BPF 백도어 매직패킷 실시간 탐지 연구

손현기*, 백승렬*, 이윤호*, 최영락*, 장승현*, 윤명근**

*국민대학교 (대학원생), **국민대학교 (교수)

Detecting Magic Packets of BPF Backdoor in Real-Time

Hyeongy Shon*, Seungryeol Baek*, Yoonho Lee*, Younglak Choi*, Seunghyun Jang*, MyungKeun Yoon**

*Kookmin University(Graduate student), **Kookmin University(Professor)

요 약

최근 BPF(Berkeley Packet Filter) 백도어 공격으로 국내 통신사 정보가 유출되는 대형 사고가 발생하였다. BPF 백도어는 공격자가 원하는 시점에 감염 서버로 매직(magic)패킷을 발송하여 은밀한 비인가 프로세스 활성화 및 원격 제어를 가능하게 한다. 일반 악성코드는 단순하게 비콘(beacon) 패킷을 주기적으로 전송하기 때문에 네트워크 레벨의 탐지가 가능하지만, BPF 백도어는 탐지가 어려운 새로운 위협이다. 본 논문에서는 BPF 백도어를 분석하여 매직패킷을 네트워크 레벨에서 실시간 탐지하는 신기술을 제안한다. 제안기술을 검증하기 위해서 BPF 백도어를 구현하여 공격 패킷과 정상 패킷을 혼합한 실험 데이터셋을 구축하였다. 실험을 통해서 제안기술이 BPF 백도어의 매직패킷을 완벽하게 탐지하는 것을 확인하였으며, 공격자가 매직패킷 탐지 시그니처를 임의로 변경시킨 변종 공격에 대해서도 여전히 탐지가 가능한 것을 확인하였다.

I. 서론

BPF(Berkeley Packet Filter) 백도어(이하 BPF도어)는 2021년 PwC 보고서로 알려졌으며 [1], 글로벌 보안기업 트랜드마이크로는 아시아 통신사를 대상으로 시도된 BPF도어 공격 정황을 2025년 발표했다 [2]. 2025년 4월 국내 통신사에서 BPF도어 변종에 의한 공격으로 가입자 식별 번호가 포함된 중요 정보가 외부로 유출되는 대형 사고가 발생했다 [3].

BPF는 1992년 미국 버클리 대학에서 개발한 네트워크 패킷 필터링 기술이다 [4]. 처음에는 유닉스 계열 운영체제에서 네트워크 패킷을 효율적으로 캡처하고 필터링할 수 있도록 만들어졌으나(Classic BPF: cBPF) 최근에는 eBPF (extended BPF)로 확장되어 범용 커널 가상 머신 형태로 진화했다.

BPF도어는 cBPF를 악용하여 포트 리스닝 없이도 특정 패킷을 감지하고 명령을 실행하는 리눅스 기반의 은밀한 백도어 악성코드이다. 일반 악성코드는 감염 서버가 C2서버(Command & Control)로 비콘(beacon) 패킷을 지속적 주기적으로 전송하여 네트워크 기반의 탐지 기술이 유용했으나, BPF도어에서는 공격자가 감염서버로 매직(magic)패킷을 보내고, BPF도어가 이를 감지하여 리버스쉘(reverse shell)을 띄우는 등 공격자가 원하는 시점에 악성 행위가 발생하기 때문에 기존의 네트워크 기반 탐지 기술로는 식별이 어렵다. 서버에 백신이나 BPF도어 감염 식별 SW를 설치하여 감염 서버를 식별할 수는 있으나 [5], 서버에 접근할 수 있는 강력한 권한 및 추가 노력이 필요하다.

본 논문에서는 네트워크 모니터링 레벨에서 BPF도어 감염 서버의 매직패킷 수신과 리버스쉘 접속을 실시간으로 탐지하는 기법을 제안한다. 제안 기법은 감염 서버에 SW를 설치하지 않고 패시브 모니터링 방식으로 매직패킷의 등장과 BPF도어의 구동을 조기에 탐지할 수 있는 최초의 기법이다.

