NTP 서버 보안 가이드

침해사고분석단 취약점점검팀

김정호 선임연구원 (jungho@kisa.or.kr)

2015. 1.

KISA 한국인터넷진흥원
Korea Internet & Security Agency

목 차

제1장 개요 ····· 1

제2장 NTP 취약점 ····· 4

1. NTP 분산 서비스 거부 취약점 (CVE : 2013-5211) ····· 5
 1) NTP Amplification ····· 5
 2) 취약점 내용 ····· 6

2. NTP 스택 오버플로우 취약점 (CVE : 2014-9295) ····· 7
 1) 스택 오버플로우 ····· 7
 2) 취약점 내용 ····· 7

제3장 대응방안 ····· 8

1. 공통 ····· 9
 1) NTP 서비스 필요성 검토 ····· 9
 2) 사설망 NTP 서버 이용 권고 ····· 9

2. 서버 보안조치 ····· 9
 1) NTP 버전 업데이트 ····· 9
 2) 설정파일 수정 ····· 10
 3) 취약점 확인 ····· 10

3. 네트워크 및 보안장비 보안조치 ····· 12
 1) 방화벽 설정 ····· 12
 2) 보안장비 설정 ····· 12

4. 보안을 고려한 제품 설계 ····· 13

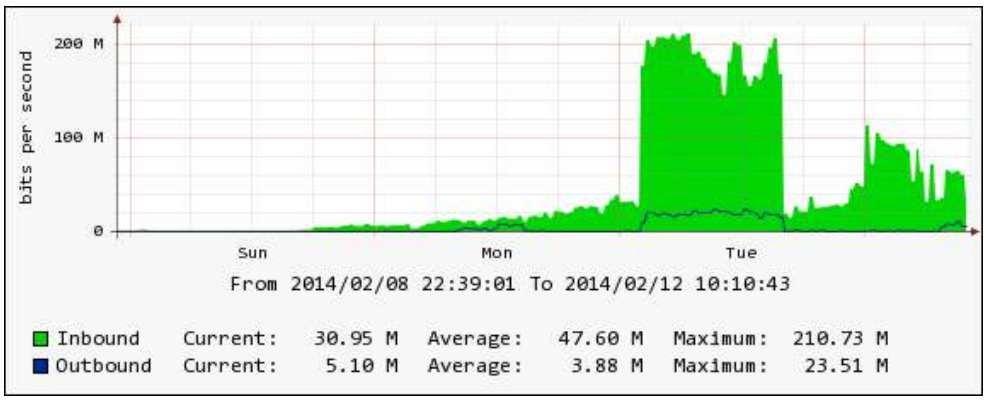
제1장
개요

제1장 개요

NTP(Network Time Protocol)는 현재까지 쓰여 온 가장 오래된 인터넷 프로토콜 가운데 하나이며, 네트워크를 통해 컴퓨터 시스템 간 시간 동기화를 위해 사용된다.

앞서 NTP 패킷이 인증을 위한 암호화 서명이 되어 있지 않아 MITM(Man In the Middle) 공격이 가능한 점과 메시지를 변조하여 클라이언트 PC의 시간을 변경할 수 있는 점 등 NTP가 사용된 오랜 기간 동안 몇몇 문제점이 발견되었으나 실제 그 파급도는 높지 않았다. 하지만 최근 발표된 NTP 분산서비스 거부 취약점(CVE-2013-5211)과 스택 오버플로우 취약점(CVE-2014-9295)의 경우 전 세계적으로 큰 이슈가 되고 있다.

<그림 1-1> NTP 취약점을 이용한 공격 트래픽 예



일례로 2014년 2월 미국의 보안업체 클라우드플레어의 고객사를 대상으로 NTP 취약점을 이용한 역대 최대 규모인 400Gbps의 DDoS 공격이 발생한 바 있으며, 국내에서도 동일한 기법의 DDoS 공격이 발견되는 등 전 세계적으로 대규모 DDoS 공격을 위한 새로운 위협으로 부상되고 있다.

