Journal of The Korea Institute of Information Security & Cryptology ISSN 1598-3986(Print)
VOL.25, NO.5, Oct. 2015 ISSN 2288-2715(Online)
http://dx.doi.org/10.13089/JKIISC.2015.25.5.1043

지능형 위협인지 및 능동적 탐지대응을 위한 Snort 침입탐지규칙 연구

한 동 희,† 이 상 진‡
고려대학교 정보보호대학원

Study of Snort Intrusion Detection Rules for Recognition of Intelligent Threats and Response of Active Detection

Dong-hee Han,† Sang-jin Lee‡
Center for Information Security Technologies, Korea University

요 약

지능형 위협을 빠르게 인지하고 능동적으로 탐지 및 대응하기 위해 주요 공공단체 및 민간기관에서는 침입탐지시스템(IDS)을 관리·운영하고 있으며, 이는 공격의 검출 및 탐지에 매우 중요한 역할을 한다. 그러나 IDS 경보의 대부분은 오탐(false positive)을 생성하는 문제가 있다. 또한, 알려지지 않은 악성코드를 탐지하고 사전에 위협을 인지·대응하기 위해서 APT대응솔루션이나 행위기반체계를 도입·운영하고 있다. 이는 가상기술을 이용해 악성코드를 직접실행하고 가상환경에서 이상행위를 탐지하거나 또는 다른방식으로 알려지지 않은 공격을 탐지한다. 그러나 이 또한 가상환경 회피, 트래픽 전수조사에 대한 성능적 문제, 정책오류 등의 약점 등이 존재한다. 이에 따라 결과적으로 효과적인 침입탐지를 위해서는 보안관제 고도화가 매우 중요하다. 본 논문에서는 보안관제 고도화의 한가지 방안으로 침입탐지시스템의 주요 단점인 오탐(false positive)을 줄이는 방안에 대해 논한다. G기관의 경험적 데이터를 근거로 실험을 수행한 결과 세 가지 유형 11가지 규칙을 도출하였다. 이 규칙을 준수하여 테스트한 결과 전반적인 오탐율이 30%~50% 이상 줄어들고 성능이 30% 이상 향상됨을 검증하였다.

ABSTRACT

In order to recognize intelligent threats quickly and detect and respond to them actively, major public bodies and private institutions operate and administer an Intrusion Detection Systems (IDS), which plays a very important role in finding and detecting attacks. However, most IDS alerts have a problem that they generate false positives. In addition, in order to detect unknown malicious codes and recognize and respond to their threats in advance, APT response solutions or actions based systems are introduced and operated. These execute malicious codes directly using virtual technology and detect abnormal activities in virtual environments or unknown attacks with other methods. However, these, too, have weaknesses such as the avoidance of the virtual environments, the problem of performance about total inspection of traffic and errors in policy. Accordingly, for the effective detection of intrusion, it is very important to enhance security monitoring, consequentially. This study discusses a plan for the reduction of false positives as a plan for the enhancement of security monitoring. As a result of an experiment based on the empirical data of G, rules were drawn in three types and 11 kinds. As a result of a test following these rules, it was verified that the overall detection rate decreased by 30% to 50%, and the performance was improved by over 30%.

Keywords: False Positive, Intrusion detection, APT attack, ISTS 2015, Network Forensic

접수일(2015년 6월 4일), 수정일(1차: 2015년 8월 11일, 2차: 2015년 9월 14일), 게재확정일(2015년 9월 21일)

† 주저자, shoal81@korea.ac.kr
‡ 교신저자, sangjin@korea.ac.kr(Corresponding author)

I. 서 론

사이버 공간에서 적으로부터의 위협을 능동적으로 인지 및 판단하여 대응할 수 있을까? 대부분 기관이나 주요 공공단체는 사이버 공격도 물리적인 공격처럼 주요 위협과 위험요소를 객관적으로 측정하여 사이버 위협을 사전에 탐지· 관리하기를 원한다. 하지만 고도화, 복잡화되고 있는 보안 위협을 기관이나 기업 스스로 사전에 방어하고 대응 전략을 짜내는 것은 이제 거의 불가능해졌다. 기업의 한정된 보안 IT 예산으로 이를 감당하기에는 보안 위협이 너무 크고 심각해졌기 때문이다. 이에 따라 정부는 종합적이고 효과적인 사이버 공격 방어체제를 구현하기 위해 각 공공분야(에너지, 국방, 교육 등)를 대상으로 사이버 보안관제센터를 설치·운영·위탁토록 '국가 사이버안전관리규정'[1]에서 정하고 있고, 민간분야도 개인정보유출방지, 산업기밀보호 등 기관의 사업특성에 맞는 정보보안을 강화하고 있다. 이렇듯 많은 공공분야 기관 및 중요 민간기관이 보안관제 서비스를 받고 있으나, 기준이 되는 보안관제 업무 평가 지표 및 제도가 부재하여 업무 관리실태를 파악할 수 없는 실정이다.[2] 주요 기관 및 기업에서 중요하게 사용되는 IDS 솔루션이 약 20 년 넘게 사용되었으나, IDS의 가장 큰 단점은 아직도 해결이 되지 못했다. IDS 이벤트의 대부분은 거짓 경고이고 정상경고는 극히 일부분이라는 것이다. 이 문제점은 꽤 오래전부터 제기되어 왔다. 과거 미국에서는 'IDS 무용론'까지 대두 되었다.[3] '무용론'이 논쟁거리가 된 후 여러 가지 새로운 움직임이 생겼다. 차단기능을 추가한 IPS나 웹서버 방어에 특화된 웹방화벽, 통합 방어시스템으로 응용서비스까지 제어할 수 있는 UTM 등 IDS와의 차별화를 주장하는 보안솔루션들이 출시되기 시작했다. 그러나 어떤 보안솔루션도 패턴 매치 기법의 문제, 특히 오탐 문제를 해결하지는 못했다. 또한 APT위협 및 알려지지 않은 공격을 탐지하기 위해 APT 대응솔루션, 스텔스공격 대응 솔루션 등을 이용해 이런 종류의 공격에 대응하고 있으나 이 또한 전수검사의 취약성, 정책설정 오류, 가상환경 회피등의 약점이 존재하여 오탐의 가능성이 항상 남아 있다. 결과적으로 그 어떤 솔루션으로 모든 공격을 방어하기는 불가능하기 때문에 보안관제의 고도화가 매우 중요하다. 오탐의 증가로 인해 전체 로그의 발생량이 증가하면 그 중에서 진짜 공격 시도를 찾아낼 가능성은 점점 줄어들게 되며, 로그가 누적될수록 공격 시도를 적시에 찾아낼 가능성은 더욱 줄어들게 된다. 즉, 이 얘기는 오탐이 증가하면 공격 시도가 있었다는 사실 자체를 아예 모를 가능성이 커진다는 뜻이다. [4] 본 논문에서는 이에 대한 연구를 수행한 과거 논문들의 장단점을 논하고 이와 차별된 본 논문의 특징을 밝힌다. 이를 위해 세 가지 관점에서 규칙을 선정하였으며, 그에 따른 규칙을 과학적으로 테스트하여 효율적인 규칙을 제안하는 데 그 목적이 있다.

II. 선행 연구

2.1 Snort 구조와 규칙

Snort[5]는 IDS를 통과하는 모든 패킷을 수집하는데 이 단계가 첫 번째 단계 sniffer, 즉 Decode 엔진이다. 다음은 preprocessor로 효율적인 공격 탐지를 위해 몇 가지 플러그인을 먼저 거쳐 매칭되는지 확인하는 단계이다. 이것이 두 번째 단계이다. 그리고 중요한 3번째 단계는 '탐지엔진' 단계이다. 여기서는 Snort의 핵심이라고 할 수 있는 규칙을 탐지하는 단계로 사용자에 의해 자동 혹은 수동으로 정의된 rule에 의해 탐지되는 단계이다. 마지막 4번째 단계는 출력단계이다. 이는 옵션을 정의하여 콘솔로 출력하거나 별도의 파일로 생성하는 것 두 가지 모두 가능하다. 혹시 별도의 관제를 직접하고 싶은 경우에는 출력을 모니터로 할 수 있고, 파일로 저장하여 네트워크 상태를 분석할 수 있다.

