

네트워크 보안

2023. 07. 25 하계 워크샵 4주차

한림대학교 정보과학대학 씨애랑

(HALLYM SECURITY TEAM SHIELD)



목차

- 대역폭 공격 – ICMP Flooding
- 대역폭 공격 – DNS 반사 공격
- 대역폭 공격 – SSDP 반사 공격
- 보안 시각화
- 로그분석 툴



대역폭 공격 – ICMP FLOODING

ICMP Flooding

■ ICMP Flooding 대한 이해

- ICMP (Internet Control Message Protocol)은 인터넷 환경에서 오류에 관한 처리를 지원하는 역할 수행
- 라우터 및 네트워크 장비에서 패킷을 전송할 때, 해당 패킷이 목적지 호스트까지 도달하지 못하거나 목적지 호스트가 정상적인 동작이 안되는 경우, 에러 메시지를 응답하여 오류 상황을 알려주는 역할 수행
 - 흔히 ping 명령어를 통한 패킷 전송 시 ICMP 프로토콜을 사용함
 - ICMP Flooding은 UDP 프로토콜 대신 ICMP 프로토콜을 사용한 다는 점을 제외하면 나머지 형태가 거의 같음.

Type 8bits	Code 8bits	Checksum 16bits
Identifier 16bits		Sequence number 16bits
Data		

Type	Code	Message
0	0	Echo reply
3	0	Destination network unreachable
3	1	Destination host unreachable
3	2	Destination protocol unreachable
3	3	Destination port unreachable
3	4	Fragmentation required, and DF flag set
3	6	Destination network unknown
3	7	Destination host unknown
4	0	Source quench (congestion control)
8	0	Echo request (used to ping)

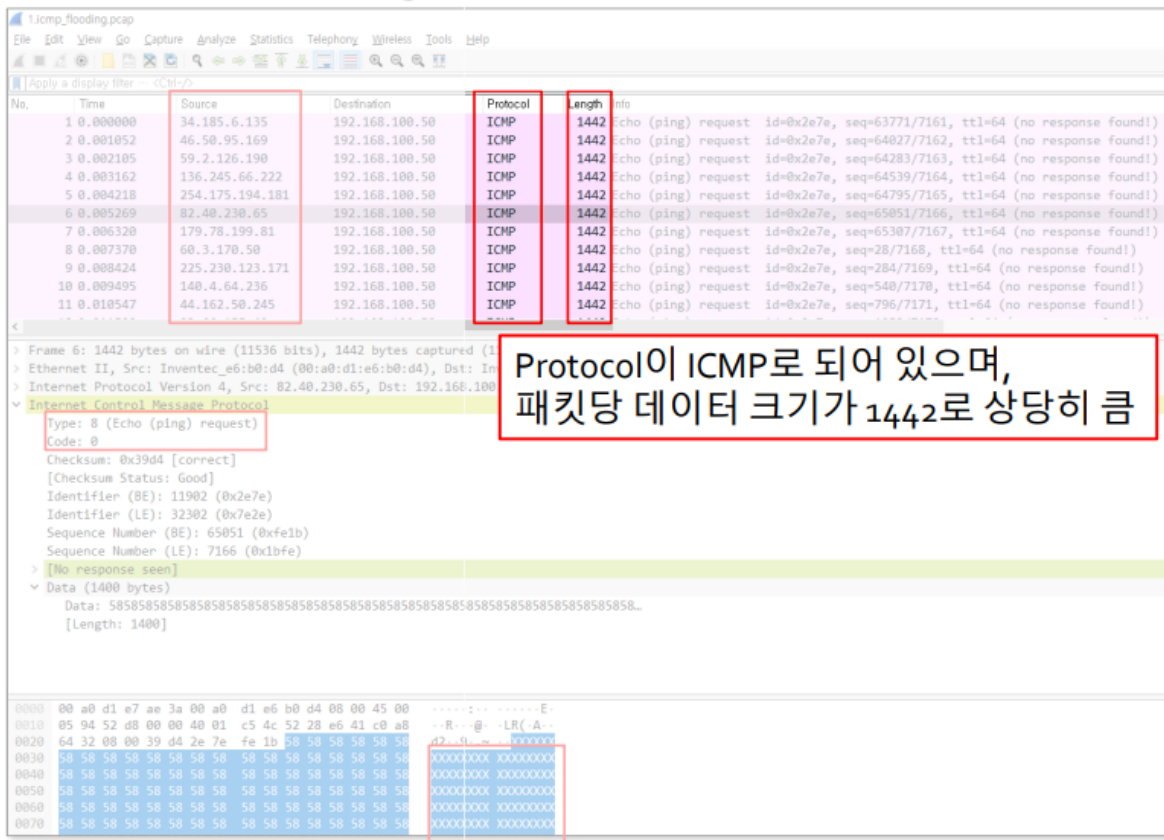


■ ICMP Flooding 트래픽



ICMP Flooding

■ ICMP Flooding 트래픽



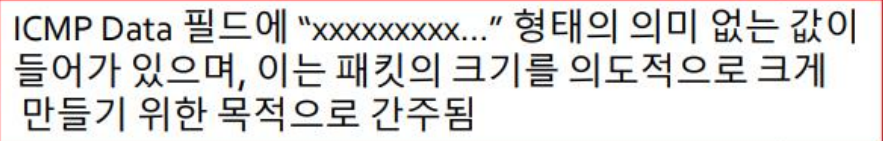
Protocol이 ICMP로 되어 있으며, 패킷당 데이터 크기가 1442로 상당히 큼



■ ICMP Flooding 트래픽

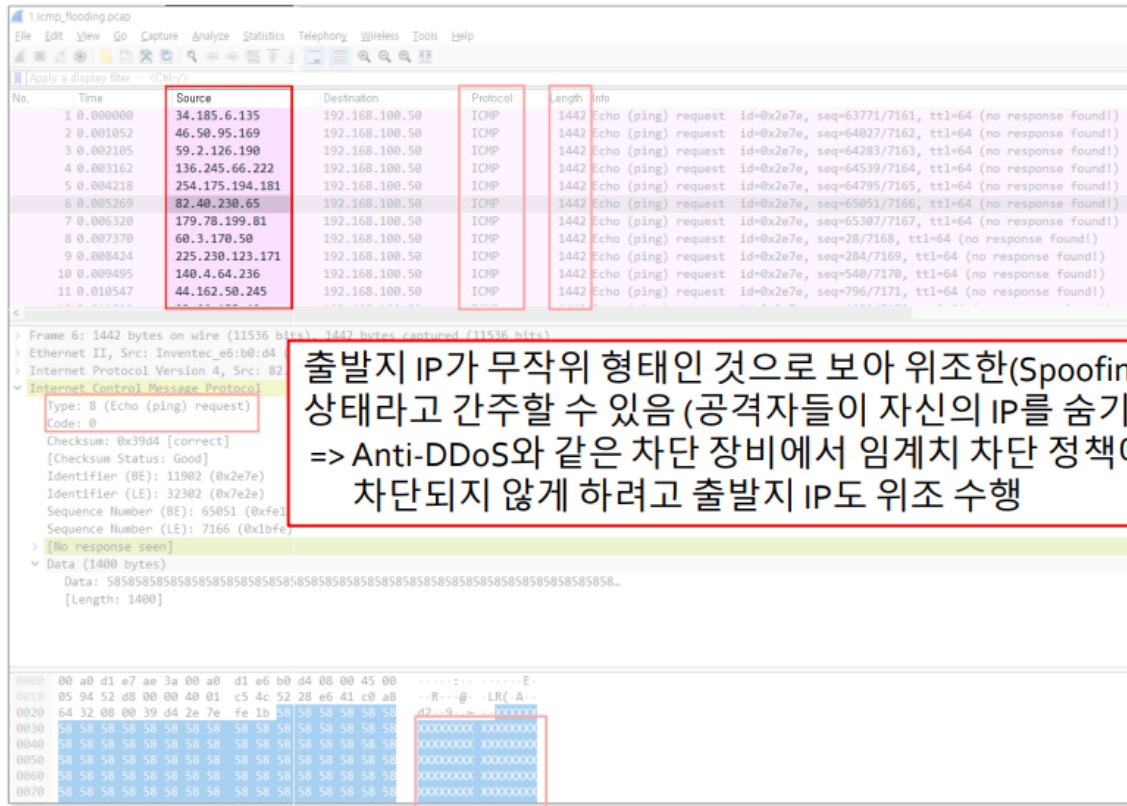
[illegible]

■ ICMP Flooding 트래픽



ICMP Flooding

■ ICMP Flooding 트래픽



출발지 IP가 무작위 형태인 것으로 보아 위조한(Spoofing) 상태라고 간주할 수 있음 (공격자들이 자신의 IP를 숨기기 위함)
=> Anti-DDoS와 같은 차단 장비에서 임계치 차단 정책에 차단되지 않게 하려고 출발지 IP도 위조 수행



ICMP Flooding

■ ICMP Flooding 대응 방안

- 충분한 네트워크 대역폭 확보
- 위조된 IP 차단
- 출발지 IP별 임계치 기반 차단
- Fragmentation 패킷 차단
- 서버 대역폭 및 가용량 확대
- Anycast를 이용한 대응
- Ingress 필터링과 Egress 필터링

- 미사용 프로토콜 필터링
 - ICMP는 헬스 체크의 목적으로만 사용하기 때문에 가능

- 패킷 크기 기반 차단
 - ICMP echo request는 64~80 바이트 정도지만, ICMP echo reply의 경우에는 약 300 바이트 전후의 큰 크기로 수신될 수 있으므로, 네트워크에 따라 적절한 임계치를 적용이 필요

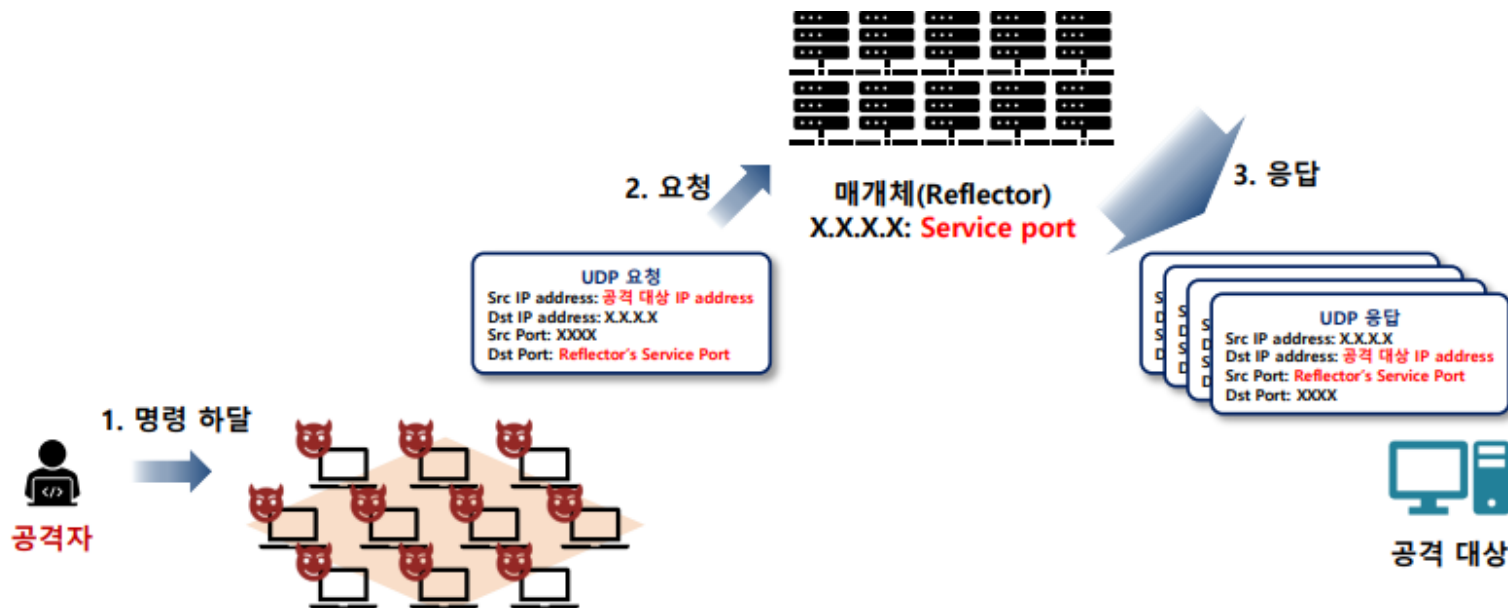


대역폭 공격 – DNS 반사 공격

Reflection Attack

■ Reflection Attack

- 인터넷에서 UDP 서비스를 사용하는 서버들을 매개체(reflector/amplifier)로 이용하여 DDoS를 발생시키는 공격
 - 좀비 PC가 아닌 실제 인터넷상에서 정상 운영중인 UDP 서비스 서버들을 공격에 악용함



Reflection Attack

■ Reflection Attack

- 요청 패킷과 응답 패킷의 형상이 반사되는 형태를 보이며 증폭되는 형태를 보이기도 하기 때문에 증폭 공격(Amplification Attack)으로도 불림
 - 1. 공격자가 C&C를 이용하여 좀비 PC들에게 공격 명령 전송
 - 2. 수많은 좀비 PC들이 출발지 IP address를 공격 대상의 IP address로 위조하고,
 - 특정 매개체가 되는 서버들로 UDP 요청 패킷들을 전송함.
 - 3. 요청 패킷을 수신한 매개체 서버들이 공격 대상이 되는 위조된 IP address로 다량의 응답 패킷들을 전송함
- 대용량의 응답 패킷으로 인해 네트워크 대역폭 고갈
- 해당 공격의 피해자는 크게 볼 때 2곳
 - 매개체 서버, 공격 대상 서버
 - 공격 대상 서버는 수많은 매개체 서버들로부터 대용량의 응답 패킷들로 인해 대역폭 고갈과 정상 서비스 운영이 불가
 - 매개체 서버는 비정상적인 요청으로 인한 상당량의 응답 패킷 생성으로 인해 아웃바운드 트래픽 고갈과 함께 서버의 부하 증가로 인한 피해 발생
 - 반사 공격에 대한 대응 방안은 공격 대상 입장과 매개체 입장 두 가지로 수립되어야 함.



Reflection Attack

■ 반사 공격 특징 - 공격 대상 입장

- 대부분 UDP 프로토콜로 구성됨
- 공격 대상으로 전송되는 대규모 패킷들은 응답 패킷임
- 출발지 포트는 특정 UDP 서비스에서 사용하는 포트 번호
- ICMP destination unreachable 패킷이 수신 될 수 있음
 - 매개체로 사용한 UDP 서버가 동작하지 않거나 포트가 닫혔을 경우, ICMP로 응답함
- 출발지 포트 정보를 통해 어떤 UDP 서비스 서버가 매개체로 사용된 것인지 알 수 있으며, 공격 유형이 정의됨
 - Ex. 출발지 포트 53 -> DNS 서버 매개체 -> DNS 반사공격
- Src IP address는 UDP가 사용되는 서버의 IP address를 의미
 - Ex. Src port 53 -> DNS 매개체 -> 공격 출발지 IP로 DNS 질의 테스트 -> 정상 응답 확인 -> 실제 DNS서버로 인지



Reflection Attack

■ 반사 공격 특징 - 매개체 입장

- 운영중인 UDP 서비스로 많은 요청 패킷이 발생함
- 출발지 IP address는 1~2개로 적음 (공격 대상의 IP address 로 위조된 IP address)
- 목적지 포트는 매개체 서버에서 운영중인 UDP 서비스의 포트 번호

■ 반사 공격 종류

- DNS 반사 공격
- SSDP 반사 공격
- SNMP 반사 공격
- NTP 반사 공격
- 기타 반사 공격



Reflection Attack

■ Reflection Attack – DNS

- DNS는 응답의 크기가 512 bytes를 초과하면 TCP로 전환되는 특징이 있음
 - DNS 질의 명령어의 요청에 대한 응답 형태
- ** TXT, ANY는 응답 값에 많은 문자열을 포함하고 있으므로 응답 크기가 큰 편에 포함됨