NTP 취약점의 위험성은 해당 서비스가 PC 및 서버에만 국한되어 실행되지 않는다는 점에 있다. 지난 5월 국내 온라인 게임업체를 대상으로 한 DDoS 공격이 발생하여 KISA에서는 그 진원지를 확인한 적이 있었는데, 공격에 사용된 시스템의 경우 좀비PC가 아닌 냉난방 관리용 셋톱박스가 원인으로 밝혀졌다. 대학 전체 냉난방을 효율적으로 관리하려고 설치한 장비로, 공격자는 이 셋톱박스에 내장된 NTP의 취약점을 악용하여 공격에 사용한 것이다. 이처럼 사물인터넷 시대에 들어서며 다양한 장비에서 NTP 서비스를 제공하고 있으나, 실제 보안 설정은 미흡하여 공격자들에게 좋은 자원이 되고 있는 실정이다.

<그림 1-2> NTP 취약점을 이용, DDoS 공격에 악용된 냉난방 셋탑박스

이에 본 가이드에서는 NTP 서비스의 주요 취약점에 다루고, 해당 취약점에 대한 점검 항목 및 대응책을 제공함으로써 사고를 예방하고 발생할 수 있는 피해를 최소화하고자 한다.

제2장
NTP 취약점

제2장 NTP 취약점

1. NTP 분산 서비스 거부 취약점 (CVE : 2013-5211)

1) NTP Amplification

NTP 분산 서비스 거부 취약점을 이해하기 위해서는 NTP 증폭(Amplification)에 대한 이해가 우선적으로 필요하다. NTP 증폭이란 분산서비스 거부(DDoS) 기법 중 하나로 공개된 NTP 서버를 통해 증폭된 UDP 패킷을 발송, 공격 대상 시스템이 정상적으로 서비스를 할 수 없도록 하는 것이다.

NTP 증폭 공격의 가장 기본적인 형태는 공격자의 IP 주소, 즉 Source IP를 공격 대상의 IP로 변조하여 NTP 서버에 다수의 monlist 명령을 요청하는 것이다. 그러면 NTP 서버는 호스트 리스트가 포함된 패킷을 변조된 IP 주소로 응답하게 된다. 이 응답 패킷은 요청 시 발생하는 패킷보다 훨씬 크게 증폭되며, 이 증폭된 트래픽은 공격 대상 서버로 향하게 되어 곧 서비스의 부하를 가져오게 된다.

NTP 증폭의 경우 근본적으로 반사 공격(Reflection Attack)에 속한다. 반사 공격은 서버의 응답 패킷을 변조된 IP 주소로 유도하도록 한다. 공격자는 자신의 IP를 공격 대상 서버 IP로 변조하여 패킷을 요청하고 NTP 서버는 그 변조된 IP주소로 응답하게 된다.

반사 공격의 경우 응답 데이터 그 자체는 정상이기 때문에 다양한 위험성이 존재한다. 여기에 증폭 기법까지 접목 되었을 때 그 위험성은 말로 설명할 수 없을 정도이다. 전형적으로 사용되는 DNS 증폭 시 응답 패킷 사이즈는 요청 패킷 사이즈에 비해 8배에서 70배 증폭된다.

NTP 증폭의 경우 더 높은 비율로 패킷을 증폭하여 전송할 수 있다. 요청 패킷 대비 응답 패킷의 비율이 1:20에서 1:200 혹은 그 이상이 될 수도 있다. 이로 인해 공격자는 Open NTP Project 등을 통해 공개된 NTP 서버를 이용, 고대역폭의 대용량 DDoS 공격을 수행할 수 있다.

