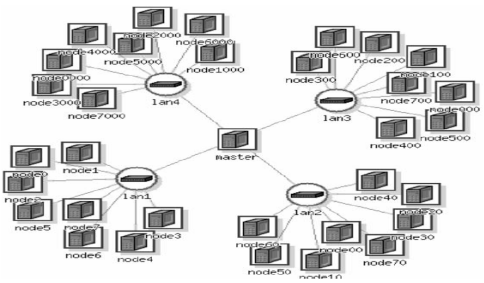

Fig.5. Experimental setup for emulating Flash-worm[19]

두 번째로 Emulab은 웜 또는 악성코드를 탐지하고 차단하는 알고리즘을 검증하는 도구로 사용되고 있다. 몇 가지 연구 사례를 소개하면 다음과 같다. Senthilkumar Cheetancheri 등은 엔드 호스트만을 이용하여 대규모 분산 웜 공격을 탐지하는 방안을 제시하고 이를 Emulab에서 테스트하였다[21]. Marie Vasek 등은 웜에 대한 상세 리포트가 실제 웜 트래픽에 어떠한 영향을 미치는지 연구하였고 이를 Emulab에서 테스트하였다[22]. Nicholas Weaver 등은 피해 노드를 스캐닝 방법으로 검색하는 스캐닝 웜을 탐지하고 이를 효과적으로 봉쇄하는 방안을 제시하고 이를 Emulab에서 검증하였다[23]. 이 외에도 [24][25] 등 많은 연구에서 웜 탐지 및 방어 방안을 Emulab에서 테스트 하고 있다.

마지막으로 Emulab이 웜 및 봇넷을 연구하기에 적합한 도구인가에 대한 검증 방법론에 대한 연구도 진행되었다. Senthilkumar Cheetancheri 등은 Emulab이 웜 탐지 및 방어 연구에 원활히 사용될 수 있는 Wrapper를 제안하고 이를 이용해서 웜 봉쇄 연구에 활용하는 것을 보였다[26]. Paul Barford는 봇넷 연구를 위해서 봇넷의 행위를 완벽히 관찰하고 분석할 수 있는 환경 구축의 중요성을 강조하면서 Emulab을 이용한 봇넷 테스트 환경인 Botnet Evaluation Environment (BEE)를 구축하고 그 효용성에 대해서 연구하였다[27].

3.3 기타 보안 연구

본 절은 상기 설명한 DDoS와 Worm, Botnet 연구 외의 Emulab이 사용되는 사이버 보안기술 연구 분야를 간단히 소개한다.

3.3.1 BGP 보안 연구

Border Gateway Protocol (BGP) 은 인터넷에서 라우터들끼리 라우팅 정보를 교환하기 위해 표준과 같이 사용되고 있는 프로토콜이다. 최근 BGP 통신을 교란함으로써 인터넷 라우팅에 혼란을 초래할 수 있는 공격이 나타남에 따라 BGP 통신 및 라우팅 정보를 보호하고자 하는 연구가 진행되고 있다 [28]. BGP 보안 연구와 관련하여 Emulab을 사용한 예는 [29][30]에서 발견할 수 있다.

3.3.2 침입 탐지 시스템 연구

사이버 보안기술 연구에서 빠질 수 없는 것이 침입

탐지 시스템 (Intrusion Detection System, IDS)과 방화벽(Firewall)이다. Emulab은 앞서 설명한 것과 같이 실제 시스템과 네트워크를 손쉽게 설정할 수 있기 때문에 IDS와 Firewall 연구에도 적합다고 할 수 있다. Calvin Ko 등은 IDS를 조직적으로 평가할 수 있는 검증방안을 제안하고 이를 Emulab을 이용하여 구현하였다. 초기 연구로 DDoS 탐지 알고리즘은 FloodWatch, D-WARD, COSSACK을 비교 평가하는 연구를 수행하였다 [31]. Ji Li 등은 여러 네트워크에서 서서히 전파하는 웜을 탐지할 수 있는 분산 침입 탐지 방안을 제안하고 이를 Emulab에 구현하였다 [32]. Clifford Neuman 등은 방화벽을 에뮬레이팅하는 연구의 효용성을 보이기 위해 Adventium Labs Distributed Firewall을 Emulab에서 구현하고 그 성능을 검증하는 연구를 수행하였다 [33]. 또 다른 방향의 연구로는 실제 Firewall이나 IDS 장비를 테스트하기 위한 용도로 Emulab을 사용하는 시도도 있다. Nicholas Weaver 등은 하드웨어/소프트웨어 혼합 방법으로 개발하는 IPS인 Shunt를 테스트하기 위해 Emulab을 사용하는 방안을 연구했다[34].