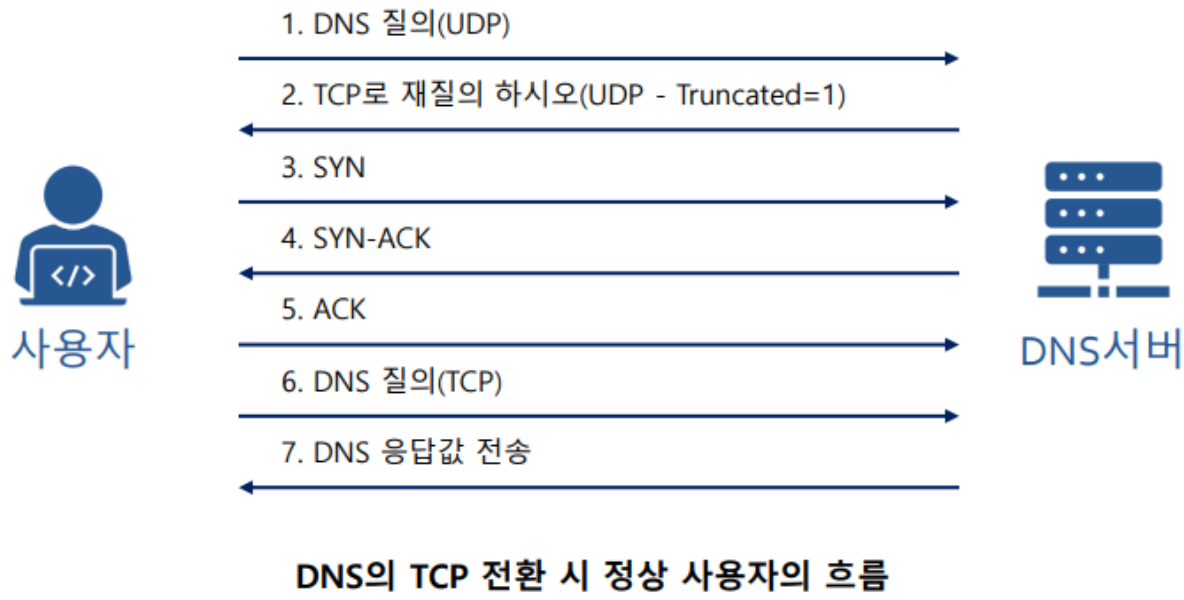
DNS 레코드	설명
A	도메인의 IP 주소(IPv4)를 의미
AAAA	도메인의 IP 주소(IPv6)를 의미
NS	도메인의 권한을 가진 네임 서버를 의미
CNAME	다른 도메인으로 위임하기 위한 별칭
TXT	형식이 지정되지 않은 임의의 텍스트 문자열의 응답
ANY	호스트가 보유한 전체 레코드의 응답을 의미



Reflection Attack

■ Reflection Attack – DNS

- DNS는 응답의 크기가 512 bytes를 초과하면 TCP로 전환되는 특징이 있음
 - DNS 질의 명령어의 요청에 대한 응답 형태



Reflection Attack

■ Reflection Attack – DNS

- DNS는 응답의 크기가 512 bytes를 초과하면 TCP로 전환되는 특징이 있음
 - DNS 질의 명령어의 요청에 대한 응답 형태 (DNS의 TCP 전환 시 패킷 흐름)

1.dns_tcp_truncate.pcap

File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help

Apply a display filter: <Ctrl-F>

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000	10.40.219.42	8.8.8.8	DNS	68	Standard query 0x7ee6 ANY iana.org
2	0.096226	8.8.8.8	10.40.219.42	DNS	68	Standard query response 0x7ee6 ANY iana.org
3	0.096757	10.40.219.42	8.8.8.8	TCP	78	58255 → 53 [SYN] Seq=0 Win=65535 Len=0 MSS=1460 WS=32 TSval=289184037 TSecr=0 SACK_PERM=1
4	0.158078	8.8.8.8	10.40.219.42	TCP	74	53 → 58255 [SYN, ACK] Seq=0 Ack=1 Win=28400 Len=0 MSS=1432 SACK_PERM=1 TSval=3581329340 TSecr=289184037 WS=256
5	0.158166	10.40.219.42	8.8.8.8	TCP	66	58255 → 53 [ACK] Seq=1 Ack=1 Win=132032 Len=0 TSval=289184098 TSecr=3581329340
6	0.158283	10.40.219.42	8.8.8.8	DNS	94	Standard query 0xd242 ANY iana.org
7	0.220593	8.8.8.8	10.40.219.42	TCP	66	53 → 58255 [ACK] Seq=1 Ack=29 Win=28416 Len=0 TSval=3581329402 TSecr=289184098
8	0.228068	8.8.8.8	10.40.219.42	TCP	1486	53 → 58255 [ACK] Seq=1 Ack=29 Win=28416 Len=1420 TSval=3581329402 TSecr=289184098 [TCP segment of a reassembled PDU]
9	0.228075	8.8.8.8	10.40.219.42	TCP	1486	53 → 58255 [ACK] Seq=1421 Ack=29 Win=28416 Len=1420 TSval=3581329402 TSecr=289184098 [TCP segment of a reassembled PDU]
10	0.228076	8.8.8.8	10.40.219.42	DNS	124	Standard query response 0xd242 ANY iana.org SOA sns.dns.icann.org RRSIG NSEC api.iana.org RRSIG NS a.iana-servers.net
11	0.228221	10.40.219.42	8.8.8.8	TCP	66	58255 → 53 [ACK] Seq=29 Ack=2841 Win=129216 Len=0 TSval=289184166 TSecr=3581329402

> Frame 2: 68 bytes on wire (544 bits), 68 bytes captured (544 bits)

> Ethernet II, Src: JuniperN_ff:10:01 (00:10:db:ff:10:01), Dst: Apple_d0:ec:ee (60:f8:1d:d0:ec:ee)

> Internet Protocol Version 4, Src: 8.8.8.8, Dst: 10.40.219.42

> User Datagram Protocol, Src Port: 53, Dst Port: 53915

> Domain Name System (response)

Transaction ID: 0x7ee6

Flags: 0x8380 Standard query response, No error

1... .. = Response: Message is a response

.000 0... .. = Opcode: Standard query (0)

... .. = Authoritative: Server is not an authority for domain

... .. = Truncated: Message is truncated

... .. = Recursion desired: Do query recursively

... .. = Recursion available: Server can do recursive queries

... .. = Z: reserved (0)

... .. = Answer authenticated: Answer/authority portion was not authenticated by the server

... .. = Non-authenticated data: Unacceptable

... .. = Reply code: No error (0)

Questions: 1

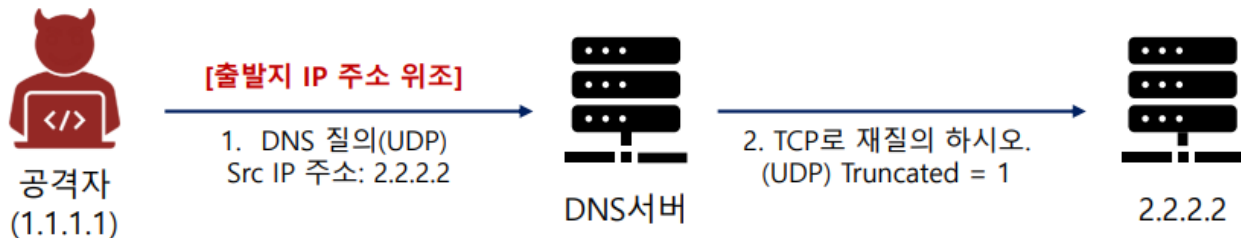
Answer RRs: 0



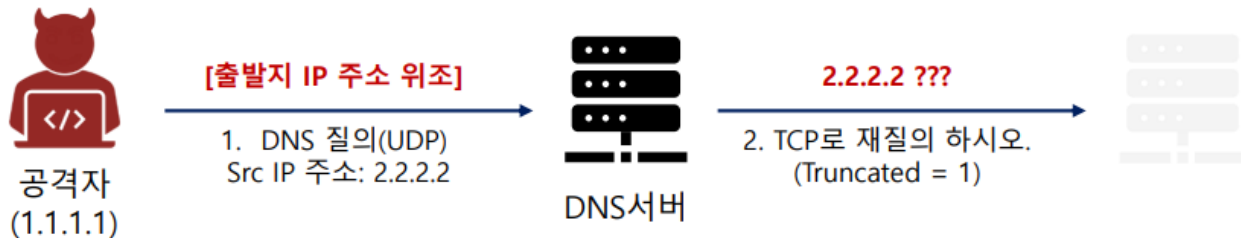
Reflection Attack

■ Reflection Attack – DNS

- DNS가 TCP로 전환될 때,
 - 위조된 IP 주소에서는 정상적인 3-way-handshake가 불가능하므로, 정상적인 DNS 질의와 응답이 발생 할 수 없음.



DNS가 TCP 전환 시, 위조된 IP 주소가 존재하는 IP 주소일 경우의 흐름



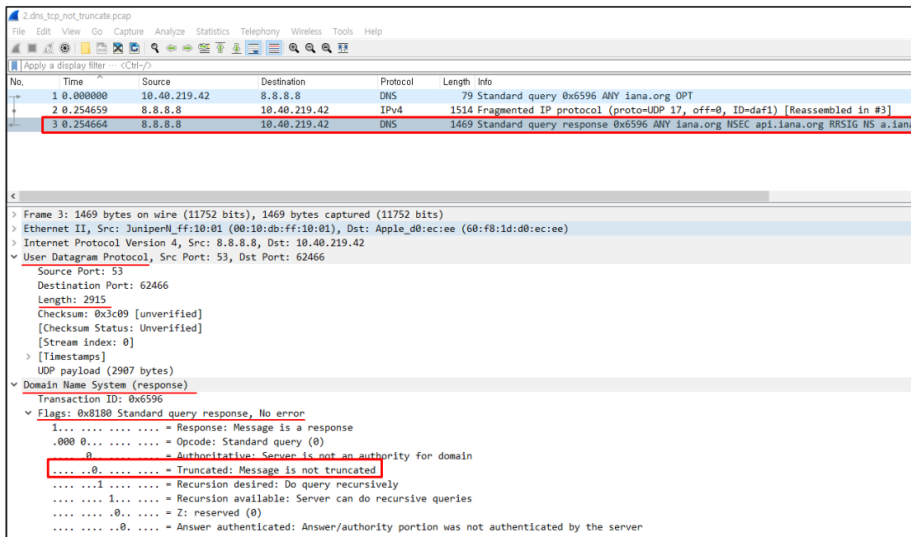
DNS가 TCP 전환 시, 위조된 IP 주소가 존재하지 않는 IP 주소일 경우의 흐름



Reflection Attack

■ Reflection Attack – DNS

- DNS가 TCP로 전환될 때,
 - 공격자 입장에서는 DNS 반사 공격을 유효하게 만들기 위해서는??
 - 응답 값의 크기를 최대한 크게 만들어야 함
 - 큰 응답값을 만들어 내기 위해서 DNS 레코드를 TXT 또는 ANY 형태로 질의 해야함
 - TCP 전환이 되지 않게 해야함(UDP인 상태를 유지)



