OpenSSL 취약점(HeartBleed) 대응 방안 권고

'14.4.14(월) / KISA 취약점분석팀

□ 개요

o 통신 구간 암호화를 위해 많이 사용하는 OpenSSL 라이브러리에서 서버에 저장된 중요 메모리 데이터가 노출되는 HeartBleed라고 명명된 심각한 버그가 발견되어 시스템 및 소프트웨어에 대한 신속한 취약점 조치를 권고

□ 취약점 정보

o 시스템 메모리 정보 노출 취약점

- CVE-2014-0160 (2014.04.07.)

o 영향 받는 버전

- OpenSSL 1.0.1 ~ OpenSSL 1.0.1f
- OpenSSL 1.0.2-beta, OpenSSL 1.0.2-beta1

o 영향 받는 시스템 및 소프트웨어

- 취약한 OpenSSL 버전이 탑재된 시스템

  ※ 서버(웹서버, VPN 서버 등), 네트워크 장비, 모바일 단말 등 다양한 시스템이 해당될 수 있음

- 취약한 OpenSSL 라이브러리가 내장된 소프트웨어 제품

o 영향 받지 않는 소프트웨어

- OpenSSL 0.9.x 대 버전
- OpenSSL 1.0.0 대 버전
- OpenSSL 1.0.1g

□ 취약점 내용

o OpenSSL 암호화 라이브러리의 하트비트(Heartbeat)라는 확장 모듈에서 클라이언트 요청 메시지를 처리할 때 데이터 길이 검증을 수행하지 않아 시스템 메모리에 저장된 64KB 크기의 데이터를 외부에서 아무런 제한 없이 탈취할 수 있는 취약점

※ 하트비트 : 클라이언트와 서버 간의 연결 상태 체크를 위한 OpenSSL 확장 모듈

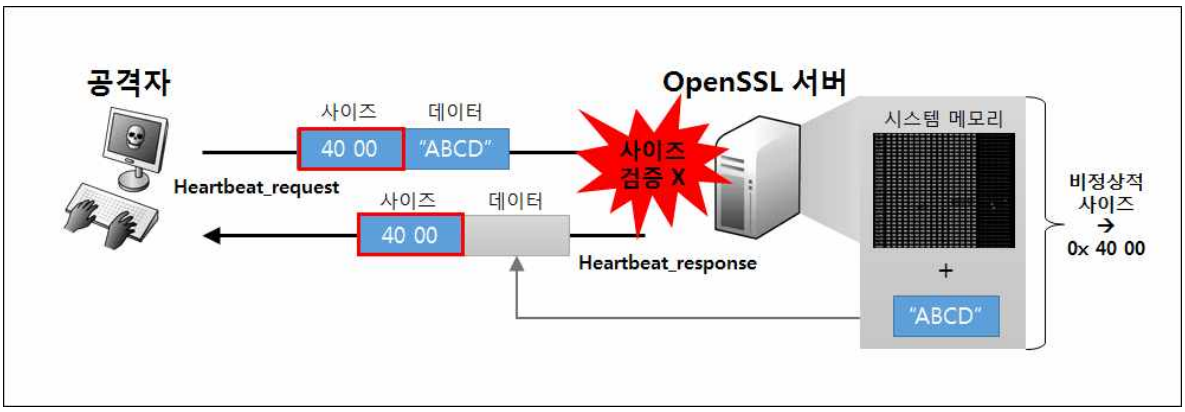
□ 공격 형태

o 본 취약점은 원격에서 발생 가능한 취약점으로, 공격자는 메시지 길이 정보가 변조된 HeartBeat Request 패킷을 취약한 OpenSSL 버전을 사용하는 서버에 전송할 경우, 정해진 버퍼 밖의 데이터를 공격자에게 전송하게 되어 시스템 메모리에 저장된 개인정보 및 인증 정보 등을 탈취할 수 있음

