PROJECT, STS

LET’S ENCRYPT USER GUIDE BOOK V1.0

PROJECT, STS

LET’S ENCRYPT USER GUIDE BOOK V1.0

목차

작성자 소개

위 가이드는

FEDORA23,

DEBIAN 8,

WINDOWS SERVER 2012 R2(예정),

RASPBERRY PI(예정)

웹 서버 사용자 모두를 위한 가이드북 입니다.

저자 소개

팀장: 김현정(세종대학교 컴퓨터공학과/ 정보처리기사)

기술팀: 심홍록(SecurityPlus Union Academy 서울경기지부, 한국산업기술대 컴퓨터공학과, 정보보안동아리 MA 회장)

김경민(SecurityInsight Research Group)

이용훈(영진전문대 컴퓨터 정보 계열)

문서팀: 유성은(SecurityPlus Union Academy 영남지부, 영남대)

자문: 김기태(한국정보기술단 본부장)

감수: 박형근(시큐리티플러스 비영리 단체 대표)

후원: 시큐리티플러스(http://www.securityplus.or.kr)

1. 소개

이 문서는 Let's Encrypt 의 설치, 구성 및 사용에 대한 가이드이다. 먼저, Let's Encrypt 의 기본 개념과 해당 기술이 대두된 배경 지식, 그리고 각종 서버 OS 와 각자의 HTTP 웹 서버에 대한 Let's Encrypt 설치 및 구성을 어떻게 진행해야 할지에 대한 내용을 전반적으로 기술하는 문서이다.

1.1 등장 배경

고속 인터넷 망의 확산과 다양한 스마트 기기의 빠른 보급으로 인터넷의 사용은 폭발적으로 증가하였다. 그 중, 인터넷을 통해 가장 많이 사용하는 서비스는 웹을 이용한 서비스이다. HTTP 프로토콜을 사용하여 HTML 문서를 주고 받는 웹 서비스는 사용자의 플랫폼과 환경에 구애 받지 않고 다양한 형태의 정보들을 손쉽게 주고 받을 수 있어 매우 편리한 정보 공유 서비스임이 틀림없다. 하지만 다양한 정보들이 공유되고 쉽게 접근할 수 있을 뿐만 아니라 플랫폼에 구애 받지 않는다는 장점은 그만큼 많은 악의적 해킹 공격에 노출될 수 있다. 실제로 대부분의 공격이 웹 취약점을 이용해 이루어지고 대부분의 악성코드와 바이러스 또한 웹을 통해 유포된다. 이러한 웹 서비스의 허점들은 안전한 정보 공유 활동을 침해하는 요소로 작용할 수 있으며, 더 나아가 사용자의 개인정보를 탈취해 제 2, 제 3 의 피해를 야기시키는 위험으로도 발전할 수 있다.

그렇지만 웹 환경에서의 안전한 통신은 여러 가지 요소들을 통하여 구축할 수 있다. 그 중 하나인 HTTPS 는 암호화되고 인증된 통신을 위하여 어느새 필수불가결한 요소가 되었다. HTTPS 는 보안 허점을 극복하기 위해 1995 년 '넷스케이프 커뮤니케이션즈 코퍼레이션'에서 SSL(Secure Socket Layer) 암호화를 적용한 프로토콜이다. HTTPS 프로토콜은 HTTP 보다 보안이 한층 강화된 프로토콜로 세션 상에서 주고 받는 데이터를 암호화하여 데이터의 적절한 보호를 보장한다. HTTPS 를 적용하여 얻을 수 있는 이점은 크게 세 가지로 정의할 수 있다.

- 데이터 암호화: TLS 프로토콜이 적용된 HTTP 프로토콜을 사용함으로써 송수신하는 데이터를 암호화하여 전송한다. 공격자에 의해 패킷이 도청/감청 당하더라도 암호화되어 있기 때문에 아무 의미 없는 데이터를 보는 것과 다름없게 된다.
- 데이터 무결성: TLS 프로토콜의 알고리즘을 이용하여 무결성이 해쳐질 수 있는 요소들을 어느 정도 보완할 수 있다.
- 인증: 송수신 측 모두 신뢰할 수 있는 통신을 하기 위해, 웹 페이지 제공자는 전자서명이 포함된 보안 인증서를 사용하여 자신의 사이트가 신뢰할 수 있는 연결을 맺을 수 있음을 증명한다.

하지만 HTTPS 프로토콜을 사용하기 위해서는 인증 기관(CA: Certificate Authority)에서 인증서를 발급 받아야 하며, 그 과정 또한 매우 복잡하다. 또한 적지 않은 금액의 수수료 또한 해마다 지불해야 한다. 이러한 불편사항 때문에 많은 웹 서비스 공급자들이 HTTPS 사용을 적극적으로 수용하지 않는 실정이다. 아래 자료는 2015 년 Websense 에서 발표한 보고서의 일부를 발췌한 것[i]이다. 표를 보면 현재 HTTPS 프로토콜은 전체 웹 프로토콜의 25%만 차지하는 것을 볼 수 있다. (HTTP = 약 75%, HTTPS = 약 25%)

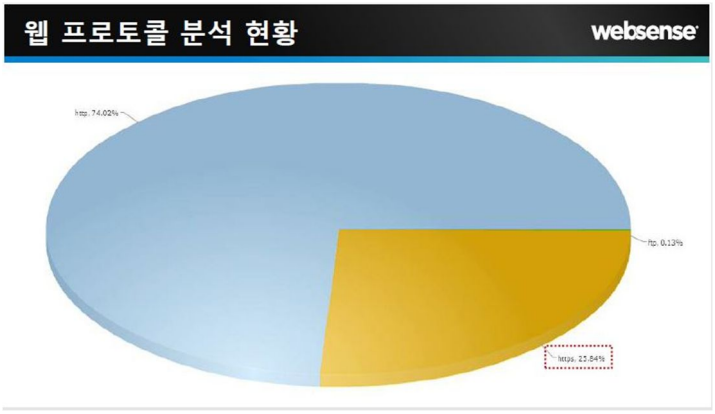


FIGURE 1 웹 프로토콜 분석 현황

저조한 HTTPS 의 보급률을 끌어 올리고, 모든 웹 서비스가 안전한 HTTPS 프로토콜을 사용하는 환경을 조성하기 위해 Mozilla, Cisco, Akamai, IdenTrust, Facebook, Google 등 다양한 글로벌 메이저 IT 기업들이 ISRG (Internet Security Research Group)라는 인증기관을 만들어 Let's Encrypt 프로젝트를 시작했다. Let's Encrypt 는 보안 인증서를 일반 유저가 구축한 웹 사이트에 쉽게 위 모델을 적용할 수 있는 오픈 서비스이다.

1.2 Let's Encrypt 란?

Let's Encrypt 프로젝트는 누구든지 HTTPS 프로토콜을 적용하여 안전한 웹 서비스 사용 환경을 조성하는데 그 목적이 있다. 이러한 목적을 수행하기 위해 Let's Encrypt 프로젝트는 웹 서비스 공급자들이 보다 쉽고 경제적으로 HTTPS 프로토콜을 공급할 수 있도록 인증서 관리 프로그램을 개발하였다. (개발한 프로그램의 이름 역시 Let's Encrypt 라고 불린다.) Let's Encrypt 를 왜 사용해야 하는지에 대한 이유를 먼저 기술한다.

i. 무료이다.

글로벌 루트 인증기관의 인증서를 발급받기 위해서는 수수료를 지불해야 한다. 금액은 2015 년 1 월 현재를 기준으로 1 년에 20 ~ 40 만원 정도의 수수료를 요구한다. (회사와 지원 서비스에 따라 천차만별의 가격대를 형성하고 있지만, 평균적 금액은 이 정도이다.)

ii. 안전하다.

사람들이 간혹 오해하는 것이 있다. 값이 저렴하고 무료이면 제품의 질이 떨어질 것이라는 것이다. 하지만 Let's Encrypt 는 그렇지 않다. Global Sign, Veri Sign, Geo Trust 와 같은 글로벌 루트 인증기관과 동일한 강도의 안전성을 보장한다.

iii. 적용하기 쉽다.

Let's Encrypt 는 쉬운 인증서 발급 절차를 지원하기 위해 별도의 프로그램을 지원한다. 해당 프로그램을 이용하면 손쉽게 인증 과정을 수행할 수 있으며 지원하는 웹 서버에 한해서 인증서의 적용 또한 손쉽게 진행할 수 있다.

이와 같이 무료이며 적용하기 쉽다는 점이 Let's Encrypt 의 가장 큰 장점이다. Let's Encrypt 의 성격과 특징을 살펴 보았으며 지원하는 기능은 다음과 같다.

Let's Encrypt 는 도메인에 따라 Let's Encrypt CA 에서 부여 받은 공개키로 구별되며, 처음 등록하는 과정을 통하여 신뢰성 있는 연결을 보장해 줌을 알려 준다. 그러면 Let's Encrypt CA 에서는 서버 측에 개인 키를 부여하며 이것을 통하여 신뢰성 있는 연결을 구축할 수 있다. 도메인 등록에 대한 도식화된 그림은 다음과 같다.