2) 취약점 내용

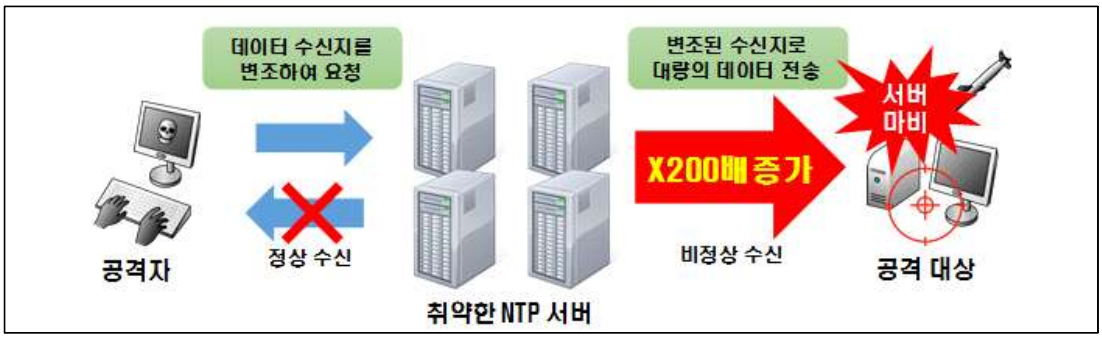
NTP 서비스 데몬 ntpd 의 4.2.7 이전 버전에서는 쿼리를 통해 해당 접속 호스트를 모니터링할 수 있는 기능을 지원한다. monlist 라는 이름의 이 명령어를 통해 NTP 서버에 쿼리를 요청하여 최근 접속한 600개 호스트 리스트를 응답 받을 수 있다.

'monlist' 의 경우 요청 IP에 대한 검증 작업을 수행하지 않기에 이를 이용하여 특정 대상을 목표로 한 DDoS 공격을 수행할 수 있으며, 공격 방법은 다음과 같다.

공격 방법

① [변조] 공격자는 IP 주소를 공격대상의 IP로 패킷을 변조
② [전송] 취약한 다수의 NTP 서버를 대상으로 monlist 이용한 다수의 쿼리 전송
③ [증폭] 쿼리를 수신한 NTP 서버는 증폭된 패킷을 변조된 IP(공격대상)로 응답
④ [장애] 공격대상 서버는 증폭된 다수의 쿼리 수신으로 네트워크 대역폭 고갈

<그림 2-1> NTP 취약점을 이용한 DDoS 공격



2. NTP 스택 오버플로우 취약점 (CVE : 2014-9295)

1) 스택 오버플로우

스택 오버플로우(Stack overflow)는 스택 포인터가 스택의 경계를 넘어설 때 발생한다. 호출 스택은 제안된 양의 주소 공간을 이루며 대개 프로그램 시작 시 결정된다. 프로그램이 호출 스택에서 이용 가능한 공간 이상을 사용하려고 할 때, 스택이 오버플로우 된다고 하며 이 경우 일반적으로 프로그램 충돌이 발생하게 된다.

이러한 오버플로우 취약점의 경우 원격 쉘코드 실행 등 다양한 형태의 공격으로 이어질 수 있어 그 위험성이 높다.

2) 취약점 내용

2014년 12월 구글의 'Stephen Roettger'는 ntpd 4.2.8 이전 버전에서 스택 오버플로우를 발생시킬 수 있는 취약점(CVE-2014-9295)을 발표하였다. 원격에서 인가되지 않은 사용자가 ntpd 관련 함수인 'crypto_recv()', 'ctl_putdat()', 'configure()'를 통해 스택 오버플로우를 발생시킬 수 있다는 내용으로 앞서 발표된 분산 서비스 거부 취약점에 이어 NTP에 대한 보안이 다시 이슈화되었다.

다만 해당 취약점의 경우 함수별로 오버플로우가 발생하기 위한 전제 조건이 존재한다. 'cryto_recv()' 함수의 경우 Autokey 인증이 활성화되어 있을 경우에만 취약점을 통하여 버퍼 오버플로우가 가능하나, 이 인증 기능은 기본으로 비활성화 되어 있어 실제 해당 기능을 강제 활성화한 시스템만이 취약점의 영향을 받게 된다.

'ctl_putdata()' 함수의 경우에도 신뢰되지 않은 호스로부터 컨트롤 메시지를 받는 것이 허용되어 있을 경우 버퍼 오버플로우가 발생 가능하나, 기본설정으로 로컬호스트에서만 해당 메시지를 받게 되어 있는 관계로 이 경우 로컬에서만 익스플로잇이 가능하다는 제약 조건이 붙게 된다. 마지막으로 'configure()' 함수의 경우 익스플로잇을 위해서는 추가적인 인증이 필요하다는 이슈가 제기되고 있는 상황이다.

