정 보 보 호 학 회 지
제 31 권 제1호, 2021. 2

콜드 부트 공격 기술 및 최신 동향 분석

원 유 승*, 한 동 국**

요 약

일반·보안 기기에서 개인 정보 및 비밀 정보가 많은 부분에서 취급된다. 보안 기기라고 할지라도 전원이 인가되지 않으면 데이터가 초기화되는 RAM의 물리적 특성을 고려하여, 보안영역으로 간과될 수 있다. 하지만 2008년 이후, RAM의 표면 온도를 현저히 낮추면 전원이 인가되지 않아도 RAM 데이터가 초기화되지 않는 특징을 활용한 콜드 부트 공격이 제기되어, 지금까지 활발히 연구되어 하나의 보안이 필요한 영역으로 자리매김하고 있다. 본 논문에서는 콜드 부트 공격의 방법론과 발전 동향, 콜드 부트 공격에 대한 물리적 특성 및 디스크 암호화 솔루션에 적용 가능한 콜드 부트 공격을 설명한다. 또한, 국내 연구에 간과될 수 있는 콜드 부트 공격을 고취시키기 위하여, 이에 대한 대응기법도 소개한다.

Ⅰ. 서 론

보안 매체 또는 일반 기기에서 개인 정보 보호가 이루어져야 하는 것 중 하나는 메모리이다. 데스크탑, 노트북, 스마트폰, IoT 장비는 모두 메모리를 탑재하고 있으며, 개인 정보가 메모리에 영구적 또는 일시적으로 저장되기 때문이다.

하지만 2008년도 RAM에 저장된 데이터를 탈취할 수 있는 기술이 처음 제기되었다 [1]. 이는 RAM의 물리적 특성을 고려하여 RAM의 데이터를 탈취하는 콜드 부트 공격(Cold Boot Attack)이다. 그 수행하는 방법을 알기 위해서는 일반적인 RAM의 물리적 특성을 알아야 한다. 이는 RAM에 저장된 데이터가 전원이 인가된 상태라면, 그 데이터를 유지하려는 특성을 지닌다. 하지만, 전원이 인가되지 않으면 메모리는 초기화 상태(일반적으로 0 값)로 돌아가게 된다.

그러나 전원이 인가되지 않는 경우라도 RAM의 표면 온도가 낮은 상태로 유지된다면, 저장된 데이터가 초기화되지 않고 그 상태를 온전히 유지한다. 이러한 물리적 특성을 이용한 공격 방법이 콜드 부트 공격으로 일컫는데, 이를 이용하여 RAM 데이터 접근에 허가되지 않은 사람이더라도 RAM 데이터를 탈취할 수 있다.

2008년 이후, 지속적으로 콜드 부트 공격에 대한 연구가 최근까지 이루어지고 있다. 블록 암호, 공개키 암호, 후양자 암호와 암호화 디스크 솔루션의 비밀 키를 콜드 부트 공격을 활용하여 취득할 수 있다. 따라서 본 논문에서는 콜드 부트 공격 방법을 간략히 살펴보고, 공격 발전 동향 및 응용 분야와 함께 그 대응기법을 살펴본다.

Ⅱ. 콜드 부트 공격

앞서 언급된 것처럼 콜드 부트 공격은 공격자에게 접근이 인가되지 않은 RAM 데이터를 탈취해 오는 것이다. 이를 간단히 하면 다음과 같은 절차로 이루어진다.

1) [희생자 입장] 허가되지 않은 사용자에게 접근이 불가능한 RAM 사용 (단, RAM에 전원이 인가된 상태를 유지)
2) [공격자 입장] 전원이 인가된 상태로 RAM의 표면 온도를 낮은 상태로 유지
3) [공격자 입장] 전원을 끈 후, 재부팅 후 RAM의 접근 권한 설정 전에 RAM 데이터를 탈취

1)에서 2)로 공격자가 RAM의 표면 온도를 전원이 인가된 상태로 낮은 상태(예, 영하 35도)로 낮추어야 한다. RAM의 표면 온도를 낮추기 위해서는 특수 장비를 활용해도 되지만, 에어 더스터 형태인 분사형 먼지 제거제로 충분히 표면 온도를 낮출 수 있다[1]. 그 이후, RAM 자체를 희생자로부터 탈취하거나 RAM이 부착

* 싱가포르 난양기술대학교 Temasek Labortories (연구원, yooseung.won@ntu.edu.sg)
** 국민대학교 정보보안암호수학과·금융정보보안학과 (교수, christa@kookmin.ac.kr)

된 기기를 모두 탈취한 뒤에 3)을 진행할 수 있다.