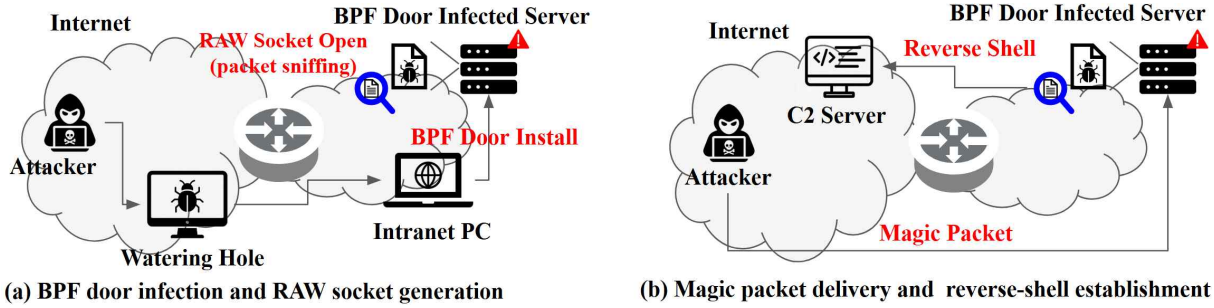

Fig. 1. BPFDoor infection and magic-packet-based C2 server communication

II. 연구 배경 및 관련 연구

BPF도어는 기존 백도어 방식의 악성코드와 다르게 네트워크 레벨 탐지가 어렵다. Fig. 1은 BPF 도어가 감염 서버에 설치되어 공격이 진행되는 과정을 보여준다. (a)단계는 BPF도어에 감염된 서버가 RAW소켓을 기반으로 모든 패킷을 스니핑하고 있는 모습이다. (b)단계는 공격자가 감염된 서버로 매직패킷을 전달하고 감염 서버에서 C2 서버로 리버스쉘을 접속하는 공격 실행과정을 보여준다. 리버스쉘 대신 다른 접속 방식을 사용하는 것도 가능하다.

공격자는 BPF도어에 감염된 서버를 제어하기 위해 BPF도어 컨트롤러를 사용하여 매직패킷을 전송하는 방식으로 공격을 수행한다. 이때 BPF 도어 컨트롤러에서 생성된 매직패킷은 Fig. 2와 같은 구조를 갖는다 [5]. BBPF도어에 감염된 서버는 수신되는 모든 패킷에 대해 백도어 내부에 하드코딩된 BPF 필터를 적용하여 프로토콜, 포트번호, 그리고 매직넘버(Magic number)를 확인한다. 필터 조건을 모두 만족하는 패킷이 감지되면 패킷에 포함된 비밀번호를 확인한다. 비밀번호가 일치하면 패킷에서 C2IP와 C2P(포트)를 추출하여 리버스쉘을 연결한다.

기존의 악성코드들은 감염 서버가 C2 서버로 주기적인 비콘 패킷을 보내게 하여 공격자의 명령을 가져오는 구조이다. 따라서 기존 네트워크 모니터링 기반의 탐지 기술들은 이러한 비콘 패킷의 단순 반복 패턴을 확인하여 감염 서버를 식별할 수 있었다 [6]. 반면에 BPF도어는 매직패킷을 사용해서 기존 탐지 기술을 우회하여 사이버보안에 큰 위협이 되고 있다.