제3장
대응 방안

제3장 대응 방안

1. 공통

1) NTP 서비스 필요성 검토

다수의 업체 및 기관에서 NTP 서비스를 설치 후 실제 사용하지 않는 케이스가 확인된다. 이렇게 방치된 서비스의 경우 공격자에 의해 악용될 소지가 다분하기에, 보안 정책상 운영이 반드시 필요한 서비스가 아닐 경우 제거하는 것이 바람직하다.

시스템 담당자들은 보유 서버 중 NTP 서비스가 실행중인 서버를 파악하여야 하며, 서비스 이용이 지속적으로 필요한지 여부에 대해 검토를 거친 후 반드시 필요하지 않을 경우 해당 서비스를 제거하여야 한다.

운영체제 내 설치된 패키지 명령에 따라 제거 명령이 조금씩 상이하나 'apt-get remove' 혹은 'rpm -e', 'yum remove' 등의 명령을 이용, 파라미터 값으로 ntp 혹은 ntpd를 지정하여 NTP 서비스를 제거할 수 있다.

2) 사설망 NTP 서버 이용 권고

업무상 필요로 인해 NTP 서비스를 구축 및 운영하여야 한다면 사설 혹은 사내망에 NTP 서버를 두는 것이 안전하다. DMZ 구간에 서버가 위치할 경우 외부로부터의 접속에 노출되어 공격에 악용될 소지가 있다. 가급적 외부에서 접속할 수 없는 사설 혹은 사내망에 NTP 서버를 두고 운영하여 외부의 위협으로부터 보호할 수 있도록 한다.

2. 서버 보안조치

1) NTP 버전 업데이트

ntpd 4.2.6 이하 버전의 경우 NTP 분산 서비스 거부 취약점이 존재하며, ntpd 4.2.7 이하 버전에는 원격코드실행을 포함한 다중 취약점이 존재한다. 이를 해결할 수 있는 가장 간단한 방법으로는 4.2.8 이상 버전으로 업데이트를 수행하는 것이다.

아래 명령을 실행하여 현재 시스템에 설치된 ntpd 버전을 확인할 수 있다.

```
ntpd -version
```

만약 설치된 버전이 권고 버전보다 낮을 경우 아래 URL에서 최신 버전의 ntpd를 다운로드 받아 설치하도록 한다.(다운로드 URL : http://www.ntp.org/downloads.html)

2) 설정파일 수정

업데이트가 가장 안전한 방법이긴 하나, 경우에 따라 업데이트를 수행할 수 없는 경우가 발생할 수 있다. 이럴 경우 NTP 분산 서비스 거부 취약점을 제거하기 위해 NTP 설정 파일 "ntp.conf 내 "disable monitor" 를 삽입, NTP 서비스를 재시작하는 방법으로 monlist 기능을 비활성화 할 수 있다.

또 다른 방법으로는 아래와 같이 "ntp.conf" 파일의 "restrict default" 항목에 "noquery" 를 추가함으로 monlist 기능을 비활성화 하는 동시에 스택 오버플로우 취약점도 함께 해결할 수 있다.

```
restrict default kod nomodify notrap nopeer noquery
restrict -6 default kod nomodify notrap nopeer noquery
```