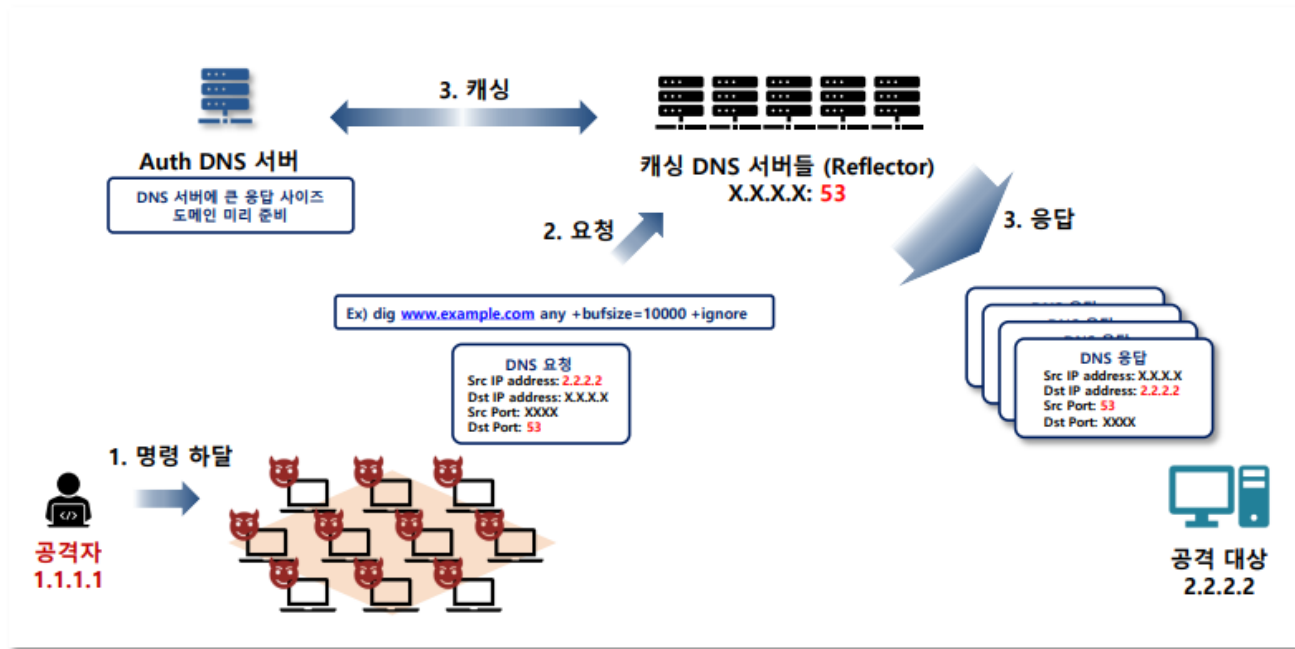
512 바이트 이상 크기에도 TCP로 전환되지 않은 DNS 응답 값



Reflection Attack

■ Reflection Attack – DNS

□ 공격 구조



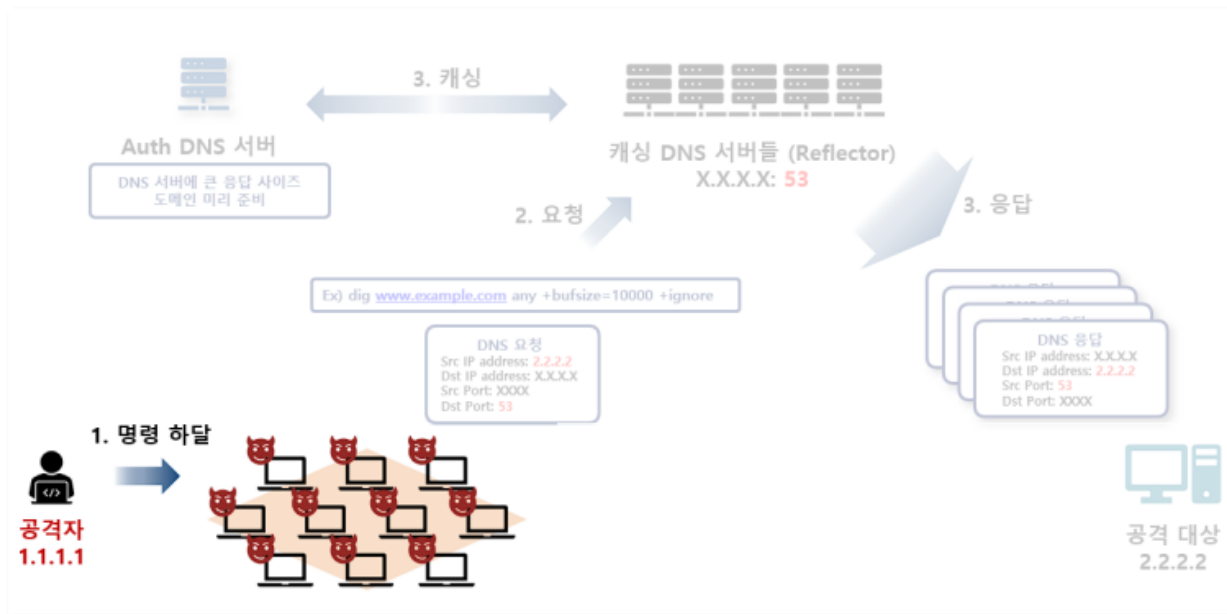
DNS 반사 공격의 구조



Reflection Attack

■ Reflection Attack – DNS

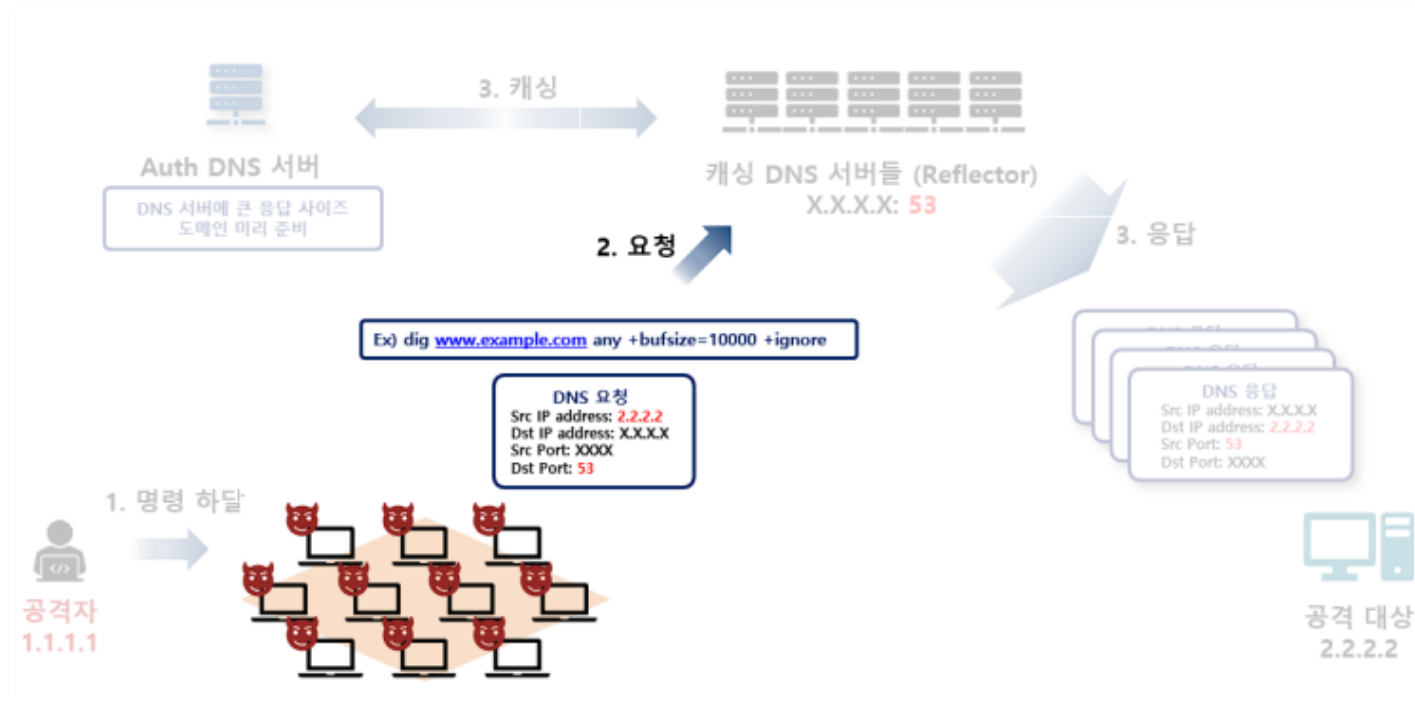
- 1. 공격자는 C&C를 이용하여 좀비 PC로 명령 전달



Reflection Attack

■ Reflection Attack – DNS

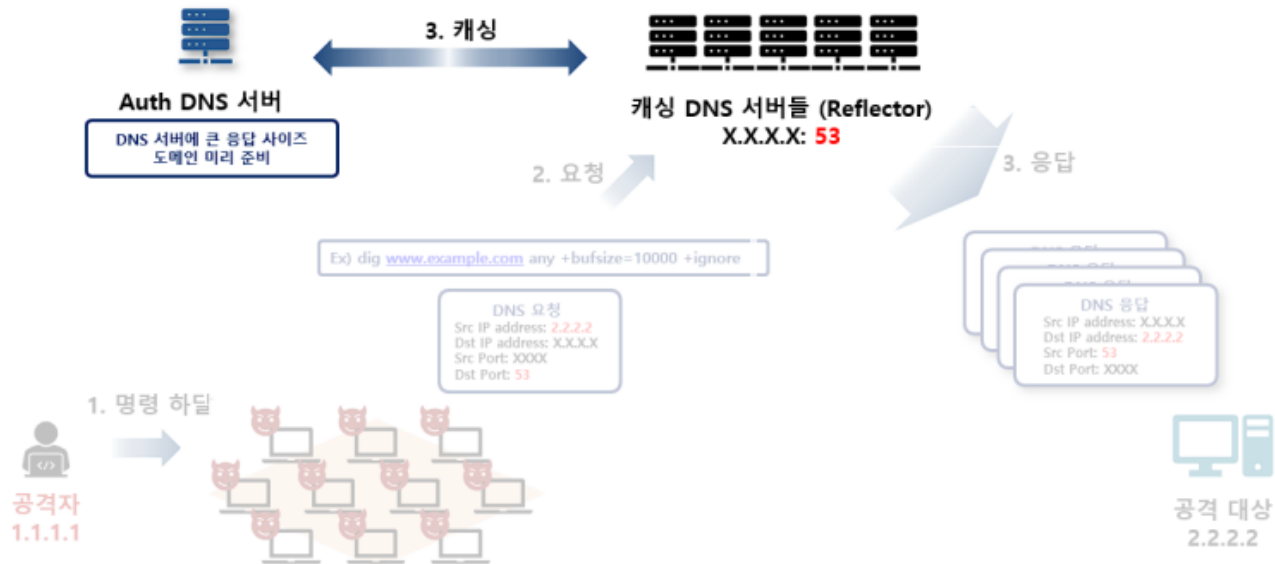
- 2. 좀비 PC들은 출발지 IP 주소를 공격 대상 IP 주소로 위조하여 다수의 캐싱 DNS에게 DNS 질의 수행
 - 큰 응답 값 생성을 위해 ANY 또는 TXT 레코드를 사용함



Reflection Attack

■ Reflection Attack – DNS

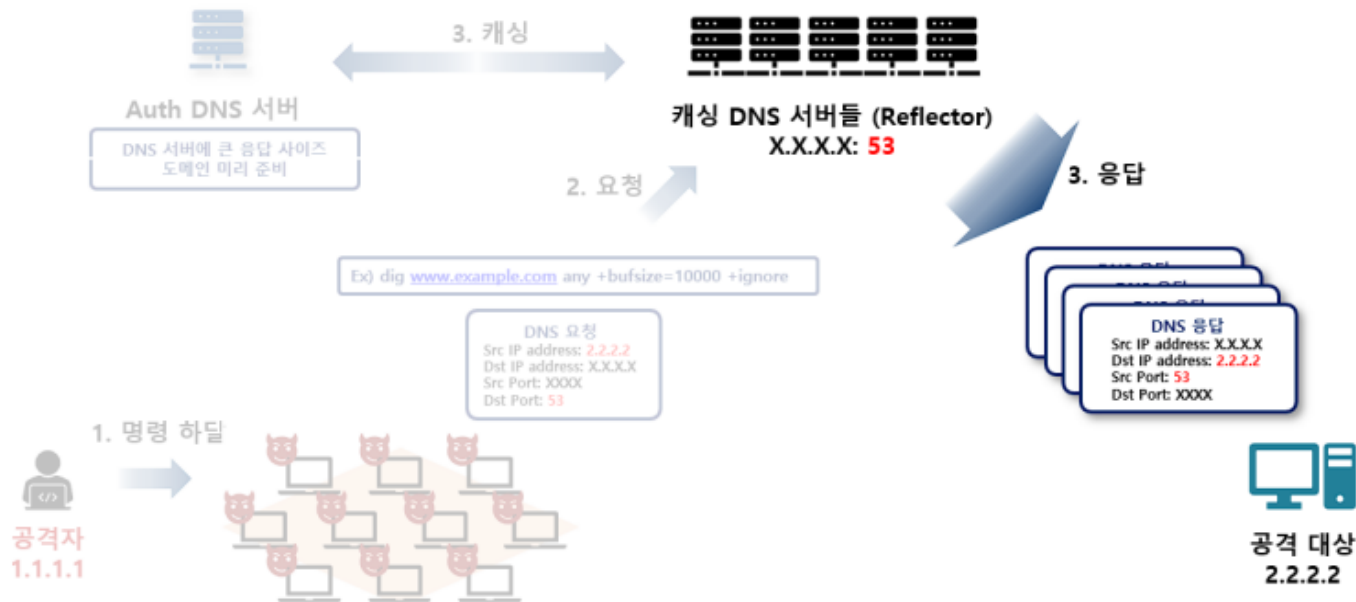
- 3. 만약 캐싱 DNS에서 사전에 캐싱이 되어 있지 않은 도메인이라면, 캐싱 DNS는 Auth DNS에게 질의를 하여 캐싱을 수행함
 - 이때, 공격자는 큰 응답 값을 만들어내기 위해 별도의 Auth DNS 서버를 구축하여 의도적인 큰 응답 값을 제작하거나, 인터넷상의 DNS에서 응답 값이 큰 도메인을 찾아서 반사 공격에 활용하기도 함



Reflection Attack

■ Reflection Attack – DNS

- 4. 질의 명령에 +bufsize와 +ignore 옵션이 사용되었기 때문에, 패킷 크기가 크더라도 UDP로 유지되며, 응답 패킷은 공격 대상에게 대역폭 공격으로 적용됨



Reflection Attack

■ Reflection Attack – DNS

□ 패킷 분석

- DNS 반사 공격은 네트워크 대역폭을 가득 채우는 것이 목적이기 때문에, 공격 대상의 서비스 종류에 상관없이 공격에 사용된 출발지 IP주소 (매개체 서버, 캐싱 DNS 서버들, Reflector)는 인터넷에서 실제 DNS를 서비스 중인 리졸빙 (DNS 쿼리에 대한 답변)이 허용된 캐싱 DNS 서버일 뿐 악성 코드에 감염된 좀비 PC는 아님

No.	Time	Source	Source Port	Destination	Destination Port	Protocol	Length	Info
41	0.001964	82.143.145.14	53	192.168.100.50	27034	DNS	1514	Standard query response 0x628a ANY cpsec.gov MX 5 stagg.cpsc.gov MX 5 hornel.cpsc.gov TXT A 63
42	0.002025	189.172.184.148	53	192.168.100.50	51292	DNS	1514	Standard query response 0x628a ANY cpsec.gov DNSKEY SOA auth00.ns.uu.net DNSKEY MX 5 stagg.cpsc
45	0.002198	89.222.175.183	53	192.168.100.50	37640	DNS	1514	Standard query response 0x9434 ANY cpsec.gov MX 5 stagg.cpsc.gov MX 5 hornel.cpsc.gov TXT A 63
46	0.002255	89.222.175.183	53	192.168.100.50	37640	DNS	1514	Standard query response 0x9434 ANY cpsec.gov MX 5 hornel.cpsc.gov MX 5 stagg.cpsc.gov TXT A 63
48	0.002368	82.151.98.242	53	192.168.100.50	64201	DNS	1514	Standard query response 0x628a ANY cpsec.gov DNSKEY DNSKEY DNSKEY AAAA 2600:803:240::2 A
49	0.002428	93.120.27.119	53	192.168.100.50	41797	DNS	1514	Standard query response 0x9434 ANY cpsec.gov RRSIG DS DS DS RRSIG RRSIG MX 5 hornel.cpsc.gov
50	0.002493	83.143.40.11	53	192.168.100.50	35051	DNS	1514	Standard query response 0x9434 ANY cpsec.gov MX 5 stagg.cpsc.gov MX 5 hornel.cpsc.gov TXT A 63
52	0.002610	202.96.85.132	53	192.168.100.50	4697	DNS	1514	Standard query response 0x628a ANY cpsec.gov SOA auth00.ns.uu.net RRSIG RRSIG RRSIG RRSIG
54	0.002732	131.8.44.50	53	192.168.100.50	61474	DNS	1514	Standard query response 0x628a ANY cpsec.gov RRSIG DS DS DS RRSIG SOA auth00.ns.uu.net RRSIG
55	0.002791	116.193.220.2	53	192.168.100.50	4284	DNS	1514	Standard query response 0x158a ANY cpsec.gov DS DS DNSKEY DNSKEY NSEC3PARAM cpsec.gov SOA auth00
58	0.002959	191.102.74.230	53	192.168.100.50	57873	DNS	1514	Standard query response 0x9434 ANY cpsec.gov NSEC3PARAM cpsec.gov DNSKEY DNSKEY DNSKEY A
65	0.003153	89.222.175.183	53	192.168.100.50	37640	DNS	1514	Standard query response 0x9434 ANY cpsec.gov MX 5 stagg.cpsc.gov MX 5 hornel.cpsc.gov TXT A 63
69	0.003577	195.149.138.3	53	192.168.100.50	7796	DNS	1514	Standard query response 0x628a ANY cpsec.gov MX 5 stagg.cpsc.gov MX 5 hornel.cpsc.gov TXT A 63
72	0.003748	94.236.230.190	53	192.168.100.50	27593	DNS	1514	Standard query response 0x9434 ANY cpsec.gov NSEC3PARAM cpsec.gov SOA auth00.ns.uu.net MX 5 stag
74	0.003854	203.176.140.138	53	192.168.100.50	1677	DNS	1514	Standard query response 0x9434 ANY cpsec.gov TXT A 63.74.109.2 AAAA 2600:803:240::2 SOA auth00

> Frame 49: 1514 bytes on wire (12112 bits), 1482 bytes captured (11856 bits)

> Ethernet II, Src: JuniperM_ca:28:01 (cc:e1:7f:ca:28:01), Dst: Dell_15:f1:f0 (44:a8:42:15:f1:f0)

> Internet Protocol Version 4, Src: 93.120.27.119, Dst: 192.168.100.50

> User Datagram Protocol, Src Port: 53, Dst Port: 41797

Source Port: 53

Destination Port: 41797

Length: 3306

Checksum: 0xc499 [Unverified]

[Checksum Status: Unverified]

[Stream index: 13]

[Timestamps]

UDP payload (1440 bytes)

> Domain Name System (response)

Transaction ID: 0x9434

> Flags: 0x8380 Standard query response, No error

Questions: 1

Answer RRs: 21

Authority RRs: 0

Additional RRs: 1

> Queries

> cpsec.gov: type ANY, class IN

> Answers



Reflection Attack

■ Reflection Attack – DNS

□ 패킷 분석

- 이전 슬라이드의 DNS 반사 공격 트래픽은 프로토콜이 UDP, 출발지 포트가 53인 것으로 보아 DNS 응답 패킷임
- ANY 레코드를 사용하여 3,306 바이트라는 응답 크기가 큰 것을 확인할 수 있음
- DNS 반사 공격의 경우, 공격 톨로 패킷을 생성하여 전송이 가능하지만 주로 인터넷상에서
 - 서비스 중인 캐싱 DNS 서버들로부터 전달된 응답 트래픽이므로 공격이 발생한 출발지 IP 주소로
 - DNS 질의를 하면 정상적인 DNS 응답이 수신되는 것도 확인할 수 있음.

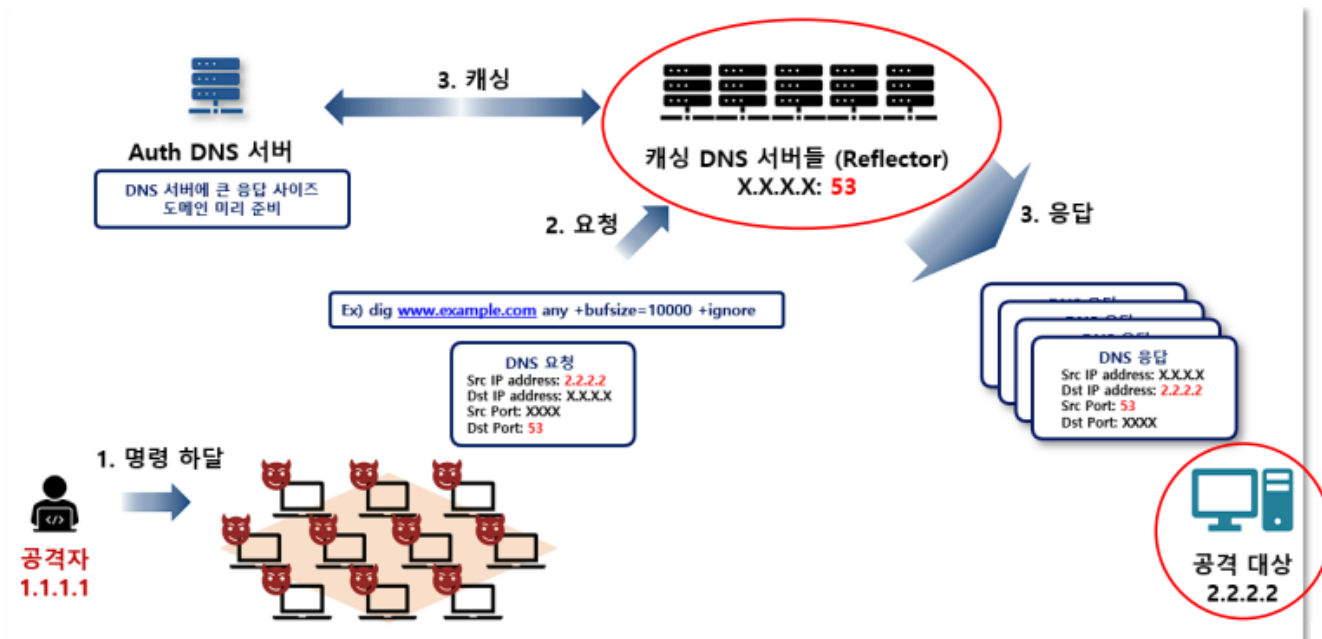


Reflection Attack

■ Reflection Attack – DNS

□ 대응 방법

- 공격 대상에서의 관점
- 매개체에서의 관점(캐싱 DNS 관점)



Reflection Attack

■ Reflection Attack – DNS

- 대응 방법 : 공격 대상에서의 관점
 - 미사용 프로토콜 차단
 - 운영중인 네트워크 내에서 UDP를 이용한 서비스가 존재하지 않을 경우,
 - 상단 라우터 또는 차단 장비에서 UDP를 우선 전면 차단
 - 그 후, "resolv.conf"에 등록된 DNS 서버의 IP주소 또는 내부망 IP 주소 대역과
 - 같은 특정 신뢰할 수 있는 IP주소들만 허용하는 것으로 대응 가능
 - 웹서비스만 운영 중인 네트워크로 DNS 반사 공격이 발생한다면 해당 정책을 수립할 수 있음.