실제 국내 통신사에서 2025년 4월 발생한 가입자 유심(Universal Subscriber Identity Module) 정보

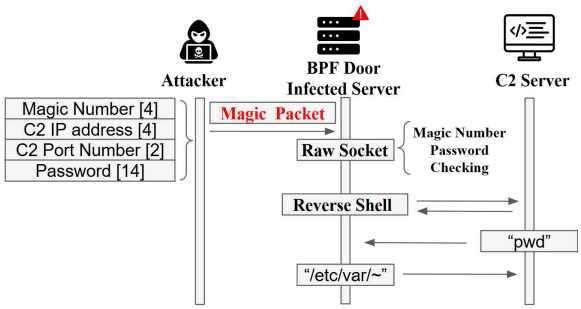

Fig. 2. BPFDoor magic packet including magic number (4 Bytes), C2IP (4 Bytes), C2P (2 Bytes), and Password (14 Bytes) [2].

유출 사고에서 BPF도어와 다수의 변종이 공격에 사용된 것으로 알려졌다 [3, 7]. 사이버보안 회사인 S2W에서는 Table 1과 같이 현재까지 알려진 BPF 도어의 변종을 크게 5가지 타입으로 분류하였다 [7].

많은 연구자들이 기존 악성코드 감염서버의 비콘 패킷 탐지 기법을 제안했으나 [6], BPF도어처럼 매직패킷을 사용하는 감염 서버를 네트워크 모니터링 방식으로 탐지하는 기술은 현재까지 거의 연구된 바 없다. 최근 국내 통신사 해킹 사고를 계기로, 침입탐지시스템에서 Table 1과 같이 알려진 BPF도어에서 사용하는 매직번호를 탐지 시그니처로 사용하여 패킷을 조사하는 방법이 사용되고 있다 [5]. 하지만 공격자가 매직번호를 변경하면, 이러한 탐지 기법은 쉽게 우회된다.

III. BPF도어 매직패킷 탐지 기술

본 논문에서는 공격자가 BPF도어 감염 서버로 전송하는 매직패킷과 리버스쉘 접속을 네트워크 레벨에서 실시간 탐지할 수 있는 RDM(Real-time Detection of Magic packet) 기법을 새롭게 제안한다.

RDM 기법은 BPF도어 감염 여부를 판별해야 할 대상 장비들(주로 내부망 호스트)이 우선

Table 1. Variants of BPF Door [7]

Type	Protocol	Magic Number
A	UDP, ICMP	0x7255
	TCP	0x5293
B	TCP	0x39393939
C	TCP	0x5293, 0x39393939
	UDP, ICMP	0x7255
D	TCP	0x5293, 0x39393939
	UDP, ICMP	0x7255
E	UDP, TCP, SCTP	0x4430CD9F5, E142766

선별되어 있다는 가정이 필요하며, 사이버보안이 중요한 사이트들은 이미 이러한 장비들이 선별되어 있다. 장비들의 IP 주소별로 멤버십 해시테이블과 블랙리스트 해시테이블을 구축한다. 본 논문에서는 두 해시테이블의 최대 크기를 고정시키고, 테이블이 가득 차는 상황에서는 가장 오래된 데이터를 삭제하여 메모리 사용량의 최대치를 일정 수준 이하로 유지한다.

RDM은 등록(Registration)과 탐지(Detection) 기능으로 구성되며, 두 기능은 Fig. 3과 같이 동시에 동작한다.

첫째, *RDM 등록* 기능에서는 모니터링 대상 IP별로 수신되는 모든 패킷을 조사한다. 패킷의 헤더로부터 출발지주소(sIP), 출발지포트(sP), 목적지주소(dIP), 목적지포트(dP)를 조합하여 해시함수로 키값(=H(sIP, sP, dIP, dP))을 생성한. 해당 키값이 처음 등장했는지 확인하기 위해서 멤버십 해시테이블을 이용한다. 멤버십 해시테이블에 해당 키값이 존재하지 않으면, 해당 패킷이 속한 세션이 처음 시작된 것으로 간주하고 블랙리스트 해시테이블에 (C2IP, C2P) 값을 저장한다. 이때 C2IP와 C2P 값은 해당 패킷을 매직패킷이라고 가정했을 때, 애플리케이션 페이로드 부분에서 5~8번째 등장하는 4바이트와 9~10번째 등장하는 2바이트에서 추출한 값으로 설정한다.

