한국게임학회 논문집
제2권 제1호 2002년 5월

모바일 네트워크 게임을 위한 침입탐지시스템의 최적화된 침입패턴 저장방법에 관한 연구

신 승 중* 김 귀 남** 이 규 호***
중부대학교 컴퓨터공학부 정보보호관리학과 *
경기대학교 정보보호공학과**
(주)시큐브***
expersin@joongbu.ac.kr

A Study on Optimized method of storing intrusion pattern of Detection System for Mobile Network game

Shin, Seung-Jung* Kim, Gui-Nam*
Joong-Bu University
Kyung-Gi University

요 약

유.무선환경에서 침입 패턴이 다양화되고, 모바일의 편리성이 강조되면서 네트워크 대역폭이 다양한 전송 기반을 요구하고 있다. 그리고 무선기반의 자료가 급증하고 있어, 무선환경에서의 침입탐지시스템 성능에 문제가 될 수 있다. 그러므로 게이트웨이를 근간으로 한 무선전송 기반을 보호하고, 컴퓨터 운영체제 상에 내재된 보안상의 결함을 보호하기 위하여 기존의 운영체제 내에 보안 기능을 추가한 운영체제이며, 커널의 핵심 부분을 인지하여 무선기반의 시스템 사용자에 대한 식별 및 인증, 강제적 접근 통제, 임의적 접근 통제, 해킹 대응 등의 보안 기능 요소들을 갖추게 하여 보안성에 강한 시스템 유지를 요구한다. 그러므로 감시대상의 정보를 미리 알고, WAP 환경하에서 감시대상에 유효한 침입패턴만을 검사하도록 침입패턴 데이터베이스를 분리하는 모델을 제시하여, 이러한 문제점에 대한 해결책을 제시하고자 하였다. 따라서 기존 침입탐지시스템의 패턴 데이터베이스를 분석하였고, 이를 적절히 분리하여 이를 다시 운영체제에 반영하는 기법이다. 그리고 이를 제시한 모델을 검증하고자 실제 구현과 실험을 통해 이를 검증하였다.

Ⅰ. 서 론

모바일을 사용하는 근거는 필요에 따라 장소에 따라 이동이 용이하고 편리하기 때문이다.[11] 그리고 최근에는 IPS의 개념과 Honey-Pot의 개념으로 매우 다양한 기술이 개발되고 있으나, 이러한 기반기술이 무선에 적용하기에는 아직 시기상조인 것으로 사료된다. 그러므로 무선기반의 네트워크 환경에서 유닉스가 가지는 "개방성"은 중요한 특성이지만 컴퓨터 내의 정보보호를 향상시키기 위한 도구는 현재의 표준 유닉스에서는 매우 부족한 실정이다. 이에, 기존 유닉스 시스템의 취약점을 보완하는 패치 버전이나 업그레이드를 통한 임시 방편적인 방법보다는 원천적으로 새로운 무선보호기반의 필요성이 대두되고 있다.

하지만, 이에 따른 역기능으로 컴퓨터 침입 및 범죄로 인한 피해도 날로 증가하고 있어, 이러한 악의적인 행위를 효율적으로 탐지하고 대응하는 기술에 대한 연구가 절실히 요구되고 있다. 본 연구는 앞으로 행해질 모바일 환경에서 보안 개념을 도입하여 인터넷을 기반으로 가상적으로 연구

할 내용이다.[12]

II. 보안 및 Secure OS

1. 보안 시스템의 정의

침입이란 권한이 없는 사용자가 발생시키는 문제 또는 합법한 사용자가 권한을 남용하는 것이라고 정의한다.[1] 또한 이와 더불어 자원의 가용성, 기밀성, 그리고 무결성 등에 저해되는 행동 집합을 침입이라고 정의하기도 한다.[2] 본 논문에서는 이와 같은 정의를 보편적으로 하여, 컴퓨터가 사용하는 자 원의 비밀성(confidentiality), 무결성(integrity), 가용성(availability)을 저해하는 일련의 행위들의 집합이다.