Reflection Attack

■ Reflection Attack – DNS

- 대응 방법: 공격 대상에서의 관점
 - Auth DNS가 공격 대상일 경우, 수신하는 응답 패킷 차단
 - DNS를 운영중인 네트워크 환경일 경우, Auth DNS는 외부로 부터 DNS 요청 질의를 받아 응답 값을 전달하는 역할을 수행
 - Auth DNS 자신이 응답 값을 수신하는 경우는 "resolv.conf"에 등록된 DNS 서버와 통신을 수행할 때임
 - Auth DNS는 "resolv.conf"에 **설정된** IP주소 이외에 DNS 응답 패킷을 수신하는 경우가 없기 때문에, 신뢰할 수 있는 IP 주소 ("resolv.conf"에 설정된 DNS IP 주소 또는 내부 IP 주소 대역) 를 **제외한 모든 DNS 응답 패킷을 차단, 또는 임계치 기반 차 단 기능을 이용하여 비정상 응답 패킷을 차단함**
- 대응 방법: 매개체에서의 관점(캐싱 DNS 관점)
 - DNS 반사 공격 시 매개체의 관점에서는 대량의 요청 패킷을 수신하게 되므로, DNS Query Flooding과 거의 흡사한 형태의 패킷을 수신하게 됨
 - 또한, ANY 또는 TXT 등 응답 값이 큰 레코드를 이용한 요청 질의가 발생한다는 점이 일반적인 Query Flooding과 다름
 - ANY, TXT 등의 레코드는 거의 사용되지 않기 때문에, 운영상 사용하지 않을 경우 원천 차단하거나 임계치 초과 시 차단하도록 설정 가능
 - 임계치 초과인 경우, 분당 10회 또는 20회 등 해당 네트워크에서의 정상상태에 대한 통계치가 필요함
 - Iptables 옵션으로 위의 정책들을 적용 가능

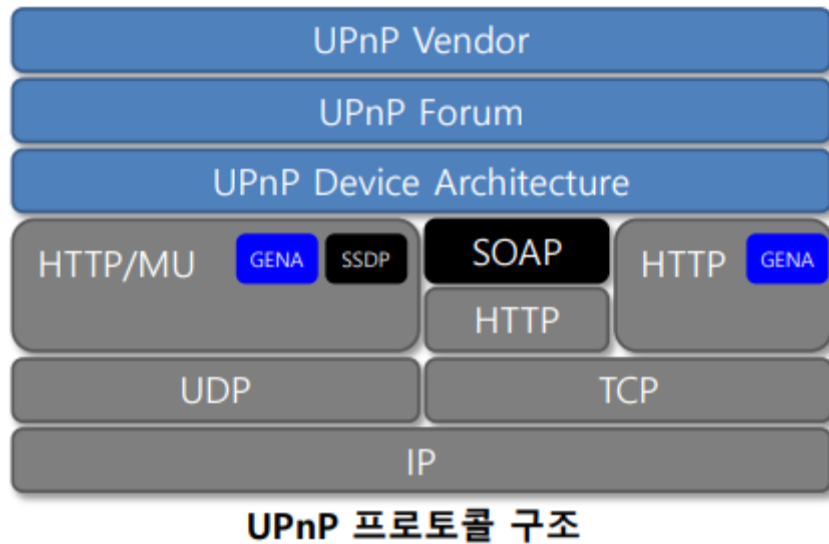


대역폭 공격 – SSDP 반사 공격

Reflection Attack

■ Reflection Attack – SSDP

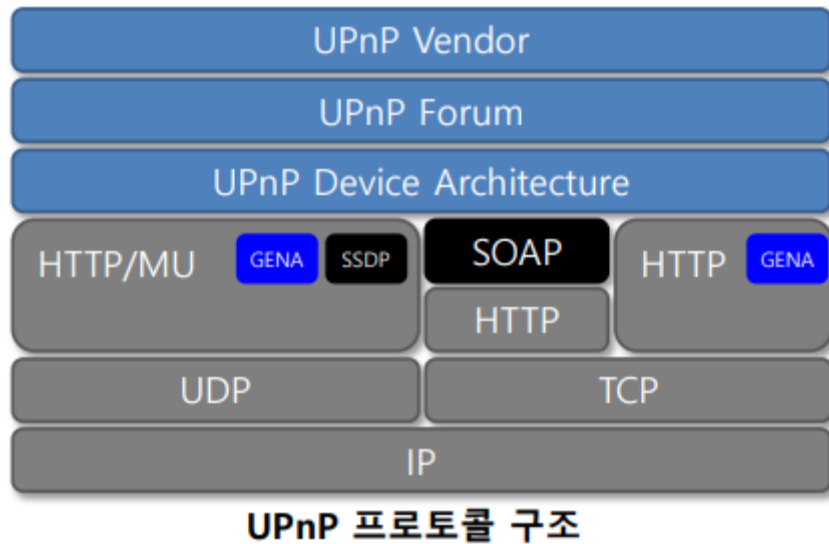
- SSDP 반사 공격은 IoT 기기를 매개체로 사용하는 반사 공격이며, UPnP 기능을 사용함
 - PnP는 특정 하드웨어를 PC에 추가할 때 운영체제가 이를 자동으로 인식하여 설치를 도와주는 기능
 - UPnP는 네트워크에 특정 기기를 추가할 때 네트워크 상에서 추가된 기기를 자동으로 인식하게 도와주는 기능



Reflection Attack

■ Reflection Attack – SSDP

- SSDP는 UDP에 속해 있지만 HTTPU/MU라는 프로토콜 내부에 포함되어 있음
 - HTTPU, HTTPMU는 UDP/IP 기반에서 HTTP를 기본 프로토콜로 하여 유니캐스트, 멀티캐스트를 지원하는 프로토콜
 - SSDP는 UDP임에도 불구하고 헤더 구조상에 HTTP 프로토콜이 사용되는 특이한 구조로 구성됨



Reflection Attack

■ Reflection Attack – SSDP

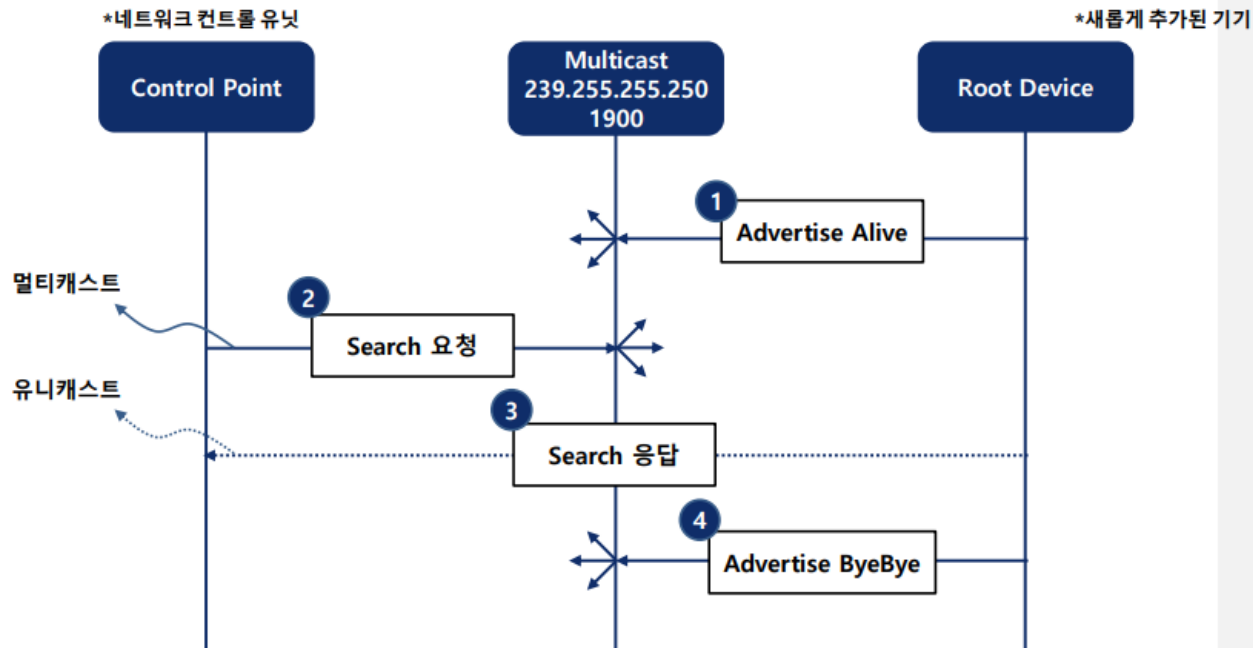
- SSDP는 네트워크상에서 서비스나 정보를 찾기 위한 프로토콜
 - 특정 기기(Root Device)가 네트워크에 추가되었을 경우, 이 장치에 대한 정보를 네트워크 기기(Control Point)에 알리고 자동으로 네트워크에 추가되는 것을 도와주는 역할을 수행
- DDoS 공격 관점에서 SSDP 프로토콜의 가장 중요한 정보는 UDP 프로토콜과 1900번 포트를 사용하는 부분임
 - SSDP는 HTTPU (UDP 기반의 HTTP) 사용
 - 모든 데이터는 text로 통신
 - 사용 프로토콜은 UDP, 포트는 1900번
 - IPv4에서 멀티캐스트 주소는 239.255.255.250, IPv6에서는 ff0x::c
 - SSDP로 통신하기 위한 기기들은 해당 IP 주소를 통해 광고 및 연결을 수행
 - UPnP 기기(또는 IoT 기기)가 네트워크 탐색을 할 때 사용



Reflection Attack

■ Reflection Attack – SSDP

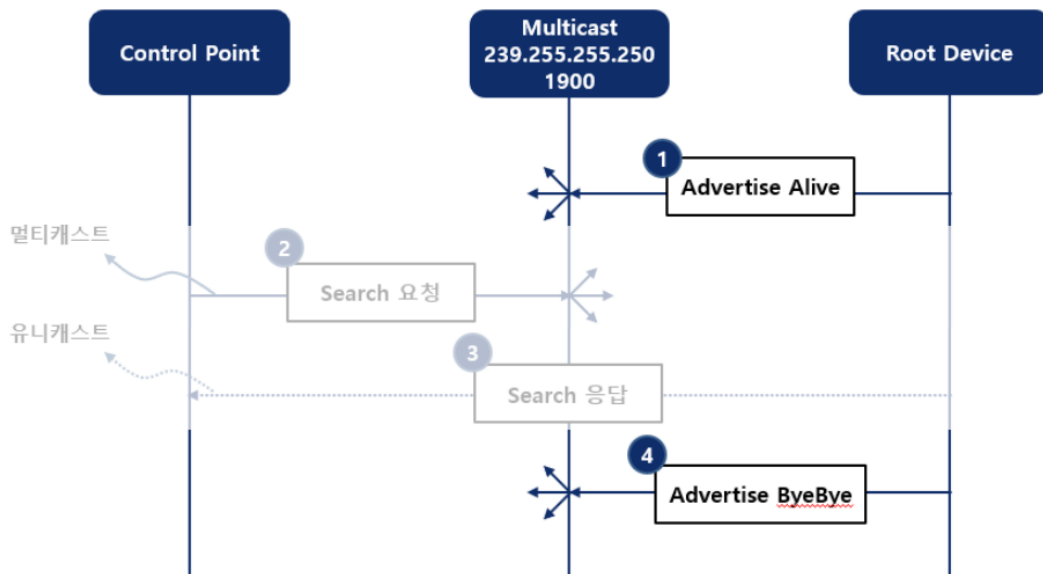
- SSDP는 네트워크상에서 서비스나 정보를 찾기 위한 프로토콜



Reflection Attack

■ Reflection Attack – SSDP

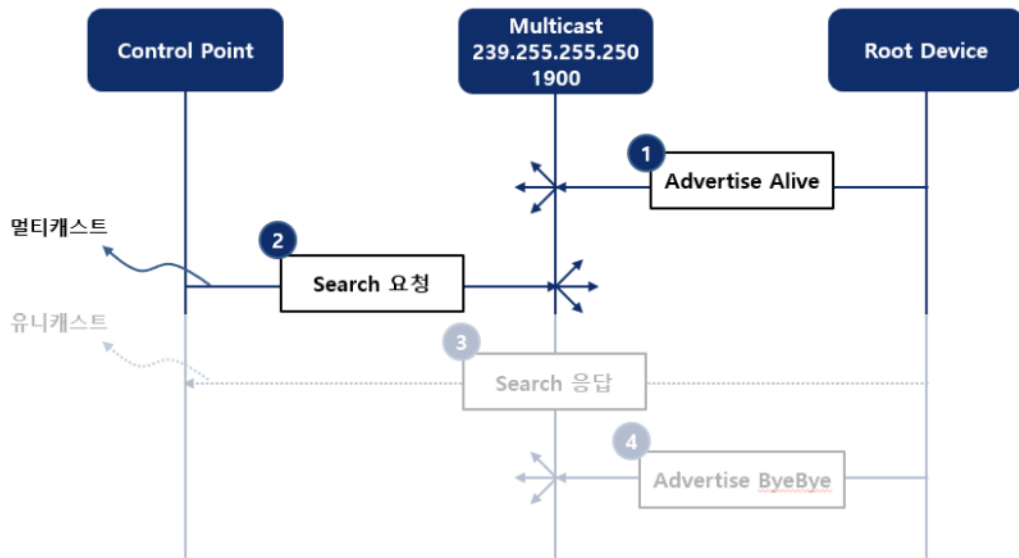
- 1. (or 4) Advertise Alive
 - NOTIFY * HTTP/1.1
 - 특정 UPnP 기기(무선라우터, CCTV, 홈캠, 냉장고, 프린터 등)와 같은 Root device 가
 - 네트워크 그룹에 연결되거나 연결 종료 정보를 알리기 위한 목적으로 사용됨
 - 멀티캐스트 주소인 239.255.255.250 또는 ff0x::c를 이용해 자신의 존재를 네트워크 그룹에 광고함



Reflection Attack

■ Reflection Attack – SSDP

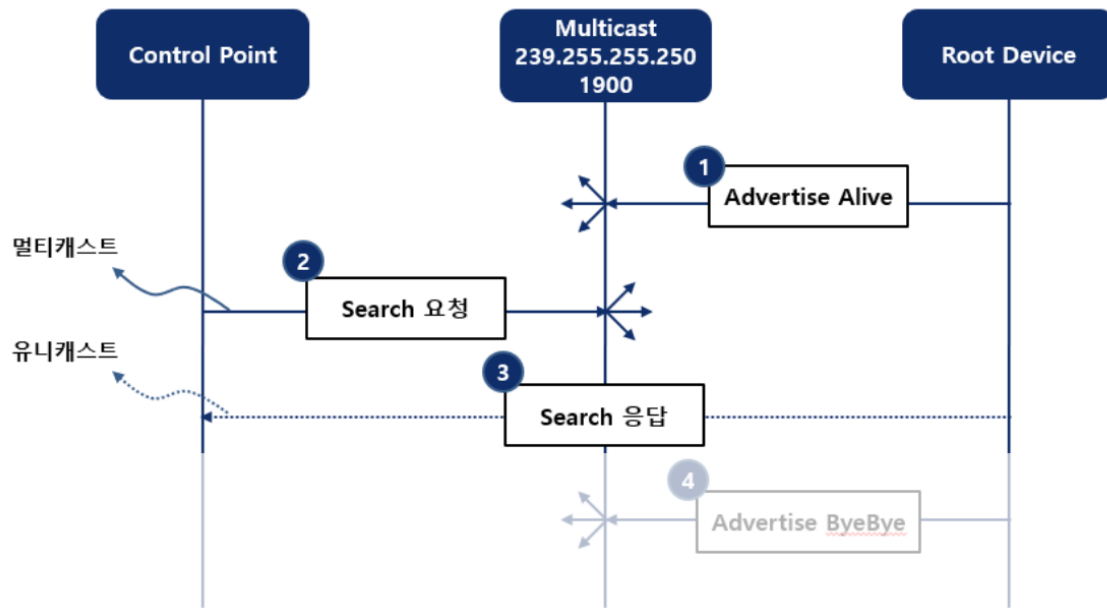
- 2. Search 요청
 - M-SEARCH * HTTP/1.1
 - 특정 Control Point (핸드폰 등 조종 주체)가 자신이 연결할 UPnP 기기(홈캠, 보일러, 냉장고, 프린터 등)를 검색할 때 M-SEARCH 메소드를 사용함
 - 멀티캐스트 주소 239.255.255.250 또는 ff0x::c를 이용하여 자신이 특정 기기를 찾고 있음을 네트워크 그룹에게 알림