3.3.3 Security Education 연구

마지막으로 소개할 Emulab을 이용한 사이버 보안 기술 연구 동향은 보안 기술 교육 분야이다. 보안기술 분야는 네트워크와 시스템 분야에서 다양한 지식을 가지고 있어야 연구가 가능한 복합 분야라고 할 수 있다. 하지만 이런 분야를 제대로 공부하기 위해서는 다양한 환경에서 여러 가지 내용을 직접 실습해 봄으로써 더 확실한 지식을 배울 수 있는데, 개인과 조직을 포함해서 보안 교육을 위해 다양한 환경을 구축하는 것은 재정적, 관리적 한계가 있을 수밖에 없다. 이러한 보안 교육 문제를 해결할 수 있는 것이 Emulab이라고 할 수 있다. 아직 보안 교육 분야에서 Emulab이 많이 사용되고 있지 않지만, 잘 활용될 경우 매우 유익한 교육 도구가 될 것으로 예상 된다 [35].

IV. Emulab을 이용한 DDoS 공격 사례 연구

본 장에서는 Emulab을 이용하여 DDoS 연구를 실제 수행해 보고 DDoS 보안 연구를 위한 도구로써 Emulab의 효용성을 검증한다.

4.1 DDoS 공격 시나리오

DDoS 공격 도구로 7·7 DDoS 공격에도 사용된 것으로 추정되는 Slowloris를 선택하였다 [36][37]. Slowloris는 웹서버와 TCP 세션을 맺은 후 GET 요청을 보낼 때 완전하지 않은 http 헤더를 보내면, 웹서버가 완전한 헤더가 올 때까지 기다리는 점을 이용하여, 완전하지 않은 http 세션을 다수 생성함으로써 웹서비스를 마비시킬 수 있다. 이 Slowloris을 이용한 DDoS 공격 생성을 테스트하고 그 파괴력을 검증해보기 위해 웹서버가 설치된 피해 시스템, Slowloris가 설치된 공격 시스템, 그리고 공격중에 피해시스템에 접근하여 웹 서비스의 가용성을 측정하는 측정 시스템이 필요하다. 첫 번째 공격은 하나의 공격 시스템을 이용해보고 두 번째 공격은 10대의 공격 시스템을 이용하여 그 결과가 다른지 확인해 본다.

4.2 DDoS 공격을 위한 Emulab 환경 설정

최대 10대의 공격 시스템과 한 대의 피해 웹서버 시스템, 그리고 한 대의 측정 시스템을 위해서 Fig.6. 과 같이 12대가 스위치 하나를 경유하여 연결된 테스트 환경을 구성하였다.

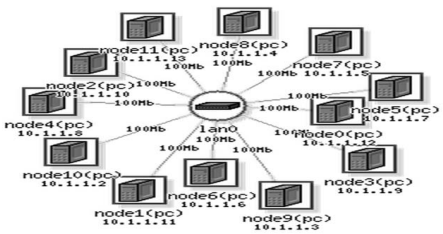

Fig.6. Experimental setup for for emulating ddos attack using Slowloris

모두 디폴트 운영체제인 CentOS 5.5를 사용하였다. 피해 시스템으로 node0를 사용하고 yum 도구를 이용하여 웹서버를 설치하였다. 공격 시스템은 node1부터 node10을 사용하고 공격도구인 Slowloris.pl을 구동하기 위한 추가 패키지인 IO-Socket-SSL-0.97와 Net-SSLeay-1.52를 설치하였다. 측정 시스템인 node11에는 특정 URL이 주어지면 이를 자동으로 다운로드 받을 수 있는 도구인 Wget를 설치하고, Wget을 피해 시스템에 설치된 웹서버의 index.html을 가져오는데 걸리는 시간을 지

속적으로 측정한다.

