ISBN : 000-00-000-0000-0

FREAK Attack(MITM) 취약점 분석 보고서

2015년

한국과학기술정보연구원
과학기술사이버안전센터

FREAK Attack(MITM) 취약점 분석 ..!!

괴짜(Freak)라 불리는 사이버보안 문제로 최근 인터넷이 떠들썩했다.

인터넷 서버와 웹 브라우저 사이의 암호화 기술을 조작해 각종 버그를 발생시키거나 데이터를 갈취할 수 있는 **Factoring attack on RSA-EXPORT Keys를 줄여서 'FREAK(괴짜)'**로 불리는 보안 취약점이 수면 위로 부상하면서 뜨거운 감자가 되었다.

미국 클린턴 행정부 시절('93~'01) 자국 소프트웨어를 수출할 때 반드시 암호화 수준을 **수출 등급**으로 낮추도록 하는 규제를 운영한 적이 있다고 한다. 당시로서는 수출 등급 암호화 기법도 충분히 안전하다고 생각했는데, 그동안 컴퓨터 성능이 월등히 좋아지고 해킹 기술이 날로 발전하면서 이제 실질적으로 보안을 위협할 만한 수준에 이르렀다.

수출 등급의 암호화 기법(512bit RSA)은 7시간 안에 서버와 클라이언트 사이의 데이터를 가로채 사용자의 비밀번호나 개인정보를 빼낼 수 있다는 사실이 확인됐다. 또한, 여러 웹 브라우저 중에서 애플의 사파리와 구글 안드로이드에 내장된 웹브라우저가 특히 취약한 것으로 알려졌다.

1. 공격 과정 및 특징

FREAK(Factoring Attack on RSA-EXPORT Key) 약자로 **Openssl s3_clnt c의 ssl3_get_key exchange 함수에서 발생하는 취약점**으로 공격자가 **MITM**(Man In The Middle Attacker)을 통해 최대 512bit RSA를 다운그레이드하여 수시간 내 정보를 유출시킬 수 있는 취약점이다.

미국 정부 사이트는 물론 **전 세계 주요 사이트*** 상당수가 이 취약점에 노출되어 언제라도 큰 보안사고로 이어질 수 있다는 지적이 나오고 있다.

* 국내 웹사이트도 1200곳 이상 포함되어 있다. 신한카드와 신한금융투자 등 금융기관은 물론이고, 올레 · LG유플러스 등 통신사, YTN · 매일경제 · 경향신문등 언론사 웹 사이트도 괴짜 버그에 취약한 것으로 드러났다.

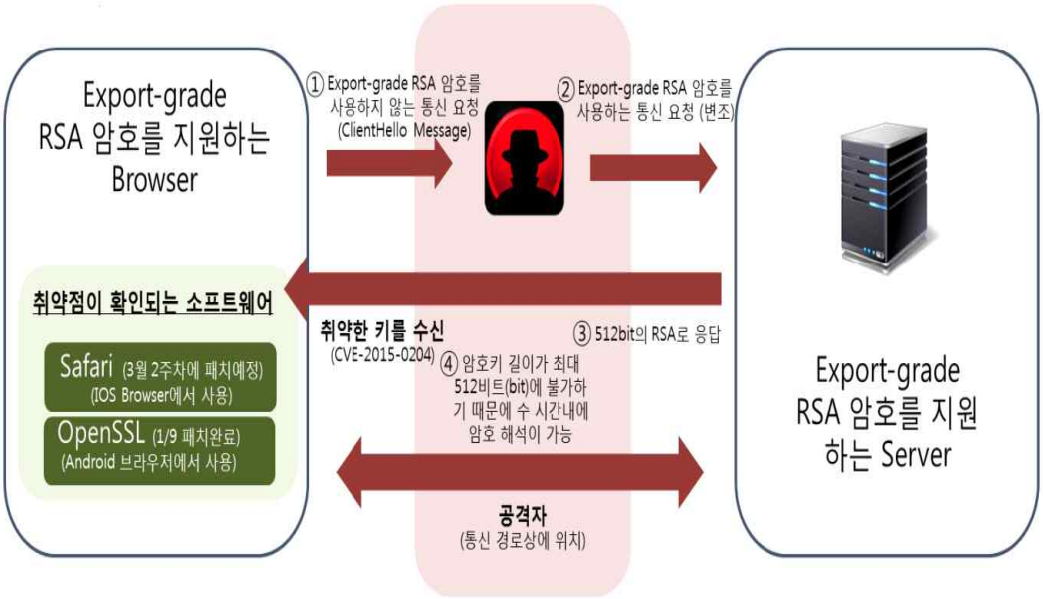
* 카페24, 에누리닷컴, 예스24, CJ몰, 알라딘, 인터파크투어 등 웹사이트도 위험에 노출돼 있다. “oo.go.kr”로 끝나는 웹 주소를 쓰는 정부기관 웹사이트 138곳, “oo.ac.kr’로 끝나는 국내 대학교 50곳도 목록에 들어 있다.