3)에서 RAM 데이터를 탈취하기 위해서는 RAM이 부착된 기기 또는 부착할 기기의 부트 시퀀스(Boot Sequence)에서 RAM을 초기화 또는 RAM 데이터 접근 권한 부여 전에, 그 값을 탈취해야 한다. 따라서 이를 위해서는 일부 부트 시퀀스를 수정하여야만 탈취할 수 있다. 만약 RAM이 탈부착이 되는 경우라면, 수정된 부트 시퀀스가 적용된 기기에 RAM을 부착하여 콜드 부트 공격을 수행할 수 있다.

Ⅲ. 기기 중심 콜드 부트 공격 발전 동향

본 장에서는 기기 중심 콜드 부트 공격 발전 동향을 설명하며, 노트북, 스마트폰, DDR3/DDR4, IoT 장비 순으로 콜드 부트 공격 동향을 살펴본다.

최초의 콜드 부트 공격[1]의 대상은 노트북으로 탈부착이 가능한 RAM에 대하여 수행하였다. 대상 노트북과 RAM의 사양은 다음 표 1과 같다.

해당 노트북에 대하여 부트 시퀀스르 3가지(PXE (Preboot Execution Environment) 네트워크 부트, USB 드라이버 부트, EFI(Extensible Firmware Interface) 부트)로 접근[1]하였다. 해당 부트 시퀀스를 일부 수정하여 부팅을 시작하자마자 USB 메모리를 RAM 데이터를 저장하는 방식으로 콜드 부트 공격을 수행하였다. 앞서 언급된 것처럼 노트북 모델 1종류만으로 수정된 부트 시퀀스를 활용하여 표 1에서의 RAM에 대하여 콜드 부트 공격이 또한 가능하다.

2003년에는 안드로이드 4.0이 설치된 갤럭시 넥서스 스마트폰 대상으로 콜드 부트 공격이 가능함을 증명하

〔표 1〕 〔1〕에서의 콜드 부트 공격 대상

메모리 종류 및 크기	칩 제조사	노트북 모델	제작 연도
SDRAM 128MB	인피니언	Dell Dimension 4100	1999
DDR 512MB	삼성전자	Toshiba Portege	2001
DDR 256MB	마이크론	Dell Inspiron 5100	2003
DDR2 512MB	인피니언	IBM T43p	2006
DDR2 512MB	엘피다	IBM x60	2007
DDR2 512MB	삼성전자	Lenovo 3000 N100	2007

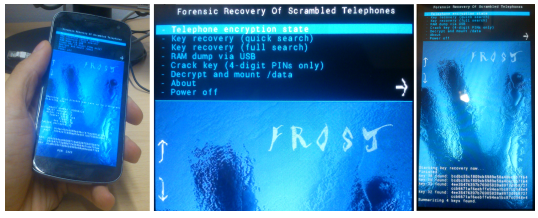
1) https://citp.princeton.edu/our-work/memory/code에서 해당 오픈 소스를 활용할 수 있다.

〔그림 1〕 에어 더스트로 RAM의 표면 온도를 -50도로 낮춰, IBM T43p 노트북에서 콜드 부트 공격을 수행

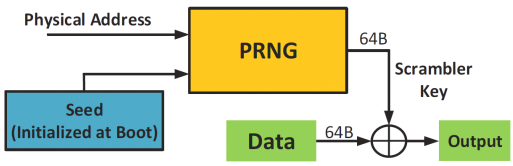
였다[2]. 앞선 노트북과는 달리 RAM의 탈착이 불가능하며 권한 설정 없이 부트 시퀀스를 수정하기는 어렵다. 부트 시퀀스를 수정하기 위해서는 “fastboot oem unlock”을 수행해야 하는데 이는 부팅 시 스마트폰 내에 “To prevent unauthorized access to your personal data, unlocking the bootloader will also delete all personal data from your phone” 라는 문구와 함께 메모리에 저장되어 있는 개인 정보가 모두 지워진다. 하지만 부팅을 수행하자마자 USB 포트를 활용하여 부트시퀀스를 수정 후(아래 부트 시퀀스 수정 권한을 가정한 경우와 같은 절차), 콜드 부트 공격을 적용하면 이메일, 사진, 방문한 웹사이트 등의 일부 개인정보가 남아있는 것을 확인할 수 있다. 즉, 부트 시퀀스를 수정할 수 있게 권한을 주면서 RAM에 있는 모든 데이터를 삭제해야 되지만, 그렇지 않아 콜드 부트 공격이 가능한 형태이다.