Snort Rule(Fig. 2.)은 두가지 유형으로 나뉘는데. 그 각각은 헤더(Header)와 옵션(Option)으로 불린다. 헤더는 7가지로 이루어져 있는데 각각의 구성요소는 다음과 같다. 헤더는 액션, Protocol, 출발지 IP, 출발지Port, 방향(Direction), 목적지

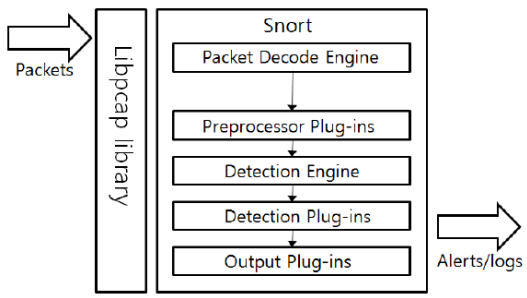

Fig. 1. Packets Process of Snort

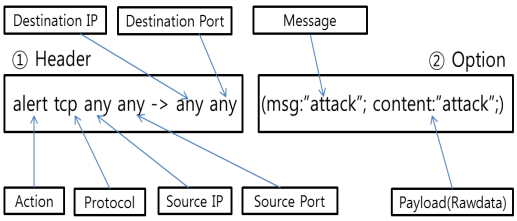

Fig. 2. Rule of Snort

IP, 목적지 Port로 구성된다. 옵션은 탐지룰명과 탐지패턴으로 구별된다.

2.2 기초 연구

FAsieh Mokarian, Ahmad Faraahi, Arash Ghorbannia Delavar[6] 연구에서는 이상징후 탐지방법 중 탐지기술(Detection Techniques) 및 처리기술(Processing Techniques)로 나누어 거짓 알람을 줄일 수 있는 연구를 하였다. Detection Techniques에서 N. B. Anuar, H. Sallehudin, A. Gani, O. Zakari[7]는 KDD Cup99의 훈련 데이터에 기초하여 데이터 마이닝과 의사 결정 트리 분류를 사용하는 하이브리드 통계적 접근 방식을 제안했다. DOS 및 R2L유형에는 의사 결정 트리가 더 적합하고, Probe 및 U2R 유형에는 규칙기반 분류가 적합함이 확인되었다. 이는 해당 유형별로 구분하여 탐지될 시 오탐율이 줄어드는 것을 제시한 성과가 있다.

Processing Techniques에서 Bakar et al[8]는 효율적이지 못한 알고리즘이 오탐을 증가 시키므로 이를 위해서 자료 이동 간의 상관관계를 통해 정탐율을 높이는 기법을 제시한다. 이를 침입경보 품질프레임워크(IAQF)로 제시하였다. 이 프레임워크(IAQF)를 사용하여 데이터의 정확성, 신뢰성, 품질 향상을 높이는 작업을 수행하였다. 위 논문들은 이상탐지(Anomaly detection)기법으로 탐지된 것이다. 이상탐지기법의 특징은 탐지 가능성은 크나, 장기간의 학습 기간이 필요하고 관리하기 어려우며 네트워크 기반 침입탐지에서는 적용이 힘들거나 실제 학습 환경을 만드는 데 어려운 단점이 있어 실무에서는 잘 사용되지 않는다.

오용탐지(Misuses detection) 기법 중 G.C.Tjhai, M.Papadaki, S.M.Furnell, N.L.Clarke[9]는 웹트래픽 관련 공격 규칙을 대부분 느슨하게 기록하였다고 표현하면서, 이를 위해 특정 문자열이나 사용자정의 맞춤형 규칙으로 규칙을 개선해야 되며, 네트워크 환경에 따라 차별화하여 관리하여야 된다고 논하였다. C.A.Obi and M.Papadaki[10]는 키워드 검색을 기반으로 공격에 일정하고 독특한 사용자 맞춤형 Content 탐지 규칙으로 규칙을 설계하였다. 스캔공격 탐지 시 임계값 설정은 모든 공격버전을 탐지 할 수 있는 임계값으로 설정하여야 한다고 강조하였으며, 특히, 시간주기가 가장 중요하고 적절하지 않은 시간주기가 설정되면 오탐이 증가한다고 하였다. 또한, 규칙의 설정 정책을 준수하지 않는 규칙은 무시되거나 적절한 조정 조치를 한뒤에 탐지되어야 한다고 제시하였다. 해당 연구는 예시된 대상 규칙에 대해서 언급하여 임계값의 특징을 제시하는 등의 성과는 있었으나, 이 두 가지 특징이 모든 탐지규칙에 공통으로 적용이 되지는 않았으며 이에 따른 적용기준이 뚜렷하지 않아 다소 미흡한 측면이 있었다. 그 외 한지용, 이인복, 한정희[11]은 시그니쳐 기반 IDS의 PCRE 탐지 성능을 높이는 방법으로 중복되는 PCRE를 가진 시그니쳐 규칙들의 병합과 PCRE를 가진 규칙을 JIT 컴파일을 제안하는 연구가 진행되었으며 상당한 수준의 성능 향상 효과를 보였다.

2.3 연관성 연구

오탐율을 낮추기 위한 기존의 연구[8.9]에서는 몇가지 개선 점을 제안하였으나 각 패턴의 유형별 연관성은 연구되지 못하였다. Elias Raftopoulos and Xenofontas Dimitropoulos[12]는 탐지시그니처의 유용성을 효과적으로 확인하기 위해 악성코드 유형별로 탐지규칙간의 어떠한 연관관계가 있는지를 수치상으로 규명하였다. Fig 3.은 탐지영역별로 시그니쳐 간 상관관계를 나타내며 유형별 Complexity features와 Signature Class에 대한 상관관계로 이 Singature Class는 다섯가지로 이루어진다.[12]

C&C Communication signature 탐지유형은 C&C 서버와 피해자 컴퓨터의 통신시도를 검사할 때 전송 패킷을 여러 조각으로 나누어 확인한다. 예를 들면, C&C서버의 루트킷의 버전과 감염된 피해자 컴퓨터의 루트킷 버전 비교를 서버로 나가는 패킷 내 전송되는 Byte 정보로 구별한 결과에서 알

수 있듯이 탐지 규칙 옵션 중 byte, field, flow가 0.94~0.95로 유사하다는 것이 확인되었다.[12]

Reporting signature 탐지유형은 공격자가 피해자에게 민감한 데이터를 유출하는 것을 탐지하는 유형이다. 피해자 PC에 정보를 분할하여 짧은 문자열을 기록하여 외부 목적 주소로 짧은 데이터를 나누어 보내는 특징이 있으며, 유출되는 데이터와 외부로 나가는 패킷의 특징을 비교하여 탐지한다. 확인결과 옵션flow가 상관관계 0.78로 유사성이 확인되었다.[12]

Egg Download signature 탐지유형은 업데이트 또는 악성 소프트웨어를 다운로드 시 해당 패킷에 특정 문자열이 일치하는지를 탐지한다. 예를 들어 HTTP요청 패킷 내 GET관련 시그니처가 악성패킷과 일치하는지를 확인하는 방법이다. 확인 결과 탐지 옵션 flow가 상관관계 0.6로 유사성이 확인되었다.[12]

Redirection signature 탐지유형은 악의적인 도메인을 탐지한다. 이를 탐지하기 위해 정규표현식을 사용하며 도메인의 약성 여부를 확인한다. 확인결과 규칙옵션 byte는 상관관계 0.88로 유사성이 확인되었다.[12]

Propagation signature 탐지유형은 스캐닝등 불법적인 IP를 탐지하는 기법을 활용한다. 확인결과 규칙옵션 IP가 상관관계 0.7로 유사성이 확인되었다.[12] 이처럼 유형별로 패킷을 탐지하는 시그니처의 종류가 다르므로 그에 대응하는 규칙 및 옵션의 종류도 다르다. 네트워크 패킷 및 상황에 맞지 않는 룰이나 옵션을 사용할 경우 오탐이 상당히 늘어나며, 상황에 맞는 룰이나 옵션이 적재적소에 필요하다는 것을 해당 연구로 알 수 있다.

2.4 본 연구의 특징

본 연구가 기존 선행연구와의 차이점은 최신 Snort 룰을 기준으로 G기관에서 탐지되고 있는 데이터를 탐지된 이벤트에 대한 분석을 통해 규칙들에 대한 문제점을 지적한다는 것이다. 이를 개선하기 위해 유형별로 3가지 관점(기본적·탐지적·성능적 관점)에서 총 11가지의 규칙을 제안했다. 이에 대한 근거는 G기관에서 탐지된 경험적 데이터를 가지고 관제사가 정·오탐을 구분하고 오탐율을 분석한 것이다. 또한, 실험에서는 제안된 규칙 11건이 ISTS DATA SET의 데이터로 테스트한 세부평가결과를 분석하여, 오탐율이 낮아지고 성능이 향상됨을 검증하였다.

