오픈 취약성 목록을 이용한 보안 위협 예측에 관한 연구★

허승표*, 이대성**, 김귀남*

요 약

최근 들어 연이어 발생하고 있는 DDoS 공격의 영향으로 정부, 기관, 기업의 보안대책과 관련 법규 제도가 강화되고 있다. 하지만 대규모 네트워크 침해사고 및 서비스 방해공격들은 앞으로도 다시 발생할 가능성이 많으며 이를 예방하기 위해선 미리 취약성을 예측할 수 있는 연구가 이루어져야 한다. 기존의 연구 방법들은 어떤 데이터를 기반으로 예측하였는지가 명확하지 않아 복잡하거나 모호하다는 한계가 있다. 따라서 본 논문은 공신력 있는 기관에서 제공하는 취약점 데이터를 기반으로 예측에 관련된 기계 학습 기술을 이용하여 이전에 발생했던 취약점을 토대로 향후 발생할 수 있는 취약점에 대해 미리 예측할 수 있는 방법을 제안하고, 실험을 통하여 효율성을 검증하였다.

A Study on The Prediction of Security Threat using Open Vulnerability List

Seung Pyo Huh*, Daesung Lee**, Kui Nam Kim*

ABSTRACT

Recently, due to a series of DDoS attacks, government agencies have enhanced security measures and business-related legislation. However, service attack and large network violations or accidents are most likely to occur repeatedly in the near future. In order to prevent this problem, researches must be conducted to predict the vulnerability in advance. The existing research methods do not state the specific data used for the base of the prediction, making the method more complex and imprecise. Therefore this study was conducted using the vulnerability data used for the basis of machine learning technology prediction, which were retrieved from a reputable organization. Also, the study suggested ways to predict the future vulnerabilities based on the weaknesses found in prior methods, and certified the efficiency using experiments.

Key words : Security Threat Prediction, Open Vulnerability List

접수일(2011년 06월 07일), 수정일(1차: 2011년 06월 17일), 게재확정일(2011년 06월 24일)

★ 본 연구는 지식경제부 지역혁신센터사업인 산업기술보호특화센터 지원으로 수행되었음.

* 경기대학교/산업보안학과
** 경기대학교/산업기술보호특화센터

1. 서 론

오늘날 네트워크 공격의 변화는 꾸준히 진화하고 있고 많은 변형 형태로 바뀌어 가고 있다. 지난 2003년 1월 25일 슬래머 웜으로 인한 인터넷 마비 현상으로부터 2011년 3월 4일 대규모 DDoS 공격까지 무차별한 공격으로 정부 및 기관, 기업들이 적지 않은 피해를 입었다. 그 당시 공격 형태를 살펴보면 2003년 발생한 1.25 인터넷 대란은 마이크로소프트(MS)의 SQL 서버 취약점을 집중 공격한 슬래머 웜 바이러스로 인해 대량의 네트워크 트래픽이 유발돼 인터넷 접속을 마비 시킨 것이다. 2009년 7월 7일과 2011년 3월 4일 발생한 공격 형태는 웜 바이러스가 아닌 이미 악성코드에 감염된 PC중에 좀비PC에 해당하는 PC들이 무차별 공격을 발생한 것이다. 현재 공격들은 좀 더 지능화 되었고 공격 목표 지점이 확실해짐으로써 이전 발생한 네트워크 공격과는 다른 양상을 띠고 있다. 최근 OWASP에 발표한 웹 취약성 2010 TOP 10에서는 지난 2007년 분류와 비슷하게 웹 애플리케이션 취약점들의 분포가 두드러지고 있다[1,2]. 또한 OWASP TOP 10 2010때와 2004년도 비교해 봤을 때 공격 위협 동향이 시스템 공격에서 웹 서비스 공격 쪽으로 변화하고 있는 것을 <표 1>를 통해 알 수 있다.