이와 달리 이미 부트 시퀀스 수정 권한을 가정한 경우에는 콜드 부트 공격이 어렵지 않게 가능하다. USB 포트를 활용하여 복구 파티션(Recovery partition)에 악의적인 코드를 삽입한 후 사용자 데이터의 암호화 키, 사용자 데이터, PIN, RAM 데이터 추출 등이 가능하다.

앞선 노트북, 스마트폰과 달리 콜드 부트 공격에 대한 대응기법이 적용된 최신 RAM(DDR3, DDR4) 대상으로 공격이 가능함을 보였다[3, 4]. 최초의 콜드 부트 공격이 제안되었을 때의 RAM은 대응기법이 존재할 수 없었기 때문에, 그 이후 콜드 부트 공격에 대한 대응기



〔그림 2〕 복구 파티션에 삽입된 악의적은 코드를 활용한 콜드 부트 공격



〔그림 3〕 DDR3, DDR4에 적용된 스크램블러 대응기법

법이 적용된 DDR3, DDR4가 출시되었다. 스크램블러(Scrambler) 대응기법이 다음과 같이 메모리 구조 내에 적용되어 있다.

그림 3에서와 같이 RAM에 데이터를 저장할 때 64바이트 단위로 스크램블러 키(Scrambler Key)를 활용하여 XOR 연산하여 암호화 저장한다. RAM에서 데이터를 불러올 때는 같은 방법으로 Output을 스크램블러 키로 XOR 연산하여 복호화하여 본 데이터를 활용할 수 있다. 공격자 입장에서는 Seed 값만 구할 수 있다면, 스크램블러 키를 알 수 있기 때문에 공격이 가능하다. 하지만 Seed 값은 메인보드로부터 받아서 사용하기 때문에, 새로운 메인보드로 RAM을 옮겨가거나 재부팅을 하면 새로운 Seed 값으로 받아서 사용하게 된다.

선형 되먹임 시프트 레지스터(Linear feedback shift register, LFSR)로 이루어진 PRNG 구조를 갖는 DDR3, DDR4에서는 각각 16개, 4096개의 구분 가능한 64바이트 키가 생성이 가능하다는 사실을 밝혀냈다. 공격자 입장에서 키를 추출하기 위해서, RAM 메모리가 대부분 0인 특성과 아래와 같은 스크램블러 키의 특징을 활용한다. 다시 말하면, 메모리가 0인 부분에 스크램블러 대응기법이 적용되면 스크램블러와 0값이 XOR된 값인 스크램블러 키가 온전히 저장된다.

$$\begin{aligned} &K[i:i+1]\oplus K[i+2:i+3]=K[i+8:i+9]\oplus K[i+10:i+11]\\ &K[i:i+1]\oplus K[i+4:i+5]=K[i+8:i+9]\oplus K[i+12:i+13]\\ &K[i:i+1]\oplus K[i+6:i+7]=K[i+8:i+9]\oplus K[i+14:i+15]\\ &K[i+2:i+3]\oplus K[i+4:i+5]=K[i+10:i+11]\oplus K[i+12:i+13]\\ &\text{for } i=0,16,32,48 \end{aligned} \quad (1)$$

$K[x:y]$는 x번째 바이트부터 y번째까지의 바이트를 의미한다. 즉, 콜드 부트 공격을 활용하여 암호화된 메모리를 얻은 후, 64바이트 단위로 식 (1)을 적용하여 만족한다면 메모리가 0인 부분에 스크램블러 키가 사용되었다는 것을 알 수 있다. 이러한 성질을 활용하여 약 16MB의 RAM 메모리가 덤프된다면 모든 스크램블러 키를 찾을 수 있다. 요컨대, 스크램블러 대응기법이 적용된다고 하더라도 기존의 콜드 부트 공격은 어렵지 않게 적용이 가능하다.