III. 탐지규칙 제안

3.1 제안 배경 및 근거

알려지지 않은 악성코드를 탐지하기 위해 많은 솔루션이 현재에도 계속 쏟아져 나오고 있다. 현재까지 출시된 APT 대응 솔루션의 장점은 가상 기술을 이용해 악성코드를 직접 실행 하고 가상환경에서 이상행위를 탐지하는 기능이 특징이다. 이를 활용해 제로데이 등 알려지지 않은 공격을 탐지한다. 또한, 근래의 APT 공격이 주로 특정 타깃을 정하고 메일 또는 SNS 기반으로 전파하는 특성이 있어 이를 탐지하기 위해서는 모든 파일 및 URL에 대한 전수조사를 수행해야 되지만 전수조사에 따른 엄청난 데이터의 유입이 성능과 비용적인 문제를 생기게 만든다. APT장비도 정책설정을 통해 이상탐지패턴을 만들어야 하며 이 정책 또한 오탐의 원인이 되어 이를

	Signature class					
Complexity feature	C&C Communication	Reporting	Egg Download	Redirection	Propagation	All signatures
Bytes checked	**0.95**	**0.52**	0.30	**0.88**	**0.65**	**0.84**
Fields checked	**0.94**	**0.60**	**0.55**	**0.87**	**0.61**	**0.83**
Byte offset is set	**0.48**	0.28	**0.54**	**0.62**	0.34	**0.51**
Regular expression is set	**0.48**	0.27	**0.59**	**0.73**	0.35	**0.52**
Regular expression size	**0.73**	0.24	**0.51**	**0.63**	0.30	0.37
Destination port is set	**0.49**	0.19	0.13	0.35	0.29	0.24
Packet size is set	0.19	0.12	0.19	0.16	**0.40**	0.12
Flow options are set	**0.91**	**0.78**	**0.60**	**0.41**	**0.48**	**0.68**
IP options are set	**0.53**	**0.42**	**0.48**	**0.53**	**0.73**	**0.56**

We highlight in bold correlation values higher than 0.4.

Fig. 3. Complexity features and their correlations with signature effectiveness[12]

관리하는 운영자의 역량이 필요하게 된다. 결과적으로 효과적인 방어를 위해서는 APT솔루션 및 스텔스 대응기술구축뿐만 아니라 이미 구축된 보안시스템에서 축적된 보안이벤트의 상관관계를 분석하거나 보안관제고도화 등 다각적인 방어체계를 구축하는 것 또한 매우 중요하다. 이에 따라 보안관제 고도화 관점에서 시그니쳐기반의 규칙의 오탐이 줄어들면, 지능형 위협을 보다 빠르게 인지하고 오탐이 줄어들어 능동적으로 탐지 및 대응하게 됨으로 APT 및 스텔스대응기술 또한 향상될 것으로 기대 된다. G기관의 경우 APT탐지 솔루션을 사용하고 있으나, 앞에서 언급된 같은 문제가 있어 이를 해결하고자 보안관제 고도화를 추진하고 있다.

G기관의 하루 평균 담당하는 이벤트 발생 건수는 6개월을 기준으로 하루 평균 20만 건 이상이며 이는 초당 2건의 이벤트를 실시간으로 처리해야 하는 트래픽 양이다. 이처럼 보안관제요원의 업무량은 상상을 초월하는 수치이며 이는 인력으로 해결할 수 있는 문제가 아니다. 그럼에도 불구하고 침해대응의 최선두 있는 보안관제는 효율적인 업무절차 및 가이드라인 부재로 침해대응이란 업무가 보안관제사의 주관 및 지식에 따라 좌우되는 실정이다. 이를 개선하기 위해 보안관제 고도화의 3대 요소를 도출하였으며 이 3가지 요소는 신뢰성, 정확성, 신속성이다. 첫 번째, 신뢰성은 믿을 수 있고 관리되어지는 규칙들이라는 것이며 가장 근본이 되는 논리이다. 규칙 자체가 신뢰가 되지 않으면 뿌리 자체가 흔들리기 때문이다. 두 번째, 정확성은 오탐이 가장 적은 가정 정확한 공격을 확실히 찾아내는 특성으로 가장 중요한 요소다, 세 번째로 신속성은 빅 데이터들을 신속하게 처리할 수 있는 속성으로 이 또한 중요하다. 왜냐하면 정확한 공격을 찾았음에도 이를 신속하게 처리하지 못하면 무용지물이기 때문이다. 이처럼, 3대 속성을 탐지 규칙과 연관 지으면 신뢰성은 Basic, 정확성은 Detection, 신속성은 Performance로 분류 될 수 있다. 이러하여 그에 맞는 탐지기준을 표(Table. 1) 같이 제안하고자 한다.

G기관의 보안이벤트 중 6개월 이상 수집된 로그를 중심으로 각 제안 유형별 11가지 규칙에 대한 오탐율을 측정하였다. 이를 근거로 각 유형이 어떠한 경우 오탐이 향상 또는 감소하는 지를 도출하였으며, ISTS 2015 SET으로 최종 실험 테스트를 하였다. 이에 따라 본 논문에서는 오탐 발생 부분을 명시하고 이에 대한 개선방안에 대한 근거 및 방안을 제시하였다.

Table 1. The proposed Rule 11

List	Content	result
Basic	Easy to read	Fig. 6.
	Purposeful activity.	Fig. 7.
Detection	Direction, Protocol, IP, Port detection	Fig. 10.
	Length of content	Fig. 11.
	Payload detection	Fig. 12.
	Fixed, non fixed	Fig. 13.
	IP Partial matching	Fig. 14.
	Mistake in word spacing	Fig. 15.
Performance	Non-payload detection	Fig. 16.
	Merge of PCRE Rule	Fig. 17.
	Location of content	Fig. 18.

3.2 제안 규칙

3.2.1 기본 규칙

· 규칙 작성 시 규칙제작자의 의도를 알기 쉽게 작성
· 룰의 탐지목적이 명확하게 드러나게 계속적 관리

첫 번째로 규칙 작성 시 목적 및 의도를 알기 쉽게 작성하여야 하며, 어려운 HEX 값으로 규칙들이 작성된 경우들이 많다. 아래 예처럼 규칙의 가독성이나 관리효율이 떨어지기 때문에 CHAR값을 주석으로 반드시 넣어 규칙을 작성하여야 한다. 아래 HEX 값을 "5F 78 5F 58 5F 42 4C 4F 43 4B 4D 4F 55 53 45 5F 58 5F 78 5F"를 CHAR값으로 표현하면 '_x_X_BLOCKMOUSE_X_x_'로 표현된다. 이를 주석으로 기재하는 것이 효율적이다. G기관의 경우 이를 제대로 지키지 않아 보안관제사가 업무파악에 애를 먹는 경우가 많았으며 밤을 새고 교대하는 경우가 많은 업무 특성상 인수인계가 제대로 되지 않고, 새로운 규칙의 경우 문서화가 되어있지 않은 경우 실제로 이런 패킷은 진·오탐 파악에 시간이 약 1.5 ~ 2배 정도 많이 걸리는 것으로 파악되었다.

두 번째로 룰의 탐지 목적이 명확하게 드러나게 룰의 생명주기(life cycle) 등을 고려하여, 계속적으로 관리하여야 한다. 아래 예처럼 룰 설명이나

```
alert tcp $EXTERNAL_NET $FILE_DATA_PORTS -> $HOME_NET any (msg:"MALWARE-BACKDOOR Rebhip.A runtime detection"; flow:to_client, established flowbits:isset,file.exe file_data content:"|5F 78 5F 58 5F 42 4C 4F 43 4B 4D 4F 55 53 45 5F 58 5F 78 5F|"; fast_pattern:only content:"|5F 78 5F 58 5F 55 50 44 41 54 45 5F 58 5F 78 5F|"; content:"|5F 78 5F 58 5F 50 41 53 53 57 4F 52 44 4C 49 53 54 5F 58 5F 78 5F|"; distance:0; metadata:service ftp-data, service http, service imap, service pop3; classtype:trojan-activity sid:21967; rev:4;)
```

reference등이 없어 해당 룰의 목적을 알기 힘든 경우가 많다. 이 같은 경우에는 룰 생명주기를 고려하여 즉, 룰의 생성, 삭제, 수정등의 일련의 과정을 관리하여야 한다. 예를 들어 해당룰의 생성(create),수정(modify),삭제(delete)등을 관리하는 별도의 형상파일을 만들어 해당룰이 언제 어떤 의도로 작성되었고 수정이 되었는지 알 수 있게 관리하는 것이 룰의 목적에 맞게 탐지될 수 있도록 추적·관리하는데 적합하다

```
alert tcp $EXTERNAL_NET any -> $HTTP_SERVERS $HTTP_PORTS (msg:"SERVER-WEBAPP view-source "; flow:to_server,established content:"/view-source"; fast_pattern nocase http_uri content:"../";http_raw_uri metadata:ruleset community, service http; classtype:web-application-attack sid:848; rev:20;))
```

