Scanning Attack 에 대한 방어기법 연구

김기훈*, 김승주*, 천영창**
*성균관대학교 전자전기컴퓨터 공학과
**삼성전자㈜ 정보통신총괄
e-mail : mykendo@gmail.com

A Study on Scanning Attack Defense System

Ki-Hoon Kim*, Seung-Joo Kim*, Young-Chang Cheon**
*Dept. of Electorial and Computer Engineering, Syung Kyun Kwan University
**Telecommunication R&D Center, Samsung Electronics

요 약

Scanning 은 불특정 사용자가 특정 시스템 혹은 네트워크에 대해 Dos/DDos Attack 을 하기에 앞서 취약성이 존재할 수 있는 서비스나 호스트를 발견하기 위해 취해지는 선행 기술 중 하나이다. 본 논문에서는 기존에 알려진 대표적인 Port Scanning 기법들에 대해 알아보고 기존에 널리 사용되고 있는 탐지 및 방어 기법과 이러한 방식이 가질 수 있는 문제점에 대해서도 살펴본 후, 이를 보완할 수 있는 기법을 논의하고자 한다.

1. 서론

근래의 인터넷은 다양한 시스템과 Host 들이 다양한 종류의 네트워크와 시스템 서비스를 제공하며 서로 혼재하여 복잡하게 얽혀서 구성되고 있다. 이렇게 다양한 서비스가 가능한 환경은 보다 편리하고, 다양한 기능을 가지게 될 수 있는 반면, 이렇게 다양한 서비스에 필요한 네트워크 보안 역시 그에 보조를 맞추어 발전하지 않는다면, 필연적으로 네트워크 환경 혹은 시스템의 취약성을 더욱 증가시킬 수 있는 토양이 될 수 있을 것이다. 실제로 네트워크 혹은 시스템에 잠재되어 있는 보안 취약성을 바탕으로 다수의 웜 / 바이러스 / DoS 공격 등이 인터넷이라는 편리함 속에서 보안 취약성을 가진 채 퍼져나가고 있는 것이 현실이다. 본 논문에서는 악의적인 사용자(Attacker)가 시스템 혹은 네트워크에 대해 악의적인 행위를 하기에 앞서 취약한 호스트 혹은 서비스를 알아내기 위해 빈번하게 수행하는 가장 일반적인 방법론 중의 하나인 Scanning 시도/공격에 대해 알아 볼 것 이다. 다양한 Scanning 기법 중 특히 Port Scan Attack 에 대해 알아보고, 네트워크 디바이스인 방화벽 및 IDS(Intrusion Detection System) 관점에서의 제안할 수 있는 Scanning 방어 기법에 대해 살펴 볼 것이다. 본문의 2 장에서 Port Scanning Attack 을 정의하고, 기존에 제시된 대표적인 탐지 및 방어 방식을 살펴볼 것이고, 3 장에서는 기존에 제시된 방어 방식의 약점에 대해 알아보고, 실제 네트워크에서 이러한 공격에 보다 능동적으로 대처할 수 있는 기법에 대해 논의할 것이다. 마지막으로 4 장에서는 실제 적용 가능성과 발전방향에 대해 살펴본다.

2. Port Scanning 공격과 방어

Port Scanning 은 공격자가 실제 시스템에 대한 공격을 수행하기에 앞서 공격이 가능하거나 취약성이 있는 시스템의 부분을 알아내고 그에 대한 정보를 알아내기 위해 수행하는 과정이다. 어떠한 서비스가 실행되고 있고, Anonymous 사용자가 접근할 수는 있는지, 적절한 Authentication 과정을 수행하는지 등을 이를 통해 밝혀 낼 수 있다. 그 외에도 Port Scanning 은 시스템의 자원 낭비를 발생시킬 뿐만 아니라, 유해 트래픽 발생으로 인해 네트워크 혼잡 발생을 유발 할 수 도 있다. [1] 다음은 기존에 제기된 대응책을 탐지와 방어(Block) 라는 관점에서 간략히 살펴보겠다.