〔표 2〕 [5]에서의 콜드 부트 공격 대상

타겟	라즈베리 파이 모델 B+
SoC	브로드컴 BCM2835 (65nm 공정)
CPU	700 MHz ARM1176JZF-S single core
GPU	브로드컴 VideoCore IV @ 250 MhzOpenGL ES 2.0 (24 GFLOPS) MPEG-2 and VC-1 (with license), 1080p30H.264/MPEG-4 AVC high-profile decoder and encoder (128 KB L2 캐시 메모리)
Memory (SDRAM)	700 MHz ARM1176JZF-S single core (SAMSUNG k4p4g324eq-rgc2)

최근 [5]에서 IoT 장비에서 최초로 콜드 부트 공격이 가능함을 보였다. IoT 장비에서 상용적으로 많이 사용되는 라즈베리 파이(Raspberry Pi)를 대상으로 하였으며 세부 스펙은 다음 표 2와 같다.

라즈베리 파이는 앞선 스마트 폰과 같이 RAM이 칩에 부착된 상태이다. 하지만, 공격 대상이 되는 라즈 베리 파이는 마이크로 SD카드에 OS가 설치되어 구동 된다. 희생자의 마이크로 SD카드에 부트 시퀀스를 수정하기 보다는, 공격자가 수정한 부트 시퀀스가 적용된 마이크로 SD카드로 교체하여 라즈베리 파이 내에 장착된 RAM에 콜드 부트 공격을 수행하였다. 즉, 앞선 공격 절차에서 공격 절차 3)의 형태가 수정되는데 이를 정리하면 다음과 같다.

3) [공격자 입장] 전원을 끈 후, 수정된 부트 시퀀스가 적용된 마이크로 SD카드로 교체 후, 재부팅 후 RAM 데이터를 탈취

수정된 절차대로 콜드 부트 공격을 수행하면 라즈베리 파이 내의 RAM 데이터(512MB)를 약 5분 안에 모두 탈취할 수 있다. 이는 라즈베리 파이가 취약하기도 하지만 SoC 칩인 BCM2835 칩 내에 콜드 부트 공격에 대한 대응기법이 존재하지 않아, 공격이 가능함을 보인 것이다.

Ⅳ. RAM 물리적 특성에 따른 콜드 부트 공격 결과

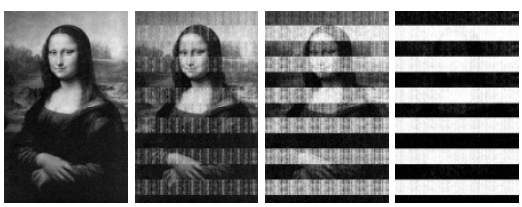
본 장에서는 콜드 부트 공격에 따른 RAM 데이터 복구 결과를 서술한다. 앞서 설명했듯이, 콜드 부트 공격

은 RAM의 전원인가가 이루어지지 않더라도 낮은 온도에서 그 데이터를 보존하려는 성질을 이용하여 공격하는 것이다. 하지만, 낮은 온도를 유지한다고 하더라도 100% 복구율을 의미하지 않는다. 물리적 특성이므로 기기에 부착된 RAM마다 차이 또는 실험(RAM의 표면 온도)에 따라 달라질 수 있다.

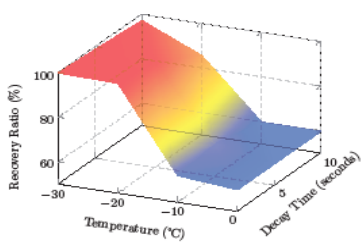
[1]에서 정의했듯이, RAM 메모리의 1비트가 디폴트 값(일반적으로 0)에서 그 반대 값(비트 1)으로 변할 확률(ρ_1)과 비트 1에서 비트 0으로 변할 확률(ρ_0)을 사용하여 콜드 부트 공격의 성공 확률을 나타낸다. 예를 들면, 콜드 부트 공격이 이루어지지 않고, RAM의 물리적 성질이 완벽히 이루어졌다면 기기를 재부팅하면 ρ_0가 1이며, ρ_1는 0일 것이다. 실제로 [1]에 따르면 콜드 부트 공격을 수행하더라도, ρ_1은 0.001(0.1%)이하로 굉장히 작으며, ρ_0에 대한 실험치는 밝히지 않았지만 0.1(10%)로 가정하여도 메모리 복구가 가능하다고 밝혔다. 즉 낮은 온도를 유지한채 메모리를 복구해와도 ρ_0와 ρ_1에 의한 비트 오류율이 존재한다.

또한, 위의 그림 4에서와 같이 영하 50도로 낮춘 후, 짧은 시간 동안의 전원 비인가는 RAM 복구 성공률 100%에 가깝다는 것을 알 수 있다.

