실시간 IP 단편 필터링 모듈 설계 및 구현

최 은 영°*, 김 용 석*, 김 현 성**, 오 재 부*, 유 기 영*

*경북대학교 컴퓨터공학과, ** 경일대학교 컴퓨터공학부

{sionchoi, shadowguys}@infosec.knu.ac.kr, kim@kiu.ac.kr, jboh@aiit.or.kr, yook@knu.ac.kr

Design and Implementation of Real Time IP Fragment Filtering Module

Eun-Young Choi*, Yong-Seok Kim*, Hyun-Sung Kim**, Jae-Bu Oh*, Kee-Young Yoo*

*Department of Computer Engineering, Kyungpook National University

**School of Computer Engineering, Kyungil University

요 약

IP 단편을 이용한 공격은 서비스 거부 공격 외에도 이를 이용하여 패킷 필터링 장비나 네트워크 기반의 침입탐지 시스템을 우회할 수 있어 그 심각성이 크다. 그러나 일부 네트워크 기반의 침입탐지 시스템은 IP 패킷단편 재조합 기능을 제공하지 않기 때문에 IP 단편의 취약점을 이용한 공격을 탐지하지 못한다. 본 논문에서는 IP 조각 패킷을 재조합 하지 않고도 단편과 관련된 헤더정보를 분석함으로써 공격을 실시간으로 탐지해 낼 수 있는 IP 단편 필터링 모듈을 설계하고 구현한다.

1. 서론

네트워크에 대한 부정침입이나 사이버테러의 문제가 심각해짐에 따라 최근 등장하고 있는 정보 시스템 기술이나 정보보호 시스템은 해킹에 대응할 수 있는 각종 방법을 고려하여 개발되고 있다. 그러나 이에 따른 해킹기술 또한 고도로 지능적이고 기술화되어 가고 있다. 이러한 공격기술들은 대규모 단위의 네트워크를 대상으로 하고 있으며 단 하나의 패킷으로도 네트워크 자체를 아예 정지시키거나 파괴해 버릴 수 있는 엄청난 위력을 보이고 있다[1]. 또한 각종 해킹 툴 이나 기법들이 대중화됨에 따라 네트워크 관련 공격들이 점점 다양해지고 있다[2]. 그 중에서도 IP 단편을 이용한 공격은 서비스 거부 공격 외에도 이를 이용하여 패킷 필터링 장비나 네트워크 기반의 침입탐지 시스템을 우회할 수 있어 그 심각성이 크다. 일부 라우터나 침입차단 시스템 그리고 네트워크 기반의 침입탐지 시스템에서 패킷 재조합 기능을 제공하지 않아 불법 사용자가 패킷을 다수의 데이터그램으로 분할하여 공격할 경우 이를 탐지하거나 차단하지 못하는 경우가 있다[5]. 이에 본 논문에서는 IP 단편을 이용한 공격 유형을 알아보고 IP 단편의 취약점을 이용한 공격에 패킷을 재조합 하지 않고도 실시간으로 패킷 헤더정보를 검사함으로써 침입을 탐지 해 낼 수 있는 IP 단편 필터링 모듈을 설계하고 구현하여 그 성능을 평가하였다.

2. IP 단편을 이용한 공격 유형

본 장에서는 이상 패킷을 이용한 서비스 거부 공격과 침입탐지시스템이나 침입차단 시스템의 탐지 룰을 우회하는 공격기술들에 대하여 알아본다[3][4][5].

2.1. Ping of Death, Jolt

Ping of Death 공격은 패킷 단편의 취약점이나 서비스 거부 공격을 이용한 공격 방법이다. 일반적으로 ping은 ICMP 메시지 타입 중 echo 요청과 echo 응답을 이용한다. 이러한 ping을 이용한 공격은 가장 손쉽게 IP 패킷을 전송 할 수 있는 공격 방법으로서 ICMP echo 요청 패킷을 상대방에게 전송 한다. 이때 공격 패킷은 표준에 규정된 길이(65,535) 이상으로 큰 IP 패킷을 전송함으로써 이 패킷을 수신받은 시스템에서 이 비정상 패킷을 처리하지 못하게 함으로써 서비스 거부 공격이 발생되도록 하는 방법이다.