2.1 Port Scanning 탐지

Port Scanning 을 탐지 하기 위해서는 Scanning 의 특징을 통한 분석이 선행되어야 할 것이다. 일반적으로 Scanning 을 그 특징을 기준으로 분류해 보면, 하나의 호스트를 향해 다수의 목적지 주소에 대한 Port Scanning 을 시도하는 Vertical Scanning 과 다수의 호스트를 향해 하나의 목적 포트만을 탐색하는 Horizontal Scanning, 그리고, 이 둘을 혼합한 방식으로 나눌 수 있다.[3] 이러한 Scanning 을 탐지하는 것은 하나의 Source 주소를 갖는 세션에 대해 특정 시간 영역상에서 정의된 임계 세션을 기준으로 판단하게 된다. 이는 주로 네트워크의 관문 역할을 하는 방화벽 / IDS (Intrusion Detection System)에서 수행되는데, 먼저 정상 트래픽의 베이스 라인을 어떠한 방식으로든 설정한 이후, 특정 Time Interval 내에서의 비정상적인 트래픽 Behavior 가 감지될 때 마다 이를 탐지하고 방어하는 방식으로, 이는 Scalable Attack 을 탐지하는데 사용되는 대표적인 방식 중 하나이다.[4] 이런 Behavior 에 기초한 Anomaly Detect 방식은 우선, 현재와 같이 다양

한 형태의 서비스가 제공되는 인터넷 환경에서 정상 트래픽의 기준이 애매모호하므로, 그에 대한 베이스라인을 설정한다는 것이 매우 어려울뿐더러, 항상 트래픽의 각 플로우에 대한 상태 정보를 관리하고 있어야 한다는 부담감으로 인해 발생하는 성능저하의 문제점에서 벗어날 수 없다.[4][5] 하지만, Traffic Behavior 를 판단근거로 하는 anomaly detection 은 Scalable Attack 의 대표적인 탐지방식으로 사용되고 있다.

2.2 Port Scanning 방어

현재 인터넷에서 일어나고 있는 Port Scanning 의 종류는 Vanilla TCP Connect Scan, Half Open Scan, FIN Scan, XMAS Scan, TCP Null Scan, ACK Scan 등의 TCP 의 포트를 Scanning 하는 Attack 과 UDP ICMP Port Scanning 등의 UDP Port Scanning 등 다양한 방식이 이용된다.[1] 이러한 Scanning 은 실제로 NMAP[2]이나 Nessus 등의 취약성을 점검하는 도구에서 현재 사용되고 있는 대표적인 공격방식이다. 이러한 다양한 형태의 Port Scanning 시도에 대해서 이를 막을 수 있는 정형화된 방법은 아직까지 존재하지 않는다. 하지만, Port Scanning 에 대해 피해를 최소화하기 위해 다음과 같은 방어 방식이 제안된다. 우선, Port Scanning 시도에 노출되는 시스템의 정보가 최소한이 되도록 시스템의 불필요한 서비스 항목에 대해서는 항상 체크하여 비활성화될 수 있도록 한다. 만약 Default Installation 으로 시스템을 설치하고 운용할 경우 기본적으로 다양한 서비스 제공을 위해 많은 포트가 열려있게 된다. 이를 우선적으로 제거하는 것은 최선의 방어가 될 수 있다. 다음으로 “TCP Wrapper”[6] 기능을 통해 원하는 Scanning 방어를 수행할 수 있다. 접속하고자 하는 호스트에 대해 허용 혹은 거부 리스트를 미리 설정하고, 리스트 검색을 통해 허용되지 않은 사용자의 접근을 제어할 수 있다. 또한 대부분의 Scanning 이 TCP 를 기반 프로토콜로 하고 행해지고 있음[3]을 고려해 볼 때 TCP 연결기반의 서비스를 감시하고 필터링 할 수 있는 TCP Wrapper 의 기능은 허가 없는 악의적인 호스트의 Port Scanning 시도에 적절히 대응할 수 있게 한다. 하지만, 오늘날처럼 복잡하고 다양한 인터넷 환경에서 누구를 허용하고, 허용치 않을지를 미리 정할 수 있느냐는 것과, Spoofing 을 더한 공격에 대한 Tcp Wrapper 의 취약점은 한계가 될 수 있다.[6] 마지막으로 제시할 수 있는 또 다른 방법은 2.1 장에서 설명했던, anomaly detection 을 이용해서 트래픽의 비정상 Behavior 를 탐지하고, 그에 기초하여 Scanning 호스트를 리스트 형식으로 업데이트하고, 이를 TCP Wrapper 의 접근제한 기능과 연동하여 사용하는 방식이 있다. 이는 보다 능동적인 Port Scanning 에 대한 방어가 될 수 있다. 실제로 이러한 형태의 접근제한은 근래에 많이 사용되고 있다.