Reflection Attack

■ Reflection Attack – SSDP

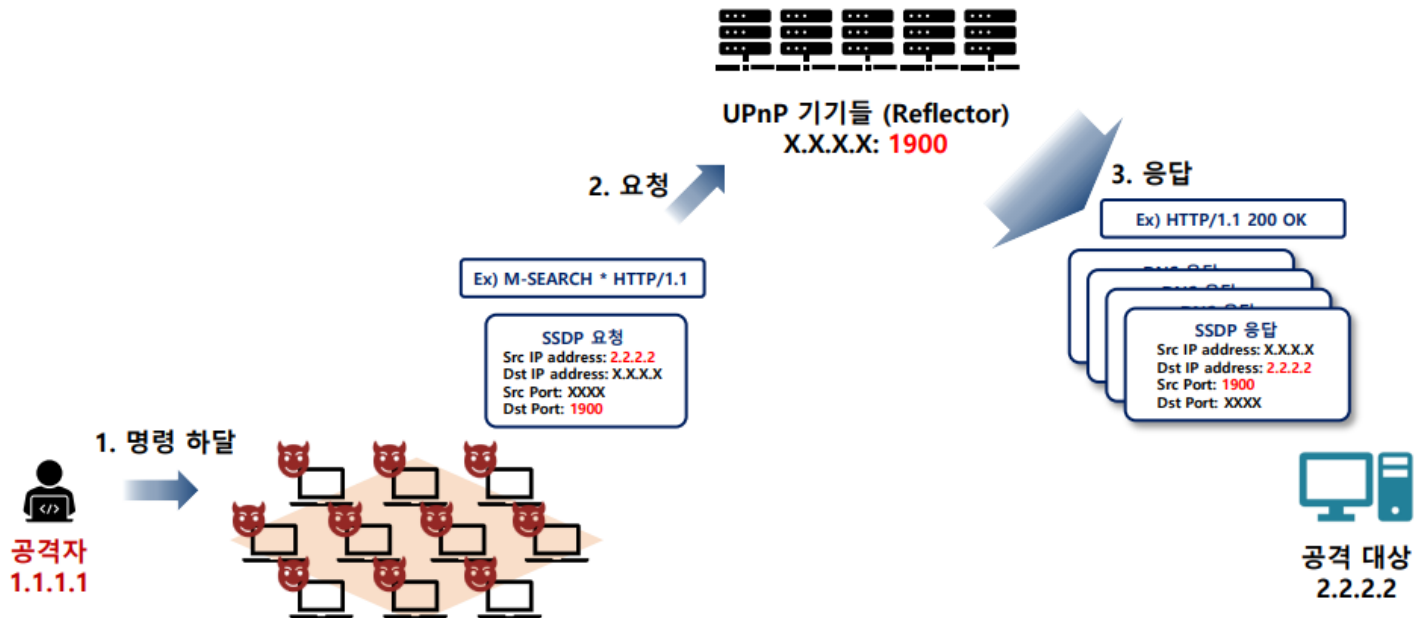
- 3. Search 응답
 - HTTP/1.1 200 OK
 - Control Point로부터 전송된 M-SEARCH 요청에 특정 UPnP 기기가 응답할 때 HTTP/1.1 200 OK를 사용
 - 응답 값에서 LOCATION 헤더에 해당 UPnP 기기의 세부 정보가 기록된 xml 파일의 URL 정보가 포함되어 있음



Reflection Attack

■ Reflection Attack – SSDP

- SSDP 반사 공격의 구조



Reflection Attack

■ Reflection Attack – SSDP

□ 공격 흐름

- 공격자는 C&C를 이용하여 좀비 PC에게 명령 전달
- 좀비 PC들은 출발지 IP를 공격 대상의 IP로 위조하여 수많은 UPnP 기기들(매개체)로
 - M-SEARCH를 이용한 요청 값을 전 송함
- M-SEARCH를 수신한 UPnP 기기들은 응답 값을 공격 대상 (위조된 출발지 IP)에게 전송하여
 - 네트워크 대역폭을 고갈시 키며, 이때 사용되는 출발지 포트 번호는 1900번으로 고정

** UPnP 기기들은 대부분 사설 IP로 특정 네트워크 내에서만 사용되는 것이 일반적임.

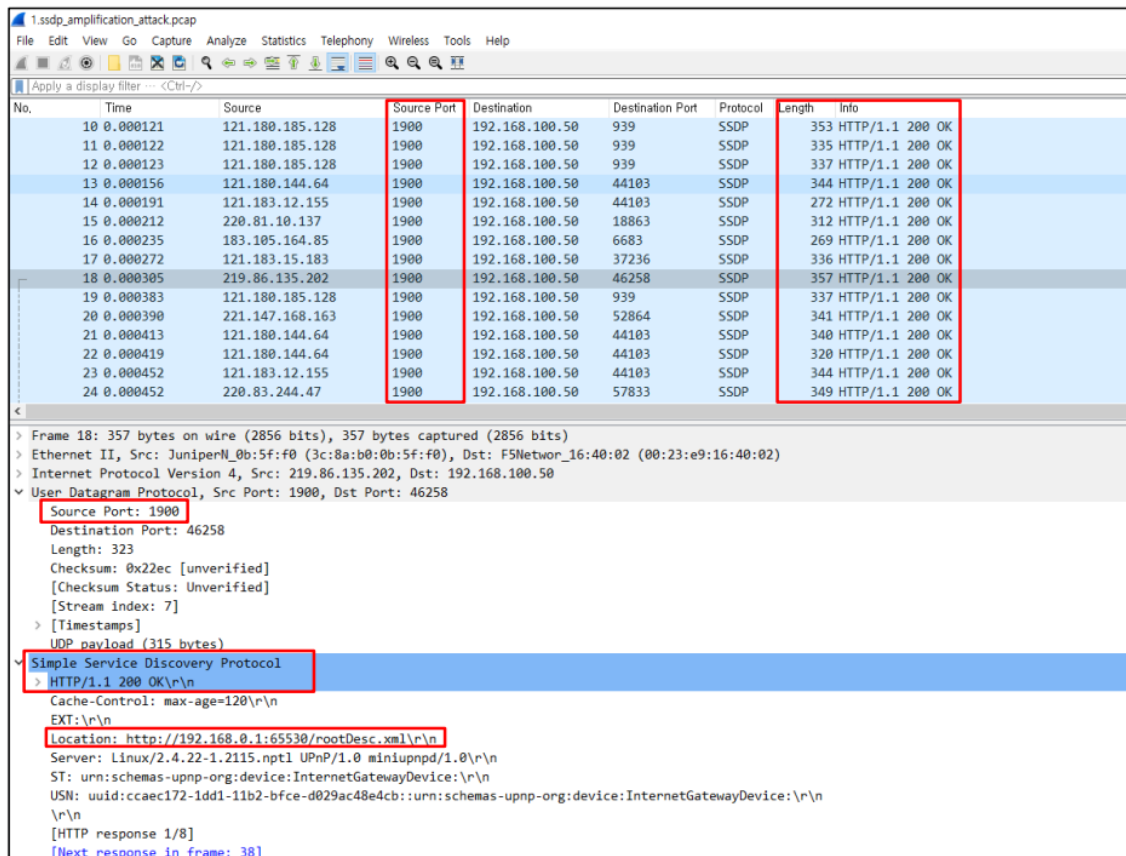
=> 반사 공격의 매개체로 사용되는 이유는??



Reflection Attack

■ Reflection Attack – SSDP

□ SSDP 반사 공격 패킷 분석



i.ssdp_amplification_attack.pcap

File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help

Apply a display filter -- <Ctrl-/>

No.	Time	Source	Source Port	Destination	Destination Port	Protocol	Length	Info
10	0.000121	121.180.185.128	1900	192.168.100.50	939	SSDP	353	HTTP/1.1 200 OK
11	0.000122	121.180.185.128	1900	192.168.100.50	939	SSDP	335	HTTP/1.1 200 OK
12	0.000123	121.180.185.128	1900	192.168.100.50	939	SSDP	337	HTTP/1.1 200 OK
13	0.000156	121.180.144.64	1900	192.168.100.50	44103	SSDP	344	HTTP/1.1 200 OK
14	0.000191	121.183.12.155	1900	192.168.100.50	44103	SSDP	272	HTTP/1.1 200 OK
15	0.000212	220.81.10.137	1900	192.168.100.50	18863	SSDP	312	HTTP/1.1 200 OK
16	0.000235	183.105.164.85	1900	192.168.100.50	6683	SSDP	269	HTTP/1.1 200 OK
17	0.000272	121.183.15.183	1900	192.168.100.50	37236	SSDP	336	HTTP/1.1 200 OK
18	0.000305	219.86.135.202	1900	192.168.100.50	46258	SSDP	357	HTTP/1.1 200 OK
19	0.000383	121.180.185.128	1900	192.168.100.50	939	SSDP	337	HTTP/1.1 200 OK
20	0.000390	221.147.168.163	1900	192.168.100.50	52864	SSDP	341	HTTP/1.1 200 OK
21	0.000413	121.180.144.64	1900	192.168.100.50	44103	SSDP	340	HTTP/1.1 200 OK
22	0.000419	121.180.144.64	1900	192.168.100.50	44103	SSDP	320	HTTP/1.1 200 OK
23	0.000452	121.183.12.155	1900	192.168.100.50	44103	SSDP	344	HTTP/1.1 200 OK
24	0.000452	220.83.244.47	1900	192.168.100.50	57833	SSDP	349	HTTP/1.1 200 OK

> Frame 18: 357 bytes on wire (2856 bits), 357 bytes captured (2856 bits)

> Ethernet II, Src: JuniperN_0b:5f:f0 (3c:8a:b0:0b:5f:f0), Dst: F5Network_16:40:02 (00:23:e9:16:40:02)

> Internet Protocol Version 4, Src: 219.86.135.202, Dst: 192.168.100.50

> User Datagram Protocol, Src Port: 1900, Dst Port: 46258

Source Port: 1900

Destination Port: 46258

Length: 323

Checksum: 0x22ec [unverified]

[Checksum Status: Unverified]

[Stream index: 7]

> [Timestamps]

UDP payload (315 bytes)

> Simple Service Discovery Protocol

> HTTP/1.1 200 OK\r\n

Cache-Control: max-age=120\r\n

EXT:\r\n

Location: http://192.168.0.1:65530/rootDesc.xml\r\n

Server: Linux/2.4.22-1.2115.nptl UPnP/1.0 miniupnpd/1.0\r\n

ST: urn:schemas-upnp-org:device:InternetGatewayDevice:\r\n

USN: uuid:ccaec172-1dd1-11b2-bfcd-d029ac48e4cb:urn:schemas-upnp-org:device:InternetGatewayDevice:\r\n

\r\n

[HTTP response 1/8]

[Next response in frame: 38]



Reflection Attack

■ Reflection Attack – SSDP

- SSDP 반사 공격 특징
 - UDP를 사용
 - 출발지 포트는 1900번
 - 출발지 IP주소들은 실제 UPnP 기기들
 - 피해자가 수신한 응답 패킷은 UDP임에도 HTTP/1.1 200 OK 라는 값이 사용됨
 - SSDP가 HTTPU에 속하기 때문
 - Location 헤더에 실제 UPnP 기기의 세부정보를 보여주는 링크가 포함됨



Reflection Attack

■ Reflection Attack – SSDP

□ SSDP 반사 공격 대응방안

• 공격 대상 관점

- 자신이 운영하는 네트워크 또는 사내 네트워크에 UPnP 기기들이 사용되지 않는 네트워크라면 출발지 포트가 1900번인 UDP 패킷(SSDP 응답 값)을 차단하여 대응이 가능함
- 또한, UDP이면서 'HTTP/1.1 200 OK' 문자열이 포함된 패킷의 경우, signature로 등록하여 해당 문자열이 포함된 패킷들을 차단하는 방법으로 대응
- 공인 IP 사용 환경일 경우, 신뢰할 수 있는 기기의 IP 주소와 IP 주소 대역을 예외 처리해야 함.

• 매개체 관점

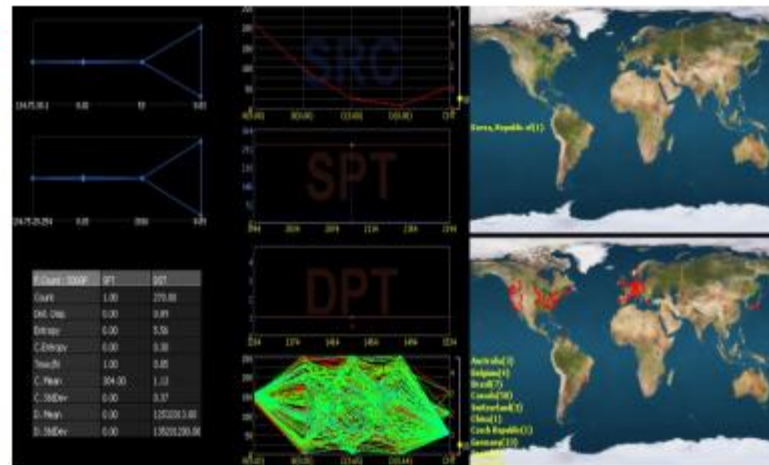
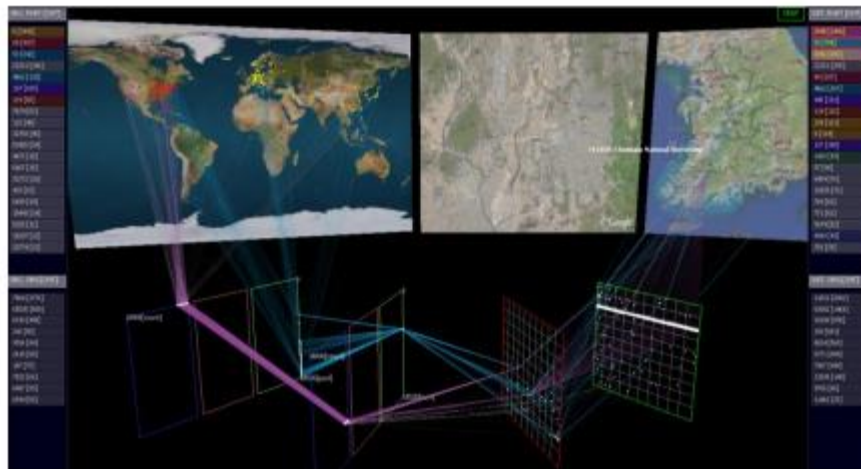
- 공인 IP 주소로 설정되어 사용하는 기기들의 경우, 사설 IP 주소로 변경하여 사용
 - 공인 IP 주소를 꼭 사용해야 하는 경우, 신뢰할 수 있는 IP 주소만 접속 가능하도록 ACL (Access Control List) 설정하여 접근 제어를 수행
- UPnP 기능을 사용하지 않는 기기들의 경우,
 - 기기들의 설정 화면을 제공하지 않게끔 비활성화 하여 매개체로 사용되는 것을 방지



보안 시각화

보안 시각화

■ ETRI VisNet and VisMon



보안 시각화

■ 데이터 시각화

- 직관적이고 그래픽적으로 형태가 명확하며, 그래프 형태별로 의미를 가짐으로써 정확한 정보를 전달

■ 시각화 전략

- 사용자
- 통계 그래프 기법
- 기술적 고려사항
- 보안 이벤트 시각화

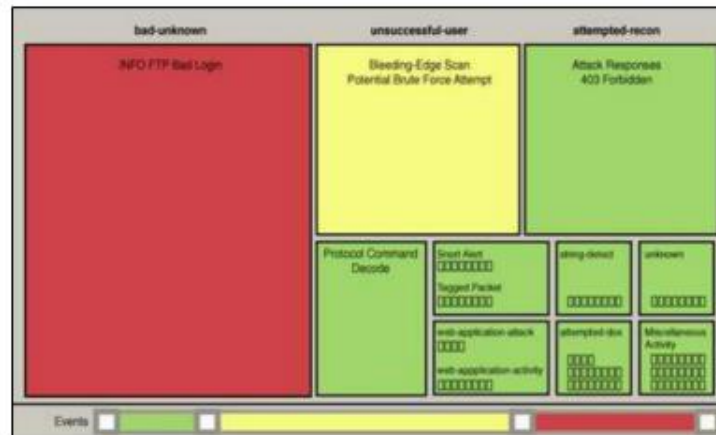
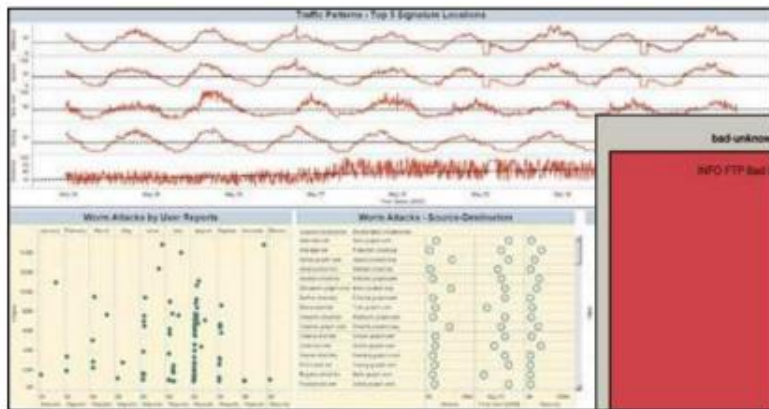