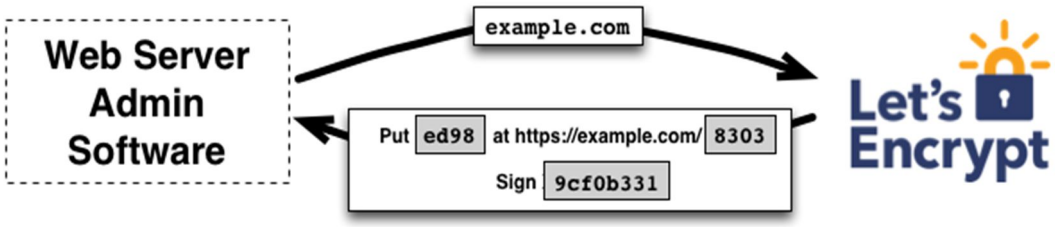


FIGURE 2 도메인을 LET'S ENCRYPT CA 에 등록하는 과정 (1)

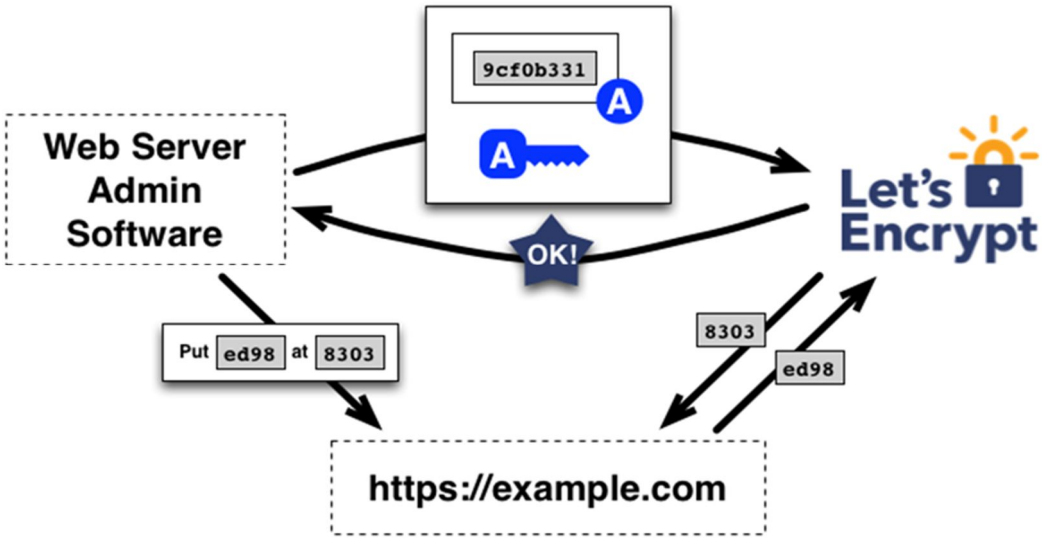


FIGURE 3 도메인을 LET'S ENCRYPT CA 에 등록하는 과정 (2)

이상의 개념으로 Let's Encrypt 의 도메인 등록에 대해 살펴 보았다. 또한 Let's Encrypt 는 크게 다음과 같은 기능을 제공한다.

- 인증서 발급: 설치 후 간단한 명령어 입력을 통하여 인증서를 쉽게 발급받을 수 있다. 즉 웹 페이지의 도메인 주소를 입력하면 그에 맞는 키를 제공받는 것이라 할 수 있다. 서버의 설정에 따라서 수동으로 발급받고 등록할 수도 있다. 해당 과정에 대하여 도식화된 그림은 다음과 같다.

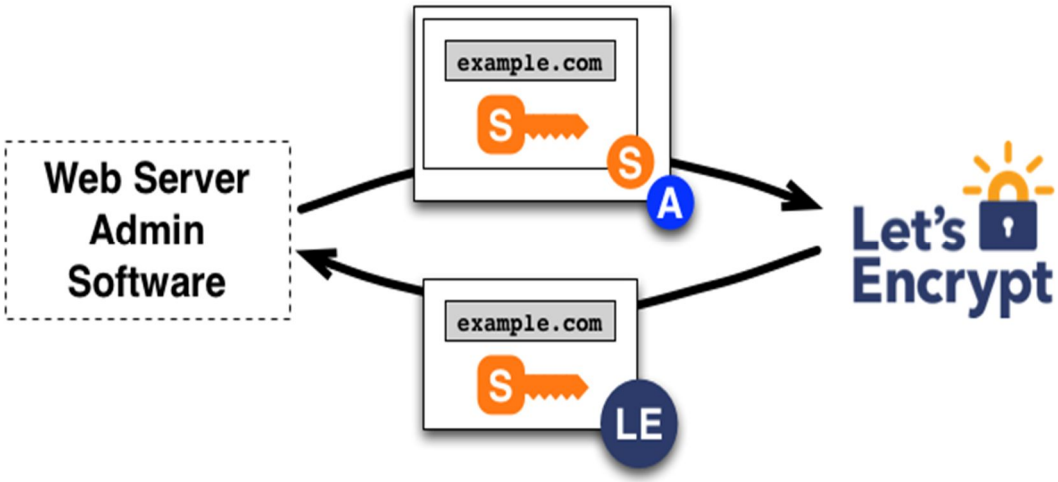


FIGURE 4 인증서를 발급 받고 보안 연결을 수립하는 과정

이처럼 웹 서버가 자신의 공개키와 도메인 정보를 포함하여 Let's Encrypt 의 인증 기관인 ISRG 에 전송하면, ISRG 는 위 두 개의 정보를 기관의 개인키로 암호화하여 서버에게 다시 전송해준다. 이때 인증 기관이 암호화하여 보내준 것이 최종적으로 사이트의 인증서가 되는 것이다. (각 기관의 공개키는 브라우저에 자체적으로 내장이 되어 있다. ISRG 기관의 공개키도 물론 존재한다.) 이로써 클라이언트가 ISRG 기관으로부터 인증 받은 웹 사이트에 접속하고자 할 때, 자신이 접속하고자 하는 사이트가 위조된 사이트가 아닌 정상적인 사이트라는 것을 확인하고 안전하게 연결을 수행할 수 있게 된다.

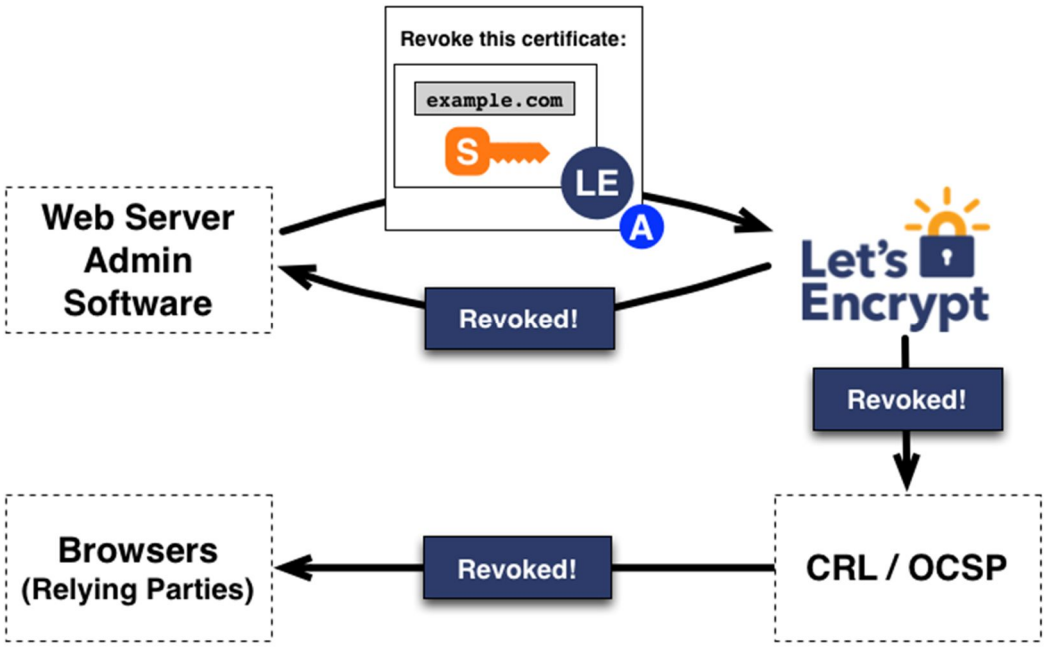
해당 작업은 웹 서비스 공급자가 사용하는 웹 서버에 따라 자동적으로 수행이 가능한 경우가 있고, 서비스 공급자가 수동으로 인증서를 발급받고 직접 적용해야 하는 경우가 존재한다. 발급 받은 인증서는 90 일에 한 번씩 재인증을 해주어야 사용이 가능하니 이 점을 유의하고 적용해야 한다.

Plug-in	Auth	Inst
Apache	Y	Y
Standalone	Y	N
Webroot	Y	N
NginX	Y	Y
Manual	Y	N

표 1 각 웹 서버 별 LET'S ENCRYPT 가 지원하는 기능

- 인증서 삭제: 인증서 삭제 또한 이와 비슷한 과정을 거친다. 웹 페이지의 도메인을 입력 한 후 해지를 위한 프로토콜을 통하여 해지한다. 해당 과정에 대하여 도식화한 그림은 다음과 같다.

FIGURE 5 LET'S ENCRYPT CA 에서 보안연결을 해제하는 과정



그 외에 인증서 갱신 등의 기능은 별도로 제공하고 있진 않지만, 스크립트를 작성하는 것을 권장하며, 90 일 안으로 한 번씩 갱신하여 사용할 수 있다.

2 Debian 8

2.1 설치 환경 설정 / 요구사항

Debian 에서 Let's Encrypt 를 설치하기 위한 별도의 준비 과정은 다음과 같다.