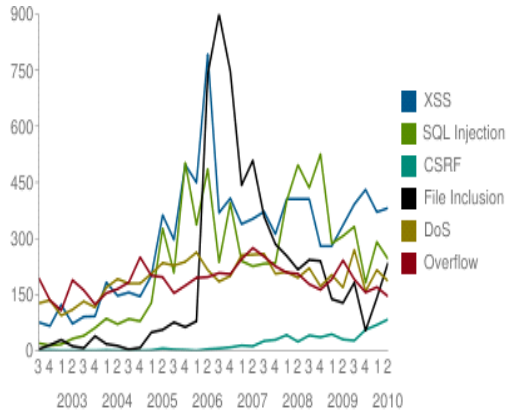
따라서 본 논문에는 취약성 목록의 데이터를 제공하는 기관으로부터 데이터를 수집하고 수집된 데이터를 바탕으로 데이터의 특징을 추출하고 이전 발생된 취약점과 현재 발생된 취약점을 기계학습을 이용해 비교함으로써 향후 발생될 취약점에 대해 예측해 본다.

2. 관련연구

과거부터 현재까지 취약점 동향을 분석해 보고 국내·외 취약점 목록을 이용한 예측에 대해 알아보고 각각 어떤 알고리즘을 사용했는지 설명한다.

2.1 취약점 동향 분석

OSVDB(Open Source Vulnerability Database)에서는 과거부터 현재까지 취약점 목록을 수집해서 데이터베이스화 하고 정확한 데이터 및 자세한 정보를 사용자들에게 제공하고 있다. OSVDB가 2003년부터 현재까지 발표한 년도별, 분기별 취약점 동향은 (그림1)과 같다[3].



(그림 1) OSVDB 취약점 동향 분석

2.2 취약점 목록을 이용한 방법

2.2.1 TOPIC 모델을 이용한 방법

취약점을 체계적으로 분석하고 관리하는 기관 중에 CVE(Common Vulnerability Exposures), NVD(National Vulnerability Database), OSVDB(Open Source Vulnerability Database) 등 여러 기관에서 취약점 목록 체계를 개발하여 새롭게 발견되는 취약점을 관리하고 있다. 이 중 CVE에서 기술되는 설명(Description)부분을 Porter Stemming Algorithm을 사용해서 어근을 추출한 뒤 비감독 학습인 LDA(Latent Dirichlet Allocation)기법을 이용하여 CVE가 갖고 있는 설명부분의 어근을 Topic을 정하고각 단어의 Topic을 정하여 각 년도 별로 취약점의 분포 경향을 나타냈다[4].

2.2.2 취약점 특징을 이용한 방법

CVE 취약점 목록에서 제공하는 Description의 문구를 이용해 접근 권한과 공격 지점을 획득한 뒤 이 두 개를 조합하여 [표 2]와 같은 Class를 분류하였다[5].

"RO"의 의미는 악의적인 사용자가 목적지 호스트의 리모트를 이용해 취약점을 공격해서 관리자 권한을

<표 1> OWASP TOP 10 2010년도와 OWASP TOP 10 2004년도 비교

OWASP TOP 10 2010	OWASP TOP 10 2004
A1 - 인젝션(Injection)	A1 - 유효하지 않은 삽입
A2 - 크로스 사이트 스크립팅(XSS)	A2 - 안전하지 않은 접근 제어
A3 - 취약한 인증과 세션 관리	A3 - 취약한 인증과 세션 관리
A4 - 안전하지 않은 직접 객체 참조	A4 - 크로스 사이트 스트립팅(XSS)
A5 - 크로스 사이트 요청 변조(CSRF)	A5 - 버퍼 오버 플로우
A6 - 보안상 잘못된 구성(신규)	A6 - 인젝션 결함
A7 - 안전하지 않은 암호 저장	A7 - 부적절한 오류 처리
A8 - URL 접근 제한 실패	A8 - 안전하지 않은 저장소
A9 - 불충분한 전송 계층 보호	A9 - 애플리케이션 DoS
A10 - 검증되지 않은 리다이렉트와 포워드(신규)	A10 - 안전하지 않은 구성 관리

획득할 수 있다. "LA"의 의미는 악의적인 사용자가 오직 로컬 지역 내에서만 관리자 권한을 얻을 수 있는 것을 의미한다[5].