[5]에서도 라즈베리 파이 내에서 RAM의 표면 온도와 전원 비인가 시간에 따른 결과 측정하였다 (그림 5).



〔그림 4〕 인피니언 SDRAM 128MB에서 표면 온도 영하 50도로 낮춘 후, 전원 비인가 시간(좌측부터 5초, 30초, 60초, 5분)에 따른 이미지 복구율



〔그림 5〕 전원 비인가 시간(Decay Time) 및 RAM 표면 온도(Temperature)에 따른 RAM 데이터 복구 성공률 (Recovery Ratio)

약 영하 30도를 짧은 시간 동안 유지할 수 있다면, RAM의 공격 복구 성공률 100%에 가깝다는 것을 알 수 있다.

이를 기반으로 라즈베리 파이 내에서 ρ_0와 ρ_1에 대한 확률을 측정한 결과는 다음과 같다.

표 3에서와 같이 전체 메모리가 512MB이지만 4169415680비트(약 497MB)만 조사된 이유는 일부 부트 시퀀스, 캐시 메모리, POST (Power-On-Self-Test) 등을 제외한 메모리이다. 앞서 [1]에서 소개된 ρ_0, ρ_1보다 현저히 작은 수치라는 것을 알 수 있다.

〔표 3〕 라즈베리 파이에서 ρ_0, ρ_1 확률 값

	ρ_0	ρ_1
값	0.0000027 (≈11373/4169415680)	0.00000009 (≈375/4169415680)

Ⅴ. 콜드 부트 공격의 응용

본 장에서는 콜드 부트 공격을 활용하여 RAM 메모리 탈취 후, 알고리즘의 비밀 키 복구 또는 디스크 암호화 솔루션의 키 복구를 설명한다.

5.1. 암호 알고리즘 비밀 키

블록 암호 또는 공개키 암호 알고리즘의 비밀 키가 RAM에 저장되어 있다면, 콜드 부트 공격을 통하여 그 비밀 키를 탈취할 수 있다. 하지만, 앞서 소개한 RAM의 물리적 특성으로 인해 100%를 복구하지 못할 수 있다. 따라서 비밀 키를 온전히 복구하기 위해서는 이러한 오류로부터 복구할 필요성이 제기된다. 이를 위해서는 블록 암호 알고리즘 같은 경우, 다수 암호화 작업(운용 모드)으로 인해 라운드 키가 일반적으로 RAM에 저장된다고 가정한다. 따라서 키 스케줄 특성을 이용하여 전체 키 복구를 수행한다. 또한, 공개키 암호 알고리즘 같은 경우에는 비밀 키의 대수적 특징을 활용하여 전체 키를 복구한다.

DES의 경우[1]에는 56비트 비밀 키에서 48비트의 16개 라운드 키를 갖는다. 단순히 Compression과 Permutation만 고려된 키 스케줄 때문에, ρ_0가 0.5이더

라도 56비트키의 틀린 비트가 있을 확률 10^{-9}으로 현저히 낮다.

AES의 경우에도 역시 키스케쥴 특성을 활용할 수 있다. SAT solver를 로 활용하여 라운드 키 간의 대수적 관계성을 이용하면, ρ_0(=0.7)가 높더라도 수초 내에 키 값을 복구할 수 있다[6].

공개키인 RSA의 경우에 공개 키와 비밀 키 간의 관계성을 활용하여 divde-and-conquer 전략으로 비밀 키를 복구할 수 있다. ρ_0가 0.4이고, ρ_1이 0.001일 때 수초내에 복구할 수 있다는 것을 보였다[7].

그 이후, NTRU 도 다항식의 곱셈과 합으로 이루어진 비밀 키의 특성을 활용하여, (ρ_0,ρ_1)가 (0.01, 0.001)일 때, 수 분에서 수 시간 내에 비밀 키를 복구하였다[8].

2018년에는 후양자 암호(Post-Qunatum Cryptography, PQC)에 대한 키 복구 방법도 연구되었다. Kyber, NewHope에 대하여 시계열/주파수계열(즉, NTT 연산 여부)에서 RAM 메모리 복구가 이루어졌느냐에 따라 공격복잡도를 계산하였다[9]. 특히, NTT를 활용하는 공격 복잡도에서는 100% 키를 복구하는 것이 아닌 일정 확률로 키를 복구할 수 있다.