Debian Linux 'apt list update

정상적으로 모든 패키지를 다운로드 받으려면, apt 레파지토리 리스트를 업데이트 하는 것이 좋다. 명령어는 다음 명령어를 이용한다.

```
$ sudo apt-get update
```

git package 설치

Let's Encrypt 는 기본적으로 소프트웨어 배포를 git 를 이용한다. 따라서 해당 패키지가 설치가 돼 있어야 정상적으로 Let's Encrypt 를 설치 할 수 있다. 다음 명령어를 이용하여 git 를 설치한다. (이미 git 이 설치가 되어 있으면 해당 과정을 건너 뛰어도 좋다.)

```
$ sudo apt-get install git
```

apache2 웹 서버 버전 확인

현재 자신이 사용하고 있는 apach2 의 버전을 확인할 필요가 있다. 안정적인 설치를 위해서 2.x 버전의 아파치 사용을 추천한다. 버전 확인 시 에는 다음 명령어를 이용한다. (현재 가이드북 에서는 apache2 version 2.4.18 을 기준으로 작성하였다.)

```
$ apache2 -v
```

설치 전 웹 서버와 Database 의 정상 종료

Let's Encrypt 를 설치하기 전, 웹 서버와 사용중인 Database 를 정상적으로 종료하고 설치하는 것이 좋다. 추후에 발생 할 수 있는 다양한 에러를 미연에 방지하기 위해서이다. 아래 명령은 웹 서버와 DB 를 각각 종료하는 명령어이다.

```
$ /etc/init.d/apache2 stop
$ /etc/init.d/mysql stop
```

2.2 Apache 2 에서의 Let's Encrypt 적용

Debian Linux 에서 Let's Encrypt 를 설치하기 위한 과정은 다음과 같다.

```
$ git clone https://github.com/letsencrypt/letsencrypt
```

먼저 Debian 상에서는 letsencrypt 를 Git 으로 다운로드 받도록 한다.

해당 명령어를 입력하면 현재 쉘에서 작업중인 디렉토리에 프로그램을 다운로드 받는다. 사용자가 인지할 수 있는 디렉토리에 다운로드 받도록 한다.

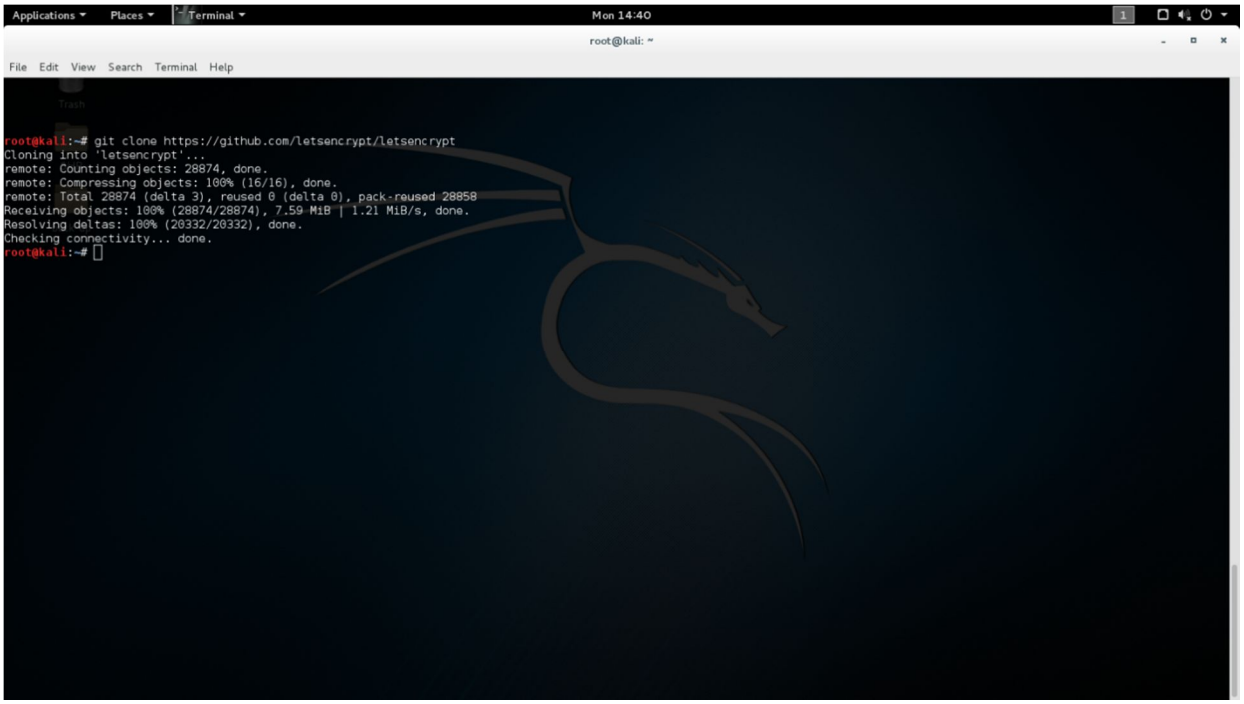


FIGURE 6 GIT 를이용하여프로그램다운로드

내려 받은 letsencrypt 를 설치할 차례이다. 해당 명령어를 실행하면 letsencrypt 를 설치할 때 프로그램의 구동에 필요한 별도의 라이브러리도 같이 설치하는 의존성 설치를 진행하게 된다. 환경에 따라 몇 분간의 시간이 요구될 수 있다. 해당 명령어는 다음과 같다.

```
$ letsencrypt-auto --help
```

설치가 완료되면 다음과 같은 화면을 볼 수 있다. 아래의 그림은 현재 letsencrypt-auto --help 명령어를 입력했을 때의 화면이다. 이미 관련 의존성 라이브러리가 다 설치되어 있어 별도로 다운로드 진행 구문이 보여지지는 않는다. 다운로드와 설치를 끝마치고 해당 화면이 보이면 정상적으로 설치를 끝마친 것이다.

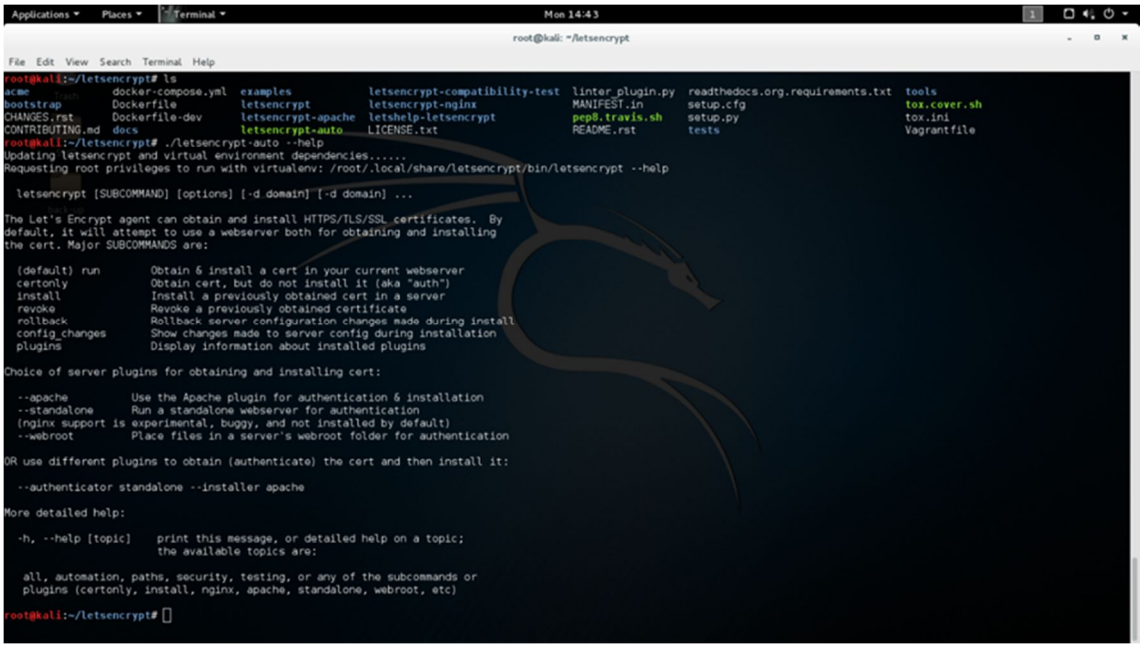


FIGURE 7 다운로드 받은 LET'S ENCRYPT 프로그램을 설치

이것으로 인증서 발급에 필요한 letsecnrypt 를 설치하였다. 해당 프로그램은 인증서의 발급, 적용과 해제까지 인증서를 통합 관리하는데 필요한 모든 기능을 수행할 수 있으므로 letsencrypt 가 설치된 폴더에서 letsencrypt-auto 라는 실행 파일을 관리자가 자주 이용하는 디렉토리에 링크시켜 놓으면 인증서와 관련된 작업을 수행할 때 편리하게 이용이 가능하다.