해당 룰의 경우 다음과 같이 수정하여 reference를 추가하고 룰 메시지에 룰이 언제 생성 또는 수정되었는지 알 수 있게 기입하는 것이 효과적이다. 이처럼 수정 또는 삭제일자를 기입하여 별도로 관리하는 것이 룰의 목적을 추적하는데 용이하다. 또한 해당 탐지룰이 CVE 1999-0174 로 '1999년도에 발견된 view-source CGI program로 어떤 파일이든 읽을수 있는 취약점공격을 탐지하기 위한 정책'으로 인지되어 목적성이 뚜렷해진다.

```
alert tcp $EXTERNAL_NET any -> $HTTP_SERVERS $HTTP_PORTS (msg:"SERVER-WEBAPP view-source directory traversal_20150102_modify"; flow:to_server,established; content:"/view-source"; fast_pattern; nocase; http_uri; content:"../"; http_raw_uri; metadata:ruleset community, service http; reference:cve,1999-0174; classtype:web-application-attack; sid:848; rev:20;)
```

G기관의 경우 목적이 분명하지 않은 규칙들로 인해 관제시 룰의 목적을 파악하는데 시간이 약 1.5배 정도 오래 걸리는 것으로 파악되었다. 다른 예로, Tor사용 IP탐지, VPN사용 IP탐지 이런 규칙들은 그 자체가 위협행위라기보다는 위협을 보다 앞서 능동적으로 탐지하고자 하는 의도로 해석되지만 이와 같은 경우는 득보다 실이 크다 즉. 대부분의 경우가 오탐이다. 또한 이런 행위를 탐지하더라도 패킷이 대부분 암호화가 되어 있어서 정·오탐 유무가 쉽지가 않으므로 이규칙을 사용하는 횟수나 빈도 등을 누적하여 행위기반탐지규칙에 활용하는 것이 바람직하다고 판단된다.

3.2.2 탐지 규칙

- Direction, Protocol, IP, Port등 정확히 입력
- 매칭데이터의 매칭 패턴의 위치 지정 필요
- 매칭데이터의 효율적인 개수 및 길이 사용
- 매칭데이터 간 고정 및 비고정 패턴 구별
- IP매칭 패턴은 전체보다 부분매칭 패턴 사용
- 매칭 데이터 단어 및 문장 간 경계 지정 필요.

첫번째로 방향(Direction), 프로토콜(Protocol), 포트(Port), IP등 기본정보에 대한 언급이 없거나, Port가 Any로 설정된 경우 오탐률이 상당히 높게 나타난다. 아래 예처럼 서버시스템에 접속하는 일반사용자가 root 계정으로 복귀할 때 "uid=0(root)"라는 시그니쳐가 표시되는 것을 탐지하는 패턴으로 목적지 서버 IP와 Port가 Any로 설정되어 있어 오탐율이 높다

```
alert ip any any -> any any (msg:"INDICATOR-COMPROMISE id check returned root"; content:"uid=0|28|root|29|"; metadata:ruleset community; classtype:bad-unknown sid:498; rev:11;)
```

이에 대한 방법으로 목적지 IP가 Any인 경우 해당 목적지를 특정서버($Server_Farm)만 탐지되게 개선하는 것이 효율적이다. G기관의 경우 해당 유형에 대한 오탐율의 발생빈도는 Port, IP, Direction, Protocol 순서대로 오탐율이 높게 나타났다. 공격 유형에 따라 스캔공격의 경우에는

Port와 IP의 오탐율에 대한 민감도가 Direction과 Protocol보다는 높게 나타났으며, C&C접속 또는 악성코드 다운로드 공격의 경우에는 Direction이 다른 옵션들보다 높게 탐지되어 공격유형에 따라 차이가 발생함을 알 수 있었다.

```
alert ip any any -> $Server_Farm any
(msg:"INDICATOR- COMPROMISE id check
returned root";  content:"uid=0|28|root|29|";
metadata:ruleset community;
classtype:bad-unknown sid:498; rev:11;)
```

두번째로, 매칭된 데이터의 세부옵션 및 위치 지정이 있는 경우와 그렇지 않은 경우 오탐의 비율이 확연히 다르다. (Table 2.)에서처럼 총 18개의 Snort에서 사용되는 탐지되는 Payload 옵션이있다. packet의 경우 해당 옵션의 내용에 따라 패킷 내 페이로드의 활용되는 요소가 구별되어 있으며, http옵션이 별도로 존재하여 http관련 내용에 따라 옵션으로 지정하여 줄 수 있다. 아래 예처럼 ICQ메신저를 사용하면 탐지하라는 규칙을 작성할 때 해당 규칙에 세부 옵션(offset〔출발위치〕, depth〔깊이〕, distance〔출발위치〕, within〔폭〕 등)을 고려하여 추가로 배열하면 오탐을 보다 줄일 수 있다

```
alert tcp $HOME_NET any -> $EXTERNAL_NET any
(msg:"POLICY-SOCIAL ICQ access";
flow:to_server,established
content:"User-Agent|3A|ICQ"; fast_pattern:only
metadata:ruleset community;
classtype:policy-violation sid:541; rev:15;)
```

아래 예에서 offset 및 depth를 추가하여 User-Agent|3A|ICQ; depth 17; offset 0; 등으로 개선할 수 있다. 또한, 예에서 http_header 옵션을 추가하여 http_header 부분만을 탐지하도록 개선하는 것이 훨씬 오탐을 줄일수 있다. G기관의 경우 옵션은 offset-depth〔절대적 위치 시작-끝〕 / distance-within 〔상대적 위치 시작 - 끝〕로 나누어 사용되며, 패턴 위치 지정을 하지 않을 경우 탐지된 이벤트 중 많게는 50% 이상이 오탐으로 판별되었다.

셋째, 매칭 데이터의 경우 효율적인 길이(length)가 오탐의 주요 관건이 된다. 특히, 너무

```
alert tcp $HOME_NET any -> $EXTERNAL_NET any
(msg:"POLICY-SOCIAL ICQ access"; flow
:to_server,established ; content: User-Agent|3A|ICQ";
depth 17; offset 0 fast_pattern:only
http_header;metadata:ruleset community;
classtype:policy-violation sid:541; rev:15;)
```

길면 룰의 갯수가 늘어나 성능에 영향을 주고 너무 짧으면 오탐에 영향을 주기 때문이다. Content 패킷의 합을 Sn, 평균을 f(n)이라고 할 때,

$$Sn = \sum_{k=1}^{n} (a1+a2+\cdots+ak\cdots+an)$$

$$f(n) = Sn / N \text{ (단, } N > 0)$$

Fig. 4.은 Content의 길이별 분표를 그래프로 나타낸 것이다. Content는 평균적으로 15byte가 가장 적당하다. 너무 작으면 오탐이 많으므로 15byte이상을 유지하는 것이 효율적이다.

Table 2. Payload Detection Rule Options

Payload Detection Rule Options	
packet(7)	http(11)
nocase	http_client_body
rawbytes	http_cookie
depth	http_raw_cookie
offset	http_header
distance	http_raw_header
within	http_method
fast_pattern	http_uri
	http_raw_uri
	http_stat_code
	http_stat_msg
	http_body

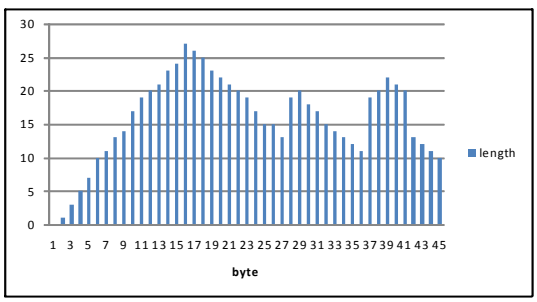

Fig. 4. Distribution by length of Contents

넷째, 매칭데이터 중 고정적으로 사용되는 패턴(SELECT, UNION 등)은 사용하지 않는 것이 탐지간 효율적이다. 아래 예처럼 특정 악성코드를 탐지하는데 고정 패턴인 SELECT을 탐지하는 것은 비효율적이다. 이는 작업을 위해서 데이터베이스에 정상 접속하여 업무하는 행위까지도 공격으로 보는 오탐이 많이 발생한다.

```
alert tcp $EXTERNAL_NET any -> $SQL_SERVERS
$ORACLE_PORTS (msg:"SERVER-ORACLE select union
attempt"; flow:to_server,established; content:"select ";
nocase; content:" union "; nocase; metadata:ruleset
community; classtype:protocol-command-decode; sid:1676;rev:7;)
```