또한, 2019년도에는 후양자 암호 중 하나인 LUOV 대상으로 비밀 키 SEED인 32바이트에 대하여 (ρ_0,ρ_1)가 (0.03, 0.001) 이어도 서명 검증을 활용하여 비밀 키를 100% 복구할 수 있음을 밝혔다[10]. 후양자 암호인 BLISS[11], McEliece[12], Supersingular Isogeny Key Encryption (SIKE)[13]에 대한 콜드 부트 공격도 유사한 방법으로 표 3와 같이 복잡도가 연구되었다.

〔표 4〕 알고리즘 기반 ρ_0, ρ_1에 따른 비밀키 복구 시간 또는 복잡도

알고리즘	ρ_0	ρ_1	시간 또는 복잡도	참고 문헌
DES	0.5	0.001	1초 이내	〔1〕
AES-128	0.7	0	1초	〔6〕
RSA-1024	0.4	0.001	2.4초	〔7〕
NTRU	0.01	0.001	수초~수시간	〔8〕
Kyber	0.2	0.1	NTT: $3 \cdot 2^{21.1}$, 95% non-NTT: $2^{38.7}$	〔9〕
NewHope	0.17	0.1	NTT: $2^{48.7}$, 84% $2^{53.7}$	〔9〕
LUOV	0.03	0.001	100% 복구	〔10〕
BLISS	0.001	0.001	100% 복구	〔11〕
McEliece	0.005	0.001	100% 복구	〔12〕
SIKE	0.01	0.001	100% 복구	〔13〕

블록 암호 및 공개키 암호 알고리즘에 대한 비밀 키가 콜드 부트 공격에 의하여 일부 에러 비트가 있는 경우, 복원하는 방법이 꾸준히 연구되고 있다. 후양자 암호의 경우에는 초기 연구로서, 알고리즘의 대수적 구조를 더 면밀히 적용할 경우 추후 개선 연구가 발생할 수 있다.

5.2. 디스크 암호화 솔루션

앞선 연구 바탕으로 암호 알고리즘의 비밀 키가 디스크 암호화 솔루션에 사용하게 될 때, 콜드 부트 공격을 통하여 암호화 솔루션의 키를 탈취할 수 있는 방법이다. 디스크 암호화 솔루션의 종류를 간략히 살펴보고, 그 방법론을 예시적으로 설명한다.

5.2.1. 디스크 암호화 솔루션의 종류 및 사용

[1]에서 디스크 암호화 솔루션에 대한 콜드 부트 공격이 최초로 이루어졌으며, 운영체제에 따라 디스크 암호화 솔루션의 종류는 다음과 같다.

표 5에서 소개된 디스크 암호화 솔루션은 기본적으로 AES 암호 알고리즘과 운용모드를 활용하여 암호화를 수행한다. 소프트웨어로 구현되어 있으며, 고속 암호

〔표 5〕 디스크 암호화 솔루션에 따른 지원 운영체제 및 알고리즘

디스크 암호화 솔루션	지원 운영체제	특징
dm-crypt	Linux	LUKS(Linux Unified Key Setup) 포맷 지원
Loop-AES	Linux	On-the-fly 디스크 암호화 솔루션
TrueCrypt	Windows, Mac OS, Linux	다양한 운영체제 및 알고리즘(예, Serpent, Twofish)을 지원
BitLocker	Windows	Trusted Platfor Module(TPM)에 키 저장을 지원
FileVault	Mac OS	솔루션 구조가 공개되지 않아, 역공학으로 구조를 알아내고, IV를 생성하기 위한 키가 따로 존재함

화를 위하여 키 스케줄을 1회만 계산하여 라운드 키를 RAM에 저장하여 지속적으로 암호화에 사용한다.

디스크 암호화 솔루션에서 dm-crypt를 예를 들면, 사용자 입력으로 비밀번호(Passphrase)를 입력받아 의사 난수 함수(Key Derivation Function, KDF)를 거쳐 그 출력을 블록 암호 알고리즘의 마스터 키로 사용한다.

5.2.2. 디스크 암호화 솔루션에 대한 콜드 부트 공격

앞서 설명한 것처럼 메모리에 저장된 블록 암호 알고리즘의 라운드 키를 콜드 부트 공격을 통해 찾아낸다면, 어렵지 않게 복호화가 가능하다. 본 절에서는 많이 사용되는 AES 암호 알고리즘 기준으로 설명한다.