FREAK 공격(MITM) 공격에 영향을 받는 시스템은 <다음>과 같다.

- OpenSSL 0.9.8zd 미만
- OpenSSL 1.0.9 버전의 1.0.0p 미만
- OpenSSL 1.0.1 버전의 1.0.1k 미만

2. 상세 분석

FREAK Attack(MITM) 방식은 <다음>과 같다.



① Export-grade RSA 암호를 사용하지 않는 일반사용자는 서버 측 Client-Hello Message 요청

② MITM 공격자는 Export-grade RSA 요청을 위해 ClientHello Message 변조 후 서버측으로 전달

③ 서버는 512비트의 Export-grade RSA 키로 응답

④ 암호키 길이가 최대 512비트에 불가하여 수시간 내 해독 가능

FREAK Attack 경우 쉘(Shell) 파일(check_freak.sh)을 이용하여 취약점 여부를 확인 가능하며, 해당 쉘 파일의 경우 CV#-2015-2014의 취약점 스캔은 아니지만 Export-grade의 암호화를 지원하는지 여부를 확인함으로써 취약점 여부 확인이 가능하다.

* 다운로드 링크(check_freak.sh)

https://gist.github.com/martinseener/d50473228719a9554e6a/download#

<check_freak.sh 스크립트 구문 분석>

```
47  check_freak() {
48    # Get the information
49    CHK=$($OPENSSL s_client -host $1 -port $2 -cipher EXPORT < /dev/null 2>/dev/null)
50    # Check if there is an export cipher
51    echo $CHK | grep "Cipher is EXP" > /dev/null
52  }
```

```
#!/usr/bin/env bash

# check_freak.sh
# (c) 2015 Martin Seener

# Simple script which checks SSL/TLS services for the FREAK vulnerability (CVE 2015-0204)
# It will output if the checked host is vulnerable and returns the right exit code
# so it can also be used as a nagios check!

PROGNAME=$(basename $0)
VERSION="v0.2"
AUTHOR="2015, Martin Seener (martin@seener.de)"

# Set the timeout how long openssl can try to connect to the service
TIMEOUT=10

print_help() {
  echo ""
  echo "$PROGNAME is a small shell script which checks remote SSL/TLS services for the FREAK vulnerability (CVE 2015-0204)"
  echo "It will return if the service is vulnerable or not and exit with 0 (OK) or 2 (CRIT) so it can be used as"
  echo "a nagios check too"
  echo ""
  echo "Usage: ./$PROGNAME <IP or Hostname> <port>"
  echo "Example: ./$PROGNAME www.google.com 443"
  echo ""
}

initialize() {
  if [ -z "$1" ]; then
    echo "The Hostname/IP Argument is missing!"
    echo ""
    print_help
    exit 3
  fi
  if [[ ! $2 =~ ^[0-9]+$ ]] || [ $2 -eq 0 ] || [ $2 -gt 65535 ] ; then
    echo "The Port argument must be a positive integer value starting at 1 up to 65535"
    echo ""
    print_help
    exit 3
  fi
  OPENSSL=$(which openssl)
  if [ "$OPENSSL" == "" ]; then
    echo "Cannot find openssl! Aborting!"
    echo ""
    print_help
    exit 3
  fi
}

check_freak() {
  # Get the information (we use the strange sleep/kill method here because timeout doesn't work on OSX by default!)
  CHK=$( $OPENSSL s_client -connect $1:$2 -cipher EXPORT < /dev/null 2>/dev/null & sleep $TIMEOUT; kill $! 2>/dev/null )
  if [ "$CHK" == "" ]; then
    echo "UNKNOWN - Timeout connecting to $1 on port $2"
    exit 3
  fi
  # Check if there is an export cipher
  echo $CHK | grep "Cipher is EXP" > /dev/null
}

case "$1" in
  --help|-h)
    print_help
    exit 3;;
  *)
    ;;
esac

# Initialize
initialize $1 $2

# Do the check
check_freak $1 $2

# Return the result
if [ $? -eq 1 ]; then
  echo "OK - $1 on port $2 is PROBABLY NOT vulnerable to FREAK (CVE 2015-0204)"
  exit 0
else
  echo "CRITICAL - $1 on port $2 IS PROBABLY VULNERABLE to FREAK (CVE 2015-0204)"
  exit 2
fi
```