이를 아래 예처럼 자주 사용되는 고정패턴을 삭제하고 비고정패턴을 사용하여 오탐을 개선한다. G기관의 경우 Sql injection 구문에서 자주 사용되는 'select'로 이루어진 규칙의 경우 구분 기호나 특정 옵션을 사용하지 않을 경우 65% 이상 오탐이 높게 나타난다. 이를 아래 예처럼 Content와 Pcre의 패턴이 동일할 경우 탐지되는 유형으로 변경하여 주는 것이 효율적이다.

```
alert tcp $EXTERNAL_NET any -> $HOME_NET 21
(msg:"PROTOCOL-FTP ProFTPD username sql injection
attempt"; flow:to_server, established; content:"|25 27|";
fast_pattern:only; content:"USER"; pcre:"/USER\s*[^\x0d]
+\x25\x27/smi"; metadata:service ftp;reference:bugtraq,
33722;reference:cve,2009-0542;classtype:attempted-admin
;sid:16524; rev:5;)
```

다섯째, IP탐지의 경우 부분탐지(범위 및 개수를 제한)할 수 있게 조정한다. 예를 들어, 해당룰의 경우 탐지패턴에 대한 룰이 IP정보만 들어가 있어 오탐의 경우가 상존한다. 이 경우 IP 정보가 CIDR형식의 서브넷마스크(/20)인 경우 대상범위는 4094개의 IP가 탐지대상이 된다. 이는 대상범위가 너무 넓어 오탐이 높게 발생한다.

```
alert tcp $HOME_NET any -> 64.28.176.0/20 any
(msg:"MALWare detection ip_list"; flow:established
metadata:ruleset community;
classtype:misc-activity sid:1428; rev:8)
```

이를 다음 예처럼 $Black_IP_list로 개선하거나 Content를 혼합하여 탐지하는 것이 효율적이다. G기관의 경우 IP를 대역대로 설정하였을 때 탐지 이벤트 중 85% 이상이 오탐으로 판별되었다. 또한 IP 탐지의 경우 미리 어떤 범위까지 관제하겠다고 하는 관제 범위가 선행되어야 한다. 예를 들어 앞에서 언급한 TOR,VPN IP를 탐지할지에 대한 부분을 확정하여 Black_list화 하고 이런유형의 IP를 관리하여야 한다. 또, 단순히 Black_list만 그룹화 하지말고 유형을 세분화하여 유포지, 경유지, 수집서버 등으로 분리하여 관리하는 것이 효율적이며 추후 연관분석 및 행위분석 자료로도 활용될 수 있다.

```
alert tcp $HOME_NET any -> $Black_IP_list any
(msg:"MALWare detection ip_list"; flow:established
metadata:ruleset community;
classtype:misc-activity sid:1428; rev:8)
```

여섯째, 단어 및 문장의 경계를 지정하여 특정 단어, 특정 문장만 매칭될 수 있도록 구분 지어 탐지하는 것이 필요하다. 아래 예는 Content 단독으로 사용되어, 특정문구(CMD.EXE)만 매칭되면 탐지되므로 오탐이 많이 발생한다.

```
alert tcp $EXTERNAL_NET any -> $HTTP_SERVERS
$HTTP_PORTS (msg:"SERVER-IIS cmd.exe access";
flow:to_server,established; content:"cmd.exe";
nocase; metadata:service http; classtype:
web-application-attack; sid:23626; rev:5
```

G기관의 경우 문장의 경계를 지정하지 않았을 때 탐지 이벤트 중 65% 이상이 오탐으로 판별되었다. 아래 예처럼 http_client_body와 문자열 분리기호(\b)를 추가하여 수정하는 것이 효율적이다. 문자열 분리기호는 분리하고 싶은 문장 앞뒤로 해당 문자를 기입해 문장이나 단어를 분리하여 주는 기능을 한다. 아래와 같이 PCRE의 http_client_body 속성을 주면 단순 특정 문구 매칭되어서 탐지되는 것이 아닌 찾고자 하는 파일에서만 탐지가 됨으로 오탐이 많이 감소한다. 따라서, 이 방법은 특정 파일을 구분하여 탐지하고자 할 경우 효과적이다.

```
alert tcp $EXTERNAL_NET any -> $HTTP_SERVERS
$HTTP_PORTS (msg:"SERVER-IIS cmd.exe access";
flow:to_server,established; content:"cmd.exe";
nocase;http_client_body;pcre:"/\bcmd\x2eexe\b/Pi
s"; metadata:service http; classtype:web-
application- attack; sid:23626; rev:5;)
```

3.2.3 성능 규칙

- Non-Payload를 Payload보다 앞에 입력
- PCRE 사용할 경우 단독으로 사용하지 말고 Content와 혼합하여 사용.
- 매칭 가능성이 낮은 데이터 및 길이가 긴 데이터 앞(미리) 매칭 하고, 빠른 매칭 옵션을 사용하여 먼저 매칭되도록 조정 필요

첫째, Non-payload를 Payload앞에 검사하는 것이 성능을 향상시키는데 많은 영향을 준다. Table. 3.는 Non-payload관련 규칙 옵션으로 IP,ID와 TCP,ICMP부분으로 구분되며 이들은 특성에 맞게 세부적으로 나누어져 있다. 예를 들어, 패킷 사이즈 크기, IP Time to Live, bit 수, TCP Sequence 등 수치적 자료를 미리 Non-payload에서 탐지하여 필터링 해주기 때문에 뒤에 탐지되는 Payload 패킷들은 수치에 맞지 않은 부분을 탐지하지 않으므로 성능을 향상시킬 수 있다.

```
alert tcp $HOME_NET any -> $EXTERNAL_NET
6666:7000 (msg:"POLICY-SOCIAL IRC nick
change"; flow:to_server,established; content:"NICK
"; fast_pattern:only; metadata:ruleset community;
classtype:policy-violation; sid:542; rev:20;)
```

아래 예처럼 Content만으로 탐지되는 경우 dsize옵션(:<140)를 추가하여 해당 크기(140byte)보다 작은 경우에만 탐지되도록 수정하는 것이 성능 향상에 좋다. G기관의 경우 주로 빈번히 사용되는 특성 중 확연히 나타나는 특징은 IP의 ttl과 TCP

```
alert tcp $HOME_NET any -> $EXTERNAL_NET
6666:7000 (msg:"POLICY-SOCIAL IRC nick change";
flow:to_server,established; dsize:<140; content:"NICK
"; fast_pattern:only; metadata:ruleset community;
classtype:policy-violation; sid:542; rev:20;)
```

의 dsize, flow 사용시 성능이 15% 이상 향상되었다는 것이다.

두번째, PCRE(Perl Compatible Regular Expressions)[13]를 단독으로 사용할 경우 성능에 치명적인 영향을 준다. 이를 Content를 같이 혼합 사용하여 Content로 패킷을 미리 필터링 하여 걸러줌으로써 PCRE 성능에 향상을 줄 수 있다.

```
alert tcp $EXTERNAL_NET any -> $HOME_NET 21
(msg:"PROTOCOL-FTP no password";
flow:to_server,established; pcre:"/^PASS\s*\n/smi";
metadata:policy max-detect-ips drop, ruleset
community, service ftp; classtype:unknown; sid:489;
rev:19;)
```

이를 아래의 예처럼 수정하여 주면 성능이 많이 향상 된다. Fig. 5.는 이를 개선한 전후의 성능 비교치를 나타내고 있다. 해당 룰로 실험한 결과 약 45%정도 성능향상 효과가 있었다. G기관의 경우 다음과 같이 개선하여 성능을 측정하였을 경우 성능 향상에 효과적이며 약 30%이상 성능이 향상됨이 확인되었다. 또한 PCRE규칙의 경우 많이 사용할 경우 성능에 매우 안좋은 영향을 미치므로, 전체 룰의 몇프로 이상은 사용하지 않아야 한다는 규칙을 정해야 된다. G기관의 경우 전체룰의 10%이상은 사용하지 않아 룰의 성능을 향상하고 있으며, 이것은 해당 네트워크 대역마다 차이가 있을 수 있으므로 해당 네트워크 대역의 트래픽, 전체룰 개수, PCRE룰 개수 등을 혼합적으로 개산하여 전체 룰중에 몇프로까지는 PCRE룰로 사용해도 네트워과 서버성능에 안정적이라는 테스트를 통해 최대치를 정해 놓고 최대치 이하로 사용될 수 있도록 조정하는 것이 효과적이다.

```
alert tcp $EXTERNAL_NET any -> $HOME_NET 21
(msg:"PROTOCOL-FTP no password";
flow:to_server,established; content:"PASS";
fast_pattern:only; pcre:"/^PASS\s*\n/smi";
metadata:policy max-detect-ips drop, ruleset
community, service ftp; classtype:unknown;
sid:489; rev:19;)
```