<표 2> Vulnerability characteristics

Class	Access route	Obtained right	CVE Examples
RA	Remote	Administrator	CVE-2005-1208
LA	Local	Administrator	CVE-2003-0188
RU	Remote	Users	CVE-2005-1214
LU	Local	Users	CVE-2005-0004
RO	Remote	Un-autorized Access	CVE-2004-0815
LO	Local	Un-autorized Access	CVE-2002-1105
S	Local and Remote	User Required	CVE-2003-0370
O	Local and Remote	Other	CVE-1999-2000

2.2.3 NVD 취약점 목록을 이용한 방법

수 많은 하드웨어, 소프트웨어 제품들의 제로데이 취약점을 분석하기 위해서 NVD에서 제공하는 데이터베이스를 이용하여 Product 필드에 공통적으로 나타나는 취약점을 확인하여 취약한 제품으로부터 나타날 수 있는 과정들을 기반으로 수동적으로 필터를 한다[6].

2.3 예측 알고리즘을 이용한 연구

2.3.1 베이지언 네트워크를 이용한 방법

베이지언 이론은 사전에 일어난 확률과 새로운 정보를 결합하여 사후 확률을 이룩하는 이론이다. 특히 Friedman의 기법은 무한의 알파벳(이벤트의 종류가 한정되지 않은)의 경우에도 잘 활용될 수 있다고 알려져 있다. 이전의 페이지 요청 순서의 빈도를 프로파일링하고 현재의 페이지 요청이 이전의 프로파일과 비교할 때 발생할 확률이 어느 정도 되는지 계산하였다. 이 계산을 통하여 확률적인 낮은 페이지 요청 순서라면 비정상적인 요청으로 판단한다[7].

2.3.2 결정트리를 이용한 방법

결정 트리는 이전 분류된 데이터 셋으로부터 귀납적으로 학습된 모델로 구성하여 결론을 도출 하는 데이터마이닝의 분류 알고리즘 방법 중 하나이다. Srinivas M 외2명[8]은 결정트리를 이용하여 트레이닝 데이터를 기반으로 오용 탐지를 할 수 있고 공격의 종류 중 하나로서 미래에 대한 공격을 예측 할 수 있다고 제안하였다. 이 논문에서는 1999년도에 DARPA에서 TCP dump data를 이용하여 744MB와 4,940,000의 레코드가 들어있는 많은 양의 데이터 셋을 SVM과 비교하여 나타내었다. 결정 트리는 많은 양의 데이터 셋임에도 불구하고 좋은 성과를 비교 분석을 통해 보여준

다[8].

2.3.3 신경 회로망을 이용한 방법

신경회로망 침입탐지(NNID)는 각 사용자에 대한 명령어 실행을 나타내는 데이터들로부터 주기적으로 구성하는 벡터에 대한 각 사용자의 감사로그를 바탕으로 트레이닝 데이터를 수집한다. 수집된 명령어 분산 벡터들로 기반한 사용자 확인을 트레이닝하고 각 새로운 명령어 분산 벡터에 대하여 네트워크 사용자를 확인한다. 이 때, 만약 네트워크 사용자 확인이 실제 사용자로부터 다르다면 이상 징후 신호이다.

3. 제안방법

3.1 오픈 취약성 목록 데이터

본 연구는 보안 취약점 목록을 무료로 열람할 수 있는 기관들의 데이터베이스를 이용하여 과거와 현재 취약점 자료를 기계학습을 통해 향 후 일어날 수 있는 취약점을 예측하고자 하는 것이다. 취약점 정보를 제공하는 기관들의 체계를 알아보고 데이터 선정 및 추출에 대해서 설명한다.