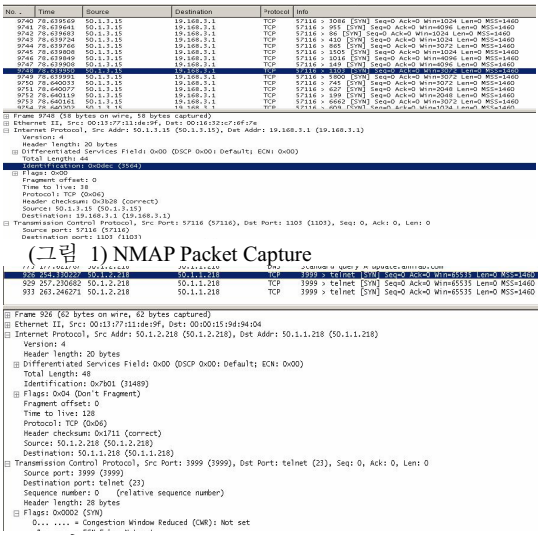
3. 능동 사전 방어 시스템

3.1 기존의 Port Scanning 방어 시스템

지금까지 Port Scanning 에 대한 탐지 및 방어방식에 대해 살펴보았는데, 이러한 방식은 주로 트래픽의 행위가 일어난 이후에서야 그를 탐지 할 수 있고, 이후 적절한 대처가 가능하다는 것을 보여주었다. 이는 Scanning 과 다른 형태의 Scalable Attack, 예를 들어, 일반적인 Scalable DoS Attack (Syn Flooding, Udp Flooding 등)에 대해서는 큰 문제가 되지 않을 수 있다. 왜냐하면, 어느 정도의 트래픽 유입을 바탕으로 그 트래픽의 특징 행위를 조사한 뒤 문제상황이라고 판단될 시 이를 탐지하고, 그 후 어떠한 조치를 취하게 될지라도, 이는 적은 시간 동안의 트래픽 유입이기 때문에 큰 문제가 될 수 없기 때문이다. 하지만, Scanning Attack 은 일반적인 DoS Attack 처럼 시스템/네트워크의 서비스 장애가 목적이 아니라, 목적이 되는 장비 혹은 네트워크의 응답을 구하는 것이 목적이 된다. 그러므로, 하나의 질의 패킷 이라도 그들이 원하는 시스템 혹은 네트워크로 유입되어 그에 대한 응답을 얻게 된다면, 이미 공격자는 소기의 성과를 달성한 것이 되는 것이다. 이러한 관점에서 볼 때 어떠한 Traffic 이 비정상 Behavior 보일 때 탐지하고, 그 이후 어떠한 방어조치를 취한다는 것은 트래픽의 유입을 허가한다는 것인데, 이는 너무 늦은 대응이 된다. 이러한 사후대응방식은 어떠한 Source IP 주소를 갖는 공격자가 어떤 일을 했는지 등에 대한 탐지 정보는 얻을 수 있으나, Scanning Attack 을 막을 수는 없다. 그러므로, Scanning Attack 에 대한 적절한 방어방식은 사후 방어가 아니라, 사전 방어방식이 되어야만 그 효과를 기대할 수 있을 것이다. 앞서 언급했던 Tcp Wrapper 를 이용하여 사전에 Deny/Allow list 를 만들어 방어하는 것도 비슷한 사전방어효과를 기대할 수 있으나, 이는 Spoofing 에 대처할 수 없는 등 여전히 문제점을 안고 있다.

3.2 Port Scanning Pre-Defense

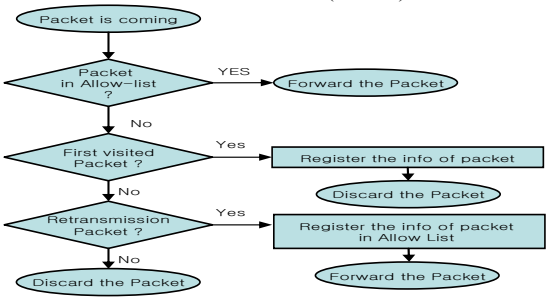
일반적인 서비스를 수행하는 정상적인 Packet 의 경우, 네트워크 혹은 Device 상황에 의해 Packet 이 유실되었을 때를 대비하여 재전송(Retransmission)이라는 메커니즘을 가지고 있다. TCP 의 경우, Reliable 한 Date 전송을 위해 RTT(Round-Trip Time)를 이용한 Timeout 을 통해 자체적인 재전송 메커니즘을 가지고 있고[7], UDP 는 자체적인 재전송 메커니즘을 지원하지 않는 Unreliable 한 프로토콜이지만, 각 Application Level 에서 서버에서 응답이 없을 경우를 대비한 Query 재전송 메커니즘을 대체로 사용하고 있다. 하지만 Scanning Attack 을 시도하는 Tool 의 경우 Packet 의 유실이 발생할 경우, 이에 대한 재전송 메커니즘은 없고, Target 시스템에서 응답이 없음이라고만 판단하게 된다. (그림 1)은 실제 NMAP[2]이 Scanning 을 시도하였을 때의 패킷을 캡쳐한 화면이고, 재전송 시도는 없음을 보여준다. (그림 2)는 정상 클라이언트가 서버로부터의 응답이 없을 때 Packet 재전송이 일어나는 것을 보여준다.