인증서 관리에 필요한 letsencrypt 를 설치하였으니, 이제 인증서를 발급받을 일만 남았다. apache2 사용자일 경우, letsecnrypt 가 지원하는 플러그인을 활용하여 설치보다 쉽게 인증서 발급과 적용이 가능하니 너무 걱정하지 않아도 된다. 우선 letsencrypt 가 설치된 디렉토리에 다음과 같은 명령어를 입력하도록 한다. 명령어를 실행하면 다음과 같은 창이 나온다.

```
$ letsencrypt-auto --apache
```

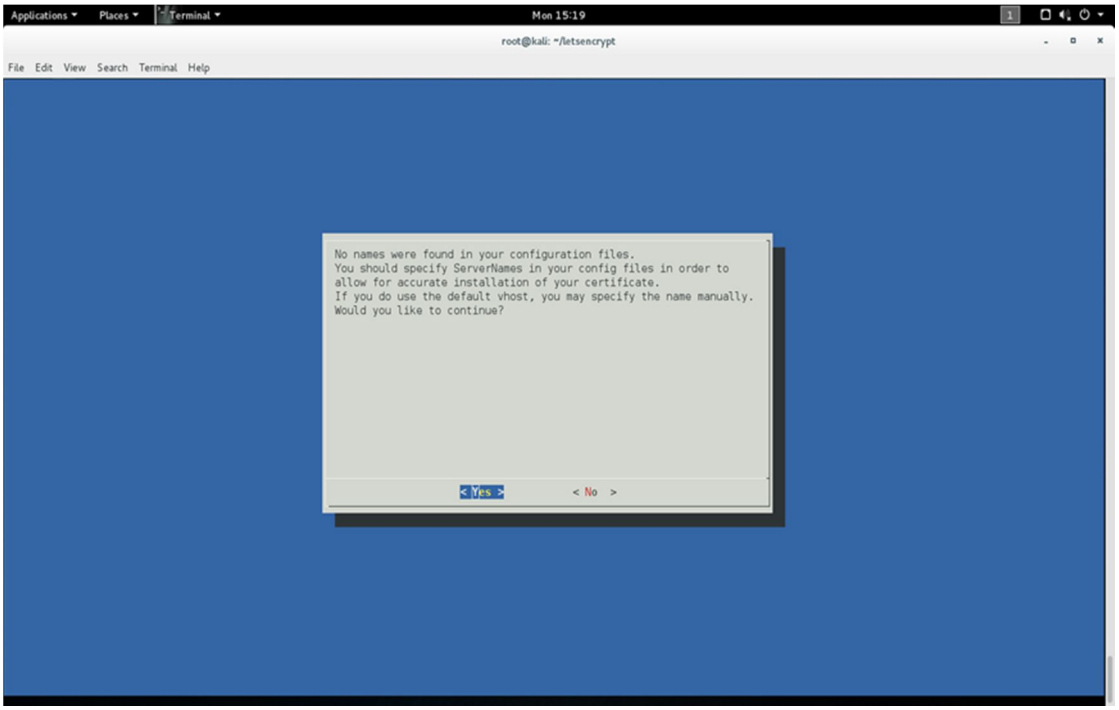


FIGURE 8 프로그램 실행 시 관리자의 이메일을 물음

해당 명령어를 실행하게 되면 파란색 화면의 창이 나온다. 현재 apache2 웹 서버의 config 파일을 조사해 보니 서버 이름이 설정되어 있지 않다는 뜻이다. YES 를 누르게 되면 Server Name 을 설정하는 창으로 이동한다.

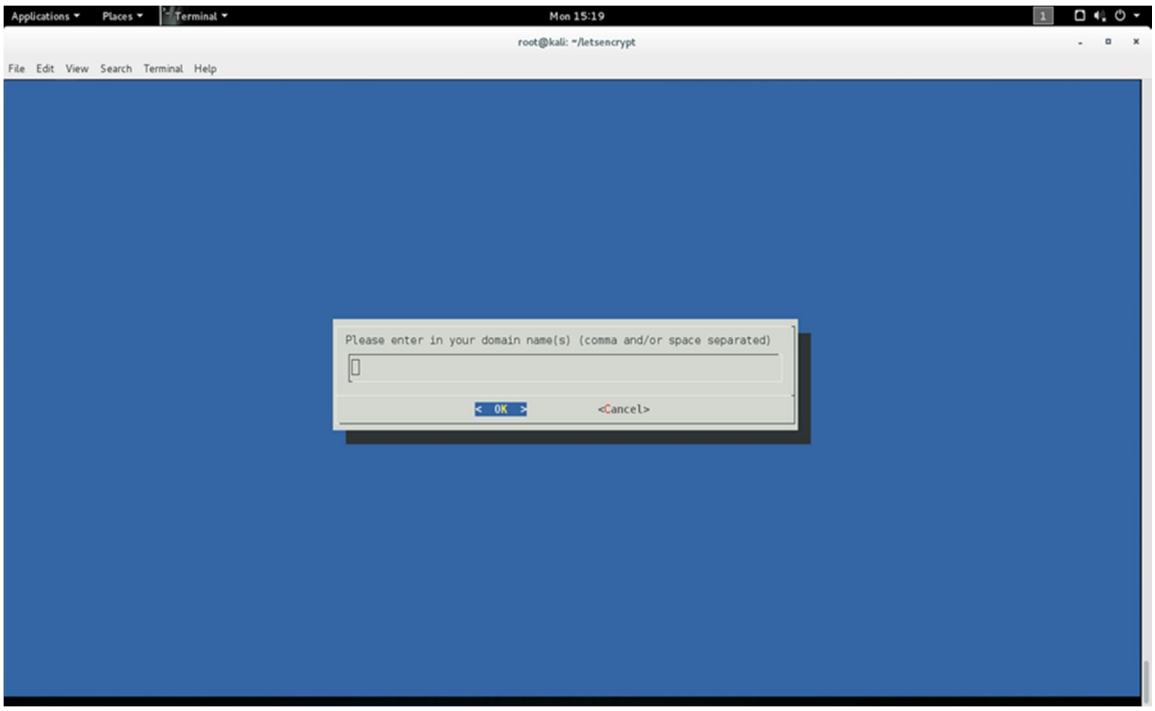


FIGURE 9 HTTPS 프로토콜을 적용시킬 도메인 이름 입력

YES 를 누른 후 볼 수 있는 Server Name 설정창이다. 이곳에 자신이 HTTPS 를 적용시킬 Domain 이름을 입력하면 된다. (현재 웹 서버에 물려있는 도메인을 입력하면 된다.) 여러 개의 도메인이 물려있는 웹 서버의 경우, 스페이스 바로 구분해서 도메인을 입력해 주면 된다.

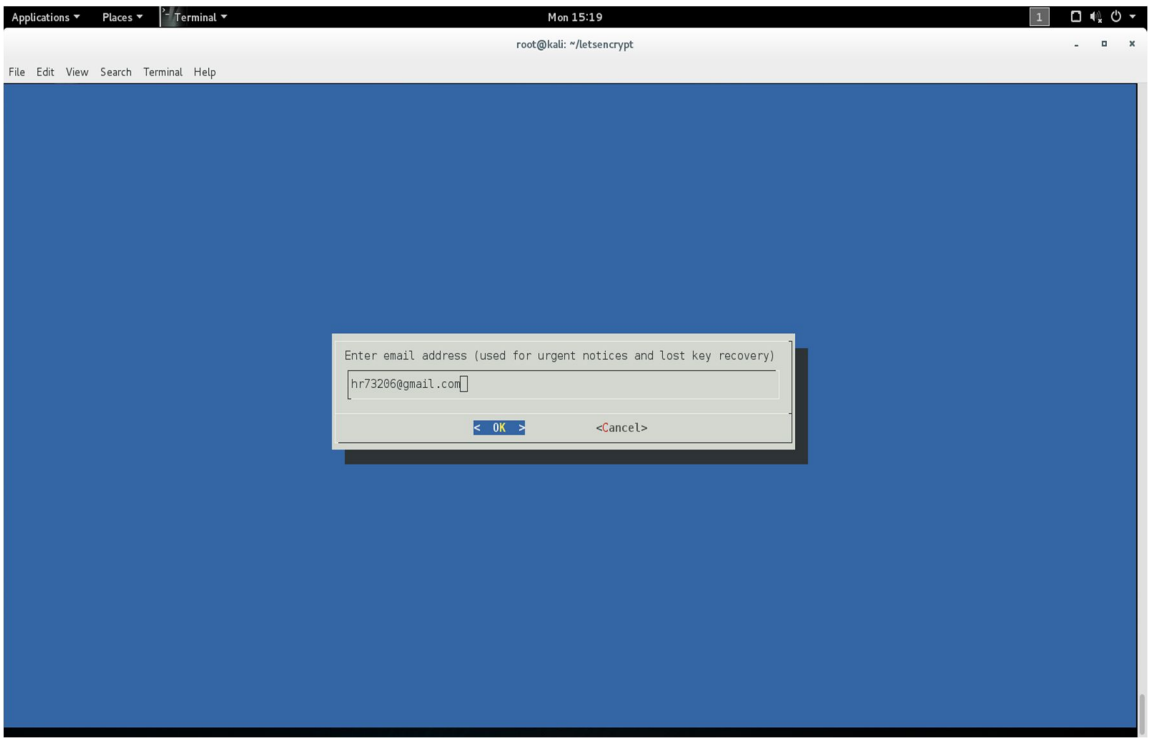


FIGURE 10 공지사항을 전달 받을 이메일 주소를 입력

도메인 등록이 끝나면 공지사항을 받을 이메일을 입력하라고 나온다. 자신의 이메일을 입력 후 OK 버튼을 누르면 된다.

