HoneyPot on AWS

AWS 클라우드 서비스 기반 허니팟 구현

91714272
임민성

Contents

1. 주제 선정 배경

증가하는 웹 기반 사이버위협, 대처 조직은 30% 미만?

좋아요 8개 | 입력 : 2022-04-26 15:23

#클라우드 #보안 #멘로시큐리티 #보안 위협 대응 현황

멘로시큐리티, '보안 위협 대응 현황' 보고서 발표

웹 기반 사이버 위협에 대처할 준비가 되어 있는 조직은 10개 중 3개 미만

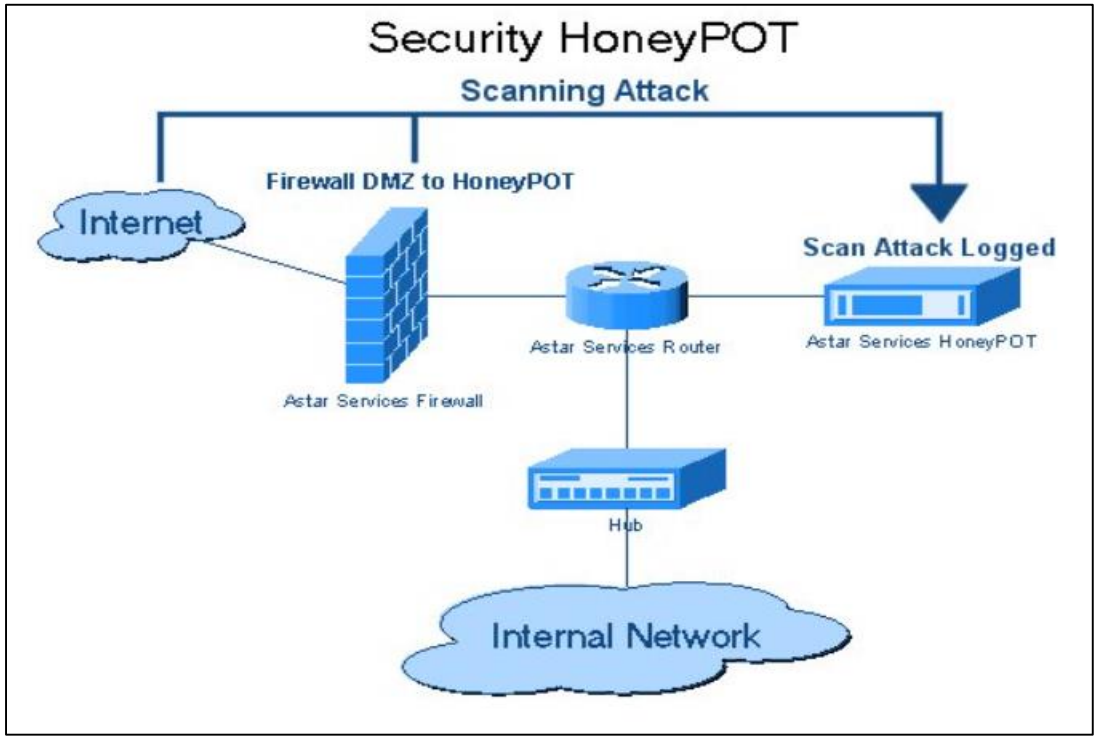
응답자의 62%가 지난 12개월 동안 웹 브라우저 기반 공격 받아

멘로시큐리티의 마크 건트립(Mark Guntrip) 사이버 보안 전략 담당 이사는 "위협 행위자들은 전통적인 보안 방어체계와 지난 10년 동안 보안기능이 실제로 바뀌지 않았다는 사실을 활용하고 있다"면서, "공격자들은 웹을 통해 HEAT 공격을 진행한다. 지난 2021년에 발생한 솔라윈즈(SolarWind) 공급망 공격의 배후에 있는 중국의 해킹 단체인 노벨륨(Nobelium)은 HTML 스머글링 기술을 사용해 악성 프로그램 설치 및 랜섬웨어 공격을 진행했다"라고 말했다. 아울러 "이러한 공격 기술이 성공하는 사례가 증가하며, 모든 기업들에게 엄청난 피해로 이어질 수 있다"라고 말했다.