2.2. Teardrop, bonk, New Teardrop

Teardrop 공격은 단편(Fragment)의 재조합 과정의 취약점을 이용한 서비스 거부 공격으로, 서로 중첩되도록 헤더를 조작한 한 쌍의 IP 패킷조각들의 재조합 과정에서 내부 버퍼를 넘치게 함으로써 수행된다. 이 과정에서 버퍼에 복사해 넣어야 할 데이터의 길이값이 음수가 되게 되고 이것을 여러 번 반복하면 시스템이 정지되거나 재부팅 된다.

2.3. 미세한 단편 공격 (Tiny Fragment Attack)

미세한 단편 공격은 최초의 단편을 아주 작게 만들어서 네트워크 침입탐지시스템이나 패킷 필터링 장비를 우회하는 공격이다.

예를 들어 TCP 헤더(일반적으로 20바이트)를 2개의 단편으로 나뉘어질 정도로 작게 쪼개서 목적지 TCP 포트번호가 첫 번째 단편에 위치하지 않고 두 번째 단편에 위치하도록 한다. 패킷 필터링 장비나 침입탐지시스템은 필터링을 위해 포트번호를 확인하는데, 이를 피하기 위하여 공격자는 포트번호가 포함되지 않을 정도로 아주 작게 단편된 첫 번째 단편을 통과시킨다. 또한 포트번호

가 포함되어 있는 두 번째 단편은 침입탐지시스템으로부터 아무런 검사 없이 통과한다. 그 결과 보호되어야 할 목적지 서버에서는 이 패킷들이 재조합되고 공격자가 원하는 포트를 통해 공격할 수 있다.

2.4. 단편 겹치기 공격 (Fragment Overlap Attack)

단편 겹치기 공격은 미세한 단편 공격기법에 비해 좀 더 정교한 공격이다. 공격자는 공격용 IP 패킷을 위해 두 개의 단편을 생성한다. 첫 번째 단편은 패킷 필터링 장비에서 허용하는 HTTP(TCP 80) 포트와 같은 포트번호를 가진다. 그리고, 두 번째 단편에서는 오프셋(Offset)을 아주 작게 조작해서 단편들이 재조합될 때 두 번째 단편이 첫 번째 단편의 일부분을 덮어쓰도록 한다(Overlap). 일반적으로 공격자들은 두 번째 단편의 오프셋을 이용하여 첫 번째 단편의 포트번호가 있는 부분까지 덮어쓴다.

3. IP 단편 필터링 모듈 설계 및 구현

IP 단편을 이용한 공격을 탐지하기 위해서 단편과 관련된 항목들이 정확한지를 검사해야 한다. 침입탐지 우회 공격은 이런 항목들을 제대로 검사하지 않는 점들을 악용하여 이루어진다[6]. 본 논문이 제안한 필터링 알고리즘인 detect_frag는 단편과 관련된 항목들을 저장하여 두고 비교 분석하게 된다. 첫 번째 패킷의 단편과 관련된 헤더정보인 ID(Identification), 헤더길이(Header Length), 전체길이(Total Length), 두 번째 패킷의 오프셋, 출발지 주소(Source Address), 목적지 주소(Destination Address)를 자료구조에 저장한다. 침입탐지 시스템의 속도 및 효율성 향상을 위하여 ID값을 해쉬하여 저장하고 나머지 정보들은 연결리스트를 생성하여 저장하였다. 패킷들은 다양한 경로들을 통해 들어오기 때문에 모든 단편들을 필터링하여 저장하기는 힘들다. 그러나 다행히도 중요한 패킷의 정보는 헤더의 시작부분에 포함하기 때문에 첫 번째 단편과 두 번째 단편만을 필터링 하면 된다[7]. 네트워크 상에서 패킷은 순서없이 무작위로 들어오기 때문에 두 번째 단편을 찾기가 힘들다. 본 논문이 제안한 알고리즘은 들어오는 패킷들의 오프셋 값을 비교하여 최소값을 가진 패킷을 찾음으로써 두 번째 단편을 찾을 수 있다. 다음은 IP단편과 관련된 헤더 정보를 저장하기 위한 구조체를 나타낸 것이다.

```
// IP 단편관련헤더정보 리스트 노드의 정의 부분
typedef struct list_node *list_pointer;
typedef struct list_node {
        u_int32_t saddr;
        u_int32_t daddr;
        u_int16_t id;
        u_int16_t frag_off;
        u_int16_t tot_len;
        list_pointer link;
};
list_pointer ptr = NULL, list[MAX];
```