셋째, 여러 가지 Content 중에 매칭 가능성이 낮은 데이터 및 길이가 긴 데이터를 앞(미리) 매칭

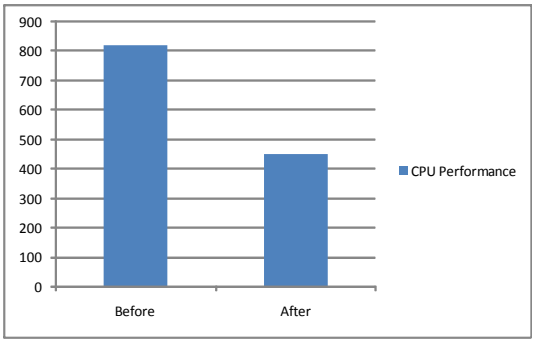

Fig. 5. Comparison of CPU usage/time before and after the PCRE Rule

Table 3. Non-Payload Detection Rule Options

Non-Payload Detection Rule Options	
IP, ID(8)	TCP. ICMP(12)
fragoffset	dsize
ttl	flags
tos	flow
id	flowbits
ipopts	seq
fragbits	ack
ip-proto	window
sameip	itype
	icode
	icmp_id
	icmp_seq
	rpc

한다. 예처럼 짧은 문장이 미리 나와 매칭이 됨으로써 성능에 안 좋은 영향을 준다. 왜냐하면, Snort의 경우 매칭데이터를 비교할 때 가장 처음 비교 대상 시그니처를 룰이 끝날 때 까지 반복해서 비교 한다. 그러므로 가장 처음에 있는 패킷이 가장 짧은 패킷이면 그 짧은 패킷을 계속 비교한다는 뜻이 되고 이는 성능에 안 좋은 결과를 초래한다.

```
alert tcp $HOME_NET 666 -> $EXTERNAL_NET any (msg:"MALWARE-BACKDOOR SatansBackdoor.2.0.Beta"; flow:to_client,established content:"Remote|3A| "; depth:11; nocase ; content:"You are connected to me.|0D 0A|Remote|3A| Ready for commands"; distance:0; nocase metadata:ruleset community; classtype:trojan-activity sid:118; rev:12;)
```

그러므로 매칭패킷 가장 앞에 긴 문장을 미리 써주어 가장 긴 문장을 젤 먼저 탐지하게 하거나 탐지패턴 간 fast_pattern 써주어 미리 탐지될 수 있게 조정할 수 있다. 또한 only 옵션을 활용하여 단 한 번만 비교되게 횟수를 조정할 수도 있다. 이를 활용 시 성능향상에 효율적이다. G기관의 경우 짧은 Content보다 길이가 긴 Content를 먼저 규칙에 매칭하는 것이 탐지시간별 탐지속도가 약 20% 성능 향상이 되는 것으로 확인되었다.

```
alert tcp $HOME_NET 666 -> $EXTERNAL_NET any ( msg:"MALWARE-BACKDOOR SatansBackdoor.2.0.Beta"; flow:to_client,established content:"You are connected to me.|0D 0A|Remote|3A| Ready for commands"; distance:0; nocase; fass_pattern:only; content:"Remote|3A| "; depth:11; nocasemetadata:ruleset community; classtype:trojan-activity sid:118; rev:12;)
```

IV. 실 험

4.1 실험 데이터 및 환경

실험 데이터 ISTS(The Information Security Talent Search)[14]는 2003년부터 뉴욕의 로체스터 연구소에서 주최하는 Security Practices and Research Student Association(SPARSA)의 대회이다. ISTS 2015 DATA SET은 ISTS 12번째 대회에서 사용된 DATASET의 이름이다. 2015년 대회는 2015년 5월 6일~ 8일까지 진행되었다. 많은 SPARSA 회원 및 RIT학생에서부터 미국 북동부, 중부 대서양과 중서부 지역의 수많은 대학을 대표하는 팀까지 다양하게 참가하였다. 해당 대회는 레드팀, 블루팀으로 나눠 진행이 되며 보통 블루팀이 레드팀의 공격을 방어하고 레드팀은 공격을 수행한다. 이 대회와 유사한 것으로는 CCDC(NATIONAL COLLEGIATE CYBER DEFENSE COMPETITION) 등이 있다. 해외 대학, 국제 유수기업, 연구소 등이 해당 대회들을 지원하여 우수한 인재를 선발하고자 노력한다. 본 논문에서는 해당 DATA SET을 1시간 단위로 샘플링하여 데이터를 테스트하였다.

국내 G기관의 특정 데이터를 활용하여 탐지결과를 일반화 하는 것이 아니라 11가지 규칙을 도출하는 수단으로 사용하였다. 그 규칙을 가지고 여러 가지 데이터 셋으로 해당 규칙을 검증하였고, 데이터 셋의 다양성을 위해서 실험을 다방면으로 수행하여 해당 규칙에 대한 신뢰성을 높이는데 주력하였다.

또한, ISTS 2015 DATA SET은 현재시점에서 가장 최신 공격 데이터 셋이며, 광범위한 네트워크에서 통신되는 다양한 환경을 모두 취합하여 데이터셋을 만들어 놓은것으로 이 데이터에 대한 활용성이 더욱 더 높다고 판단된다.

실험환경은 Intel(R) Xeon 2 Core 3.5GHz, 2GRAM, Windows7, PHP, Apache, MySQL, Snort2.9.7.0, Snort 2.9.7.0 snapshot, BASE 등 이와 같이 환경을 구축한 후 실행을 요일별 실행 및 결과를 확인하였다.

4.2 실험 결과

4.2.1 기본규칙

기본규칙의 경우는 실험환경에서 테스트하여 오탐을 낮추는 것이 아니라 관리적인 측면으로 얼마나 오판을 적게하는지에 대한 실험을 하는 것이다. 이는 전체 규칙을 나열하여 가독성 및 목적성이 얼마나 뚜렷하게 잘 나타나있는지를 식별할 수 있으며 실험환경에서 룰의 개선 전과 후의 오판율을 비교해 다음 실험을 테스트 한 것이다. 실험환경에서는 오판율감소를 오탐율 감소와 동일하다고 가정하였다.

첫 번째로 가독성 측면에서 확인해 본 결과이다. Fig. 6.에서 알 수 있듯이 가독성이 좋은 규칙이 오판율이 낮게 나타나며 약 8% 정도 오판율 감소 효과를 나타냄을 알 수 있다.

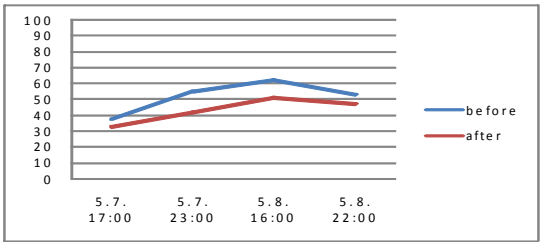

Fig. 6. Easy to read

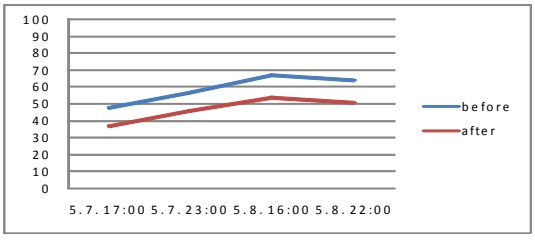

Fig. 7. Purposeful activity

기본규칙 두 번째 목적성이 뚜렷하게 수정한 결과 약 10% 정도 오판율 감소 효과를 나타냈다. Fig. 7.에서 알 수 있듯이 목적성이 뚜렷해지면 오판이 낮아진다. 이는 탐지하기 위한 목적이 분명해져 오판율이 낮아지는 것으로 해석된다.

4.2.2 탐지규칙

ISTS DATA SET 탐지규칙이 적용된 후 오탐률이 상당히 감소(Fig. 8.~ Fig. 9.)하였으며, 시간별 탐지 규칙에 대한 오탐율은 다음과 같다. 이 데이터는 2015년 5월 7일부터 8일까지 실험데이터를 기준으로 하였다.