공격 방법

① **(전송)** 공격자는 하트비트 패킷 헤더에서 페이로드 길이 필드를 조작하여 서버에 전송

② **(응답)** 서버는 공격자가 요청한 길이(최대 64KB)만큼 메모리에서 데이터를 추출하여 공격자에게 응답



※ 노출 가능한 정보: SSL 서버 비밀키, 세션키, 쿠키 및 개인정보(ID/PW, 이메일주소 등) 등

※ 노출되는 정보는 서비스 환경에 따라 다를 수 있음

□ 취약점 확인 절차

o 점검 대상 선정

- 서버, 네트워크, 보안 장비 등의 시스템에서 OpenSSL 설치 여부 확인
- 웹 서버의 경우 서브 도메인을 운영하는 시스템도 점검 대상에 포함

  ※ 서브 도메인 : mail.example.com, blog.example.com 등
- 시스템뿐만 아니라 소프트웨어 제품 자체에 OpenSSL 라이브러리가 내장되어 있을 경우 버전 확인 후 점검 대상에 포함

o 취약점 노출 여부 확인 방법

- 명령어를 통한 OpenSSL 버전 정보 확인
  - openssl이 설치된 시스템에서 아래 명령어를 입력하여 취약점에 영향 받는 버전을 사용하는지 확인

```
root@server:~# openssl version –a
OpenSSL 1.0.1 14 May 2012
| 취약 버전 정보 |
```

- OpenSSL 하트비트(HeartBeat) 활성화 여부 확인
  - 취약한 버전의 OpenSSL을 사용하는 시스템 중 HeartBeat 기능 사용 여부 확인 방법 (단, 패치된 최신 버전(1.0.1g)은 활성화 여부를 확인할 필요 없음)
  - 취약한 버전이 HeartBeat를 사용하지 않은 경우 취약점에 영향 받지 않음

```
root@server:~# openssl s_client –connect domain.com:443 –tlsextdebug –debug –state | grep
                -i heartbeat
```

※ 명령어 실행 방법 : domain.com에 점검 대상 URL 정보로 수정

※ HeartBeat 기능이 활성화되어 있는 경우 heartbeat 문자열이 검색됨

```
TLS server extension "heartbeat" (id=15), len=1
0000 - 01                                                .
read from 0x95cb888 [0x95d0e33] (5 bytes => 5 (0x5))
0000 - 16 03 02 0b cc                                    .....
read from 0x95cb888 [0x95d0e38] (3020 bytes => 3020 (0xBCC))
0000 - 0b 00 0b c8 00 0b c5 00-05 9d 30 82 05 99 30 82   ..........0...0.
0010 - 04 81 a0 03 02 01 02 02-08 11 bb ec db 00 00 39   ...............9
0020 - d0 30 0d 06 09 2a 86 48-86 f7 0d 01 01 05 05 00   .0...*.H........
0030 - 30 5e 31 0b 30 09 06 03-55 04 06 13 02 4b 52 31   0^1.0...U....KR1
0040 - 12 30 10 06 03 55 04 0a-0c 09 43 72 6f 73 73 43   .0...U....CrossC
```

※ HeartBeat 기능이 활성화되지 않은 경우 heartbeat 문자열이 검색되지 않음

```
TLS server extension "session ticket" (id=35), len=0
read from 0x9349888 [0x934ee33] (5 bytes => 5 (0x5))
0000 - 16 03 02 13 6f                                    ....o
read from 0x9349888 [0x934ee38] (4975 bytes => 4975 (0x136F))
```

- OpenSSL에서 사용하는 소스코드 확인

- OpenSSL 취약점이 발생된 소스코드를 열람하여 아래와 같이 보안 패치 코드가 추가되었는지 확인을 통해 취약 여부 판별
- 패치된 버전에서는 아래와 같이 사용자 요청 메시지에 대한 길이를 검사하도록 코드가 추가됨

취약점 코드(ssl/d1_both.c)	**보안 패치 코드**(ssl/d1_both.c)
	/* Read type and payload length first */ if (1 + 2 + 16 > s->s3->rrec.length) return 0;

hbtype = *p++;
 	
n2s(p, payload);
 	
pl = p;
	hbtype = *p++;
	
	n2s(p, payload);
	
	if (1 + 2 + payload + 16 >
	  s->s3->rrec.length)
	  return 0;
	
	pl = p;