2. Secure OS의 정의

시스템에 대한 보안은 기본적으로 구조를 변경하지 않고 여러 가지 방법으로 개선될 수 있으나 아주 민감한 정보를 보호하고자 한다면, 강력한 개발 전략과 특별한 시스템 구조가 요구된다. 보안 커널 방법은 일반 운영체제에 내재되어 있는 보안 문제점을 해결하기 위하여 운영체제를 설계하는 방법을 말한다. 그 기본 기능으로는 강제적인 접근 제어와 신뢰할 수 있는 경로, 그리고 보호된 경로 등을 들 수 있다. 여기에서 접근 제어는 운영체제 내에서 자원 사용의 주체가 되는 사용자 또는 프로세스가 객체인 파일, 파일 시스템, 디스크 등을 접근할 때 신분이나 규칙에 의하여 해당 객체에 대한 접근을 통제한다.

그리고 강제적인 접근 제어(Mandatory Access Control)로는접근 제어 정책(Access Control Policy), 인증 사용 정책(Authentication Usage Policy), 암호 사용 정책(Cryptographic Usage Policy), 추가 사용 정책있으며, 신뢰할 수 있는 경로와 보호된 경로(Trusted Path & Protected Path)로는 신뢰할 수 있는 경로(Trusted Path)와 보호된 경로(Protected Path)가 있다.

3. 침입 탐지 시스템 분류

3.1 Kumar 분류

Purdue 대학의 S. Kumar는 자신의 박사학위 논문 "Classification and Detection of Computer Intrusions"[3]에서 침입을 행위의 결과에 따라 비정상 침입과 오용 침입의 두 가지로 분류하였다.

구분	비정상침입탐지					오용침입탐지				
침입탐지방법	통계적인 방법(Statistical approaches)	특징 추출(Feature Selection)	비정상적인 행위 측정 방법들(anomaly measures)의 결합	예측 가능한 패턴 생성(Predictive Pattern Generation)	신경망(Neural Network)	조건부 확률(Conditional Probability)	전문가 시스템(Production/Expert System)	상태 전이 분석(State Transition Analysis)	키-스트로크 관찰(Keystroke Monitoring)	룰에 근거한 침입탐지(Model-based Intrusion Detection)

[표 1] kumar의 침입탐지 기법 분류

3.2. COAST 분류

COAST[4]에서는 침입탐지시스템을 크게 다음의 두 가지 분류 기준으로 나누고 있다.

구분	데이터의 소스(Source)를 기반으로 하는 분류			침입의 모델을 기반으로 하는 분류	
분류	(가)단일 호스트 기반(Host Based)	(나)다중 호스트 기반(Multihost Based)	(다)네트워크 기반(Network Based)	비정상 행위 탐지(Anomaly Detection)	(1)오용 탐지(Misuse Detection)

[표 2] COAST의 침입탐지시스템 분류

3.3. ICSA IDSC 분류

ICSA IDSC(Intrusion Detection Systems Consortium)[5]에서는 침입탐지시스템을 크게 다음의 세 가지 분류 기준으로 나누고 있다.

모니터링 접근방식에 따른 분류
- 응용 기반(Application Based)
- 호스트 기반(Host Based)
- 대상 기반(Target Based)
- 네트워크 기반(Network Based)
- 통합(Integrated)

감사데이터의 분석 시점에 따른 분류
- 일괄/인터벌 분석(Batch/Interval Analysis)
- 실시간 분석(Real Time Analysis)

침입 분석 기법에 따른 분류
- 시그너쳐 분석(Signature Analysis)
- 무결성 분석(Integrity Analysis)

- 통계적 분석(Statistical Analysis)

System Design Features		Monitoring Approach				Timing of Analysis	Type of Analysis	
What can it do? "D" = Detects "P" = Prevents "R" = Repairs "S" = Supports		Application Based	Host Based	Target Based	Network Based	Batch/Interval Mode	Signature Analysis	Integrity Analysis
Type	Examples of Security Problems							
Confidentiality	Unauthorized access to files and system resources		D				D	
	Violation of corporate system use policies	D	D				D	
	Violation of coporate security polocies	D	D	D	D		D	D
	Weak or nonexistent passwords	D	D				D	
Integrity	Placement of trojan horses and malicious software		D	P		D	D	P
	Presence of trojan horses and malicious software			D			D	D
	Network service-based attacks				D		D	
	CGI-based attacks	D					D	
Availability	Denial of services attacks				D		D	
	Failure or misconfiguration of firewalls	D			D		D	
	Attacks occurring over encrypted networks	D	D				D	
	Unusual activity or variations from nomal use patterns							
Other	Errors in system or network configuration		D				D	
	Liability exposure associated with attacks using organiztional resources to attack others	P	P	P	P	P	P	P
	Post-incident damage assessment	S	S	S	S	S		S