본 논문에서는 BPF도어 소스 코드 분석과 기존 위협리포트 내용을 바탕으로 정보 선택 위치를 결정하였다 [5]. 여기서 두 가지 상황이 발생할 수 있다. 첫째, 조사 대상 패킷이 매직패킷이었다면, 감염서버가 접속할 대상의 IP 주소와 포트 번호를 블랙리스트 해시테이블에 등록하게 된다. 둘째, 매직패킷이 아니었다면 의미

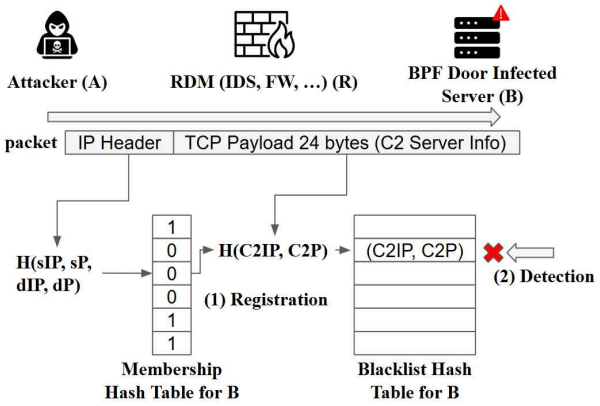

Fig. 3. Architecture of proposed RDM (Real-time Detection of Magic packet)

없는 값이 블랙리스트 해시테이블에 등록된다. RDM은 두 가지 경우에 모두 적절한 대처가 가능하다.

둘째, *RDM 탐지* 기능에서는 등록 단계에서 블랙리스트 해시테이블에 추가된 (C2IP, C2P) 값들을 이용해서 감염서버로부터 C2서버로 접속되는 리버스쉘 등에 사용되는 패킷을 탐지한다. 등록 단계에서 실제 매직패킷에 포함되었던 (C2IP, C2P) 값이 접속 시도에서 사용된다면, 매직패킷과 그로 인해 활성화된 리버스쉘 접속이 실시간 탐지된 것으로 간주한다. 만약 등록단계에서 매직패킷이 아닌 패킷에서 추출된 (C2IP, C2P) 정보가 블랙리스트 해시테이블에 있다면, 탐지 기능에서는 매칭이 발생하지 않으므로 문제가 발생되지 않는다.

IV. 실험

본 논문에서는 BPF도어 매직패킷 탐지를 실험하기 위해서 BPF도어를 재현하여 자체적으로 실험 환경을 구축하였다.

실험에는 두 가지 버전의 악성코드를 사용했다. 원본 BPF도어 악성코드를 그대로 재현한 버전과 재현된 원본 BPF에서 매직넘버를 수정하여 변종 코드를 구현한 버전이다. 각각을 Origin BPF도어와 Mutant BPF도어로 칭한다.

공격 트래픽은 로컬 환경과 리모트 환경 모두에서 수집했다. 로컬 환경은 가상 머신을 활용하여 감염서버와 C2서버를 동일 호스트 내에 구성하였다. 반면, 리모트 환경에서는 C2서버를 감염서버와 다른 외부망에 위치시켰다. 두 환경 모두 감염서버는 BPF도어 악성코드에 감염된 상태이며, C2서버는

TLS(Transport Layer Security) 기반으로 통신한다.

공격은 공격자가 감염서버로 전송한 매직패킷에 의해서 시작된다. 본 논문에서는 매직패킷 전달 프로토콜로 TCP, UDP, ICMP를 모두 구현하여 실험하였다 (Table 1). 매직패킷이 감염서버에 도달하면, BPF도어가 이를 인식하고 활성화되어 C2서버로 리버스쉘을 연결한다. 공격 시작부터 리버스쉘 연결 과정까지 발생하는 모든 패킷을 와이어샤크(WIreshark)로 수집하여 데이터셋에 반영했다.