하지만, 콜드 부트 공격을 수행하더라도 전체 메모리에서 일부 에러 비트가 존재하는 라운드 키를 찾아야 한다. 라운드 키가 산발적으로 저장되지 않고 RAM 메모리에 순차적으로 저장된다[1]. 오류비트를 감안하여 첫 16바이트를 1라운드 키로 가정하고, 2번째 16바이트가 2라운드 키인지를 검사하면 어렵지 않게 메모리 내에 라운드 키를 찾아낼 수 있다. [1]에서 이에 대한 소스 코드를 공개하고 있다.

Ⅵ. 콜드 부트 공격에 대한 대응기법

2008년 이후 콜드 부트 공격이 실용적으로 가능해짐에 따라, 이에 대한 대응기법이 제안되고 있다. RAM에서의 대응기법, 부트 시퀀스에서 대응기법, 디스크 암호화 솔루션과 같이 응용 솔루션에서의 대응기법이 있다.

6.1. RAM 에서의 하드웨어 대응기법

앞서 소개한 [3, 4]에서와 같이 .DDR3, DDR4에는 스크램블러 하드웨어 대응기법이 존재한다. 하지만, RAM 데이터가 대부분 0이라는 점과 스크램블러 키의 복잡도 부족으로 암호화 된 키가 어렵지 않게 누출되었다. [4]에서 제안된 것과 같이 부트 때 생성된 키를 기반으로 하드웨어로 구현된 AES-CTR 또는 ChaCha 알고리즘을 활용하여 암·복호화를 수행할 수 있다. 특히, 최소 DDR4 읽기 지연 시간(12.5ns)을 고려하여 ChaCha8을 제안하였다.

6.2. 부트 시퀀스 대응기법

장치에서 부트 시퀀스가 이루어질 때, 부트 초기에 RAM 데이터를 모두 초기화하는 방법이 하나의 대응기법이 될 수 있다. 특히, RAM이 기기에 부착된 경우에는 부트초기 하드웨어적으로 초기화 대응기법이 있다면 콜드 부트 공격이 불가능하다.

하지만 RAM이 탈부착이 가능한 경우에는 취약한 기기로 RAM을 부착하면 여전히 공격이 가능하다.

6.3. 디스크 암호화 솔루션 대응기법

AES의 라운드 키를 모두 RAM에 저장함으로써 콜드 부트 공격을 통해 복화하가 가능하였다. 이를 막기 위해서는 라운드 키를 on-the-fly 형식으로 수행하는 방법이 하나의 대응기법이 될 수 있다. 특히, [14,15]에서는 CPU의 하드웨어 내부 레지스터에 마스터 키를 저장함으로써 콜드 부트 공격에 대하여 안전성을 입증할 수 있다.

Ⅶ. 결 론

물리적 특성을 활용하여 RAM 데이터의 값을 탈취할 수 있는 콜드 부트 공격은 최초 제안된 2008년 이후에 지속적으로 연구되고 있다. 콜드 부트 공격을 수행하기 위해서는 RAM의 표면 온도를 낮춰야 될 뿐만 아니라, 부트 시퀀스를 파악해야 공격을 수행할 수 있다. 또한, 물리적 특성으로 인해 RAM을 탈취한다고 해도 에러 비트가 존재하기 때문에, 비밀 키를 온전히 복구하기 위해서는 알고리즘 특성을 활용하여 그 후처리 하는 방법론도 연구되고 있다.

이를 기반으로 디스크 암호화 술루션과 같은 응용 어플리케이션의 비밀 키가 누출될 수 있음이 연구되어, 각 특성에 맞게 대응기법이 줄곧 제안되고 있다. 하지만, 하드웨어 스크램블러 같은 대응기법의 취약점이 밝혀지면서 지속적으로 대응기법의 연구가 필요한 실정이다. 특히, 라즈베리 파이처럼 하드웨어 칩으로 이루어진 장비에 취약점이 밝혀질 경우, 그 취약점을 보완하기에는 쉽지 않다. 그러므로 하드웨어 대응기법을 적용할 때는 많은 경우의 수를 고려해야 한다.

따라서 부트 시퀀스의 다양한 공격법과 알고리즘 상

의 복잡도를 줄이는 방법 및 대응기법이 지속적으로 연구될 것으로 사료된다.

참 고 문 헌

[1] Halderman, J. Alex, et al. "Lest we remember: cold-boot attacks on encryption keys." Communications of the ACM 52.5 (2009): 91-98.

[2] Tilo, Muller, Spreitzenbarth Michael, and Felix C. Freiling. "Frost: forensic recovery of scrambled telephones." Proceedings of the International Conference on Applied Cryptography and Network Security. 2014.