[표 3] ICSA IDSC의 침입탐지시스템 분류

3.4. IBM Zurich Research Lab. 분류

IBM Zurich Research Lab.[6]에서는 침입탐지시스템의 기능적 특성과 비기능적 특성으로 분류하고 기능적 특성으로 세 가지 분류기준을, 비기능적 특성으로 한 가지 분류기준을 제시하고 있다. 이 분류는 99년 4월 ISO/IEC JTC1/SC27 "IT 침입탐지프레임워크" 회의에서 독일 NB(National Body)가 침입탐지 분류 기준[7]으로 제안한 바 있다.

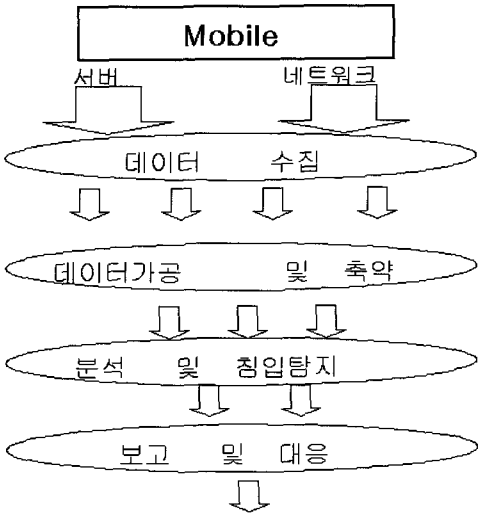
3. 침입탐지시스템 기술적 구성요소

침입탐지 시스템은 [그림 1]와 같이 크게 데이터수집 단계, 데이터의 가공 및 축약 단계, 침입 분석 및 탐지 단계, 그리고 보고 및 대응 단계의 4 단계 구성 요소를 갖는다.

III. 침입 패턴 분리 모델

1. 기존 침입 탐지 시스템의 침입 패턴 데이터베이스

기존 침입 탐지 시스템의 침입 패턴 데이터베이스는 다음



[그림 1] 침입탐지 시스템의 기술적 구성요소

과 같이 세 가지 모델로 분류할 수 있다.

1.1. 침입 패턴 데이터베이스가 존재하지 않는 경우

이 경우는 침입 패턴 데이터베이스가 따로 존재하지 않고, 침입 탐지 시스템 자체에 하드 코딩(Hard coding)되어 있는 경우이다. 이 경우 각 패턴마다 탐지 기능이 최적화 되어있어, 처리 속도가 빠르고, 정형화된 패턴으로 표현할 수 없는 공격도 탐지할 수 있는 장점이 있으나, 오용 탐지 기법을 이용한 침입 탐지 시스템은 항상 새로운 패턴이 추가될 가능성이 존재하므로, 새로운 패턴이 추가될 때마다, 프로그램을 수정해야 하는 단점이 있다.

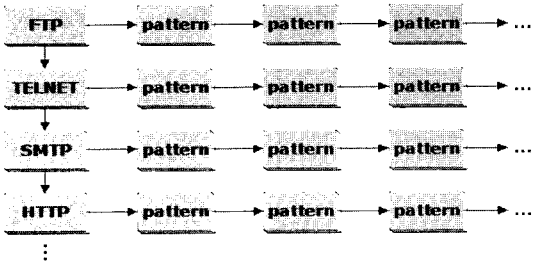
1.2. 하나의 침입 패턴 데이터베이스로 구성된 경우

정형화된 패턴 표현 규칙을 가지고 패턴을 생성하여, 이를 침입 탐지 시스템이 읽어들인다. 한번 제작된 침입 탐지 시스템으로 계속 새로운 침입 패턴을 추가/갱신하여, 새로운 공격에 대해 빠른 대응을 할 수 있으나, 패턴이 많아질수록 검색하는 공간이 늘어나 성능 저하가 급격히 일어난다.