- 보안 분야는 최신 트렌드에 가장 민감해야 할 분야 중 하나

- 99개의 과거에 발견된 취약점을 조치해도, 1개의 새로운 취약점에 무력해지는 것이 보안

1. 주제 선정 배경



- 최신 공격 기법을 탐지하고, 로깅할 수 있는 허니팟 개념의 창설

1. 주제 선정 배경

허니팟, 중요한 기술이지만 그 만큼 많은 단점 존재해

허니팟은 사이버 공격 탐지, 예방과 대응에 중요한 기술인 것은 분명하나 단점도 존재한다. 첫째로 구축의 어려움이다. 허니넷 같이 일정 규모의 허니팟 집합을 구성하기까지 많은 시간과 자원의 소모가 필요하다. 또한, 실제 시스템과 동일한 환경을 구축하기 위해서는 비용도 만만치 않게 들어간다. 구축 규모에 따라 많은 네트워크 장비와 서버 장비가 필요하며 고액의 라이선스 비용도 지불해야 하는 상황도 감수해야 한다.

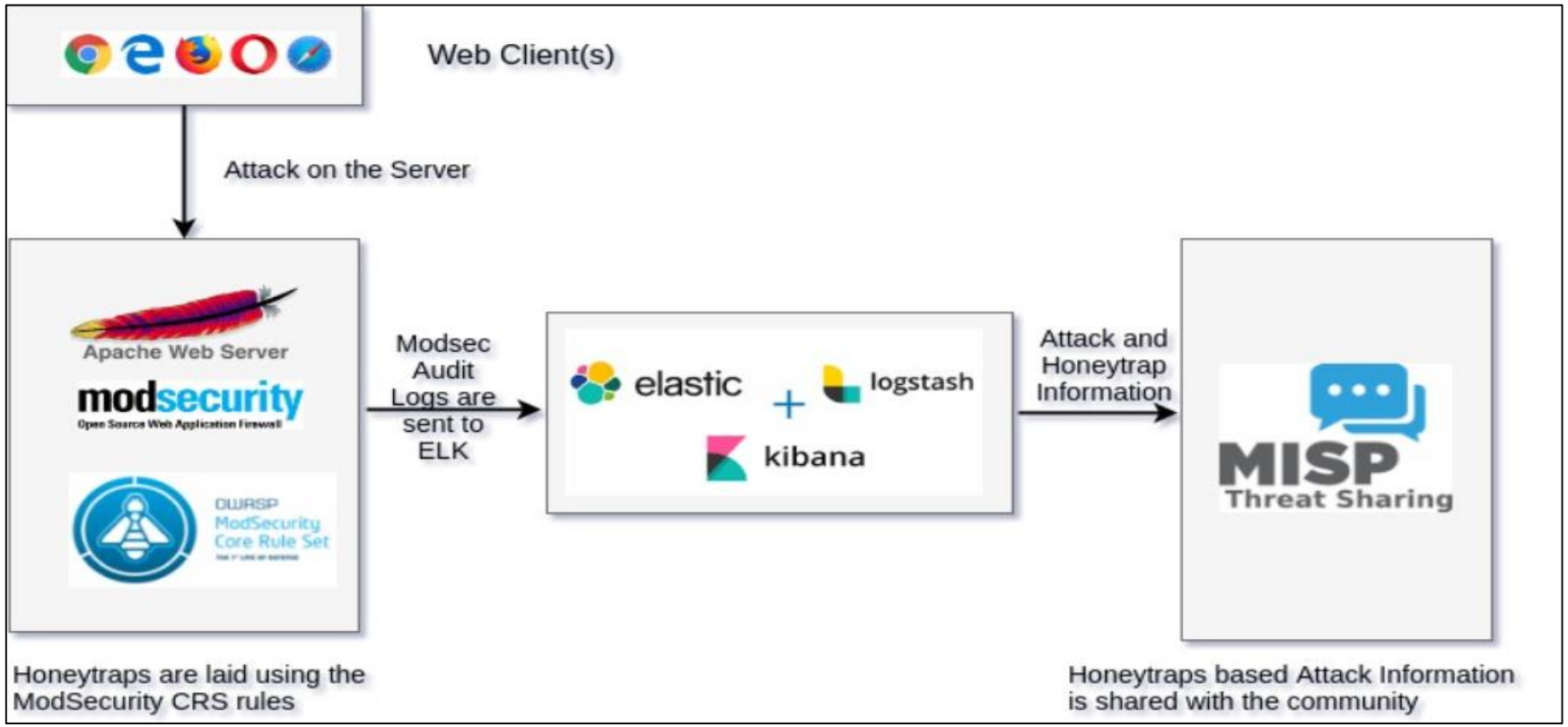
두 번째로 허니팟은 공격자와 깊은 상호작용이 어렵다는 점이다. 최근의 악성코드는 허니팟을 탐지하기도 하고, 허니팟 자체를 속이기 위해 행동을 변경하는 다형성 악성코드도 존재한다. 하지만, 허니팟은 공격자를 유인한 후에도 그들이 계속해서 실제 시스템에 있다고 착각하게 만드는 능력이 부족해 최근의 악성코드 대응에는 미흡한 부분이 있다.