3.1.1 CVE(Common Vulnerability Exposusre)

미 영리 기관인 MITRE에서는 국제표준으로 제공하고 있는 취약성 목록인 CVE(Common Vulnerability Exposures)는 잠재적인 보안 취약성 목록을 관리하고 있다. 개별적은 취약성 데이터베이스 및 보안도구에 포함된 각종 취약성 정보를 공유하기 편하도록 표준화하는데 목적이 있다. 또한 동일한 취약성에 대해 해커와 보안 업체간 다르게 사용해온 명칭을 표준화한 목록이며 오랜 기간 테스트를 거쳐 호환성이 충족된 보안 제품과 서비스에 한해 등록이 가능하다.

CVE는 주로 Intrusion Detection System, Assessment Tools, Vulnerability Databases, Academic Reseach, Incident Report과 같은 곳에서 사용되고 있다.

CVE 구조는 각 취약점 별로 나와 있는 참조 CVE 번호와 CAN 번호로 나뉘어져 있으며 CAN 번호는 Candidate의 약자로서 CVE목록에 들어가기 위해 완전하게 검증되지 않은 후보 목록이다. 각 CVE취약점 참조는 NIST(National Institute of Standard and Technology)의 ICAT 취약점 목록 서비스로 링크되어 있다.

3.1.2 OSVDB(Open-Source Vulnerability Database)

OSVDB는 컴퓨터의 전반적인 보안취약점 수집을 관리하고 있다. 또한 데이터베이스 이용에 대해 자유롭게 배포하고 있다. OSVDB의 수집 분야는 운영체제의 취약성 보안, 소프트웨 제품, 프로토콜, 하드웨어, 디바이스 및 세계 정보기술 사회 공공기반 시설들의 다른 요소들의 정보를 포함하고 있다.

3.2 데이터 추출

3.2.1 취약점 목록 속성 추출

위 두 개의 취약성 목록 데이터 중 OSVDB 기관에서 제공하는 데이터를 이용하여 정보를 추출하였다. <표 4>은 OSVDB에서 자체적으로 취약점을 분류한 속성으로써 취약점의 발생 근원지, 취약점의 영향성, 위협 종류, 취약점 공시 여부 등 여러 가지 정보들을 갖고 있기 때문에 본 연구의 예측 데이터를 실험 하기 위해 쓰였다. <표 3>는 각 속성들이 갖고 있는 데이터들로서 각 데이터를 설명하고 수치화로 나타내었다.

3.2.2 기계학습(Machine Learning) 적용 방법

본 연구는 학습 데이터 셋(Data Set)을 갖고 있기 때문에 사전 확률을 통해 사후 확률을 예측 할 수 있는 베이지언 네트워크 방법과 관심있는 데이터를 분류하기 위해 평가하는 절차를 그래픽으로 표현한 이진트리방법으로 나뉘어 적용할 수 있다. 현존하는 오픈데이터마이닝 툴 중에 weka는 데이터마이닝 문제를 해결하기 위해 만들어진 기계학습 알고리즘을 모아놓은 툴이다. weka에서 제공하는 일반적으로 90%의 학습 셋과 10%의 테스트 셋으로 이뤄진 10 fold 교차 검증(cross-validation) 기법을 주로 사용한다. 이것은 무작위로 교차 검증 테스트를 할 때 학습 셋에 하나의 클래스 학습정보가 터무니없이 부족하거나 없