1.3. 서비스별로 분리된 경우

2번의 경우와 마찬가지로 정형화된 패턴 표현 규칙을 가지고 패턴을 생성하여, 이를 침입 탐지 시스템이 읽어들이되, 아래 [그림 2]와 같이 서비스별로 구분된 데이터베이스로 구성된 경우이다. 동적으로 침입 패턴 데이터베이스를 구축할 수 있어 새로운 공격에 대해 빠르게 대응할 수 있으

며, 패턴의 수가 늘어나도 성능 저하가 완만하게 일어난다. 공개된 침입 탐지 시스템인 SNORT[7] 등 많은 침입 탐지 시스템에서 사용되는 메커니즘이다.



[그림 2] 서비스 기준으로 분리된 패턴데이터 베이스 구조

2. 서버 정보를 이용하여 분리된 패턴 데이터베이스

2.1 침입 패턴의 분류

세계적으로 컴퓨터 시스템 취약성과 관련된 정보들을 취합하여 정리하고 있는 인터넷 사이트에 존재하는 침입 패턴들을 분석한 결과 침입 패턴들을 다음 [표 4]과 같은 기준으로 분류할 수 있었다.[8,9,10]

기준	공격방법	서비스	Secure OS(OS)
구분	- 서비스 거부 공격(DoS), 포트 프로빙(Port Probing) 공격 - OS 커널, 응용 프로그램, 프로토콜의 취약성을 이용한 공격	- HTTP - SMTP - FTP - TELNET - 기타	- SunOS - HP-UX - AIX - DG/UX - LINUX - Windows - 기타

[표 4] 침입 패턴의 분류

2.2. 침입

패턴 데이터베이스 분리

분리된 침입 패턴 데이터베이스는 각 데이터베이스에 균등하게 침입 패턴이 할당될 경우 최대 효과를 얻을 수 있다.

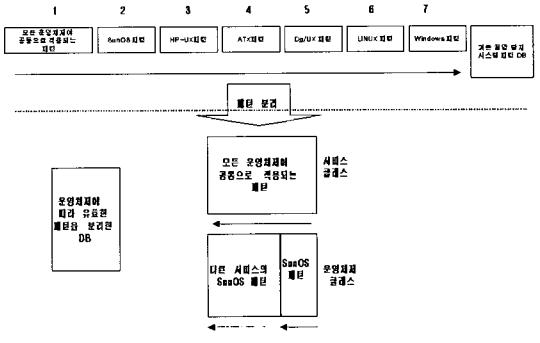
2.3. 분리된 침입 패턴 데이터베이스의 효과

[그림 3]와 같이 기존 침입 탐지 시스템에서는 최악의 경우 ①+②+③+④+⑤+⑥+⑦의 공간을 검색해야 한다. 하지만 본 논문에서 제안한 데이터베이스 모델의 경우 만약 수집된 패킷이 SunOS를 사용하는 호스트를 목적지로 하는 패킷이었다면, ①+②+[다른 서비스의 SunOS 패턴]의 공간만을 검색하면 된다. 여기서 [다른 서비스의 SunOS 패턴] 즉,

클래스	패턴 DB	설 명	비고
서비스	HTTP 패턴 DB	각 프로토콜에 관련된 공격에 대한 패턴 저장	모든 Secure OS에 공통적으로 유효한 패턴을 서비스별로 분리하여 저장
	SMTP 패턴 DB		
	FTP 패턴 DB		
	TELNET 패턴 DB		
	기타	기타 프로토콜 관련 공격에 대한 프로토콜 저장	
운영체제	SunOS 패턴 DB	각 Secure OS에 관련된 공격에 대한 패턴 저장	특정 Secure OS에 유효한 공격 패턴 저장
	HP-UX 패턴 DB		
	AIX 패턴 DB		
	DG/UX 패턴 DB		
	LINUX 패턴 DB		
	Windows 패턴 DB		
	기타	기타 Secure OS 관련 공격에 대한 프로토콜 저장	

[표 5] 분리된 침입 패턴 데이터베이스

[그림 3]의 오른쪽에 점선 화살표로 표시된 부분은 4장에서 설명될 '분석 및 침입 판정 모듈의 동작 과정' 에서 알 수 있듯이, 분석 및 침입 판정 모듈은 가장 먼저 포트 정보를 비교하므로, 그 뒤에 수행하게 되는 프로토콜, 플래그, 패킷 데이터(data / payload) 비교를 수행하지 않아 실제로 ③+④+⑤+⑥+⑦을 검색하는 계산량보다 훨씬 적은 적은 계산량을 필요로 하게 된다.