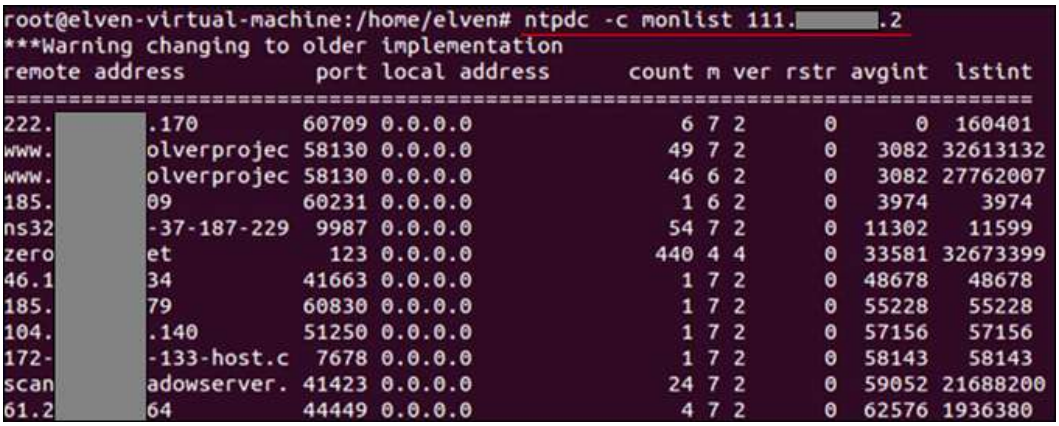
3) 취약점 확인

분산 서비스 거부 취약점에 영향을 받는 서버인지 또는 적절하게 조치가 되었는지에 대해 확인하기 위해 원격에서 아래 명령어를 실행해보도록 한다.

```
ntpdc –c monlist <점검 대상 NTP 서버 IP>
```

위 명령어를 실행하였을 때 timed out 메시지가 출력되면 이는 취약점에 영향을 받지 않는 것이나, 아래와 같이 리스트가 출력되면 조치가 필요한 상황으로 볼 수 있다.

<그림 3-1> NTP 분산 서비스 거부 취약점 확인방법



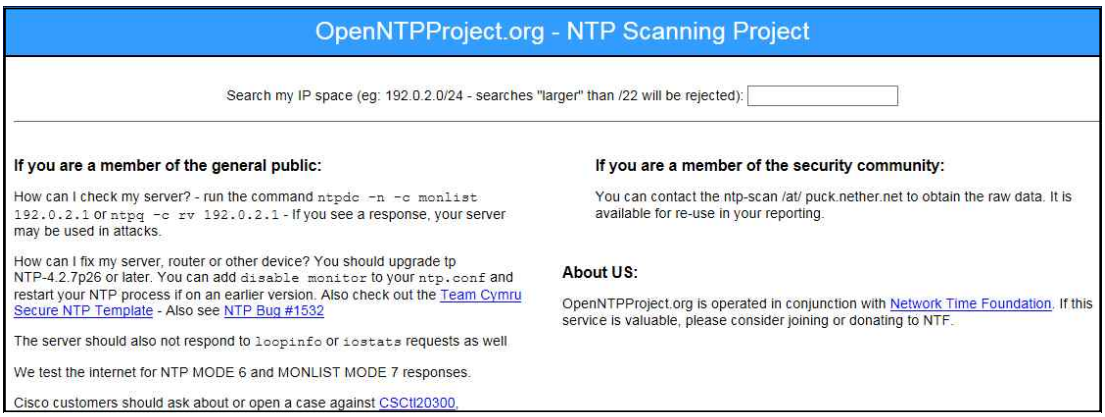
보안 스캐너 "Nmap" 을 이용하여 분산 서비스 거부 취약점을 점검할 수도 있다. "Nmap" 에서는 NSE 스크립트를 제공하는데, 이 중 "ntp-monlist" 를 통해 monlist 활성화 여부에 대한 확인이 가능하다.

```
nmap -sU -pU:123 -Pn -n —script=ntp-monlist <점검 대상 NTP 서버 IP>
```

위 명령어를 실행하였을 때 ntp-monlist 관련 리스트가 출력된다면 취약점이 존재하는 것으로 볼 수 있다.

이외에도 "OpenNTPProject" 를 통해 특정 네트워크에서 실행중인 NTP 서버 리스트를 확인할 수 있으며, 이 리스트를 점검에 활용할 수도 있다. (URL : http://openntpproject.org)

<그림 3-2> OpenNTPProject 사이트 화면



3. 네트워크 및 보안장비 보안조치

1) 방화벽 설정

방화벽을 통해 외부 트래픽으로부터 NTP 유입을 접근 통제할 수 있다. iptables 명령어를 이용하여 인가된 IP에 대해서만 UDP 123 포트로 접근을 허용하도록 하는 것이 바람직하다.