<표 3> 취약성 분류 수치화 정보

수치화	Location	Impact	Exploit	Disclosure	Attack Type
1	Physical Access Required	Loss of Confidentiality	Exploit Public	OSVDB_verified	Authentication
2	Remote Network Access	Loss of Integrity	Exploit Rumored	Vendor_disputed	Crytographic Vulnerability
3	Dial-up Access Required	Loss of Availability	Unknown	Vendor_verified	DoS
4	Local Access equired	Unknown	Exploit Wormified	Coordinated disclosure	SQL injection
5	Location Unknown	1 and 2	Exploit Commercial	Uncoodinated disclosure	XSS
6	Wireless Vector	1 and 3	Exploit Private	Third-party verified	Overflow
7	Mobile Phone Hand held Device	2 and 3		Third-party disputed	Misconfiguration
8	Local Remote			Discovered in the wild	CSRF, XSRF
9	Context Dependent			Unknown	Other

어서 테스트 결과에 이상한 결과가 나오는 것을 방지할 수 있기 때문이다[9]. 또한 향 후 일어날 수 있는 취약점에 대해선 혼동 행렬(Confusion Matrix) 결과값의 수치 정보를 기반으로 이전 년도와 현재 년도 중에 어느 취약점이 많이 발생했고 적게 발생 했는지 알 수 있으므로 예측할 수 있게 된다.

<표 4> 취약점 속성 분류 및 설명

속성	설명
Location	해당 취약점이 어디서 발생했는지 알 수 있다.
Impact	해당 취약점의 영향성을 알 수 있다. (기밀성, 무결성, 가용성)
Exploit	해당 취약점이 무엇을 위협하는지 알 수 있다.
Disclosure	해당 취약점이 어디서 공시했는지 알 수 있다.
Attack Type	해당 취약점이 어떤 위협 종류인지 알 수 있다.

4. 실험 결과

4.1 실험 방법

오픈 데이터마이닝 툴인 weka는 해당 툴에서 제공하는 포맷 형식이여야만 인식 가능하기 때문에 데이터 수집에 있어서 포맷에 맞춰 수집해야 한다. 포맷 방식은 ARFF(Attribute-Relation File Format)으로 데이터 속성을 명확하게 정의할 수 있는 장점이 있으며 포맷은 헤더와 본문부분으로 크게 나뉠 수 있다. (그림 2)는 ARFF포맷 형식에 맞춰 변화한 결과이다. @attribute 라인을 통해 각 속성에 대한 정보를 기록한다. (그림 2)는 <표 3>의 수치화 정보를 토대로 헤더 부분에 정의돼 있으며 각각의 데이터 수치화 정보는 본문부분에 정의돼 있다. @data는 한 줄에 하나의 레코드(인스턴스)를 기록한다. data{2, 1, 3, 9, 7}은 data{remote access required, loss of confidentiality, exploit unknown, disclosure unknown, misconfiguration}을 의미하는 것이다.

OSVDB 데이터 중 2009년도 데이터 2000개와 2010년도 데이터 2010년도 데이터 2000개 내외의 데이터를 (그림 2)와 같은 형식으로 데이트를 수집하였으며 weka툴을 이용해 기계 학습인 베이지언 네트워크와 결정트리 알고리즘을 적용하여 결과를 도출해 낸다.

4.2 실험 결과

Weka를 사용한 실험 결과는 제안 방법에서도 언급했듯이 10 fold 교차 검증(cross-validation) 방법을 이용하여 분류기의 정확성을 높혔다. (그림 3)과 (그림 4)는 각각 2009년도와 2010년도 취약점 데이터를 Naiv

```
@relation vulnerability

@attribute Location {1, 2, 3, 4, 5, 6, 7, 8, 9}
@attribute Impact {1, 2, 3, 4, 5, 6, 7, 8}
@attribute Exploit {1, 2, 3, 4, 5, 6, 7}
@attribute Disclosure {1, 2, 3, 4, 5, 6, 7, 8, 9}
@attribute Attack_type {1, 2, 3, 4, 5, 6, 7, 8, 9}

@data
2, 1, 3, 9, 7
2, 5, 3, 9, 4
2, 5, 3, 9, 4
2, 5, 1, 5, 4
2, 5, 1, 9, 4
2, 5, 3, 9, 4
2, 2, 3, 3, 2
2, 5, 3, 9, 4
2, 5, 3, 9, 4
2, 5, 3, 3, 4
2, 2, 1, 3, 7
4, 3, 3, 3, 3
```