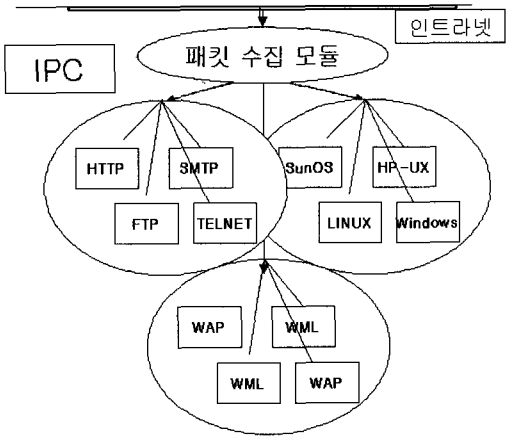


[그림 3] 침입 패턴 데이터베이스 검색 공간 비교

3. 분리된 침입 탐지 패턴 데이터베이스 기반 네트워크 침입탐지 시스템

3.1. 전체 시스템 구성

분리된 침입 탐지 패턴 데이터베이스를 기반으로 하는 네트워크 침입 탐지 시스템의 전체 구성도는 다음 [그림 4]와 같다.



[그림 4] 전체 시스템 구성도

V. 분리된 침입 탐지 패턴 데이터베이스 기반 네트워크 침입탐지 시스템 구현 및 성능 시험

1. 시스템 구현

1.1. 구현 환경

본 논문에서 제안한 모델을 시험하기 위해 제작한 프로그램의 구현환경은 아래와 같다.

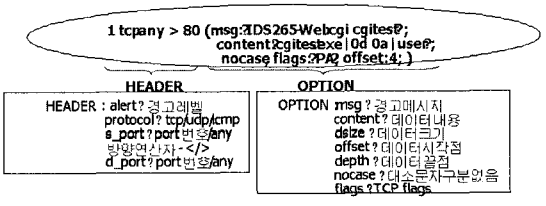
- 운영 체제 : Solaris 7
- 플랫폼 : Sun Ultra 60 (CPU : 450MHz, RAM : 512Mbytes)
- 언어 : C language
- 컴파일러 : GCC 2.95.2
- LEX & YACC : 패턴 데이터베이스 문법에 따른 패턴 구문 분석(Syntax Analysis)을 위해 사용

1.2. 시스템 구현

전체 시스템은 앞장에서 설계한대로 각 모듈을 구현하였으며, 이 때 각 기능 모듈은 다음과 같은 두 프로그램에 구현되었다.

- frontend : 호스트 정보 수집 모듈, 패킷 수집 및 축약 모듈
- engine : 분석 및 침입 판정 모듈, 침입 탐지보고 모듈

또한 이 시스템 구현에 사용된 패턴 데이터베이스의 각 패턴 구성은 아래 [그림 5]와 같으며, 이를 입력으로 읽어들여 구문 분석을 하기 위해 LEX와 YACC을 사용하였다.



[그림 5] 침입 패턴 형식

2. 성능 시험 시나리오

성능 시험 시나리오는 제시된 모델의 성능을 정확히 검증하기 위해, 두 가지 측면, 즉, 침입 패턴을 가지고 있지 않은 패킷을 처리할 경우와 침입 패턴을 가지고 있는 경우를 고려하여, Random 패킷 처리 능력 시험과, 공격 sequence 처리 능력 시험으로 구성하였다.

2.1. 시험 대상

- Linear Model : 패턴 데이터베이스의 분리 없이 선형으로 구성
- Service Model : 서비스별로 패턴 데이터베이스 구성
- OS Model : 본 논문에서 제시한 모델로, 서비스별OS별로 패턴 데이터베이스 구성

2.2 공격 Sequence 처리 능력 시험

가. 시험 목적 : 패턴 데이터베이스에 존재하는 공격 sequence에 대한 각 시험 대상의 처리 능력을 측정한다.