※ 참고 사이트 : http://git.openssl.org/gitweb/?p=openssl.git;a=commitdiff;h=96db902

- KISA(한국인터넷진흥원)를 통한 취약점 여부 확인

※ 자체적인 확인이 어려울 경우 KISA 전문가로부터 점검을 요청

성명	연락처	메일주소
손기종	02-405-5223	skj@kisa.or.kr
김유홍	02-405-5488	uhong@kisa.or.kr

□ 해결 방안

<시스템 측면 대응 방안>

o OpenSSL 버전을 1.0.1g 버전으로 업데이트

o 서비스 운영환경에 따른 소프트웨어 의존성 문제를 고려하여 업데이트 방법을 선택하고 반드시 먼저 테스트 수행

※ 아래 보안 패치 방법은 CentOS/Fedora 및 Ubuntu의 예제로 각 운영체제 별로 업데이트 방법이 상이할 수 있음

- CentOS/Fedora

· 전체 시스템 업데이트(OpenSSL을 포함한 시스템 내의 소프트웨어 전부 업데이트)

```
yum update
```

· OpenSSL 업데이트

```
sudo pacman -Syu
```

- Ubuntu

· 전체 시스템 업데이트 (OpenSSL을 포함한 시스템 내의 소프트웨어 전부 업데이트)

```
sudo apt-get update
sudo apt-get dist-upgrade
```

· OpenSSL 업데이트

```
sudo apt-get install --only-upgrade openssl
sudo apt-get install --only-upgrade libssl1.0.0
```

o 운영환경의 특수성 때문에 패키지 형태의 업데이트가 어려운 경우, Heartbeat를 사용하지 않도록 컴파일 옵션을 설정하여 재컴파일 가능

- OpenSSL 소스코드를 처음 다운받아 컴파일하는 경우 라이브러리 의존성 문제가 발생하여 추가적인 작업이 필요한 경우도 존재

```
./config –DOPENSSL_NO_HEARTBEATS
make depend
make
make install
```

<네트워크 보안 장비 측면 대응 방안>

o 취약점 공격 탐지 및 차단 패턴 적용

- 아래의 Snort 탐지 룰(rule)을 참고하여 침입탐지시스템 및 침입차단 시스템에 패턴 업데이트 적용 권고

※ 차단 패턴 적용은 서비스 및 네트워크 영향도를 고려하여 적용

[OpenSSL HeartBeat 취약점 탐지 Snort Rule]
- **SSL 서비스 포트에 대해 공격 요청시 전송되는 |18 03 ??| 탐지 패턴**

```
alert tcp any any < > any
[443,465,563,636,695,898,989,990,992,993,994,995,2083,2087,2096,2484,8443,8883,9091]
(content:"|18 03 00|"; depth: 3; content:"|01|"; distance: 2; within: 1;
content:!"|00|"; within: 1; msg: "SSLv3 Malicious Heartbleed Request V2";
sid: 1;)
alert tcp any any < > any
[443,465,563,636,695,898,989,990,992,993,994,995,2083,2087,2096,2484,8443,8883,9091]
(content:"|18 03 01|"; depth: 3; content:"|01|"; distance: 2; within: 1;
content:!"|00|"; within: 1; msg: "TLSv1 Malicious Heartbleed Request V2";
sid: 2;)
alert tcp any any < > any
[443,465,563,636,695,898,989,990,992,993,994,995,2083,2087,2096,2484,8443,8883,9091]
(content:"|18 03 02|"; depth: 3; content:"|01|"; distance: 2; within: 1;
content:!"|00|"; within: 1; msg: "TLSv1.1 Malicious Heartbleed Request V2";
sid: 3;)
```

※ 출처 : FBI

<서비스 관리 측면 대응 방안>

o 서버 측 SSL 비밀키(Secret Key)가 유출되었을 가능성을 배제할 수 없기 때문에 인증서를 재발급 받는 것을 운영자가 검토

o 취약점에 대한 조치가 완료된 후 사용자들의 비밀번호 재설정을 유도하여 탈취된 계정을 악용한 추가 피해를 방지하는 방안도 고려

※ 야후 메일의 경우 접속한 사용자의 계정정보가 유출되는 것이 확인되어 현재 비밀번호 변경을 안내 중