위의 예는 탐지규칙을 적용하였을 경우 오탐율이 평균 평균 30~50%가 낮아짐이 증명되었다. 이후, 규칙별로 해당규칙에 대한 규칙결과를 나타내고자 한다.

첫째, Direction, Protocol, Port와 오탐율과의 상관관계를 나타낸다. Direction과 Protocol을 개선하였을 경우 오탐율이 소폭 감소하였으나 port가 ANY로 설정된 것을 특정 Port로 매칭시켰을 때 오탐율이 대폭 감소하는 것을 확인할 수 있었다. Protocol 및 Port는 5%내외 이였으나, Port의 경

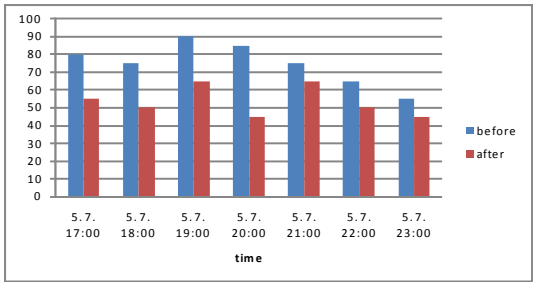

Fig. 8. March 7th of ISTS 2015

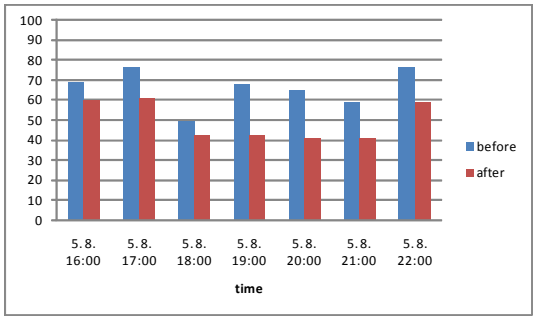

Fig. 9. March 8th of ISTS 2015

우 감소 폭이 40% 이상이고 IP의 경우 20% 이내로 감소하는 것을 알 수 있다.

둘째, Content의 길이가 15바이트 아래인 경우 오탐율이 대폭 상승하였으며, 평균 15바이트 이상인 경우 오탐율이 감소하는 것을 확인할 수 있다. 이는 Content의 길이에 따라 오탐율이 상승하는 것을 나타내며, Fig. 11.에서와 같이 평균 15byte를 기점으로 15바이트 이상인 경우에는 오탐율 38% 이내 이나, 15byte 이하인 경우 오탐율이 62% 이상 상승한다는 것을 확인할 수 있었다.

셋째, Fig. 12.은 Payload 옵션별 오탐율을 나타내고 있다. 그림에서 보이는 것 같이 규칙이 특정 페이로드 옵션 없이 단독으로 쓰이는 경우 오탐율이 70% 정도로 높지만 rawbytes 및 nocase등의 옵션은 오탐율 감소에 큰 차이가 없으나 depth, offset과 distance, with 등 콘텐츠의 범위를 나타내는 옵션과 같이 쓰일 경우 오탐율이 42% 이하로 떨어지는 것을 확인 할 수 있다.

넷째, Fig. 13.은 고정 패킷(fixed)과 비고정 패킷(non fixed)에 따른 오탐율은 나타내고 있다. 그림에서 보이는 것 처럼 고정패킷(SELECT, UNION, PUT, POST, GET) 등을 사용한 패킷과 그렇지 않은 패킷의 오탐율을 수치상으로 비교해 본 결과 약 30% 이상의 차이가 남을 알 수 있었다.

다섯째, Fig. 14.은 IP 그룹 매칭과 혼합 매칭에 따른 오탐율을 나타내고 있다. IP 그룹 단독으로 쓰였을때는 오탐율이 80%를 넘는다. 그러나 혼합매칭으로 바뀔 경우 그 절반인 40% 정도로 오탐이 낮아지는 것을 확인할 수 있다.

여섯째, 문자열 일부 매칭, 띄어쓰기 실수, 판단 오류에 따른 오탐에 대한 결과는 Fig. 15.에서처럼 문자열 일부 매칭과 띄어쓰기 실수는 문자열에 범위와 한계를 지정하지 않을 경우 아주 빈번하게 나타나며 규칙을 개선하면 확연하게 좋아지는 것을 알 수 있다. 그러나 판단 오류 같은 경우는 특정 경우를 제외하고는 구별이 쉽지가 않아 결과가 좋게 좋아지지는 않았다.

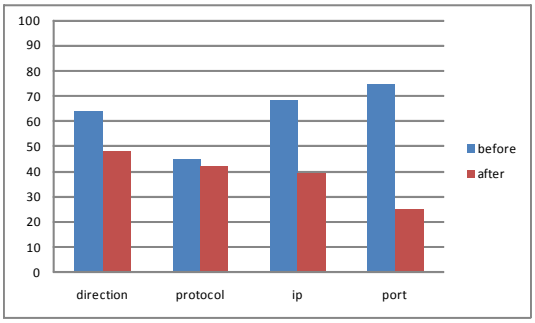

Fig. 10. Direction, Protocol, IP, Port detection

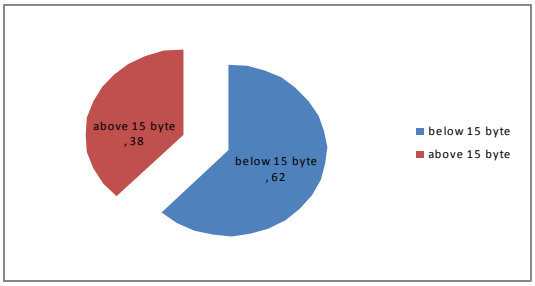

Fig. 11. Length of content

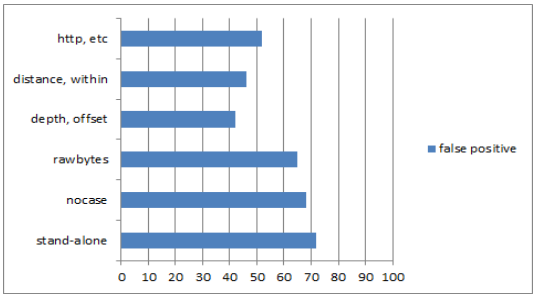

Fig. 12. Payload detection

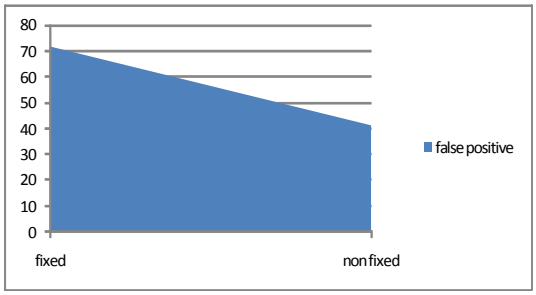

Fig. 13. Fixed, non fixed detection

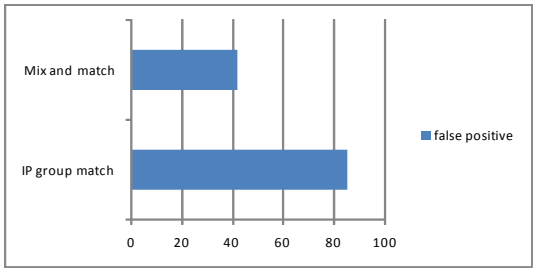

Fig 14. IP Partial matching

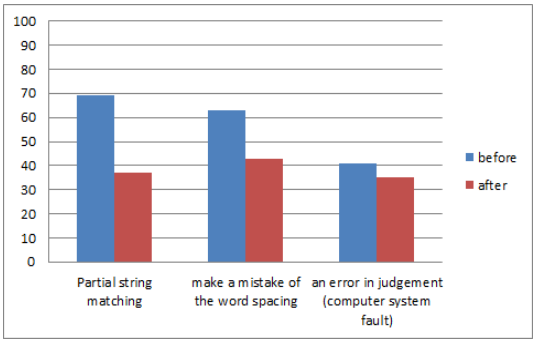

Fig 15. Mistake in word spacing

4.2.3 성능규칙

마지막으로 성능 규칙의 경우 해당 룰에 따른 성능수치를 룰 적용 전 과 후를 비교하는 것으로 테스트 하였으며 그 결과는 아래와 같다.

첫 번째 성능 규칙은 Non-payload를 Payload 앞에 입력하여 수치적으로 미리 구분될 수 있는 것은 미리 구분하여 검사하는 것이다. 이렇게 걸러진 패킷들이 성능에 향상을 준다. Fig. 16.에서 규칙 적용 전과 후 시간이 2/3 정도만에 처리됨을 확인할 수 있다.