나. 시험 방법

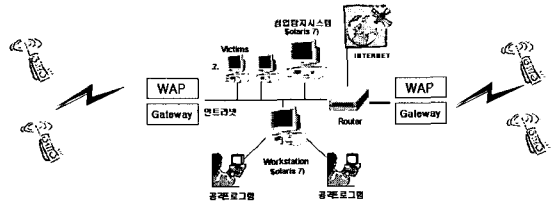
- 정해진 시간동안(1분) 구현된 공격 프로그램을 통해 패턴 데이터베이스에 존재하는 공격 signature를 가지고 있는 패킷을 생성하여 네트워크에 흐르게 한다.
- 전송된 패킷 수와 각 시험 대상이 침입으로 탐지한 패킷의 수를 측정한다.
- 시험에 사용된 패턴은 총 112개로 [표 6]과 같이 가급적 균등하게 분배된 패턴 데이터베이스를 구성하여 시험한다.
- 같은 시험을 10회 반복하여, 평균값을 계산한다.

모델명	데이터베이스	패턴 수
Linear Model	단일 데이터베이스	270 개
Service Model	HTTP	35 개
	FTP	30 개
	SMTP	20 개
	ETC	22 개
OS Model	HTTP	11 개
	FTP	12 개
	SMTP	13 개
	ETC	15 개
	UNIX	10 개
	LINUX	9 개
	Windows	4 개
	ETC	6 개

[표 6] 시험 대상 패턴 데이터베이스 구성

3. 시험 환경

앞에서 제시한 시스템의 성능 시험 시나리오를 위해 구성된 시험 환경은 아래 [그림 6]과 같다.



[그림 6] 실험 환경 구성도

4. 시험 결과 및 분석

본 연구의 목적은 시나리오를 통한 평가 후 결과를 분석하여 문제를 해결하는데 있다. 그러므로 앞에서 기술한 시험 시나리오에 따라 시험한 결과를 표로 정리하면 아래 [표 7]와 같다.

시나리오	공격 sequence 처리 능력 시험
총 전송 패킷 수	약 217,421 개
Liner Model	21,500 탐지
Service Model	52,470 탐지
OS Model	65,401 탐지

[표 7] 성능 시험 결과표

공격 sequence 처리 능력 시험 결과를 살펴보면, OS Model이 가장 높은 결과를 보여주었으며, 그 다음으로 Service Model과 Linear Model의 순서로 결과가 나왔다. OS Model의 경우 Linear Model에 비해 약 3.4배, Service Model에 비해 8.0% 높은 성능을 보여 본 논문에서 제시한 모델의 성능을 검증할 수 있었다. 하지만, 앞에서 기술한 분리된 패턴 데이터베이스 모델의 효과대로 계산하였을 경우 약 20% 이상의 성능 향상을 예상할 수 있으나 이에는 미치지 못하였다. 이는 앞에서 설명한대로 침입 패턴의 수가 충분히 많지 않아, 메시지 큐 분배에 따른 overhead가 크게 작용하여, 제시한 모델의 장점을 충분히 발휘할 수 없었기 때문으로 분석된다. 향후, 정확한 공격 패턴 분석을 통한 패턴 데이터베이스의 효과적인 분리에 대한 연구를 지속하여 좀 더 많은 침입 패턴을 가지고 시험할 경우 충분히 만족스러운 결과를 얻을 수 있을 것으로 생각된다.

V. 결론 및 향후 연구

1. 결론

기존의 침입 탐지 시스템들은 침입 패턴 데이터베이스를 유지함에 있어, 모든 패턴을 하나의 데이터베이스 혹은 서비스 별로 분리한 데이터베이스에 유지하여, 검색 공간이 너무 크고, 감시 대상 호스트에 유효하지 않은 패턴들도 비교하는 불필요한 작업들이 많이 수행되었다. 이를 해결하기 위해 본 논문에서는 침입 패턴 데이터베이스를 호스트 정보(OS 타입)를 기준으로 다시 분리하여, 수집된 데이터에 대해 검색하는 침입 패턴 데이터베이스 공간을 축소시켰다. 이를 통하여 수집된 데이터를 빠른 시간 안에 처리하여, 더 많은 패킷을 수집 및 분석할 수 있어, 결과적으로 침입 탐지 시스템의 탐지 효율을 높이는 효과를 가져왔다. 따라서 같은 성능을 가진 하드웨어 플랫폼 상에서 침입 탐지 시스템이 보다 효율적으로 침입을 탐지할 수 있는 방안을 제시하였다.