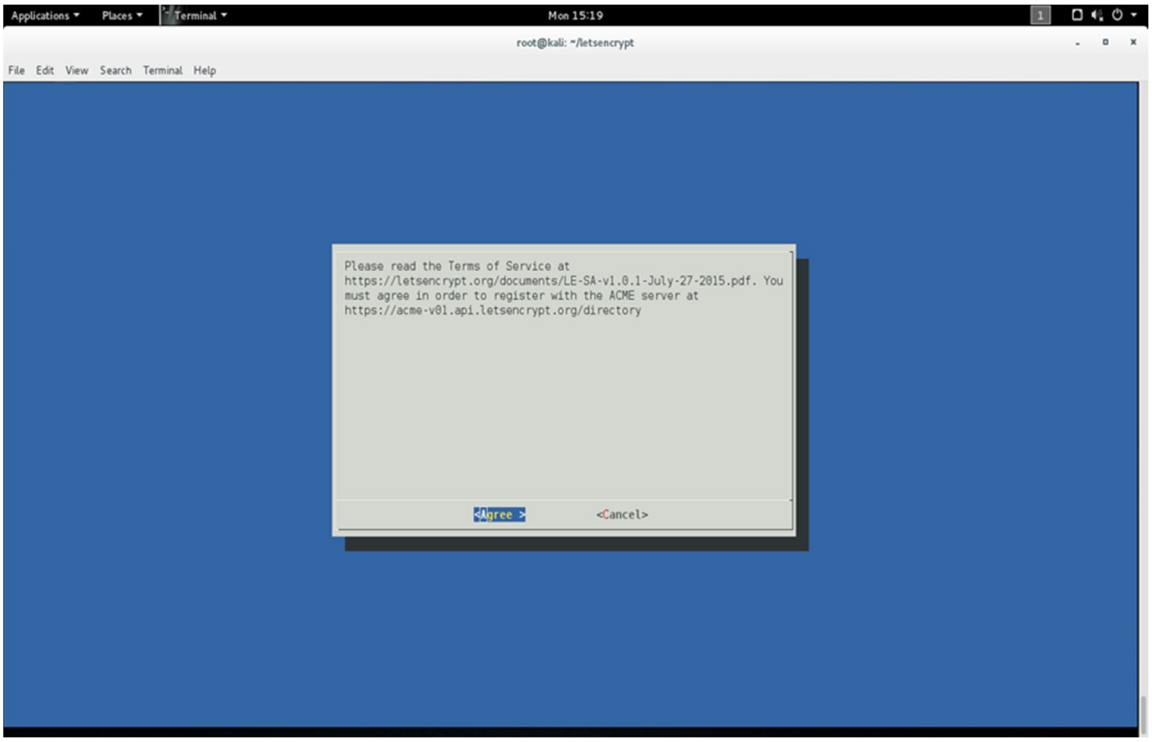


FIGURE 11 LET'S ENCRYPT 약관 동의

이메일 등록을 끝마치면 약관 동의 창이 나온다. Agree 를 눌러 약관에 동의 후, 계속 진행하도록 한다.

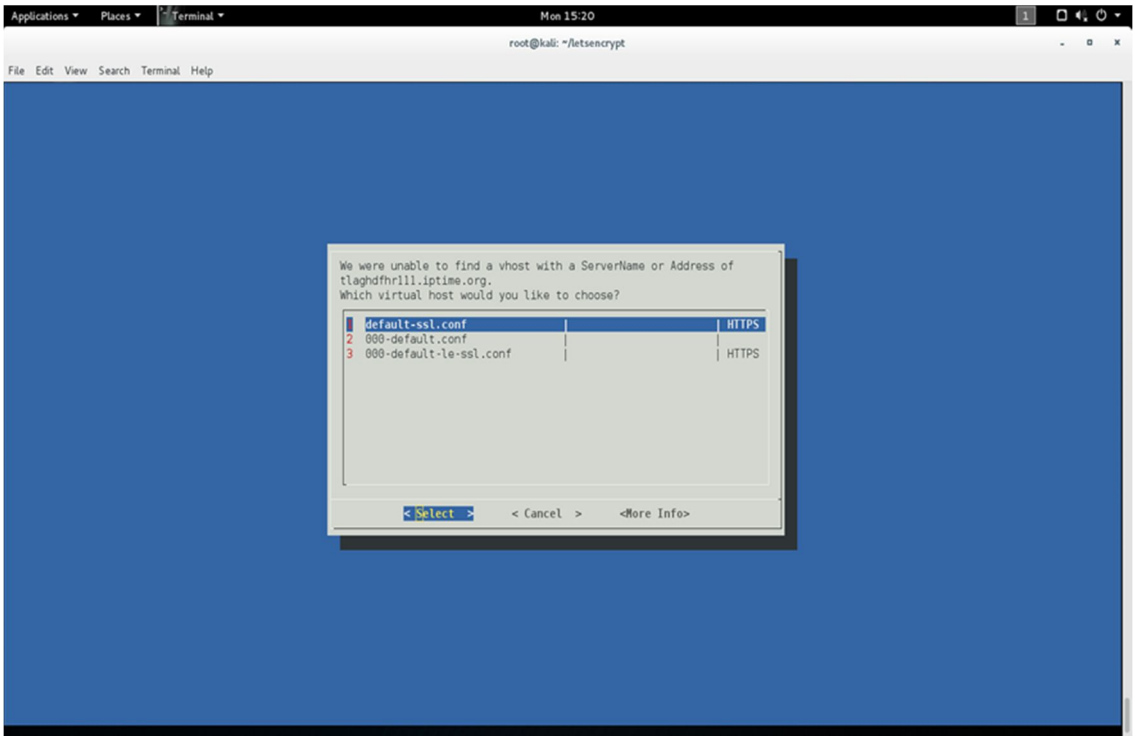


FIGURE 12 가상 호스트 옵션 설정

특별한 경우가 없는 경우 1 번을 선택하도록 한다.

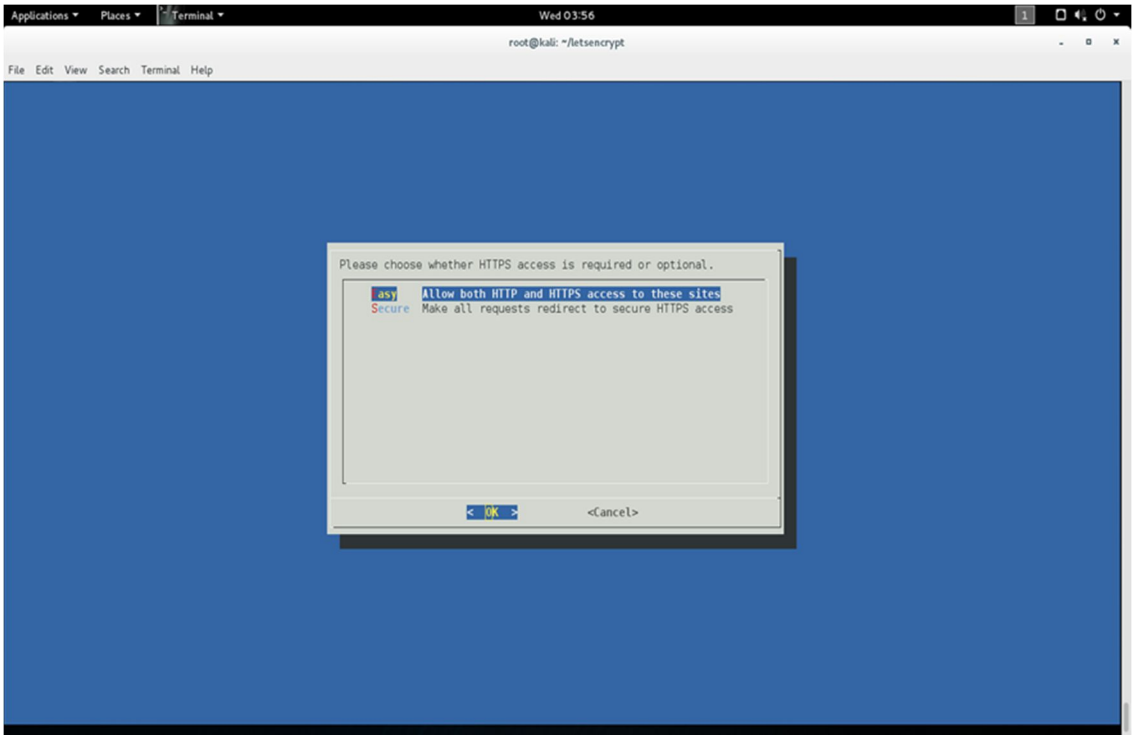


FIGURE 13 HTTPS 접속 옵션 설정

그 다음으로는 HTTPS 접속 옵션 창이 나오게 된다. 이곳에서 2 가지 옵션을 고를 수 있다. 첫 번째 옵션은 HTTPS:// ~ 로 들어오는 요청만 HTTPS 를 제공하는 옵션이고, 두 번째 옵션은 해당 도메인으로 들어오는 모든 요청을 HTTPS 로 만들어주는 옵션이다. 필요에 따라 옵션을 선택하도록 한다.

해당 과정까지 모두 끝마치면 HTTPS 가 정상적으로 적용될 것이다. 정상적으로 HTTPS 요청이 이루어지는지 확인하기 위해 자신의 도메인으로 이동해 URL 옆에 녹색 자물쇠 모양이 있는지 확인하는 것이 필요하다. (진행 중 오류가 발생하지 않으면, 정상적으로 적용이 되어 있을 것이다.)

두 번째 옵션을 선택 함에도 불구하고 URL Redirecting 이 이루어지지 않는 경우가 존재한다. 그럴 경우 직접 apache2 의 설정 config 파일을 수정해 주어야 한다. 방법은 Fedora Linux 에 적용해 준 방식과 거의 같다. 통상의 경우에는 */etc/apache2/apache2.conf* 로 가서 맨 아랫줄에 다음과 같은 구문을 추가하여 준다.

```
NameVirtualHost *:80
<VirtualHost *:80>
      ServerName YourServerName.org
      Redirect permanent / https://YourServerName.org/
</VirtualHost>
```

해당 옵션을 apache2.conf 파일에 추가해 주면 성공적으로 작동하는 것을 볼 수 있다.