[3] Bauer, Johannes, Michael Gruhn, and Felix C. Freiling. "Lest we forget: Cold-boot attacks on scrambled DDR3 memory." Digital Investigation 16 (2016): S65-S74.

[4] Yitbarek, Salessawi Ferede, et al. "Cold boot attacks are still hot: Security analysis of memory scramblers in modern processors." 2017 IEEE International Symposium on High Performance Computer Architecture (HPCA). IEEE, 2017.

[5] Won, Yoo-Seung, et al. "Practical Cold boot attack on IoT device-Case study on Raspberry Pi." 2020 IEEE International Symposium on the Physical and Failure Analysis of Integrated Circuits (IPFA). IEEE, 2020.

[6] Kamal, Abdel Alim, and Amr M. Youssef. "Applications of SAT solvers to AES key recovery from decayed key schedule images." 2010 Fourth International Conference on Emerging Security Information, Systems and Technologies. IEEE, 2010.

[7] Paterson, Kenneth G., Antigoni Polychroniadou, and Dale L. Sibborn. "A coding-theoretic approach to recovering noisy RSA keys." International Conference on the Theory and Application of Cryptology and Information Security. Springer, Berlin, Heidelberg, 2012.

[8] Paterson, Kenneth G., and Ricardo Villanueva-Polanco. "Cold boot attacks on NTRU." International Conference on Cryptology in India. Springer, Cham, 2017.

[9] Albrecht, Martin R., Amit Deo, and Kenneth G. Paterson. "Cold boot attacks on ring and module lwe keys under the ntt." IACR Transactions on Cryptographic Hardware and Embedded Systems (2018): 173-213.

[10] Villanueva-Polanco, Ricardo. "Cold Boot Attacks on LUOV." Applied Sciences 10.12 (2020): 4106.

[11] Villanueva-Polanco, Ricardo. "Cold Boot Attacks on Bliss." International Conference on Cryptology and Information Security in Latin America. Springer, Cham, 2019.

[12] Polanco, Ricardo Villanueva. Cold Boot Attacks on Post-Quantum Schemes. Diss. Royal Holloway, University of London, 2019.

[13] Villanueva-Polanco, Ricardo, and Eduardo Angulo-Madrid. "Cold Boot Attacks on the Supersingular Isogeny Key Encapsulation (SIKE) Mechanism." Applied Sciences 11.1 (2021): 193.

[14] Müller, Tilo, Felix C. Freiling, and Andreas Dewald. "TRESOR Runs Encryption Securely Outside RAM." USENIX Security Symposium. Vol. 17. 2011.

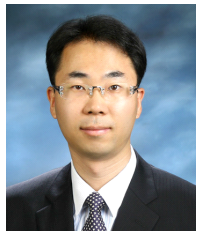
[15] Götzfried, Johannes, and Tilo Müller. "ARMORED: CPU-bound encryption for Android-driven ARM devices." 2013 International Conference on Availability, Reliability and Security. IEEE, 2013.

〈저 자 소 개〉

원 유 승 (Yoo-Seung Won)

학생회원

2012년 2월 : 국민대학교 수학과 졸업
2014년 2월 : 국민대학교 수학과 석사
2018년 2월 : 국민대학교 금융정보보안학과 박사
2018년 3월~2020년 11월 : 삼성전자 파운드리 사업부 IP개발팀, Staff Engineer
2019년 11월~현재 : 싱가포르 난양공과대학교 Temasek Laboratories, Research Scientiest
<관심분야> 정보보호, 부채널 분석, 오류 주입 분석, 블록암호/공개키 부채널 및 오류주입 대응기법, 머신 러닝 공격, 콜드 부트 공격



한 동 국 (Dong-Guk Han)

종신회원

1999년 2월 : 고려대학교 수학과 학사
2002년 2월 : 고려대학교 수학과 이학석사
2005년 2월 : 고려대학교 정보보호대학원 공학박사
2004년 4월~2005년 4월 : 일본 Kyushu Univ., 방문연구원
2005년 4월~2006년 4월 : 일본 Future Univ.-Hakodate, Post.Doc.
2006년 6월~2009년 2월 : 한국전자통신연구원 정보보호연구단 선임연구원
2009년 3월~현재 : 국민대학교 정보보안암호수학과 정교수
<관심분야> 공개키 암호시스템 안전성 분석 및 고속 구현, 부채널 분석 및 대응법 설계, IoT 정보보호 기술