2. 시사점

본 논문에서는 감시 대상 호스트의 Secure OS 종류 정보만을 가지고, 기존 침입 패턴 데이터베이스에서 패턴을 분

리해 냈으나, 특정 공격이 성공하기 위해서는 Secure OS 종류 및 버전 그리고, 그 위에서 동작하는 응용 프로그램 버전, 패치 여부 등이 매우 중요한 요소로 작용한다. 이러한 점을 감안할 때, 감시 대상이 되는 호스트에 대해 좀 더 자세한 정보를 알고, 침입 패턴 데이터베이스를 더 자세히 분류하고, 이때 필요한 적절한 침입 탐지 엔진 프로세스의 개수를 도출해낼 경우 본 논문에서 제시한 것보다 더 좋은 성과를 얻을 수 있을 것으로 생각된다.

또한, 네트워크 트래픽 및 해킹 추세(Hacking Trends)를 고려하여, 발생 빈도에 따라 패턴 데이터베이스 내의 침입 패턴의 배열을 다르게 구성하는 방법도 침입 탐지 엔진의 성능 향상에 도움을 줄 수 있을 것으로 생각된다. 앞으로 이러한 부분들에 대한 연구가 진행되어야 할 것이다.

VI. 참고 문헌

[1] B.Mukherjee, T.L. Heberlein, and K.N.Kevitt, "Network intrusion Detection", IEEE Network, 8(3):26-41, May/June 1994

[2] R. Heady, G.Luger, A.Maccabe, and M.Servilla, "The Architecture of a Network Level Intrusion Detection System", Technical Report, Computer Science Department, University of New Mexico, August 1990.

[3] S. Kumar, Classification and Detection of Computer Intrusions, Purdue University, Aug, 1995

[4] http://www.cs.purdue.edu/coast/intrusion-detection/welcome.html

[5] http://www.icsa.net/services/consortia/intrusion/

[6] H. Debar, M. Dacier, and Andreas Wespi, Towards a Taxonomy of Intrusion Detection Systems, IBM Research Division, Zurich Research Lab., Research Report RZ 3030, June 1998.

[7] http://www.snort.org/

[8] http://www.whitehats.com/

[9] http://cve.mitre.org/

[10]http://www.cert.org/

[11]N.Bambos, Toward power-sensitive network architectures in wiress communications: Concepts, issues and design aspects, IEEE personal Communications (June 1998)50-59

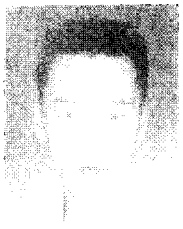
[12] IEEE draft standard P802.11D2.1 Wiress MAC and Physical Layer Working Group(1995)

신승중

1984. 2 한성대학교 경영학과 졸업
1988. 8 세종대학교 대학원 경영학과 졸업
1994. 2 건국대학교 산업대학원 전자계산학과 졸업
2000. 8 국민대학교 대학원 정보관리학과 졸업
현재 중부대학교 컴퓨터공학부 정보보호관리학 주임교수
현재 한국사이버테러정보전학회 이사
현재 한국SI학회 이사
현재 한국진흥정보학회 종신회원
관심분야 : 모바일보안, 데이터전송알고리즘, 네트워크보안, 모바일게임

김귀남

미국 캐자스대학 수학과 (응용수학사)
미국 콜로라도주립대학 통계학과 (통계학석사)
미국 콜로라도주립대학 기계,산업공학과 (기계,산업공학박사)
현재 경기대학교 정보보호기술공학과 주임교수



이규호

1999년 아주대학교 정보및 컴퓨터공학부 (공학사)
2001년 아주대학교 컴퓨터공학과 (공학사)
2001년 경기대학교 정보보호기술공학과 (박사과정)
2001년 ~ 현재 (주)시큐브 선임연구원
관심분야 : 정보보호