마지막으로 허니팟 자체가 공격자에게 이용당해 다른 내부 시스템들까지 위험에 처할 수 있는 가능성도 존재한다. 허니팟 시스템이 장악되면 이 시스템이 거점이 돼 다른 시스템으로 공격자가 이동할 수 있는 빌미를 제공하게 된다. 이러한 단점들로 인해 허니팟은 악성코드를 분석하고 연구하기 위한 수동적 용도로 많이 사용됐다.

- 트렌드에 민감하기 위해서 '허니팟'이라는 개념이 도입. 그러나 한계점이 명확

- 구성에 대한 리소스 낭비. 내부 시스템 피해에 대한 우려 등

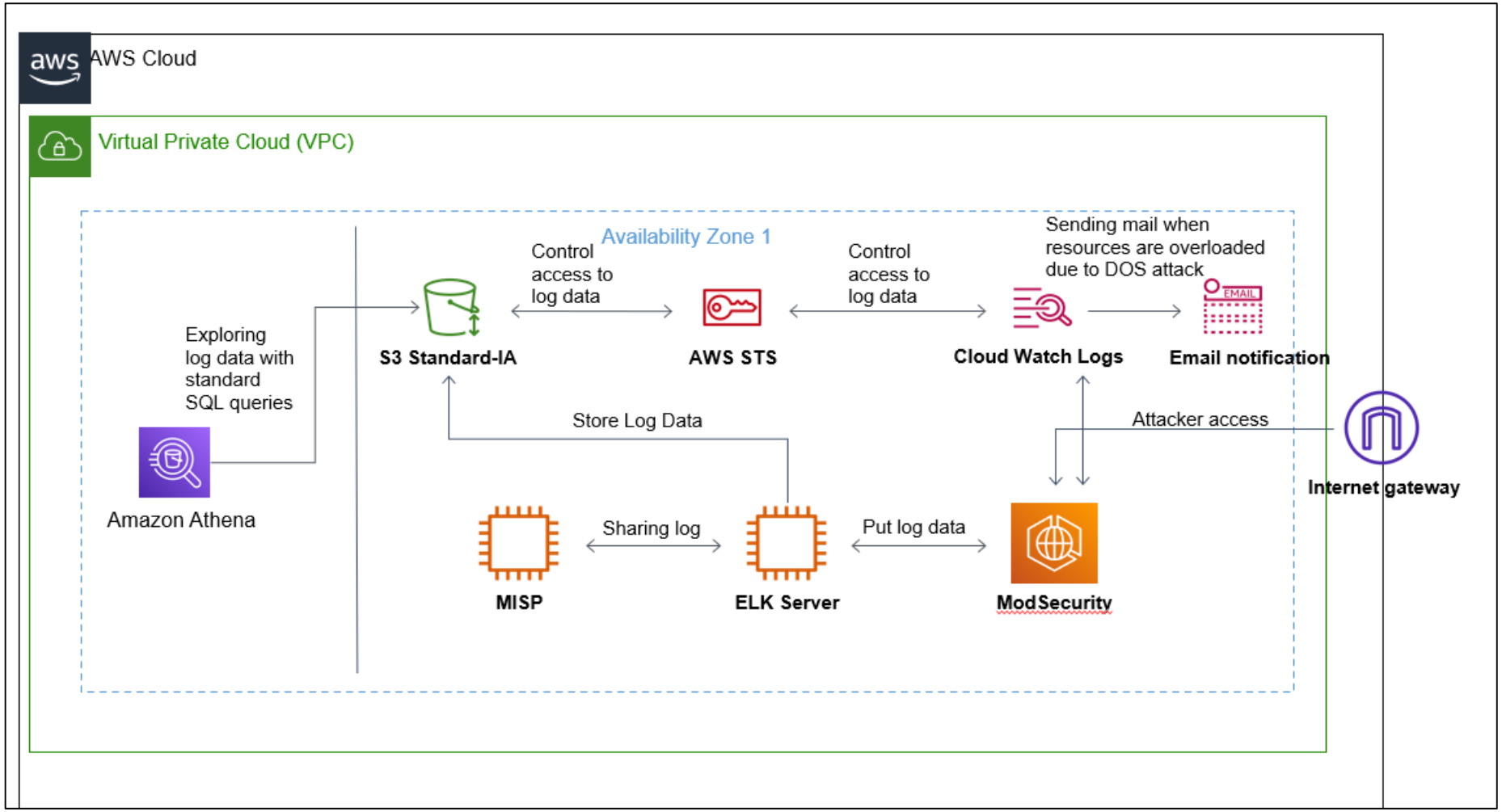
1. 허니팟 구성



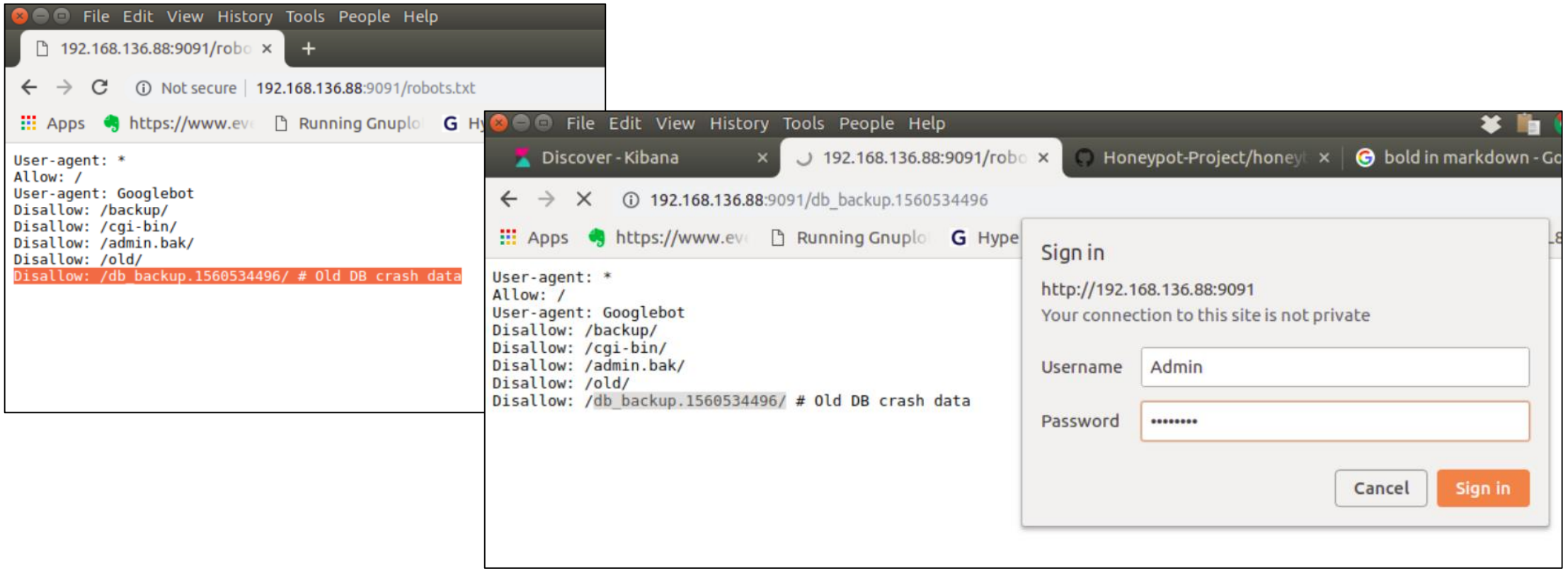
- OWASP HoneyTrap 오픈 소스 활용

- ModSecurity(방화벽), ELK Stack(로그 분석), MISP(페이로드 공유)

2. AWS 구성


AWS Cloud
Virtual Private Cloud (VPC)
Availability Zone 1
Control access to log data
Control access to log data
Sending mail when resources are overloaded due to DOS attack
Exploring log data with standard SQL queries
S3 Standard-IA
AWS STS
Cloud Watch Logs
Email notification
Store Log Data
Attacker access
Amazon Athena
Internet gateway
Sharing log
Put log data
MISP
ELK Server
ModSecurity