3 Fedora 23

3.1 설치 환경설정 / 요구사항

Fedora Linux' DNF list update

별다른 오류 없이 설치하기 위해서 DNF 리포지토리(repository)를 업데이트 하도록 한다. DNF 리포지토리(repository) 리스트 업데이트는 다음 명령어를 사용하도록 한다.

```
$ sudodnf update
```

Git package 설치

Let's Encrypt 는 기본적으로 소프트웨어 배포를 git 를 이용한다. 따라서 해당 패키지가 설치가 돼 있어야 정상적으로 Let's Encrypt 를 설치할 수 있다. 다음 명령어를 이용하여 git 를 설치한다. (이미 git 가 설치가 되어 있으면 해당 과정을 건너뛰어도 좋다.)

```
$ sudodnfinstall git
```

Apache2 웹 서버 버전 확인

현재 자신이 사용하고 있는 apach2 의 버전을 확인할 필요가 있다. 안정적인 설치를 위해서 2.x 버전의 아파치 사용을 추천한다. 버전 확인 시에는 다음 명령어를 이용한다. (현재 가이드북에서는 Apache2 ver. 2.4.18 을 기준으로 작성하였다.)

```
$ apache --version
```

웹 서버와 Database 서비스 데몬 종료

Let's Encrypt 를 설치하기 전, 웹 서버와 사용 중인 Database 를 정상적으로 종료하고 설치하는 것이 좋다. 추후에 발생할 수 있는 다양한 에러를 미연에 방지하기 위해서이다. (각각 Apache, mysql 을 종료하는 명령어 이다.)

```
$ systemctl stop httpd.service
$ systemctl stop mysqld.service
```

3.2 Apache 2 에서의 Let's Encrypt 적용

Apache 2 웹 서버에서 Let's Encrypt 를 설치하는 과정은 다음과 같다.

먼저 Let's Encrypt 를 DNF 패키지 관리자로 설치를 진행한다. 설치를 위해서 다음 명령어를 입력하도록 한다.

```
$ sudodnfinstall letsencrypt
```

해당 명령어를 실행시키면 다음과 같이 설치할 패키지들에 대하여 사용자에게 묻는 화면이 뜬다. 확인 후 설치를 진행하도록 한다.

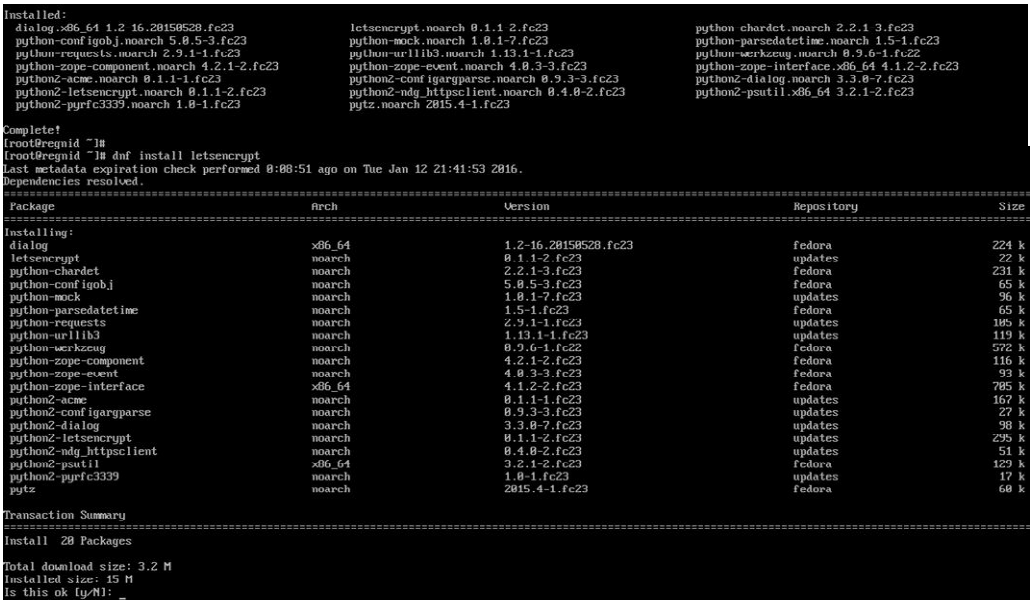


FIGURE 14 DNF 패키지 매니저로 설치하는 과정 (1)

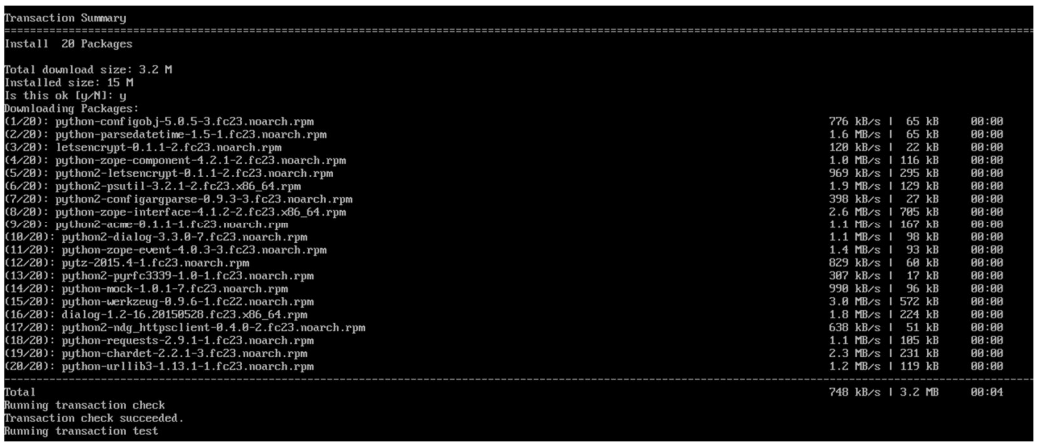


FIGURE 15 DNF 패키지 매니저로 설치하는 과정 (2)

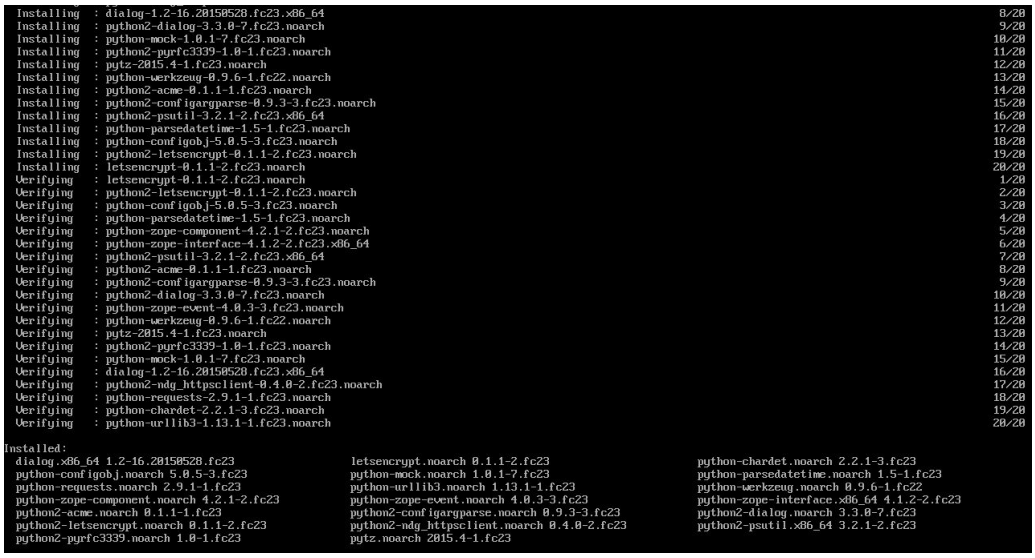


FIGURE 16 DNF 패키지 매니저로 설치하는 과정 (3)

그 다음, 정상적으로 설치되었는지 확인하기 위하여 --help 명령어를 입력하여 본다.

```
letsencrypt -- help
```

다음 명령이 뜬다면 정상적으로 설치에 성공한 것이다.

```
[root@localhost ~]#
[root@localhost ~]# letsencrypt --help

  letsencrypt [SUBCOMMAND] [options] [-d domain] [-d domain] ...

The Let's Encrypt agent can obtain and install HTTPS/TLS/SSL certificates.  By
default, it will attempt to use a webserver both for obtaining and installing
the cert. Major SUBCOMMANDS are:

  (default) run        Obtain & install a cert in your current webserver
  certonly             Obtain cert, but do not install it (aka "auth")
  install              Install a previously obtained cert in a server
  revoke               Revoke a previously obtained certificate
  rollback             Rollback server configuration changes made during install
  config_changes       Show changes made to server config during installation
  plugins              Display information about installed plugins

Choice of server plugins for obtaining and installing cert:

  (the apache plugin is not installed)
  --standalone      Run a standalone webserver for authentication
  (nginx support is experimental, buggy, and not installed by default)
  --webroot         Place files in a server's webroot folder for authentication

OR use different plugins to obtain (authenticate) the cert and then install it:

  --authenticator standalone --installer apache

More detailed help:

  -h, --help [topic]    print this message, or detailed help on a topic;
                        the available topics are:

   all, automation, paths, security, testing, or any of the subcommands or
   plugins (certonly, install, nginx, apache, standalone, webroot, etc)
```