(그림 2) ARFF 포맷 형식

```
=== Run information ===

Scheme:       weka.classifiers.bayes.NaiveBayesSimple
Relation:     vulnerability
Instances:    1991
Attributes:   5
              Location
              Impact
              Exploit
              Disclosure
              Attack_type
Test mode:    10-fold cross-validation

=== Classifier model (full training set) ===
Time taken to build model: 0.02 seconds

=== Stratified cross-validation ===
=== Summary ===

Correctly Classified Instances        1454               73.0286 %
Incorrectly Classified Instances       537               26.9714 %
Kappa statistic                          0.6275
Mean absolute error                      0.1081
Root mean squared error                  0.2334
Relative absolute error                 50.3236 %
Root relative squared error             71.2353 %
Total Number of Instances             1991
```

(그림 3) 2009년도 나이브 베이지언 도출 결과

e Bayes 방법으로 도출한 결과이다. 2009년도 결과를 보면 데이터 예측할 수 있는 비율이 약73%를 보였고 나머지는 부정확한 데이터로 분류 되었다. 또한, 2009년도 취약점 데이터 중 1454개의 취약점 데이터는 Attack Type의 관측 값에 따라 나머지 속성 정보와 함께 향 후 발생할 취약점에 대해 예측 할 수 있는 이전 데이터로서 신뢰도를 확보할 수 있다.

(그림 4)는 2010년도 취약점 데이터를 Naive Bayes 방법으로 결과를 도출한 결과이다. 2009년도와 비슷한 비율로 데이터의 예측 비율을 보였다. 2009년도에 예측된 값과 거의 유사하다는 것을 알 수 있었다. 즉, 2009년도와 2010년도의 관측값인 Attack Type 속성은 서로 유사하다는 결과를 도출 할 수 있었다.

(그림 5)와 (그림 6)은 2009년도와 2009년도의 혼동행렬 결과로서 정확, 부정확 레코드들의 개수를 나타낸다. (그림 5)는 정확도가 높은 값을 기준으로 SQL Injection, XSS, Misconfiguration 순으로 월등히 높았고 2010년도에도 마찬가지로 XSS공격 SQLInjection 보다 다소 높았지만 주로 웹 서비스 공격 비율이 많이 차지하는 것을 볼 수 있었다.

```
=== Run information ===

Scheme:       weka.classifiers.bayes.NaiveBayesSimple
Relation:     vulnerability
Instances:    2064
Attributes:   5
              Location
              Impact
              Exploit
              Disclosure
              Attack_type
Test mode:    10-fold cross-validation

=== Classifier model (full training set) ===
Time taken to build model: 0.02 seconds

=== Stratified cross-validation ===
=== Summary ===

Correctly Classified Instances        1453               70.3973 %
Incorrectly Classified Instances       611               29.6027 %
Kappa statistic                          0.6099
Mean absolute error                      0.121
Root mean squared error                  0.247
Relative absolute error                 54.2313 %
Root relative squared error             73.9702 %
Total Number of Instances             2064
```

(그림 4) 2010년도 나이브 베이지언 도출 결과

2009나도 나이브 베이지언 confusion matrix

a	b	c	d	e	f	g	
17	0	0	28	3	21	10	a = Authentication
2	72	1	0	1	0	1	b = DoS
1	0	741	8	0	1	1	c = SQL Injection
1	0	8	477	0	29	1	d = XSS
2	1	3	53	46	3	10	e = Overflow
5	0	29	205	5	90	7	f = Misconfiguration
8	3	1	64	10	10	11	g = Other