2. AWS 구성



- robots.txt 파일 내 DB백업파일 경로 노출

- 해당 경로에 대한 접근제어 BUT Admin/Admin 으로 기본 계정 사용 중

2. AWS 구성

```
▼<body>
    <h3>Add entry</h3>
    <p> Add another Article</p>
...  <!-- DEBUG - the source code for the old login page is login.php.bak --> == $0
  ▶<form action="login.html" method="post">…</form>
  </body>
</html>
```

{"transaction":{"time":"17/Jun/2019:14:46:40 +0000","transaction_id":"XQen0G-vetL1qCSlAhmDmgAAAA
A","remote_address":"192.168.112.210","remote_port":51984,"local_address":"172.17.0.3","local_por
t":80},"request":{"request_line":"GET /login.php.bak HTTP/1.1","headers":{"Host":"192.168.136.88:9
091","Connection":"keep-alive","Upgrade-Insecure-Requests":"1","User-Agent":"Mozilla/5.0 (X11; Lin
ux x86_64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/71.0.3578.98 Safari/537.36","DNT":"1","Ac
cept":"text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,image/apng,*/*;q=0.8","Acc
ept-Encoding":"gzip, deflate","Accept-Language":"en-GB,en-US;q=0.9,en;q=0.8","Cookie":"(null)=Admi
n:0"}},"response":{"protocol":"HTTP/1.1","status":404,"headers":{"Set-Cookie":"(null)=Admin:0","Co
ntent-Length":"211","Keep-Alive":"timeout=5, max=100","Connection":"Keep-Alive","Content-Type":"te
xt/html; charset=iso-8859-1"},"body":"<!DOCTYPE HTML PUBLIC \"-//IETF//DTD HTML 2.0//EN\">\n<html>
<head>\n<title>404 Not Found</title>\n</head><body>\n<h1>Not Found</h1>\n The requested URL /log
in.php.bak was not found on this server. \n</body></html>\n"},"audit_data":{"messages":["Warnin
g. String match \"/login.php.bak\" at REQUEST_FILENAME. [file \"/etc/httpd/modsecurity.d/modsecuri
ty.conf\"] [line \"274\"] [id \"999008\"] [msg \"HoneyTrap Alert: Fake HTML Comment Data Use
d.\"]","Warning. Pattern match \"^[\\\\d.:]+$\" at REQUEST_HEADERS:Host. [file \"/etc/httpd/modsec
urity.d/owasp-crs/rules/REQUEST-920-PROTOCOL-ENFORCEMENT.conf\"] [line \"810\"] [id \"920350\"] [r
ev \"2\"] [msg \"Host header is a numeric IP address\"] [data \"192.168.136.88:9091\"] [severity
 \"WARNING\"] [ver \"OWASP_CRS/3.0.0\"] [maturity \"9\"] [accuracy \"9\"] [tag \"application-multi
\"] [tag \"language-multi\"] [tag \"platform-multi\"] [tag \"attack-protocol\"] [tag \"OWASP_CRS/P
ROTOCOL_VIOLATION/IP_HOST\"] [tag \"WASCTC/WASC-21\"] [tag \"OWASP_TOP_10/A7\"] [tag \"PCI/6.5.10
\"]"],"error_messages":["[file \"apache2_util.c\"] [line 273] [level 3] [client %s] ModSecurity: %
s%s [uri \"%s\"]%s","[file \"apache2_util.c\"] [line 273] [level 3] [client %s] ModSecurity: %s%s
 [uri \"%s\"]%s"],"stopwatch":{"p1":964,"p2":1979,"p3":61,"p4":415,"p5":156,"sr":104,"sw":99,"l":
0,"gc":0},"response_body_dechunked":true,"producer":["ModSecurity for Apache/2.9.1 (http://www.mod
security.org/)","OWASP_CRS/3.0.2"],"server":"Apache/2.4.27 (Fedora)","engine_mode":"ENABLED"}}