FIGURE 17 정상 설치 여부 확인

이후, 서버의 도메인을 CA 에 등록하기 위하여 다음 명령어 형식을 입력하여 요청한다. 명령어의 구성은 다음과 같다.

```
letsencrypt --text --email (user-email-address) --renew-by-default -d (domain-name)
 --agree-tos --webroot-path (letsencrypt-main-directory) certonly
```

```
[root@localhost ~]# letsencrypt --text --email rudals531@gmail.com \
> --renew-by-default --agree-tos \
> -d regnid.securityplus.or.kr \
> --webroot --webroot-path /var/www/html certonly

IMPORTANT NOTES:
 - Congratulations! Your certificate and chain have been saved at
   /etc/letsencrypt/live/regnid.securityplus.or.kr/fullchain.pem. Your
   cert will expire on 2016-04-28. To obtain a new version of the
   certificate in the future, simply run Let's Encrypt again.
 - If you like Let's Encrypt, please consider supporting our work by:

   Donating to ISRG / Let's Encrypt:   https://letsencrypt.org/donate
   Donating to EFF:                    https://eff.org/donate-le

[root@localhost ~]#
```

FIGURE 18 서버의 도메인을 LET'S ENCRYPT CA 에 등록

이어서, 유효한 설정 값이 복사된 파일에 대하여 심볼릭(symbolic) 링크를 만들고 설정파일을 복사한다. 해당 명령어는 다음과 같이 구성된다.

```
ln -s /etc/letsencrypt/live/www.example.com/cert.pem /etc/pki/tls/certs/www.example.com.crt
ln -s /etc/letsencrypt/live/www.example.com/chain.pem /etc/pki/tls/certs/www.example.com.chain.crt
ln -s /etc/letsencrypt/live/www.example.com/privkey.pem /etc/pki/tls/private/www.example.com.key
cp /etc/httpd/conf.d/ssl.conf{,.backup}
sed -i 's@\(SSLCertificateFile\) .*@\1 /etc/pki/tls/certs/www.example.com.crt@'
/etc/httpd/conf.d/ssl.conf
sed -i 's@\(SSLCertificateKeyFile\) .*@\1 /etc/pki/tls/private/www.example.com.key@'
/etc/httpd/conf.d/ssl.conf
sed -i 's@#\(SSLCertificateChainFile\) .*@\1 /etc/pki/tls/certs/www.example.com.chain.crt@'
/etc/httpd/conf.d/ssl.conf
```

```
[root@localhost ~]#
[root@localhost ~]# ln -s /etc/letsencrypt/live/regnid.securityplus.or.kr/cert.pem \
> /etc/pki/tls/certs/regnid.securityplus.or.kr.crt
[root@localhost ~]# ln -s /etc/letsencrypt/live/regnid.securityplus.or.kr/chain.pem \
> /etc/pki/tls/certs/regnid.securityplus.or.kr.chain.crt
[root@localhost ~]# ln -s /etc/letsencrypt/live/regnid.securityplus.or.kr/privkey.pem \
> /etc/pki/tls/private/regnid.securityplus.or.kr.key
[root@localhost ~]#
[root@localhost ~]# cp /etc/httpd/conf.d/ssl.conf{,.backup}
[root@localhost ~]#
```

```
[root@localhost ~]#
[root@localhost ~]# sed -i 's@\(SSLCertificateFile\) .*@\1 /etc/pki/tls/certs/regnid.securityplus.or
.kr.crt@' /etc/httpd/conf.d/ssl.conf
[root@localhost ~]# sed -i 's@\(SSLCertificateKeyFile\) .*@\1 /etc/pki/tls/private/regnid.securitypl
us.or.kr.key@' /etc/httpd/conf.d/ssl.conf
[root@localhost ~]# sed -i 's@#\(SSLCertificateChainFile\) .*@\1 /etc/pki/tls/certs/regnid.securityp
lus.or.kr.chain.crt@' /etc/httpd/conf.d/ssl.conf
```

FIGURE 19 별도의 추가 설정 적용

해당 명령어를 입력한 후, 위의 명령어에 적용된 SELinux 정책에 대한 익스플로잇을 방지하기 위해 해당 명령어를 추가로 입력해주도록 한다.

```
semanagefcontext -a -t cert_t '/etc/letsencrypt/(archive|live)(/.*)?'
restorecon -Rv /etc/letsencrypt
```

해당 명령어를 입력하면 다음과 같이 적용되는 것을 볼 수 있다.

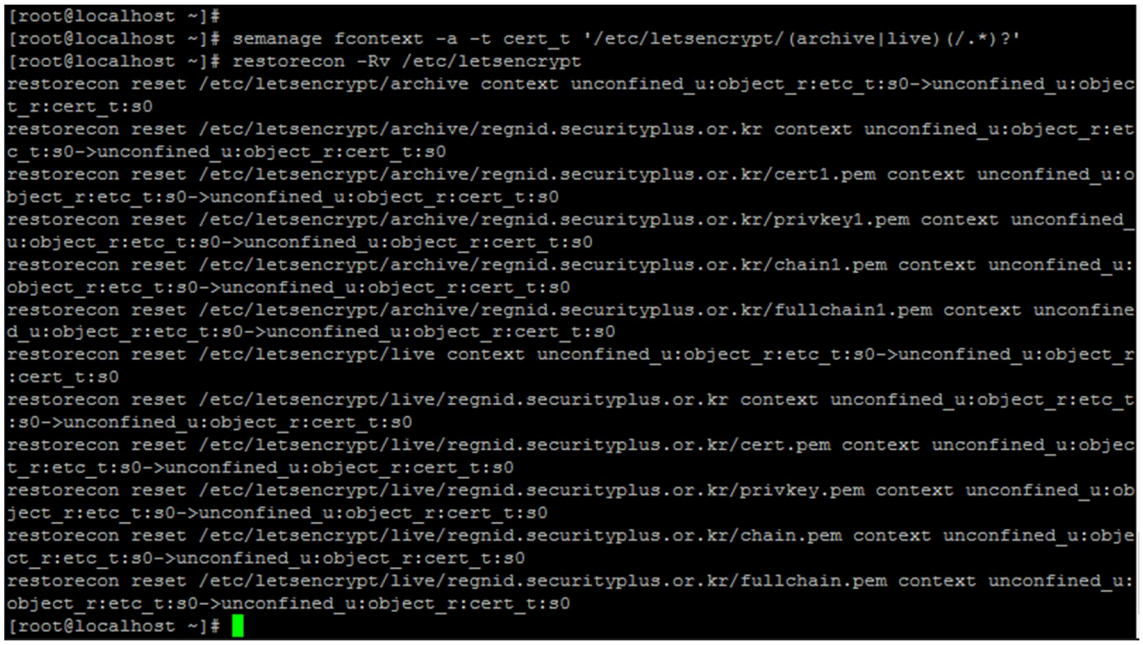


FIGURE 20 SELINUX 관련 이슈 해결을 위한 명령어 입력

추가로, HTTP 주소로 접근하더라도 HTTPS 프로토콜이 적용된 웹 서버로 리다이렉트 시켜주기 위하여 해당 설정값을 httpd.conf 에 추가하도록 한다.

```
<VirtualHost *:80>
RewriteEngine On
RewriteCond %{HTTPS} off
RewriteRule (.*) https://%{HTTP_HOST}%{REQUEST_URI}
</VirtualHost>
```

그리고 위에서 변경한 사항들을 적용시키기 위해서 다음 명령어를 통해 http 서비스를 재 시작한다.

```
systemctl restart httpd.service
```