또한 100 byte 이상 NTP 패킷에 대한 차단을 설정하도록 한다. 통상적으로 NTP 패킷은 100byte 이하(48byte)이며 공격 패킷은 400 byte 내외(주로 monlist)이므로 패킷 크기 값을 통해 아래와 같이 제한할 수 있다.

```
iptables -A OUTPUT -p udp --sport 123 -m length --length 100: -j DROP
```

이외에도 'ntpquery' 에 대한 'rate-limit' 을 설정하는 방안이 있는데, 동일 IP에서 1분 이내 3번 이상의 쿼리 발생 시 차단하도록 설정한다.

```
iptables -A INPUT -p udp --dport 123 -m recent --set --name ntpquery

iptables -A INPUT -p udp --dport 123 -m recent --name ntpquery --rcheck
--seconds 60 --hitcount 3 -j DROP
```

2) 보안장비 설정

IDS와 IPS의 경우 아래와 같이 NTP monlist 패킷에 대한 시그니처 기반 필터링을 적용, 분산 서비스 거부 취약점을 이용한 공격을 방어할 수 있다.

```
MON_GETLIST-|00 03 2a|
```

Anti-DDoS 장비의 경우에는 아래와 같이 설정 변경을 통해 보안을 강화하도록 한다.

① 비정상 UDP 패킷 차단

- 비정상적인 UDP Header 값을 가진 패킷을 차단하도록 한다.

② LAND2 공격 UDP 패킷 차단

- 출발지와 목적지 포트가 동일한 UDP 패킷을 차단하도록 한다.
- 단 NTP 서버를 운영하지 않는 기관 및 업체에서 적용이 가능하다.

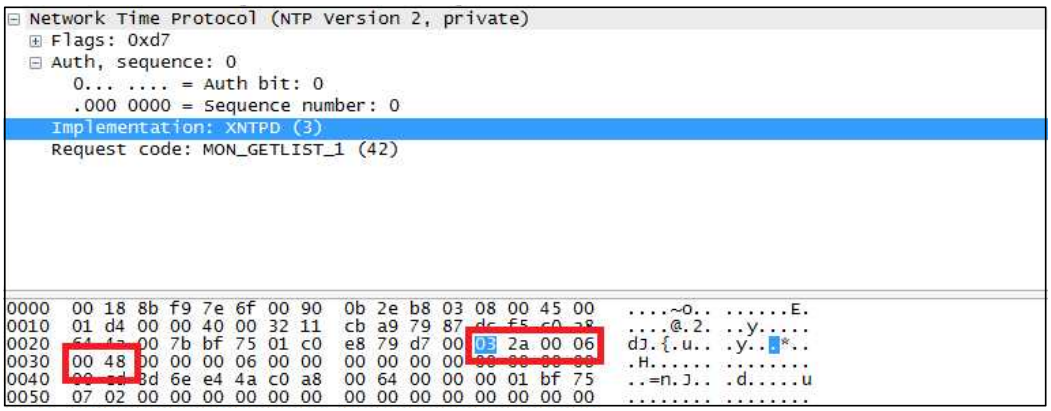
③ 소스 IP 기반의 Flooding 방어 설정

- UDP PPS 임계치(10 PPS)를 넘은 Source IP를 Black list에 등록한다.

④ NTP 시그니처 방어 설정

- UDP 프로토콜의 목적지 포트 기준 any에 대해 NTP monlist 요청 패킷의 헥사값 (032a00060048)을 시그니처로 등록한다.

<그림 3-3> NTP monlist 요청 패킷



4. 보안을 고려한 제품 설계

사물인터넷이 활성화됨에 따라 다양한 기기가 출시되고 있으며, 이 중 상당수에서 NTP 서비스가 실행되고 있다. 이러한 기기들의 경우 공격에 취약할 뿐만 아니라 취약점이 확인되더라도 신속하게 보안 패치를 적용하기가 어려운 문제점이 존재한다.

이에 제조사는 제품을 설계하는 초기 단계부터 보안을 고려하여 공격에 악용되지 않도록 주의하여야 한다. 제품 운용에 있어 NTP 서비스가 반드시 필요하지 않을 경우 해당 서비스를 설치하지 않도록 하며, 반드시 필요할 경우 차후 해당 서비스에 대한 보안 패치가 지속적으로 이루어질 수 있도록 설계 후 설치하여야 할 것이다.