보안 시각화

■ 시각화 전략

□ 사용자

- 시각화를 프로그래밍 및 작성하는 사람은 사용자 그룹이나 개인 사용자를 위해서 그래프를 작성해야 함
 - 보여주기 식이 아닌, 실제 이용자를 대상으로 작성



보안 시각화

■ 시각화 전략

□ 사용자

사용자	시각화 요구사항
관리	전체 네트워크 또는 전반적인 사항에 대한 요약을 해주는 시각화가 필요
운영	실시간 우선 순위 경보, 기록 데이터, 이벤트 상관 관계, 관리 부분에 지원을 할 수 있도록 개별적인 보안 시각화가 필요
개발	개별 또는 그룹 장비, 어플리케이션, 프로토콜, 일반적인 성능에 관한 시각화가 필요 해당 시각화에는 단기 및 장기 로그, 에러 로그 및 에러 시나리오, 네트워크 모델 등이 포함
고객	세부적인 사항 보다 일반적인 사항으로 응답 시간, 네트워크 가용성, 애플리케이션 가용성, 평균 복구 시간 등의 다른 관리/운영/개발쪽에서의 확인 내용보다 일반화된 시각화가 필요



보안 시각화

■ 시각화 전략

- 통계 그래프 기법

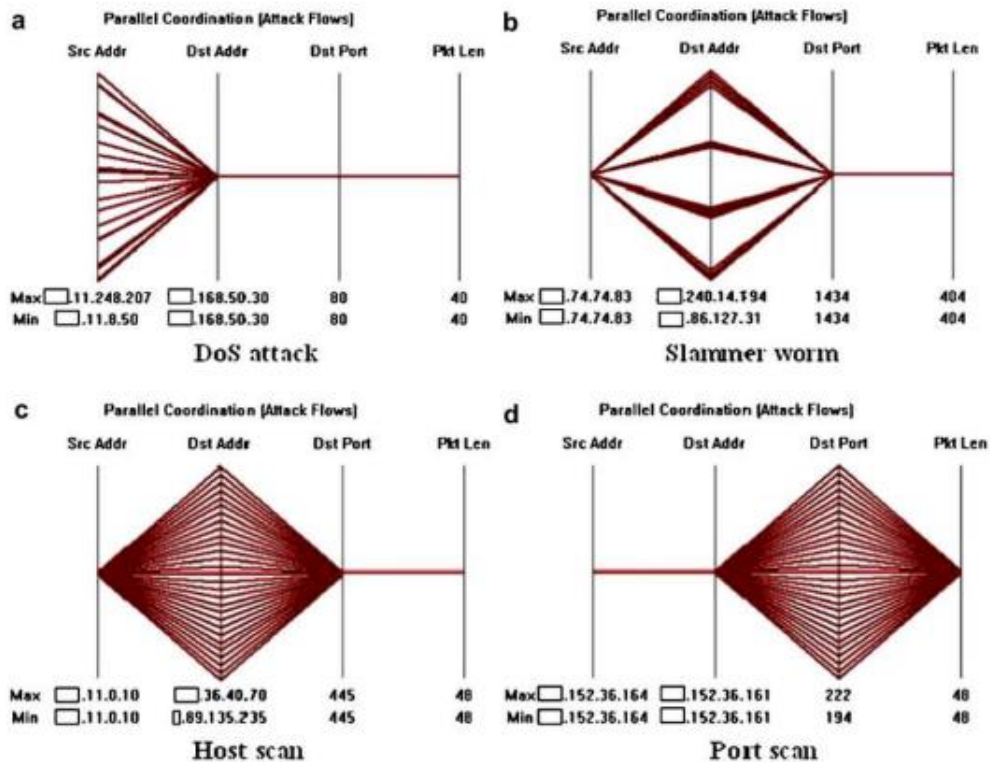


Fig. 1 – Rescaled attack graphs.

[Source Choi, H., Lee, H., & Kim, H. (2009). Fast detection and visualization of network attacks on parallel coordinates. computers & security, 28(5), 276-288.]



보안 시각화

■ 시각화 전략

- 통계 그래프 기법

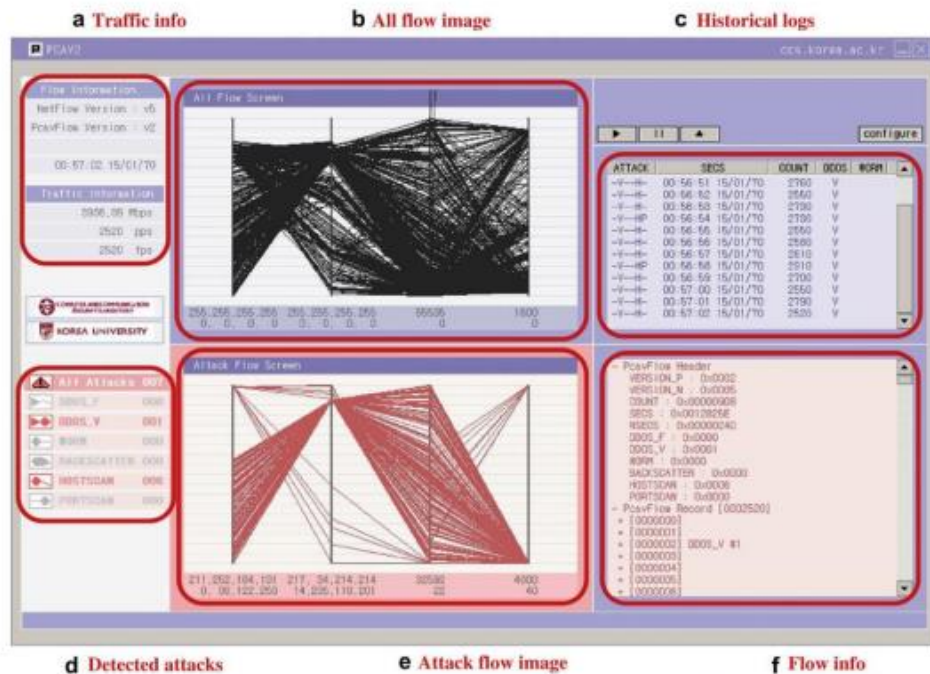


Fig. 5 - Screenshot of the PCAV application running on backbone traffic.

Table 1 - Graphical signatures of nine attacks.

Implied Attack	Signature	Divergences
Portscan		1:1:m:1
Hostscan		1:m:1:1
Worm		1:m:1:1
Source-spoofed DoS (port fixed)		m:1:1:1
Backscatter		1:m:m:1
Source-spoofed DoS (port varied)		m:1:m:1
Distributed hostscan		m:m:1:1
Network-directed DoS		m:m:m:1
Single-source DoS		1:1:1:1



보안 시각화

■ 기술적 고려사항

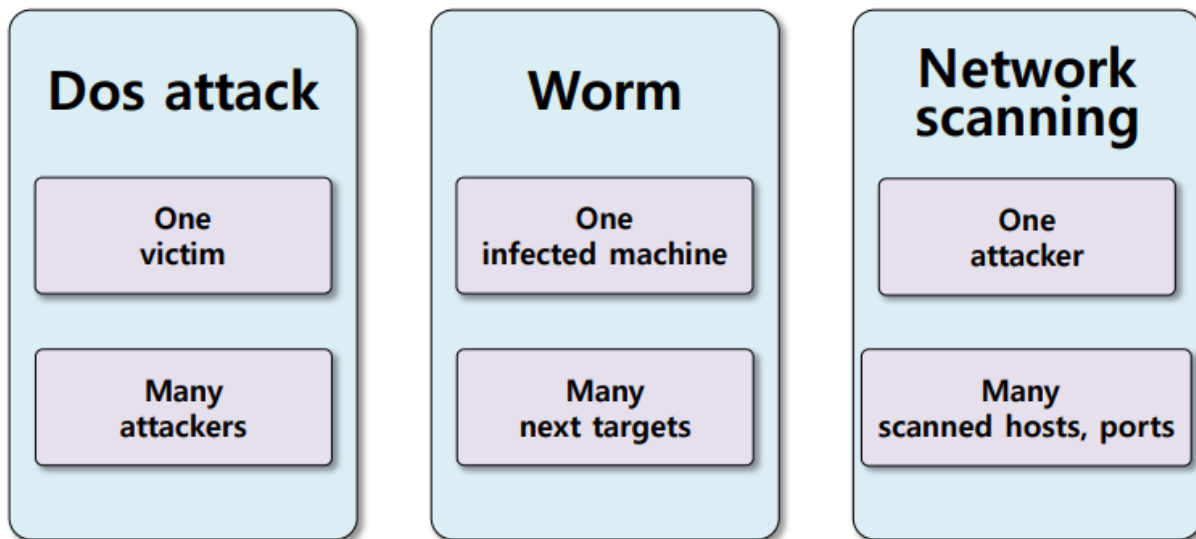
□ 확장성 (Scalability)

- 그래프 작성 시, 시각화 프로그램의 경우 많은 양의 트래픽 또는 로그들을 전반적으로 표현
 - 데이터의 크기 및 양에 관련됨
- 그래프를 단순하게 할 경우, 의사결정을 할 때 적용하기 어려울 수 있음
 - 예) 초당 패킷의 수(PPS)만 시각화한 그래프에서 포트 스캐닝 인지 아닌지 확인하는 것은 제한적임
 - 보안 시각화 설계 시 고려할 사항
 - 확인하고자 하는 문제를 표현해줄 수 있는 적절한 차트
 - 차트에서 현재 표현하는 Feature의 추가
 - 차트의 복잡성을 증가시키지 않음 (너무 많은 정보는 의사 결정을 내리기 어렵게 만듦)
 - 시각화 차트/그래프에서 색상 또는 모양 적용



보안 시각화

■ 공격 특징



보안 시각화

■ 공격관련 4개의 중요 변수

- Source 및 Destination IP 주소
 - 공격자 호스트와 피해자 호스트를 확인할 수 있음

- 목적지 포트번호
 - 공격 대상의 서비스를 식별하고 포트 스캐닝 공격을 검증

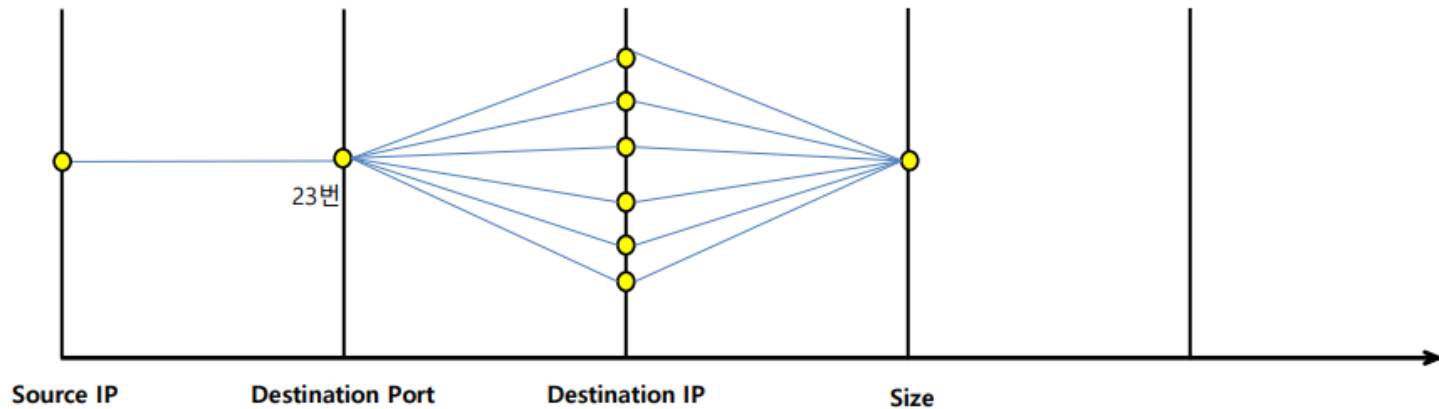
- 패킷의 평균 크기
 - 대부분의 웜은 일정한 페이로드로 전파되며, 웜의 평균 패킷 크기는 고정된 길이로 지정할 수 있음

- TCP 헤더의 플래그 및 IP 헤더의 프로토콜 필드
 - 다른 Feature로 사용할 수 있음



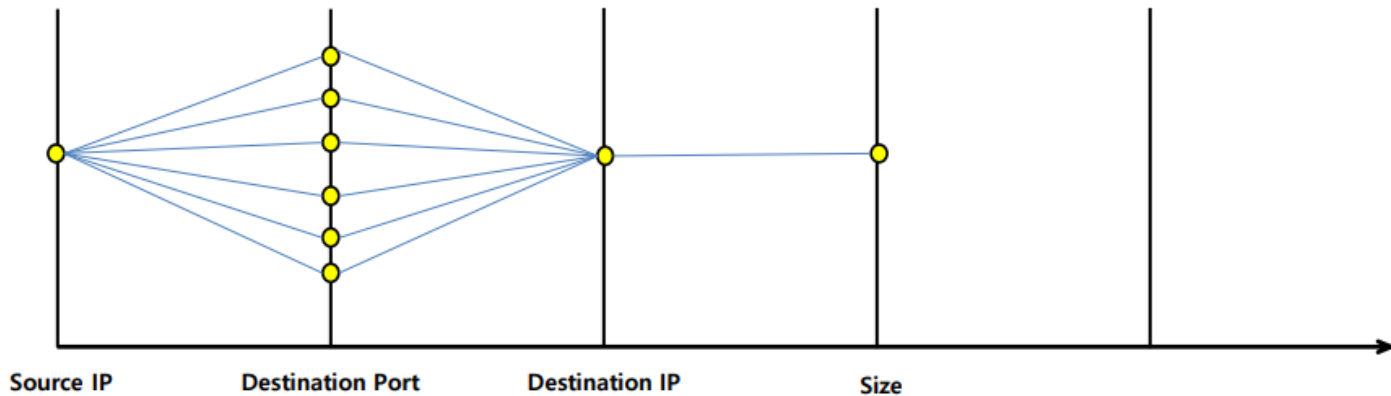
보안 시각화


■ Example



보안 시각화

■ Example

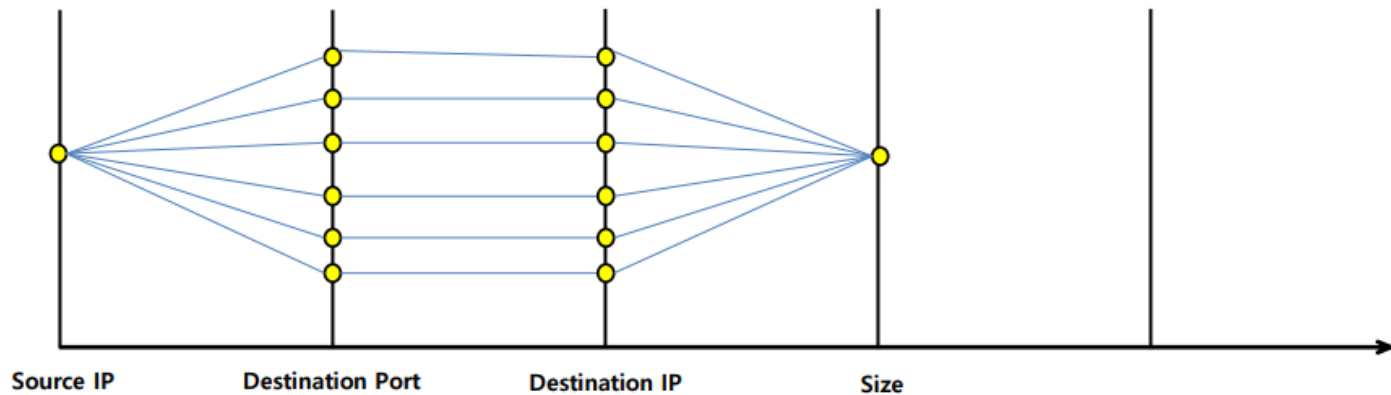


 : Port scanning (for single target)
ex) Nmap -sT 143.248.1.1



보안 시각화

■ Example



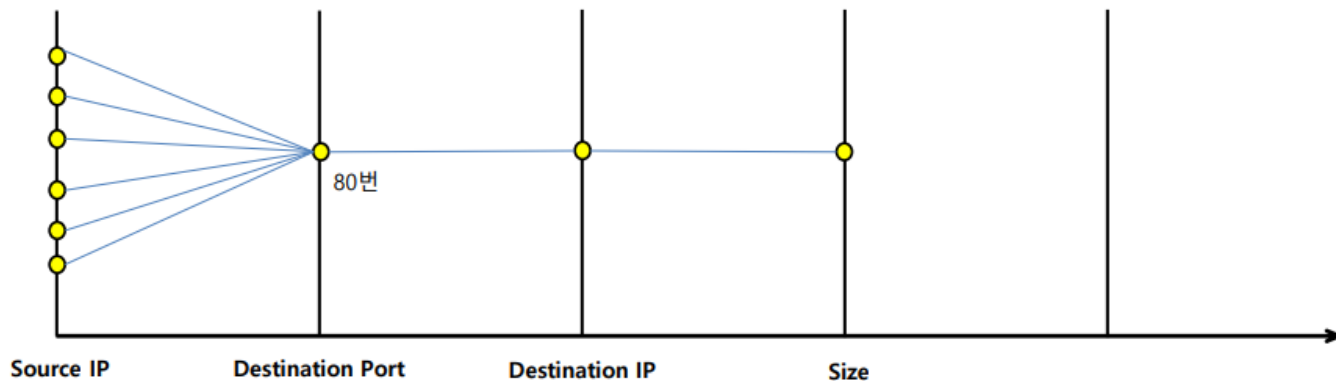
: Port scanning (for many IP)