- 공격자가 접근할만한 5가지의 미끼 준비

- 공격자가 해당 공격 벡터에 접근하면 태그와 함께 로깅

1. 기대 효과

기존 허니팟의 두 가지 문제점에 대한 해결

과도한 리소스 비용과 서버 관리

- S3 : 탄력적인 스토리지 서비스로 저장한 만큼의 비용을 지불하므로 로그 저장 공간이 부족해 스토리지 추가 증설이 필요 없음

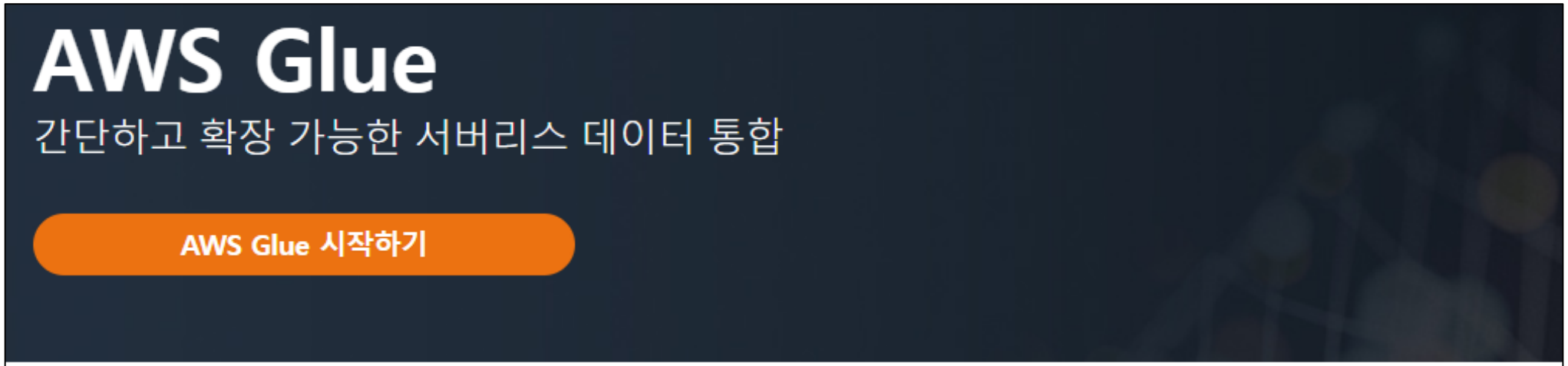
- ECS : Elastic Container Service 로 서버리스 서비스 중 하나. 컨테이너를 올려 실행시켜주고 해당 컨테이너에 대한 트래픽 비용만 발생하므로 비용효율적

- CloudWatch Alarm : CloudWatch를 사용해 CPU 사용량을 추적하여 Ddos 공격 등 비용이 대량 발생할 수 있을 때 AWS SNS 와 연동하여 바로 알람이 울리게 설정

허니팟의 취약성으로 인해 내부 시스템에 대한 피해 가능성

- 클라우드 서비스를 사용 시 허니팟은 클라우드 서비스를 제공하는 기업의 온프레미스 환경에 존재하기 때문에 사용자의 온프레미스 네트워크와는 무관함. 따라서 온프레미스에 허니팟을 설치하는 것 보다 훨씬 안전함

2. 향후 계획



AWS Glue는 분석, 기계 학습 및 애플리케이션 개발을 위해 데이터를 쉽게 탐색, 준비, 그리고 조합할 수 있도록 지원하는 서버리스 데이터 통합 서비스입니다. AWS Glue에서는 데이터 통합에 필요한 모든 기능을 제공하므로, 몇 개월이 아니라 몇 분 안에 데이터 분석을 시작하고 해당 내용을 활용할 수 있습니다.

데이터 통합은 분석, 기계 학습 및 애플리케이션 개발을 위해 데이터를 준비하고 결합하는 프로세스입니다. 이 작업은 다양한 소스에서 데이터 검색 및 추출, 데이터 강화, 정리, 정규화 및 결합, 데이터베이스, 데이터 웨어하우스 및 데이터 호수에 데이터 로드 및 구성 등의 여러 작업을 포함합니다. 이러한 작업은 종종 각자 다른 제품을 사용하는 다른 유형의 사용자가 취급합니다.

2. 향후 계획

a. AWS Athena 및 AWS Glue 서비스를 활용한 최근 공격 동향 보고서 자동화 시스템

b. 로그 백업 데이터 서비스 구축

c. 더 많은 공격자를 속이기 위해 웹서버 Docker Image 내 서버 파일 수정 후 DB 연결
->
DB 샘플 데이터 삽입 후 공격자의 DB 공격 페이로드 확인

감사합니다