(그림 5) 2009년도 취약점 데이터 혼동행렬 결과

2010년도 나이브 베이지언 confusion matrix

a	b	c	d	e	f	g	
13	0	1	17	0	16	5	a = Authentication
0	77	0	2	2	0	1	b = DoS
0	0	494	8	0	1	0	c = SQL Injection
0	0	6	564	0	11	2	d = XSS
1	0	0	24	21	2	24	e = Overflow
5	0	69	242	2	183	10	f = Misconfiguration
7	0	4	130	4	14	101	g = Other

(그림 6) 2010년도 취약점 데이터 혼동행렬 결과

5. 결 론

본 논문은 공신력 있는 기관에서 제공하는 취약점 데이터 목록을 기반으로 예측에 관련된 기계 학습 기술을 이용하여 이전에 발생했던 취약점을 토대로 향후 발생할 수 있는 취약점에 대해 미리 예측할 수 있는 방법을 제안하였다.

많은 분야에서 예측 방법 활용을 무수히 사용함에도 불구하고 정보보안 분야에서는 연구되지 않았다. 본 제안 방법은 취약점 데이터의 정확도를 검증하고 예측 학습 방법들과 비교함으로써 효율성을 확인 하였다. 취약점 종류를 살펴보면 이전 데이터에서는 시스템을 향한 공격이 많았다면 현재는 웹 서비스를 향한 공격이 대부분이고 위협 종류도 사생활 및 상업적인 위협이 증가 하고 있다는 것을 본 연구를 통해 알아보았다.

본 논문의 제안 방법을 통해 향후 발생할 취약점을 미리 예측함으로써 공격에 대한 대응 방안을 마련할 수 있는 계기를 마련한다. 취약점 목록 데이터의 신뢰도가 70%에 해당하지만 나머지 30%의 부정확한 예측 비율을 감소시키기 위한 데이터의 정밀한 정규화 및 전처리 과정이 필요하고 다양한 기계 학습 방법을 적용한 연구가 필요하다.

참고문헌

[1] The Open Web Application Security Project, "OWASP TOP 10 Project",. http://www.owasp.org/ .

[2] SANS, “@Risk: The consensus security alert,” http://www.sans.org/newsletters/risk/, September 2009.

[3] OSVDB, " Vulnerabilities in OSVDB disclosed by type by quarter", http://www.osvdb.org/ .

[4] N. Stephan and Z. Thomas, "Security Trend Analysis with CVE Topic Models",

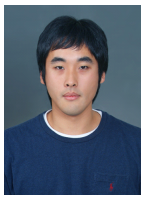
[5] Yeu-Pong L and Po-Lun H, "Using the vulnerability information of computer systems to improve the network security", Computer Communication, pp2032-2047, Mar 2007.

[6] M. McQueen, T. Mcqueen, W. Boyer, S.Mcbride, "Emprical Estimates of 0Day Vulnerabilities in Control System", Proceedings of the SCADA Security Scientific Symposium, Jan 2009

[7] 조상현, 김한성, 이병희, 차상덕, “베이지언 추정을 이용한 웹 서비스 공격 탐지”, 정보보호학회 논문지 13권 2호, 2003, 4

[8] Srinivas Mukkamala, Guadalupe Janoski, An drew Sung, “Intrusion Detection: Support Vector Machines and Neural Networks", Proceedings of the IEEE international joint conference on neural networks, pp1702-1707, 2002

[저 자 소 개]



허 승 표 (Seung-pyo Huh)

2009년 2월 남서울대학교
컴퓨터학과 학사
2011년 현재 경기대학교
산업보안 석사 과정

email : huhspunk@naver.com

이 대 성 (Dae-sung Lee)

1999년 2월 인하대학교
전자계산공학과 학사
2001년 2월 인하대학교
전자계산공학과 석사
2008년 2월 인하대학교
정보공학과 박사

email : xdillema@naver.com

김 귀 남 (Kui-nam Kim)

1989 Univ. of Kansas 수학과 학사
1993 Colorado State Univ
통계학과 석사
1994 Colorado State Univ
산업공학과 박사

email : harap@hanmail.net