다음의 IP 단편 필터링 모듈의 알고리즘을 살펴보면 MF 플래그가 켜져 있는 최초의 단편(오프셋 == 0)이 도착하면 ID를 해쉬하여 해당 항목에 단편의 헤더정보를 기록한다. 오프셋이 0이 아닌 패킷이 들어왔을 때는 동일한 ID를 매칭하여 들어온 패킷이 두 번째 단편인지를 확인한다. 두 번째 단편임이 확인이 되면 오프셋이 패킷의 전체길이보다 크기를 비교하여 겹치는 부분이 있다면 단편 겹치기 공격으로 간주하고 적절한 대응을 수행한다. 버퍼 오버플로우(Buffer Overflow)를 방지하기 위하여 패킷들간의 전송되는 시간 간격을 측정하여 두 번째 패킷 전송후 일정시간이 소요된 패킷은 버퍼에서 삭제한다.

```
detect_frag() {
  if (flag == MF && offset == 0) {
     패킷의 ID, 전체길이, 출발지주소, 도착지주소를
저장 ;
  }
  else if (flag == MF && 오프셋 != 0) {
      ID를 검색하여 동일한 것을 찾음 ;
       if (두 번째 단편 오프셋 == null) {
           if ((전체길이/8) > 오프셋) {
               alert 하고 버퍼에서 삭제 ;
            }
           else
             오프셋 최소값으로 저장 ;
       }
       else if (두 번째 단편 오프셋 != null) {
            if (최소값 > 현재 오프셋) {
               if ((전체길이/8) > 오프셋) {
                 alert 하고 버퍼에서 삭제 ;
                 }
              else
                현재 오프셋을 최소값으로 저장 ;
              }
        }
   }
}
```

4. 실험 및 성능평가

4.1. 실험 환경

제안한 IP 단편 필터링 모듈은 Linux Kernel 2.4.7 운영체제와 Intel Pentium-II 333Mhz, 128M 메모리의 컴퓨터에서 구현하였다. 패킷을 수집하기 위해 LibPcap 0.6.2[8]를 이용하였고 공격 시뮬레이션을 위하여 공격용 해킹툴인 Teardrop과 Nmap을 사용하였다.

4.2. 실험 시나리오

제안한 IP 단편 필터링 모듈의 성능을 평가하기 위해 해킹툴을 가지고 다양한 조건에서 공격을 시도해 보았다. 먼저 단일 호스트에서 Teardrop을 이용하여 IP단편 겹치기 공격을 하고 nmap을 사용하여 미세한 IP 단편 공격을 하였다. 그리고 이 두 개의 공격유형을 혼합하여 공격하고 탐지여부를 조사하였다. 또한 여러 곳에서 동시에 공격을 할 경우의 침입탐지여부를 조사하기 위하여 3개의 호스트에서 각각 IP단편 겹치기 공격과 미세한 IP 단편 공격하고 이 두 가지를 혼합하여 동시에 공격을 수

그림 1 시뮬레이션을 위한 시스템 구성

행하였다. 단일 호스트일 경우 30차례에 걸쳐 공격을 시도하였고 멀티 호스트일 경우 각 호스트마다 10차례씩 동시에 공격을 하여 탐지율을 조사하였다.