(그림 1) NMAP Packet Capture

(그림 2) Normal Service Capture

이러한 차이점을 이용하여, 정상적인 서비스 시도와 Scanning Attack 시도를 구분할 수 있고, 이를 Tcp Wrapper 기능과 연동한다면, 허용 List 를 보다 능동적으로 작성할 수 있다. 예를 들어 처음 전송되는 질의 Packet 에 대해서는 Scanning 시도 인지 일반 클라이언트의 정상 질의 시도인지 판단할 수 없으므로, 일단 모든 페킷을 폐기 처리한 이후, 방어 시스템 내부에 폐기 처리되었던 Packet 에 대한 Source IP 주소 및 Destination Port 번호, Sequence Number 등의 필요한 정보를 저장한다. 정당한 클라이언트라면, Packet 전송 이후 아무런 응답을 받지 못했을 때, Timeout 이 발생하게 되고, 이를 Packet 유실이라고 판단한 후, 재전송 Packet 을 보낼 것이다. 그 재전송 Packet 이 방어 시스템으로 다시 유입 되었을 때, 처음에 유입되었을 때는 폐기 처리되었지만, 기 저장되어 있던 Packet 의 정보와 비교하여 이것이 정당한 재전송 Packet 이라는 것을 판단할 수 있고, 이를 Tcp Wrapper 의 허용 List 에 등록하면서, Packet 전송을 허가할 수 있다. 만약 Scanning 시도였다면, (그림 1)에서 볼 수 있듯이 재전송 시도 없이 종료할 것이므로, Scanning Query Packet 은 먼저 폐기되게 되어 목적 시스템으로 유입자체가 불가능하게 된다. 즉, Scanning 시도는 아무런 응답을 얻지 못하여 관련 정보 획득 없이 종료되게 되는 것이다. 이를 간단히 도식화 하면 (그림 3)와 같다.



(그림 3) Scanning 시도에 대한 사전 방어 시스템

4. Conclusion

앞서 살펴본 바와 같이 Port Scanning 은 기본적으로 어떠한 목적이 되는 시스템의 응답을 바탕으로 그 취약성 정보 파악이 목적이 된다. 이러한 공격은 기존에 알려진 Scalable Attack 의 대표적인 방어방식인 Traffic Behavior 를 바탕으로 하는 사후 방어방식과는 다른 접근이 필요함을 확인하였다. Scanning 공격은 Scanning 패킷의 유입자체를 막아 공격자의 목적 달성을 막는 방어방식을 사용해야 한다. 3 장에서 언급했던 패킷 재전송을 이용한 Scanning 공격의 방어는 Scanning 의 요청시도가 유입되는 것 자체를 막을 수 있는 사전 방어방식의 하나의 예가 될 수 있다. 물론, 본 논문에서 예를 든 방어기법은 TCP 의 RST 나 FIN Flag 를 이용한 공격[1]같이 모든 Scanning Attack 에 공통적으로 적용 될 수 있는 것은 아니다. 하지만, 앞서 살펴본 바와 같이 Vanilla TCP Connect Scan, Half Open Scan 같은 TCP 의 SYN 을 사용한 Scanning[1]에는 효과적으로 대응할 수 있다. UDP 를 이용한 Scanning 역시 Application 별로 구분하여 각각의 재전송 방식을 이용하여 적용할 수 있다면, 유사한 방식의 적용도 가능할 수 있다. 날이 갈수록 다양해지는 Scanning 공격의 진화에 발맞추어 Scanning Packet 이 유입되기 이전에 미리 Scanning 인지 아닌지 판단하고, 이를 차단할 수 있는 보다 다양한 형태의 방어 기법 진화가 필요한 시점이다.

참고문헌

[1] Roger Christopher. “Post Scanning Techniques and Defense Against Them” (2001)
[2 Fyodor. http://www.insecure.org/nmap
[3] Cynthia Bailey Lee, Chris Roedel, Elena Silenok. “Detection and Characterization of Port Scan Attacks”
[4] Ramana Rao Kompella, Sumeet Singh, George Varghese. “On Scalable Attack Detection in the Network”
[5] Kirill Levchenko, Ramamohan Paturi, George Varghese. “On the Difficulty of Scalably Detecting Network Attacks”
[6] Wietse Venema. “TCPWRAPPER Network monitoring, access control, and booby traps.”
[7] James F..Kurose, Keith W.Ross “Computer Networking – A top-down Approach Featuring the Internet”, 2nd Edition, p228 ~p238