ex) Nmap -sT 143.248.1.1/24



보안 시각화

■ Example

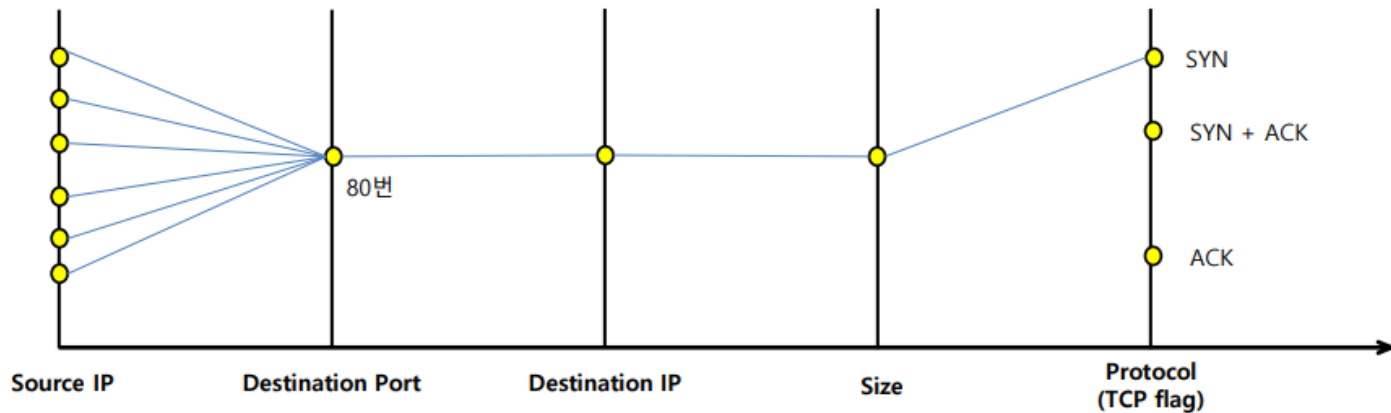



 : DoS attack (web attack)



보안 시각화

■ Example

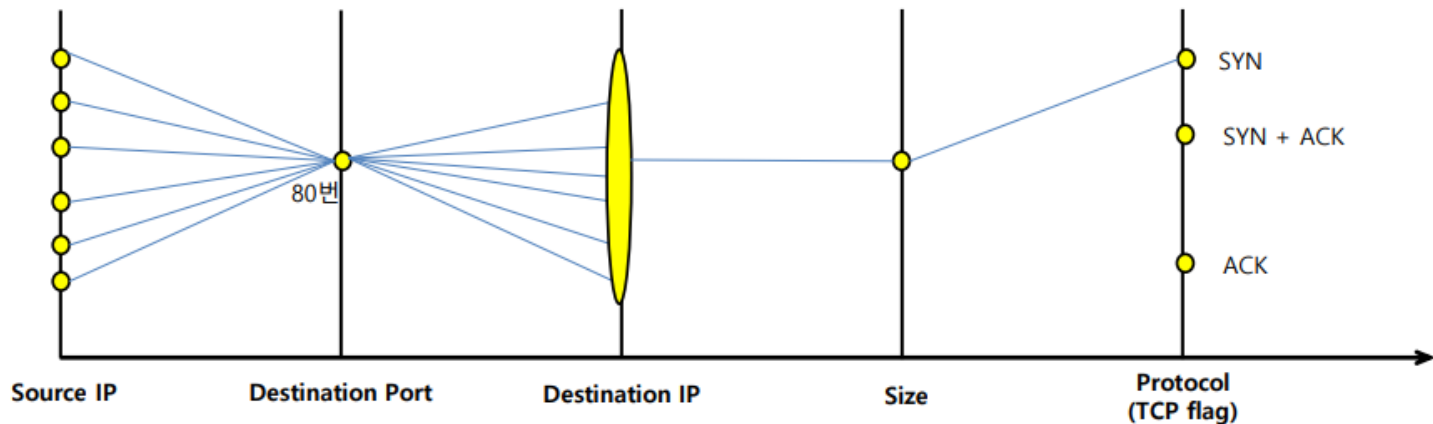


 : DoS attack (web attack)
(if port == 80 AND flag == SYN)
HTTP SYN Flooding attack



보안 시각화

■ Example



: DDoS attack



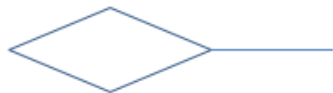
보안 시각화

■ 시각화를 통한 장점

- 많은 양의 정보를 종합적으로 해석할 수 있음
- 시각화된 결과를 인식하는데 있어 전문화된 인력이 요구되지 않음
- 빠르게 상황을 인식할 수 있음

■ Signature

- IDS → signature (패턴 DB와의 비교)
 - 상황별 고유한 signature가 나와야 오탐을 줄일 수 있음
- Visual signature (시각적 패턴)
 - 상황별 고유한 도형(시각적 패턴)이 나와야 오탐을 줄이고 상황인식을 정확히 할 수 있음



< port scanning >



< massive scanning >



< DDoS >



보안 시각화

■ Signature 정교화 작업

- DDoS와 대규모 이벤트로 인한 대량 웹 접속의 구분 을 더 정교하게 해야 할 때, size 축을 하나 더 두어서 해결



: DDoS



: Web Request



보안 시각화

■ Signature 정교화 작업

- 정보(factor) 에는 도형 모양, 두께 등이 포함
 - 축 같은 정보 보다 (nominal 한 추가정보) 색상이나 두께를 더 직관적으로 인식



If average_traffic_size
> 1kbytes
→ **web event**



If average_traffic_size
< 40 bytes
→ **DDoS**



로그분석 툴

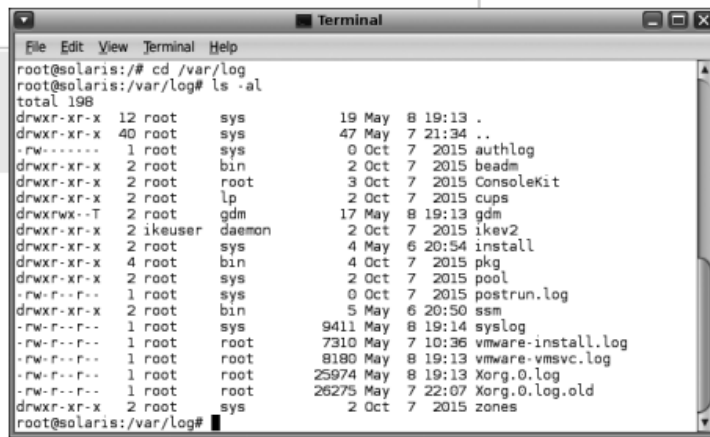
로그 분석

■ HIDS에 필요한 정보

- 네트워크 데이터 (네트워크 관련 로그, 패킷 모니터링) + 시스템 로그 데이터
- 리눅스의 주요 로그 디렉토리 위치

경로	적용 시스템
/usr/adm	초기 유닉스, BSD 계열 : HP-UX 9.X, SunOS 4.x
/var/adm	최근 유닉스, SVR 계열 : 오라클 솔라리스, HP-UX 10.x 이후, IBM AIX
/var/log	일부 BSD 계열 : BSD, FreeBSD, 오라클 솔라리스, 리눅스
/var/run	일부 리눅스

```
cd /var/log
ls -al
```



```
root@solaris:~# cd /var/log
root@solaris:/var/log# ls -al
total 198
drwxr-xr-x 12 root  sys      19 May  8 19:13 .
drwxr-xr-x 40 root  sys      47 May  7 21:34 ..
-rw-r--r--  1 root  sys         0 Oct  7 2015 authlog
drwxr-xr-x  2 root  bin         2 Oct  7 2015 beadm
drwxr-xr-x  2 root  root        3 Oct  7 2015 ConsoleKit
drwxr-xr-x  2 root  lp          2 Oct  7 2015 cups
drwxrwx--T  2 root  gdm       17 May  8 19:13 gdm
drwxr-xr-x  2 ikeuser daemon    2 Oct  7 2015 ikev2
drwxr-xr-x  2 root  sys         4 May  6 20:54 install
drwxr-xr-x  4 root  bin         4 Oct  7 2015 pkg
drwxr-xr-x  2 root  sys         2 Oct  7 2015 pool
-rw-r--r--  1 root  sys         0 Oct  7 2015 postrun.log
drwxr-xr-x  2 root  bin         5 May  6 20:50 ssm
-rw-r--r--  1 root  sys      9411 May  8 19:14 syslog
-rw-r--r--  1 root  root     7310 May  7 10:36 vmware-install.log
-rw-r--r--  1 root  root     8180 May  8 19:13 vmware-vmxvc.log
-rw-r--r--  1 root  root    25974 May  8 19:13 Xorg.0.log
-rw-r--r--  1 root  root   26275 May  7 22:07 Xorg.0.log.old
drwxr-xr-x  2 root  sys         2 Oct  7 2015 zones
root@solaris:/var/log#
```

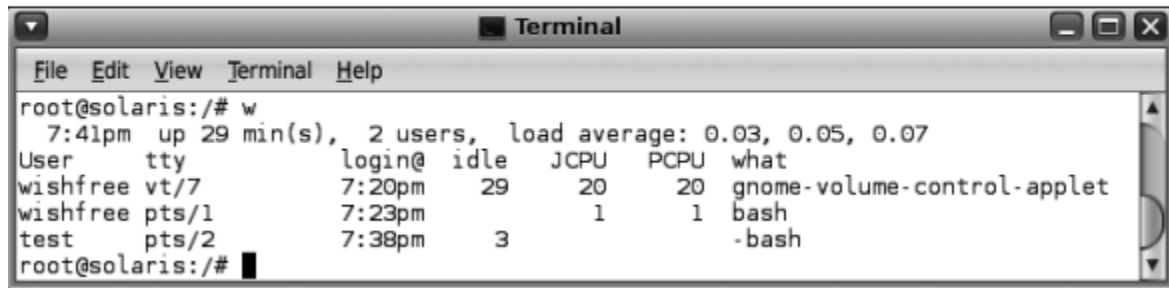


로그 분석

■ utmp(x) 로그

- utmp 데몬 : utmp(x) 파일에 로그 남기는 프로그램
- 리눅스의 가장 기본적인 로깅을 제공하는 데몬(/etc/lib/utmpd) 현재 시스템에 로그인한 사용자의 상태 출력
- utmp 데몬에 저장된 로그를 출력하는 명령 : w, who, users, whodo, finger 등
- w 명령 : 현재 시스템에 로그인된 사용자 계정과 로그인 셸 종류, 로그인 시간, 실행 중인 프로세스의 종류

```
W
```



```
root@solaris:/# w
 7:41pm up 29 min(s),  2 users,  load average: 0.03, 0.05, 0.07
User      tty          login@   idle   JCPU   PCPU   what
wishfree  vt/7         7:20pm   29     20     20    gnome-volume-control-applet
wishfree  pts/1        7:23pm    1      1      1    bash
test      pts/2        7:38pm    3      1      1    -bash
root@solaris:/#
```

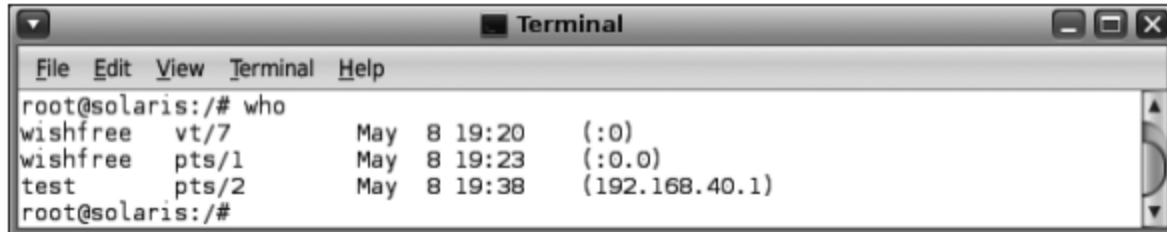


로그 분석

■ utmp(x) 로그

- who 명령 : 접속한 시스템의 IP 확인

```
who
```



A terminal window titled "Terminal" showing the output of the 'who' command. The prompt is 'root@solaris:/#'. The output lists three users: 'wishfree' on 'vt/7' at 'May 8 19:20' with IP '(:0)', 'wishfree' on 'pts/1' at 'May 8 19:23' with IP '(:0.0)', and 'test' on 'pts/2' at 'May 8 19:38' with IP '(192.168.40.1)'. The prompt returns to 'root@solaris:/#'.

User	Terminal	Date	Time	IP Address
wishfree	vt/7	May 8	19:20	(:0)
wishfree	pts/1	May 8	19:23	(:0.0)
test	pts/2	May 8	19:38	(192.168.40.1)

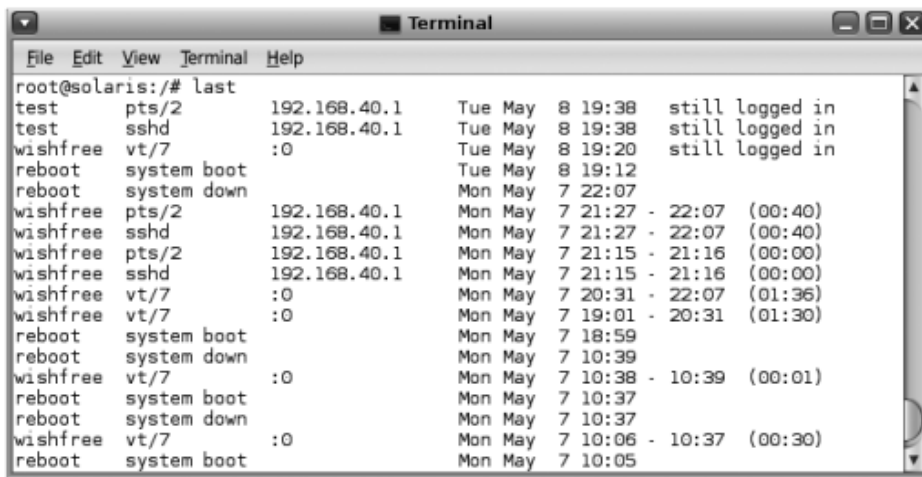


로그 분석

■ wtmp(x) 로그

- wtmp 데몬: wtmp(x) 파일에 로그 남김, /usr/include/utmp.h 파일 구조체 사용
- utmp 데몬과 비슷한 역할, 사용자들의 로그인, 로그아웃, 시스템 재부팅 정보 수록
- last 명령 이용 확인

last



```
root@solaris:~# last
test pts/2 192.168.40.1 Tue May 8 19:38 still logged in
test sshd 192.168.40.1 Tue May 8 19:38 still logged in
wishfree vt/7 :0 Tue May 8 19:20 still logged in
reboot system boot Tue May 8 19:12
reboot system down Mon May 7 22:07
wishfree pts/2 192.168.40.1 Mon May 7 21:27 - 22:07 (00:40)
wishfree sshd 192.168.40.1 Mon May 7 21:27 - 22:07 (00:40)
wishfree pts/2 192.168.40.1 Mon May 7 21:15 - 21:16 (00:00)
wishfree sshd 192.168.40.1 Mon May 7 21:15 - 21:16 (00:00)
wishfree vt/7 :0 Mon May 7 20:31 - 22:07 (01:36)
wishfree vt/7 :0 Mon May 7 19:01 - 20:31 (01:30)
reboot system boot Mon May 7 18:59
reboot system down Mon May 7 10:39
wishfree vt/7 :0 Mon May 7 10:38 - 10:39 (00:01)
reboot system boot Mon May 7 10:37
reboot system down Mon May 7 10:37
wishfree vt/7 :0 Mon May 7 10:06 - 10:37 (00:30)
reboot system boot Mon May 7 10:05
```

last 명령 실행하여 시스템에 로그인한 사용자의 최근 목록 확인

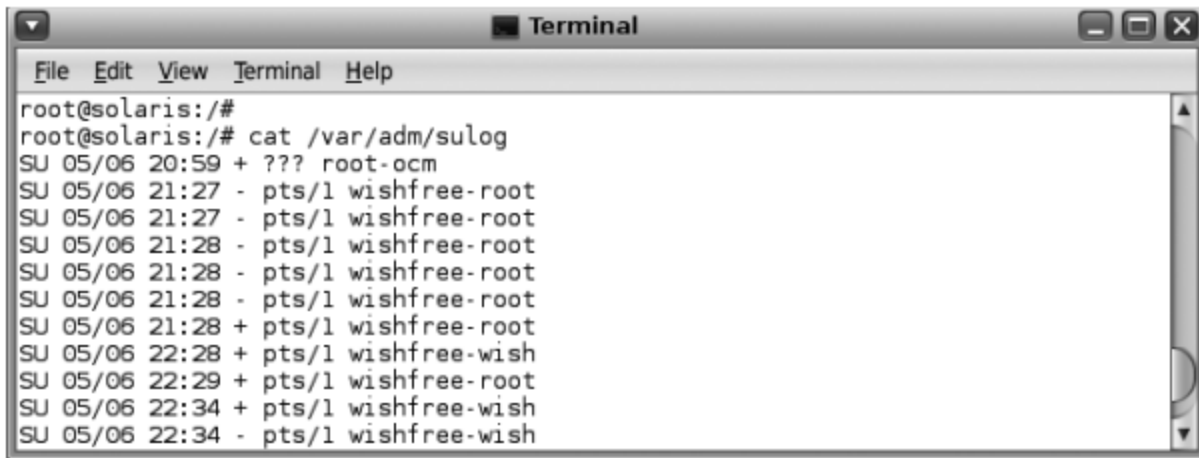


로그 분석

■ su 로그

- su(switch user)는 권한 변경에 대한 로그
- 출력 형식 : [날짜][시간][+(성공) or -(실패)] [터미널 종류][권한 변경 전 계정 - 변경 후 계정]
- su 로그에 대한 설정 파일 : /etc/default/su

```
cat /var/adm/sulog
```



A terminal window titled "Terminal" with a menu bar (File, Edit, View, Terminal, Help). The terminal shows the command `cat /var/adm/sulog` and its output, which lists su login attempts. The output shows a successful login for root-ocm and several failed attempts for wishfree-root and wishfree-wish.