Slowloris을 위한 공격은 1단계로 먼저 node0에 설치된 웹서버의 timeout 시간을 알아내기 위해 ./slowloris.pl -dns node0 -port 80 -test 을 실행했다. timeout 값으로 240초를 사용하라는 결과 값을 얻었다. 이는 웹서버가 완성되지 않은 http 헤더를 받은 후, 해당 스레드를 제거하기 전까지 기다리는 시간인데 Slowloris 공격 시 완성되지 않은 헤더를 보낸 후, 해당 스레드가 제거되지 않도록 다시 헤더패킷을 다시 보내는 간격이다. 2단계 공격은 slowloris.pl -dns node0 -port 80 -timeout 240 -num 500 -tcpto 5 을 실행하는데, 최대 500개의 http 연결을 생성하며 TCP timeout은 5초로 사용한다는 뜻이다. Slowloris는 순차적으로 500개의 연결을 모두 생성한 후, 240초간 sleep한 후 다시 깨어나서 같은 공격을 반복한다.

4.3 DDoS 공격 결과 분석

Fig.7.은 1개의 공격 노드를 이용하여 웹서버를 1시간 동안 공격하고 이때 웹서버에 생성되는 http 연결의 개수를 측정한 것이다. Slowloris가 500개의 http연결을 시도했으나 실제로는 약 440개까지 만들어졌다. 흥미로운 결과는 http 연결이 제거되기 전에 다시 http 헤더를 보내는 Slowloris 공격 설명에 의거하여 http 연결 개수가 줄어들지 않을 것이라 예상했지만, 240초를 기다리는 동안 대부분의 http 연결이 제거되었고 바로 이어서 시도된 공격은 최대 60여개의 http 연결을 생성하였다. 이후에는 시도된 공격은 비슷한 패턴을 보였다.

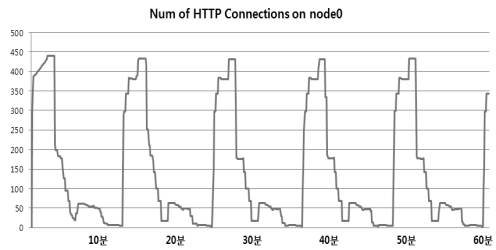

Fig.7. The number of concurrent connections under Slowloris dos attack by one node

공격을 받는 웹서버의 성능을 측정하기 위해 상기 공격중에 node11에서 웹서버로 웹서비스 요청을 하여 index.html을 받아오는데 걸린 시간을 측정하였다. Fig.8.은 웹서버에 생성된 http 연결 개수에 따른 index.html 응답시간을 나타낸다. 약 250개의 http 연결까지는 지연이 거의없이 서비스가 가능하며, 250개~350개 사이는 10초 이내로 서비스가 이루어졌다. 하지만 350여개 이상부터 응답시간이 50초 이상으로 급격히 늘어나면서 웹서비스가 거의 불가능한 상태가 되었다. 전체 공격시간 중 350개 연결개수가 지속된 시간은 14.35분 이었으며, 이는 전체 시간의 약 24%였다. 결과적으로 한 개의 공격노드를 이용한 Slowloris DoS 공격으로 웹서버의 가용성을 76%로 떨어지게 할 수 있었다.

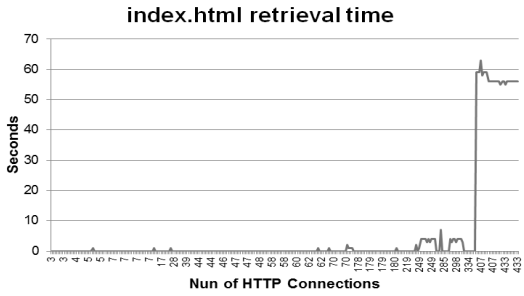

Fig.8. Retrieval time of index.html under Slowloris dos attack by one node

Fig.9.와 Fig.10.은 공격 노드를 10개로 늘인 후 같은 공격을 시도한 결과다. http 연결개수는 500~600개였으며 이때 시도한 웹서비스 접근은 거의 실패하였고 성공하더라도 100초 이상 걸리는 경우가 대부분이었다. 결과적으로 10개의 공격노드를 이용한 Slowloris DDoS 공격으로 웹서버의 가용성을 0%로 떨어지게 할 수 있었다.