* 해당 쉘 파일을 사용하여 암호화방식 Export-grade 암호화 방식 지원 가능 여부를 체크하여 해당 취약점 점검이 가능하다.

<스크립트 사용법>

```
명령어: check_freak.sh 점검 도메인(IP) 점검 Port
명령어 예) #sh ./check_freak www.test_freakattack.com 443
```

<스크립트 점검 결과 확인>

```
File  Edit  View  Search  Terminal  Help
[root@localhost Desktop]# sh ./check_freak.sh   점검 도메인 or IP     443 (점검 포트)
CRITICAL - The Service at      점검 도메인      on port 443 IS PROBABLY VULNERABL
 to FREAK (CVE 2015-0204)
```

일반 사용자 경우 https://freakattack.com 사이트 접속을 통해 <다음>과 같이 웹 브라우저 취약 여부를 확인할 수 있다(클라이언트 취약점 확인 방법).

<Windows PC 환경에서의 Chrome Browser 점검 결과>

* 해당 윈도우 PC 환경에서 Chrome Browser 점검 결과는 취약하지 않습니다.

<Google Android 기본 Browser 점검 결과>

* 안드로이드 기본 Web Browser를 점검한 결과 취약성을 확인하였습니다.

<Apple IOS Safari 점검 결과>

* IOS 기본 Web Browser를 점검한 결과 취약성을 확인하였습니다.

3. 취약점 대응 방법 및 해결 방안

<운영체제(OS) 레벨> 취약한 Open SSL 버전 업데이트

- OpenSSL 0.9.8 버전 ⇒ OpenSSL 0.9.8zd 업데이트
- OpenSSL 1.0.9 버전의 1.0.0 버전 ⇒ OpenSSL 1.0.9 버전의 1.0.0p 업데이트
- OpenSSL 1.0.1 버전의 1.0.1 버전 ⇒OpenSSL 1.0.1 버전의 1.0.1k 업데이트

<Client 레벨>

- 일반 사용자 경우 기본 브라우저 문제 해결 시까지 크롭 등 타 브라우저 사용
- 취약 Web Browser(Google Android 기본 Browser, Apple Safari) 패치 예정

<Network 레벨>

- “DTLS_Excessive_Handshakes(탐지패턴)” 시그니처로 탐지 · 대응

참고 링크

https://gist.github.com/martinseener/d50473228719a9554e6a

https://freakattack.com/

http://blog.cryptographyengineering.com/2015/03/attack-of-week-freak-or-factoring-nsa.html

http://www.washingtonpost.com/blogs/the-switch/wp/2015/03/03/freak-flaw-undermines-security-for-apple-and-google-users-researchers-discover/

http://d.hatena.ne.jp/Kango/20150304/1425448983

http://thehackernews.com/2015/03/freak-openssl-vulnerability.html