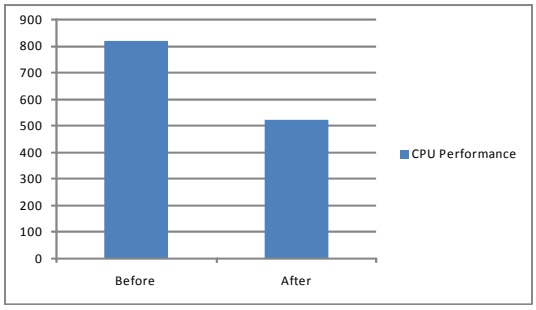

Fig. 16. Non-payload detection

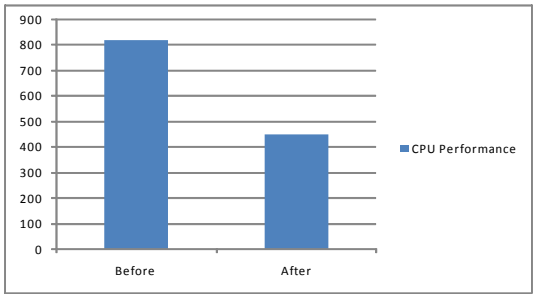

Fig. 17. Merge of PCRE Rule

두 번째 성능 규칙은 PCRE(PERL 기반 정규표현식)을 이용하여 PCRE 단독으로 사용된 규칙을 Content 규칙과 혼합하여 사용하였을 경우 얼마나 성능에 영향을 미치는지를 테스해본 것으로 적용 전과 후를 비교했을 시 Fig. 17.에서 알 수 있듯이 기존 탐지시점에서 1/2정도의 시간 정도만 걸릴 정도로 성능향상에 많은 도움을 주는 것으로 확인 되었다.

세 번째 성능 규칙은 성능 규칙중 패킷의 길이에 따른 성능을 측정한 것(Fig. 18.)으로 패킷의 길이가 짧은 것이 앞에 있으면 성능에 악영향을 미치므로 그 순서를 조정하여 맨 뒤로 조정하여 가장 길이가 긴 패킷을 맨 먼저 탐지하도록 비교 했을시 15% 성능향상이 확인되었다.

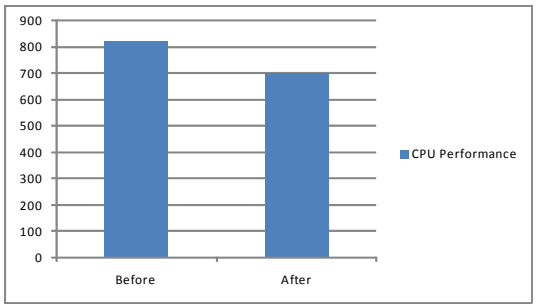

Fig. 18. Location of content

V. 결 론

기본규칙은 규칙들의 활용측면으로 룰의 생명주기와 룰의 가독성 과 목적성을 잘 고려하여 탐지규칙을 작성해야 된다. 아래는 그 결과는 Table. 4.을 나타낸다. 다소 오탐률이 크게 향상 되지는 않았으나, 룰의 관리적인 측면도 오탐율에 영향을 미치는지가 객관적으로 제시되었고, 룰 규칙을 생성할 때 단순히 탐지의 목적만이 아닌 관리의 목적도 같이 생각해 주어야 된다는 좋은 예시가 되었다.

탐지규칙은 dataset(ISTS 2015)을 기초로 오탐(false positive)와 관련된 규칙을 효율적으로 개선

Table. 4. Improvement of Basic Rules

Date	Data	False Positive (Before)	False Positive (After)
5.7	1,947	46%	**37%**
5.8	1,795	57.5%	**43%**
Sum	3,742	51.75%	**43.25**

해 보았다. 본 규칙으로 탐지될 시 오탐이 30~50% 이상 줄어들어 효율적인 탐지가 됨을 Table. 5.에서 확인하였다.

성능 규칙의 경우 미리 우선적으로 탐지 및 검토될 수 있는 부분은 우선 처리하여 적용될 경우 평균 33% 성능이 향상되었으며 그 결과 Table. 6.와 같다.

위에서 논한 규칙 들을 적용 시 상당 부분 효율적으로 대응할 것으로 판단되지만 더 중요한 것은 조직이나 단체에서 보안 정책 미리 수립하여 사원에서 CEO까지 보안정책을 준수하는 것이 필요하다. 또한, 중요자산에 대한 객관적 위협 감수 지표를 도출하여 허용범위 내에서 위협을 감수할 수 있게 감당하는 것 또한 중요 하다. 이는 규칙으로 작성될 수 없으며 조직의 특성, 기능, 비전을 활용하여 지속적으로 관리되어야 한다. 향후 연구에서는, 알려지지 않은 공격을 탐지하는 행위기반기법을 활용해 오탐을 줄이는 방법에 대해서 연구해보고자 한다.

Table. 5. Improvement of Detection Rules

Date	Data	False Positive (Before)	False Positive (After)
5.7	1,947	75%	**53%**
5.8	1,795	66%	**49%**
Sum	3,742	70.5%	**51.5%**

Table. 6. Improvement of Performance Rules

Rule	Before (microsec)	After (microsec)	Avg	Sum (microsec)
Fig.16	193,762	123,846	63%	131,630
Fig.17	193,762	106,389	54%	
Fig.18	193,762	164,656	85%	

References

〔1〕 Law.go.kr〔online〕 http://law.go.kr/admRulInfoP.do?admRulSeq=2000000093285

〔2〕 Hyundo Lee and Sang Jin Lee, "A Study on development of evaluation indicators on the Managed Security Service(MSS)" Journal of the Korean Institute of Information Security and Cryptology, pp.1133-1143, v.22, 2012

〔3〕 Gartner〔online〕 http://web.archive.org/web/20031204111139/http://www.gartner.com/5_about/press_releases/pr11june2003c.jsp

〔4〕 Myeong-Hun Gang, "The completion of the IDS and Security control seen in Big data Analytics," wow Books, . pp.80-102, 2013

〔5〕 Snort〔online〕 http://www.Snort.org

〔6〕 Asieh Mokarian, Ahmad Faraahi, Arash Ghorbannia Delavar, Payame Noor University, Tehran, IRAN "False Positives Reduction Techniques in Intrusion Detection Systems-A Review," IJCSNS International Journal of Computer Science and Network Security, pp.128-134, VOL.13 No.10, October 2013

〔7〕 N. B. Anuar, H. Sallehudin, A. Gani, O. Zakari, "Identifying false alarm for network intrusion detection system using hybrid data mining and decision tree," Malaysian journal of Computer Science, pp.101-115, Vol. 21(2), 2008.

〔8〕 N.A. Bakar, B. Belaton, A. Samsudin, "false positives reducetion via intrusion alert quality framework," 13th IEEE International Conference on Communication, pp16-18. Nov. 2005.

〔9〕 G.C.Tjhai, M.Papadaki, S.M.Furnell, N.L.Clarke, Investigating the problem of IDS false alarms:An experimental study using Snort, Proceedings of IFIP TC 11 23 rd International Information Security Conference, pp. 253-268, Sep. 2008.

〔10〕 C.A.Obi and M.Papadaki "Guidelines/Recommendations on Best Practices in Fine Tuning IDS Alarms," Advances in Communications, Computing, Networks and Security pp.107-114, Vol 6, 2008

〔11〕 Jiyoung Han, Inbok Lee, Junghee Han

"Accelerating PCRE Performance of Signature-Based IDS" Journal of Information Science: Systems and Theory pp.53-60, Apr No. 2, 2013
〔12〕 Elias Raftopoulos and Xenofontas Dimitropoulos, "A quality metric for IDS signatures: in the wild the size matters," EURASIP Journal on Information Security, Dec. 2013
〔13〕 PCRE 〔online〕 http://www.pcre.org
〔14〕 ISTS 〔online〕 http://ists.sparsa.org/

〈저자소개〉

한 동 희 (Donghee Han) 학생회원
2009년 2월 : 대구 가톨릭대학교 컴퓨터공학과 졸업
2012년 2월~현재 : 고려대학교 정보보호대학원 석사 과정
〈관심분야〉 보안관제, 포렌식, 빅데이터

이 상 진 (Sangjin Lee) 종신회원
1994년 8월: 고려대학교 수학과 박사
2006년 2월~2011년 12월: 암호연구회 위원장
2008년 3월~현재: 고려대학교 정보보호연구원 디지털포렌식센터장
2006년 1월~현재: 한국디지털포렌식학회 이사
現 고려대학교 정보보호대학원 교수
〈관심분야〉 암호이론, 디지털포렌식