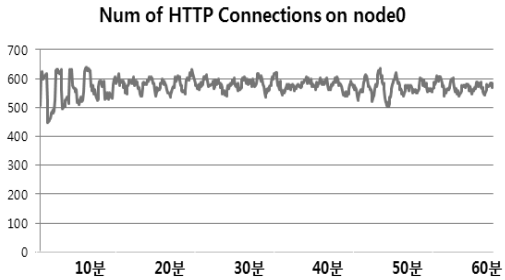

Fig.9. The number of concurrent connections under Slowloris dos attack by ten nodes

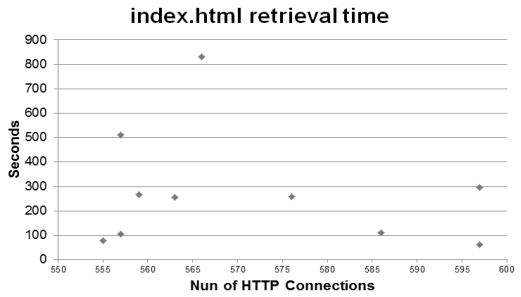

Fig.10. Retrieval time of index.html under Slowloris dos attack by ten nodes

4.4 DDoS 공격 연구를 위한 Emulab 사용 효용성 평가

Slowloris를 이용한 DDoS 공격 연구를 위한 Emulab 사용의 가장 큰 장점은 12개의 리눅스 시스템을 손쉽게 구성한 것이다. 시스템 할당 및 실험 환경 구성이 1시간 이내에 완료 가능하였다. 또한 측정 결과값은 NFS로 연결된 개인 계정에 저장되어 쉽게 접근이 가능하였다.

Emulab을 연구에 사용하기 위한 가장 큰 단점은 각 시스템의 현재 상태를 저장하는 기능이 없다는 것이다. 즉, 한 노드에 관련 패키지를 설치를 한 후, Emulab에서 Swap-Out이 되면 해당 내용이 사라진다. 따라서 테스트를 할 때마다 패키지를 설치하는 어려움이 있었다.

Emulab에 필요한 기능으로는 배치작업 모드로 판단된다. 10대의 시스템에서 DDoS 공격을 수행하기 위해 가상터미널 10대를 사용하여 같은 명령어를 일일이 수행하였다. 지정한 여러 시스템에 동일한 명령어를 실행할 수 있는 배치작업 모드가 있으면 연구 수행에 매우 유익할 것으로 판단된다.

V. 결 론

본 연구는 국내 최초의 Emulab이 구축된 것을 기점으로 Emulab을 소개하고, 특히 Emulab을 이용한 사이버 보안기술 연구 동향을 살펴보는데 목적이 있다. Emulab은 사용성, 제어성, 실제성에 중요성을 두어, 실험자가 자신이 원하는 실험 환경을 구축하고 제어하기 쉽고, 또한 실제 네트워크와 운영체제를 포함한 실제 시스템에서 실험하도록 자동으로 환경을 구성하여준다. Emulab의 가장 큰 장점은 기존 시뮬레이션 연구에서 실제 네트워크와 시스템을 추상화하는 과정에서 발생하는 시뮬레이션과 실제 환경에서의 차이점을 상당부분 없앨 수 있다.

사이버 보안기술 연구 분야 활용 부분에 있어서는 DDoS와 웜 및 봇넷 연구 분야가 가장 활발하다. 연구 방향을 크게 분류한다면 사이버 공격을 탐지 및 차단할 수 있는 방안을 제시하고 이를 검증하는 도구로써 사용되는 방향과 사이버 공격 자체에 관한 분석을 위해 Emulab 내에서 사이버 공격을 시도해 보는 방향으로 나눌 수 있다. Emulab을 사이버 보안 연구에 적극 활용하기 위해서는 DETER 프로젝트에서 수행했던 연구와 같이 Emulab 내부에서 실험하던 악성 트래픽 및 코드가 외부로 유출될 수 있는 내부적 위협, Emulab 외부에서 내부로 침투 및 자료 탈취 등의 외부적 위협 등을 효과적으로 차단할 수 있는 추가적 방편들도 제시되어야 더욱 안전하고 원활한 연구가 진행될 것이다.

추후 연구로는 시뮬레이션 및 소규모 네트워크에서 진행했던 기존 보안연구를 Emulab 환경을 이용하여 재실험하여 기존 결과와의 차이점을 분석하여 기존 연구 결과들을 재검증하고자 한다. 또한 이를 통해 얻은 노하우를 축적하고 정리함으로써 Emulab을 이용한 사이버 보안기술 연구 체계를 구축하고자 한다.