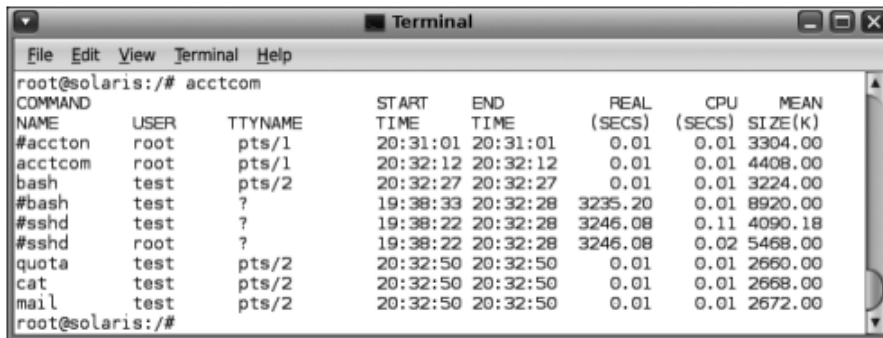
```
root@solaris:/#  
root@solaris:/# cat /var/adm/sulog  
SU 05/06 20:59 + ??? root-ocm  
SU 05/06 21:27 - pts/1 wishfree-root  
SU 05/06 21:27 - pts/1 wishfree-root  
SU 05/06 21:28 - pts/1 wishfree-root  
SU 05/06 21:28 - pts/1 wishfree-root  
SU 05/06 21:28 - pts/1 wishfree-root  
SU 05/06 21:28 + pts/1 wishfree-root  
SU 05/06 22:28 + pts/1 wishfree-wish  
SU 05/06 22:29 + pts/1 wishfree-root  
SU 05/06 22:34 + pts/1 wishfree-wish  
SU 05/06 22:34 - pts/1 wishfree-wish
```



로그 분석

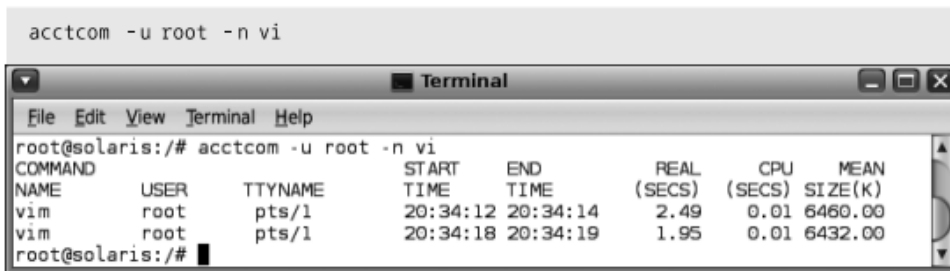
■ pacct 로그

- 시스템에 로그인한 모든 사용자가 수행한 프로그램에 대한 정보 저장하는 로그
- acctcom 명령 실행을 통해 확인 가능



NAME	USER	TTYNAME	START TIME	END TIME	REAL (SECS)	CPU (SECS)	MEAN SIZE(K)
#accton	root	pts/1	20:31:01	20:31:01	0.01	0.01	3304.00
acctcom	root	pts/1	20:32:12	20:32:12	0.01	0.01	4408.00
bash	test	pts/2	20:32:27	20:32:27	0.01	0.01	3224.00
#bash	test	?	19:38:33	20:32:28	3235.20	0.01	8920.00
#sshd	test	?	19:38:22	20:32:28	3246.08	0.11	4090.18
#sshd	root	?	19:38:22	20:32:28	3246.08	0.02	5468.00
quota	test	pts/2	20:32:50	20:32:50	0.01	0.01	2660.00
cat	test	pts/2	20:32:50	20:32:50	0.01	0.01	2668.00
mail	test	pts/2	20:32:50	20:32:50	0.01	0.01	2672.00

- root 계정으로 vi 에디터 실행한 기록 출력하는 명령



```
acctcom -u root -n vi
```

NAME	USER	TTYNAME	START TIME	END TIME	REAL (SECS)	CPU (SECS)	MEAN SIZE(K)
vim	root	pts/1	20:34:12	20:34:14	2.49	0.01	6460.00
vim	root	pts/1	20:34:18	20:34:19	1.95	0.01	6432.00

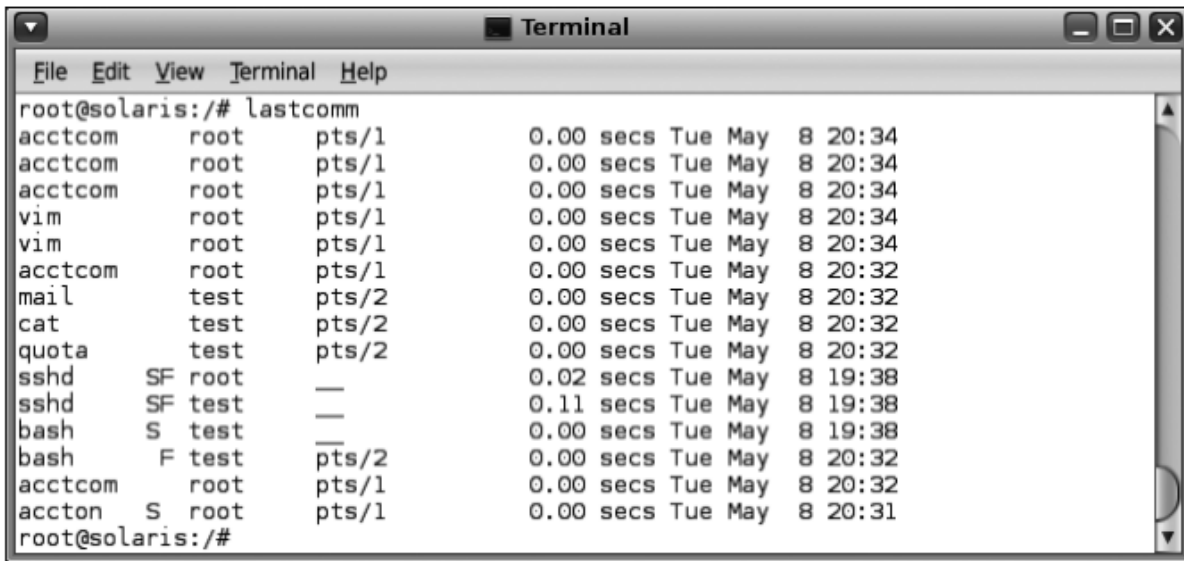


로그 분석

■ pacct 로그

- lastcomm 명령 : 실행된 날짜 출력

```
lastcomm
```



A terminal window titled "Terminal" showing the output of the `lastcomm` command. The output lists various processes, their users, and the terminals they are running on, along with the time they were last executed. The processes include `acctcom`, `vim`, `mail`, `cat`, `quota`, `sshd`, `bash`, and `accton`. The users are `root`, `test`, and `S`. The terminals are `pts/1` and `pts/2`. The times are in the format `HH:MM:SS`.

```
root@solaris:/# lastcomm
acctcom    root    pts/1    0.00 secs Tue May  8 20:34
acctcom    root    pts/1    0.00 secs Tue May  8 20:34
acctcom    root    pts/1    0.00 secs Tue May  8 20:34
vim        root    pts/1    0.00 secs Tue May  8 20:34
vim        root    pts/1    0.00 secs Tue May  8 20:34
acctcom    root    pts/1    0.00 secs Tue May  8 20:32
mail       test    pts/2    0.00 secs Tue May  8 20:32
cat        test    pts/2    0.00 secs Tue May  8 20:32
quota      test    pts/2    0.00 secs Tue May  8 20:32
sshd       SF root    —        0.02 secs Tue May  8 19:38
sshd       SF test   —        0.11 secs Tue May  8 19:38
bash       S test   —        0.00 secs Tue May  8 19:38
bash       F test   pts/2    0.00 secs Tue May  8 20:32
acctcom    root    pts/1    0.00 secs Tue May  8 20:32
accton     S root    pts/1    0.00 secs Tue May  8 20:31
root@solaris:/#
```



로그 분석

■ Syslog

- 시스템의 로그 정보를 대부분 수집하여 로깅
- 해당 로그의 종류와 로깅 수준은 /etc/syslog.conf 파일에서 확인

■ authlog/loginlog

- loginlog는 실패한 로그인 시도에 대한 로깅 수행
- loginlog 파일에 실패한 로그인 기록이 저장되도록 설정
- 이 설정은 /etc/default/login 파일에 저장, 시스템 재부팅할 때 적용



로그 분석

■ 시스템별 로그 상세 경로

로그 파일	리눅스(레드햇)	솔라리스	HP-UX (10.x 이상)	IBM-AIX
utmp, wtmp	/var/run(utmp) /var/log(wtmp)	/var/adm	/var/adm	/var/adm
utmpx, wtmpx	존재하지 않음	/var/adm	존재하지 않음	존재하지 않음
btmp	/var/log	존재하지 않음	/var/adm	존재하지 않음
syslog	존재하지 않음	/var/log	/var/adm/syslog/syslog.log	/var/adm
secure	/var/log	존재하지 않음	존재하지 않음	존재하지 않음
sulog	존재하지 않음	/var/adm	/var/adm	/var/adm
pacct	/var/log	/var/adm	/var/adm	/var/adm
authlog	존재하지 않음	/var/log	존재하지 않음	존재하지 않음
messages	/var/log	/var/adm	/var/adm	/var/adm
loginlog	존재하지 않음	/var/adm	존재하지 않음	존재하지 않음
lastlog	/var/log	/var/adm	/var/adm	/etc/security
access_log	/var/log/httpd	/var/log/httpd	/usr/local/etc/httpd/logs	/usr/local/etc/httpd/logs
error_log	/var/log/httpd	/var/log/httpd	/usr/local/etc/httpd/logs	/usr/local/etc/httpd/logs
shutdownlog	존재하지 않음	존재하지 않음	/etc/shutdownlog	존재하지 않음
failedlogin	존재하지 않음	존재하지 않음	존재하지 않음	/etc/security



로그 분석

■ 그 외 사용 가능한 속성

- CPU 사용량 / 비율 / 시간
- 메모리 사용량 / 비율 / 시간
- 네트워크 사용량 / 비율 / 시간
- 디스크 사용량 / 비율 / 시간
- 프로세스별 사용 시간 및 빈도
- ...



로그 분석 - 보안 시각화

■ Wtmp를 보안시각화로 적용할 경우

- wtmp (UNIX / Linux log) && pacct (모든 명령어 기록; lastcomm)
 - who, when, where, how (FTP, ...) 얼마동안 세션을 맺었다.
 - → where (회사 내부 : 파란색 / 회사 외부로부터 : 빨간색)
 - → when (업무 시간 이외 : 빨간색 / 업무시간 내 : 파란색)
 - R, F, I, M (요약테이블)
 - R (최근 로그인 시간)
 - F (로그인 빈도)
 - I (로그인 인터벌)
 - M (로그인 횟수)
 - ex)
 - 1시간 동안 로그인을 100번 vs. 3달 동안 로그인을 100번
 - 한번 로그인 하고, 다음 로그인까지 평균 16시간 간격이 있는 사람 vs. 한번 로그인과 다음 로그인 사이 평균 3분 간격이 있는 사람



로그 분석 - 패킷

■ 네트워크 환경에서의 대용량 패킷 분석

- 전체적인 프로토콜 분포
- 트래픽을 많이 발생시킨 IP 관련 통계
- HTTP와 같은 요청 및 응답에 대한 통계

■ 대용량 패킷을 분석하는데 있어서 일정 통계 값을 통한 빠른 상황 인지가 가능함

- 하지만, 보다 빠른 의사 결정을 위해서는 그래프 형태로써 시각적으로 인지하기 쉽게 표현하는 것이 필요함



로그 분석

■ 시스템 이벤트 로그

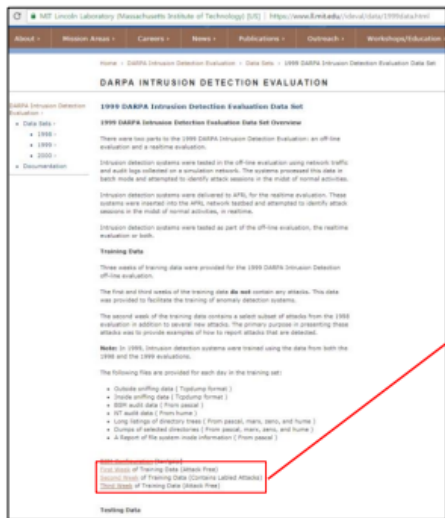
- Window 시스템에서 시스템로그가 이벤트 형식으로 관리됨
 - 이벤트 로그 : 응용프로그램 로그, 보안 로그, 시스템 로그, 디렉토리 서비스 로그, 파일 복제 서비스 로그, DNS 서버 로그 등 (기본적으로 응용프로그램, 보안, 시스템 로그 이벤트를 기록)

이벤트 ID - Type1	이벤트 ID - Type2	내용
624	4720	사용자 계정 생성
625		사용자 계정 유형 변경
627	4723	암호 변경 시도
628	4724	사용자 계정 암호 설정
630	4726	삭제된 사용자 계정
680	4776	로그인 성공 정보
681	4777	로그인 실패 정보
528	4624	성공적인 로그인
529	4625	잘못된 암호를 이용한 로그인 시도



■ DARPA Dataset

□ Attack-Free 상태의 Dataset



DARPA INTRUSION DETECTION EVALUATION

DARPA Intrusion Detection Evaluation >

- Data Sets >
 - 1998 >
 - 1999 >
 - 2000 >
- Documentation

1999 Training Data - Week 1

The simulation network normally collected data twenty-two hours a day. The topology program was used to examine the outside topology data files and the actual times of the first and last packet were extracted. These times are shown below. During the first week of training data the simulation network did not experience any unscheduled down time.

First Packet Time			Last Packet Time		
Mon	Mar 1	08:00:02	Tue	Mar 2	06:00:02
Tue	Mar 2	08:00:02	Wed	Mar 3	06:00:01
Wed	Mar 3	08:00:03	Thu	Mar 4	06:00:01
Thu	Mar 4	08:00:03	Fri	Mar 5	06:00:02
Fri	Mar 5	08:00:02	Sat	Mar 6	06:00:02

Monday

outside.treedump.data	159,432 kb gzipped
inside.treedump.data	165,264 kb gzipped
Solaris BSM audit.data	5,599 kb gzipped
NT.audit.data	4,501 kb tarred & gzipped
Selected directory dumps	3,118 kb tarred & gzipped
File system listing & inode record	6,851 kb tarred & gzipped

Tuesday

outside.treedump.data	155,620 kb gzipped
inside.treedump.data	163,009 kb gzipped
Solaris BSM audit.data	2,869 kb gzipped
NT.audit.data	82,712 kb tarred & gzipped
Selected directory dumps	2,482 kb tarred & gzipped
File system listing & inode record	6,347 kb tarred & gzipped

Wednesday

outside.treedump.data	180,585 kb gzipped
inside.treedump.data	186,631 kb gzipped
Solaris BSM audit.data	2,238 kb gzipped
NT.audit.data	306 kb tarred & gzipped
Selected directory dumps	3,057 kb tarred & gzipped
File system listing & inode record	6,758 kb tarred & gzipped

Thursday



THANK

YOU