정상 패킷 데이터는 CICIDS2017 공개 데이터셋(https://www.unb.ca/cic/datasets/ids-2017.html)과 국내 주요 통신망에서 수집한 사설(Private) 데이터셋을 사용하였다 (Table 2).

네트워크 침입탐지 실질적 표준 기술인 스노트 (Snort)와 수리카타 (Suricata)를 SOTA (State-of-the-art) 기술로 도입하여 RDM과 비교한다. 해당 기술들은 알려진 매직넘버를 시그니처로 사용하여 매직패킷을 탐지한다[5].

실험 결과, Origin BPF도어 공격은 RDM과 SOTA 기술이 모두 탐지하였다. 그러나 SOTA 기술들은 변형된 Mutant BPF도어 공격을 탐지하지 못하는 한계를 보였다. 반면 RDM은 세 가지 다양한 프로토콜 환경 모두에서 Mutant BPF도어 공격을 오탐 없이 효과적으로 탐지하였으며, Table 3는 실험 결과를 요약한다. 실험을 통해 기존 시그니처 기반 기법의 한계를 확인함과 동시에 RDM이 매직번호를 변경한 변종까지 탐지하는 것을 확인할 수 있었다.

Table 2. Number of Packets in Datasets

Protocol	Dataset	
	Private	CICIDS2017
TCP	38,819,864	10,718,485
UDP	5,151	934,997
ICMP	12,673,864	671
Others	13,750	55,818
Total	51,519,657	11,709,971

Table 3. RDM vs SOTA

Solution	Type	Protocol		
		TCP	UDP	ICMP
Snort	Origin	O	O	O
	Mutant	X	X	X
Suricata	Origin	O	O	O
	Mutant	X	X	X
RDM	Origin	**O**	**O**	**O**
	Mutant	**O**	**O**	**O**

V. 결론

본 논문은 BPF도어의 매직패킷을 네트워크 수준에서 실시간 탐지하는 RDM 기법을 제안했으며, 감염서버에 소프트웨어 설치 없이 BPF도어의 활성화 징후를 식별하는 새로운 탐지 방향을 제시했다.

Acknowledgement

이 논문은 2023년도 정부(과학기술정보통신부)의 재원으로 정보통신기획평가원의 지원(RS-2023-00235509, ICT융합 공공 서비스·인프라의 암호화 사이버위협에 대한 네트워크 행위기반 보안관제 기술 개발)과 정부(교육부)의 재원으로 한국연구재단의 지원을 받아 수행된 기초연구사업임(No. NRF-2023R1A2C1004653).

[참고문헌]

[1] PwC, “Cyber Threats 2021: A Year in Retrospect”, 2021.

[2] Trend Micro, “BPFDoor’s Hidden Controller Used Against Asia, Middle East Targets”, 2025.

[3] 조재호, “[SKT 해킹 사태] KISA, SKT 해킹 악성코드 공개...보안 점검 권고”, https://www.boannews.com/media/view.asp?idx=137004&page=1&kind=1, 2025.

[4] D. Jin, A. J. Gaidis, and V. P. Kemerlis, “BeeBox: Hardening BPF against Transient Execution Attacks,” USENIX Security , 2024.

[5] KISA,”BPFDoor 악성코드 점검 가이드”, https://www.boho.or.kr/kr/bbs/view.do?searchCnd=&bbsId=B0000133&searchWrd=&menuNo=205020&pageIndex=1&categoryCode=&nttId=71754, 2025.

[6] M. Abu Talib, et al. "APT beaconing detection: A systematic review." Computers & Security 122 (2022): 102875.

[7] S2W, “Detailed Analysis of BPFDoor targeting South Korean Company”, https://medium.com/s2wblog/detailed-analysis-of-BPFDoor-targeting-south-korean-company-328171880a98, 2025.