References

[1] S. Floyd and V. Paxson, "Difficulties in simulating the internet," IEEE/ACM Transactions on Networking (TON), vol. 9, no. 4, pp. 392-403, Aug. 2001.

[2] X. Chen, J. Andersen, Z. Mao, M. Bailey, J. Nazario, and F. Jahanian, "Towards an understanding of anti-virtualization and anti -debugging behavior in modern malware," Proceedings of the 38th Annual IEEE/IFIP International Conference on Dependable Systems and Networks (DSN'08), pp. 177-186, Jun. 2008.

[3] B. White, J. Lepreau, L. Stoller, R. Ricci, S. Guruprasad, M. Newbold, M. Hibler, C. Barb, and A. Joglekar, "An integrated experimental environment for distributed Systems and Networks," Operating systems design and im-

plementation 2002, pp. 255-270, Oct. 2002.
[4] KISTI Emulab: Network Emulation Testbed Home, http://www.emulab.kreonet.net/
[5] Utah Emulab: Network Emulation Testbed Home, http://www.emulab.net/
[6] DETER Network Security Testbed, http://www.isi.deterlab.net/
[7] Y. Chen, K. Hwang, and W. Ku, "Collaborative detection of ddos attacks over multiple network domains," IEEE Transactions on Parallel and Distributed Systems, vol. 18, no. 12, pp. 1649-1662, Dec. 2007.
[8] M. Natu and J. Mirkovic, "Fine-grained capabilities for flooding ddos defense using client reputations," Proceedings of the 2007 Workshop on Large Scale Attack Defense, LSAD '07, pp. 105-112, 2007.
[9] J. Mirkovic, M. Robinson, P. Reiher, and G. Oikonomou, "Distributed defense against ddos attacks," Technical Report CIS-TR-2005-02, University of Delaware CIS Department, 2005.
[10] M. Beaumont-Gay, "A comparison of syn flood detection algorithms," Proceedings of the Second International Conference on Internet Monitoring and Protection (ICIMP), pp. 9, Jul. 2007.
[11] J. Mirkovic, S. Fahmy, P. Reiher, and R. Thomas, "How to test dos defenses," Cybersecurity Applications and Technology Conference For Homeland Security, pp. 103-117, Mar. 2009.
[12] J. Mirkovic, E. Arikan, S. Wei, S. Fahmy, R. Thomas, and P. Reiher, "Benchmarks for ddos defense evaluation," DETER Community Workshop on Cyber Security Experimentation and Test, Jun. 2006.
[13] J. Mirkovic, B. Wilson, A. Hussain, S. Fahmy, P. Reiher, R. Thomas, and S. Schwab, "Automating ddos experimentation," DETER Community Workshop on Cyber Security Experimentation and Test, Aug. 2007.
[14] J. Mirkovic, S. Fahmy, P. Reiher, R. Thomas, A. Hussain, S. Schwab, and C. Ko, "Measuring impact of dos attacks," DETER Community Workshop on Cyber Security Experimentation and Test, Jun. 2006.
[15] H. Hazeyama, M. Suzuki, S. Miwa, D. Miyamoto, and Y. Kadobayashi, "Outfitting an inter-as topology to a network emulation testbed for realistic performance tests of ddos countermeasures," Proceedings of the Conference on Cyber Security Experimentation and Test, Jul. 2008.
[16] R. Chertov, S. Fahmy, and N. Shroff, "High fidelity denial of service(dos) experimentation," DETER Community Workshop on Cyber Security Experimentation and Test, Jun. 2006.
[17] A. Hussain, S. Schwab, R. Thomas, S. Fahmy, and J. Mirkovic, "Ddos experiment methodology," DETER Community Workshop on Cyber Security Experimentation and Test, Jun. 2006.
[18] C. Neuman, C. Shah, and K. Lahey, "Running live self-propagating malware on the deter testbed," DETER Community Workshop on Cyber Security Experimentation and Test, Jun. 2006.
[19] S. Hanna and D. Nicol, "Implementation and instrumentation of a flash-worm," DETER Community Workshop on Cyber Security Experimentation and Test, Jun. 2006.
[20] L. Li, S. Jiwasurat, I. Hamadeh, G. Kesidis, C. Neumann, and P. Liu, "Emulating sequential scanning worms on the deter testbed," Proceedings of IEEE/Create-Net TridentCom, Jun. 2006.
[21] S. Cheetancheri, J. Agosta, D. Dash, K. Levitt, J. Rowe, and E. Schooler, "A distributed host-based worm detection sys-