4.3. 성능 평가

단일 호스트에서 다양한 공격유형으로 공격한 결과 모두 False-positve 오류율 0%에 탐지율 100%를 보여주었다. 그리고 3개의 멀티 호스트에서 공격한 결과 역시 동일한 성능을 보여주었다. 단편을 이용한 서비스 거부 공격을 효율적으로 탐지 할 수 있음을 알 수 있다.

표 1 IP 단편 필터링 모듈 성능평가

공격호스트의 개수	공격 유형	탐지율
Single Host	Tiny Fragmentation attack	100%
	Fragment Overlap attack	100%
	mixed Tiny & Overlap	100%
Multi Host (3개)	Tiny Fragmentation attack	100%
	Fragment Overlap attack	100%
	mixed Tiny & Overlap	100%

그림 2는 단편 겹치기 공격과 미세한 단편 공격을 동시에 수행했을 때 나타난 침입을 탐지한 결과이다.

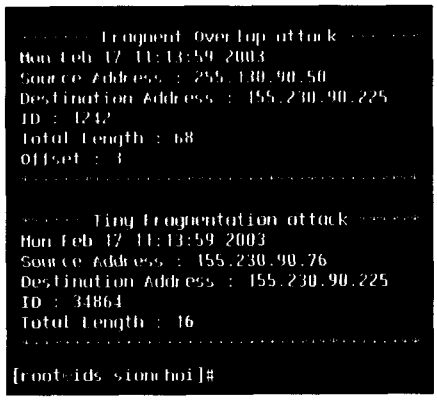

그림 2 IP 단편 필터링을 통한 탐지 결과

5. 결론 및 향후 연구과제

IP 단편을 이용한 공격들은 비정상적인 패킷들을 통하여 서비스 거부 공격을 하거나 패킷 필터링 장비나 침입차단시스템을 우회하기 때문에 탐지해 내기가 어렵다. 본 논문에서는 네트워크 기반의 침입탐지 시스템이 탐지해 내기 어려운 IP 단편을 이용한 침입탐지 시스템 우회 공격을 탐지해 낼 수 있는 IP단편 필터링 모듈을 설계하고 구현하였다. 실험결과 False-positve 오류율 0%에 탐지율 100%를 보여주어 정확한 탐지가 이루어진다는 것을 알 수 있다. 향후 연구과제로는 좀더 많은 이상 패킷 사례 연구를 통해 분석 패턴항목을 더 상세하고 정확하게 나누고 탐지한 모듈의 추가하거나 성능을 향상시킬 수 있는 연구가 필요하다.

[참고문헌]

[1] Paul E.proctor, *Practical Intrusion Detection Handbook*, Prentice Hall PTR, 2001.

[2] Marina Bykova, Shawn Ostermann, Brett Tjaden, "Detection Network Intrusions via Statistical Analysis of Network Packet Characteristics", *33rd Southeastern Symposium on System Theory (SSST)* , 2001.

[3] Stephen Northcut, Judy Novak, *Network Intrusion Detection An Analyst's Handbook Second Edition*, New Riders, 2001.

[4] Ed Skoudis, *Counter Hack*, Prentice Hall PTR, 2002.

[5] 정현철, IP Fragmentation을 이용한 공격기술들, 한국 정보보호 센터, 2001.

[6] Thomas H. Ptacek , Timothy N. Newsham, "Insertion, Evasion, and Denial of Service : Eluding Network Intrusion Detection", *Technical Report*, Secure Networks Inc., 1998.

[7] Ziemba, Reed & Traina, "Security Considerations for IP Fragment Filtering", RFC1858, 1995.

[8] http://www.cet.nau.edu/~mc8/Socket/Tutorials/section4.html