tem," Proceedings of the 2006 SIGCOMM Workshop on Large-Scale Attack Defense (LSAD), pp. 107-113, Sep. 2006.
〔22〕 M. Vasek and T. Moore, "Do malware reports expedite cleanup? an experimental study," 5th USENIX Workshop on Cyber Security Experimentation and Test (CSET), Jun. 2012.
〔23〕 N. Weaver and S. Staniford, "Very fast containment of scanning worms, Revisited," Malware Detection, Springer Verlag, Vol. 27, pp. 113-145, 2007.
〔24〕 L. Li, P. Liu, Y. Jhi, and G. Kesidis, "Evaluation of collaborative worm containments on deter testbed," DETER Community Workshop on Cyber Security Experimentation and Test, Aug. 2007.
〔25〕 L. Briesemeister and P. Porras, "Formally specifying design goals of worm defense strategies," DETER Community Workshop on Cyber Security Experimentation and Test, Arlington, Virginia, Jun. 2006.
〔26〕 S. Cheetancheri, D. Ma, K. Levitt, and T. Heberlein, "Towards a framework for worm-defense evaluation," Proceedings of the 25th IEEE International Performance, Computing, and Communications Conference (IPCCC), pp. 559-565, Apr. 2006.
〔27〕 P. Barford and M. Blodgett, "Toward botnet mesocosms," Proceedings of First Workshop on Hot Topics in Understanding Botnet, USENIX Association, Apr. 2007.
〔28〕 K. Butler, T. Farley, P. McDaniel, and J. Rexfod, "A survey of bgp security issues and solutions," Technical Report TD-5UGJ33, ATandT Labs - Research, Feb. 2004.
〔29〕 S. Tseng, S. Wu, K. Ma, C. Chuah, S. Teoh, K. Zhang, and X. Zhao, "Elisha: a visual and interactive tool for bgp anomaly detection and analysis," DETER Community Workshop on Cyber Security Experimentation and Test, Jun. 2006.
〔30〕 K. Butler and P. McDaniel, "Testing large scale bgp security in replayable network environments," DETER Community Workshop on Cyber Security Experimentation and Test, Jun. 2006.
〔31〕 C. Ko, A. Hussain, S. Schwab, R. Thomas, and B. Wilson, "Towards systemic ids evaluataion," DETER Community Workshop on Cyber Security Experimentation and Test, Jun. 2006.
〔32〕 J. Li, D. Lim, and K. Sollins, "Dependency-based distributed intrusion detection," DETER Community Workshop on Cyber Security Experimentation and Test, Aug. 2007.
〔33〕 C. Neuman, D. Dayama, and A. Viswanathan, "Emulating an embedded firewall," deter community workshop on Cyber Security Experimentation and Test, Aug. 2007.
〔34〕 N. Weaver and V. Paxon, "Stress-testing a gbps intrusion prevention device on deter," DETER Community Workshop on Cyber Security Experimentation and Test, Jun. 2006.
〔35〕 P. Peterson and P. Reiher, "Security exercises for the online classroom with deter," Proceedings of the 3rd international conference on Cyber security experimentation and test (CSET'10), Aug. 2010.
〔36〕 Slowloris, http://en.wikipedia.org/wiki/Slowloris
〔37〕 Slowloris HTTP DoS, http://ha.ckers.org/slowloris

〈저 자 소 개〉

이 만 희 (Manhee Lee) 종신회원
1995년: 경북대학교 컴퓨터공학과 공학사
1997년: 경북대학교 공학석사
2008년: Texas A&M 대학교 컴퓨터공학과 공학박사
1997년~2003년: 한국과학기술정보연구원 연구원
2008년~2009년: Cisco Systems, San Jose
2010년~2011년: 국가보안기술연구소 선임연구원
2012년~현재: 한남대학교 조교수
〈관심분야〉 네트워크/시스템/스마트폰 보안, 고성능 시스템, 컴퓨터교육

석 우 진 (Woojin Seok) 정회원
1996년: 경북대학교 컴퓨터공학과 학사
2003년: Univ. North Carolina, Computer Science 석사
2008년: 충남대학교 컴퓨터공학과 박사
2003년~현재: 한국과학기술정보연구원 선임연구원
〈관심분야〉 무선/이동 QoS, TCP 성능 분석