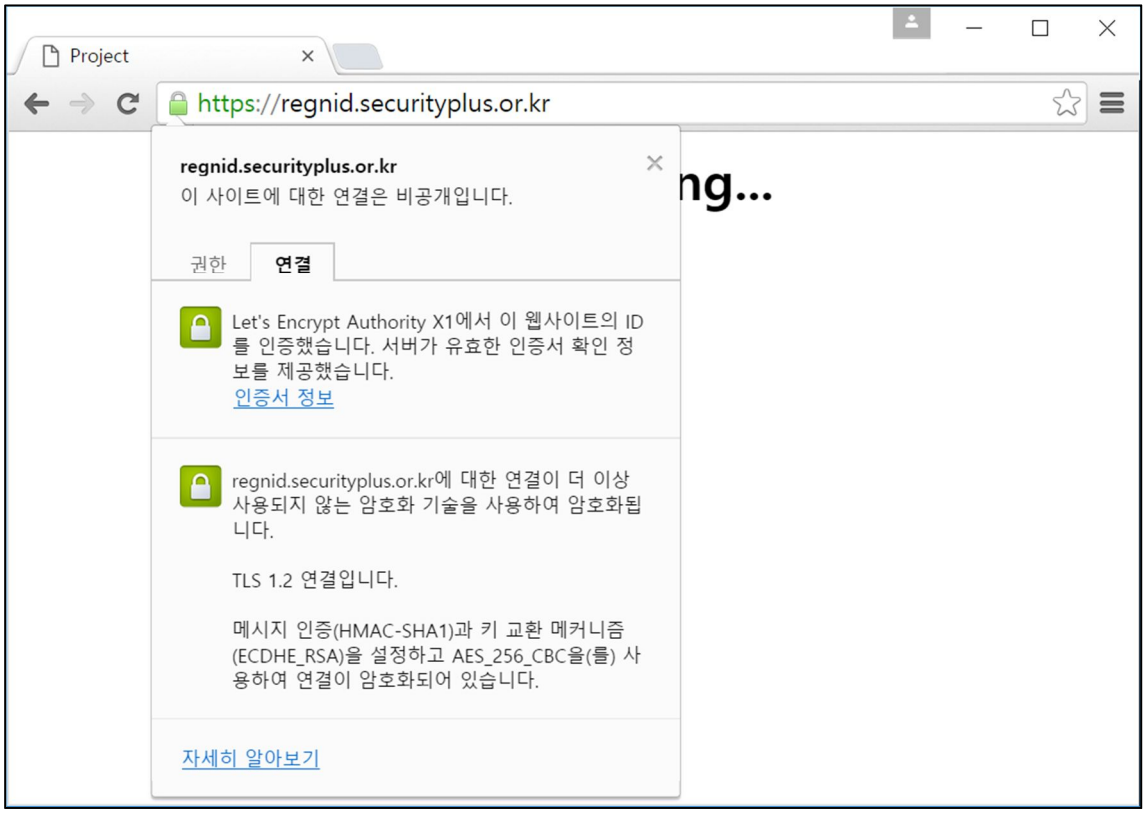
3.3 HTTPS 연결 작동 테스트

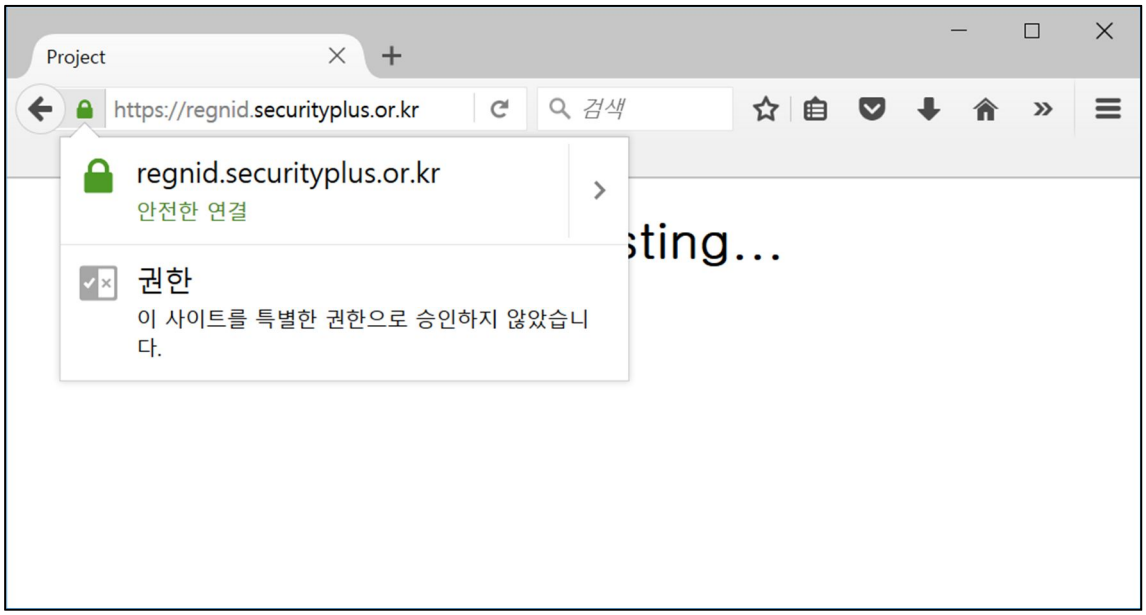
이후, 클라이언트에서 서버에 각 브라우저로 접속하여 HTTPS 연결이 되어있는지를 살펴본다.

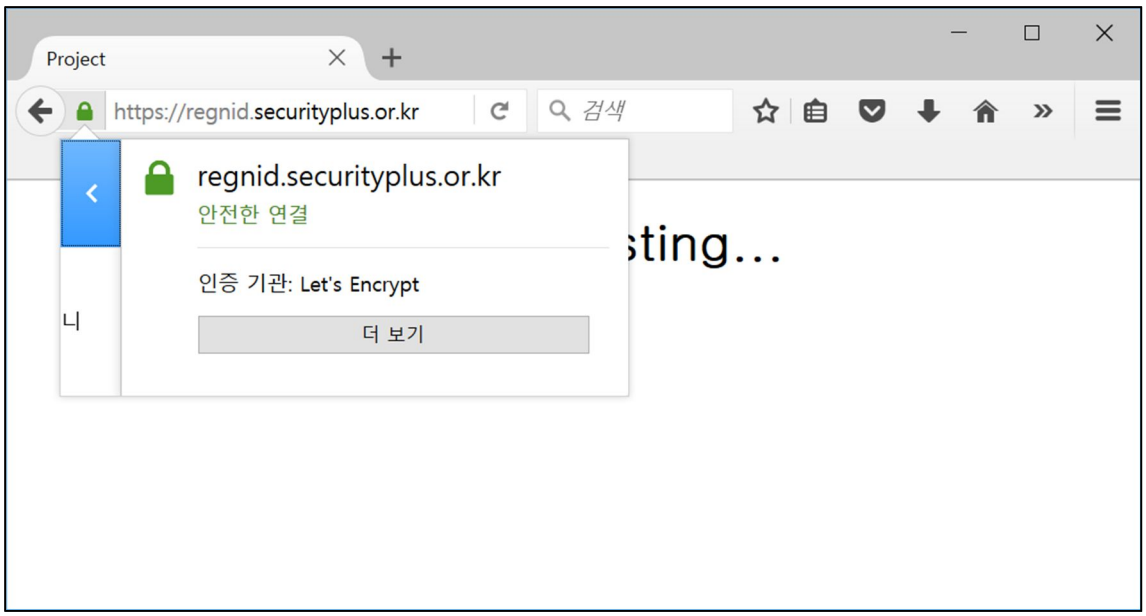
3.3.1 개인용 컴퓨터(PC) HTTPS 연결 작동 테스트

다음 그림들은 순서대로 크롬, 파이어폭스, 엣지, 인터넷 익스플로어 브라우저로 서버에 접속하여 인증서를 확인한 그림들이다.

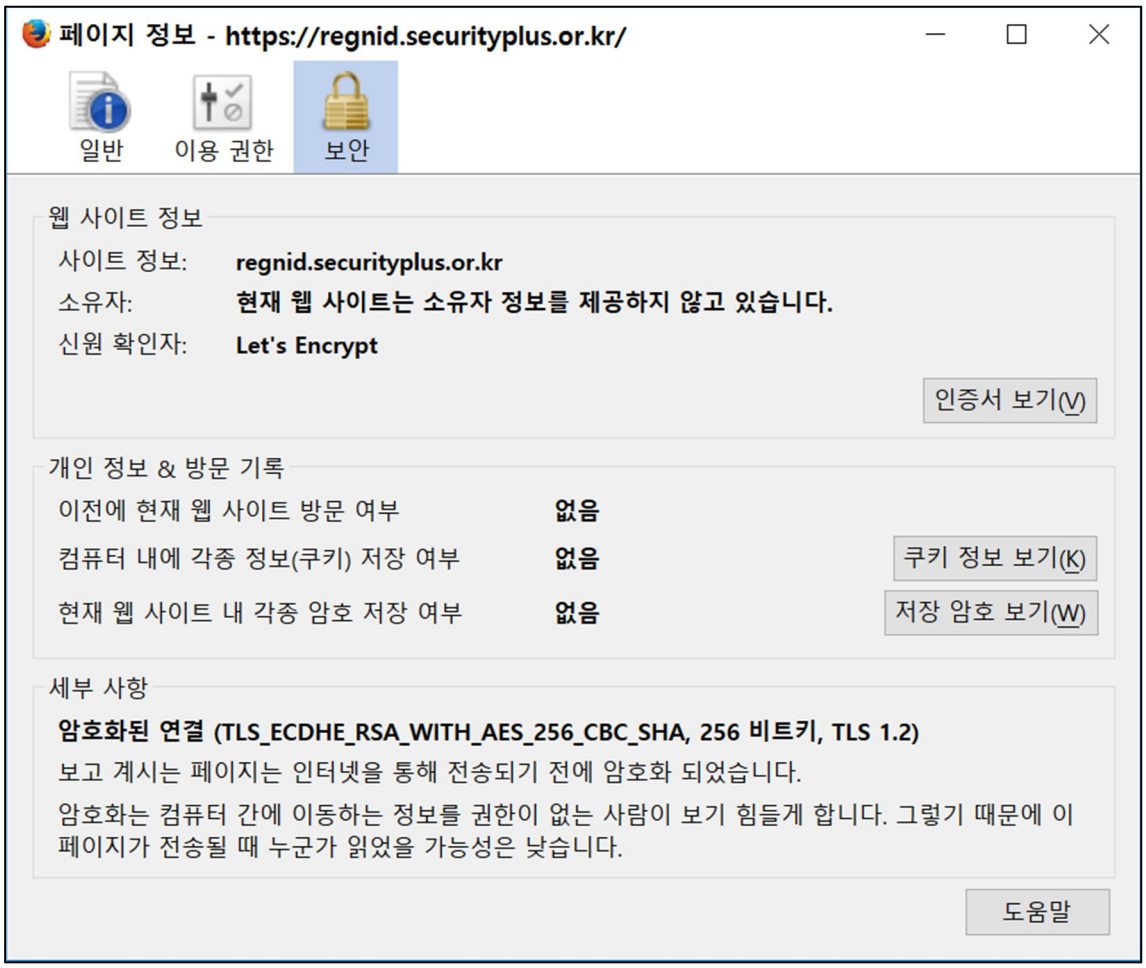
▣ 크롬 브라우저

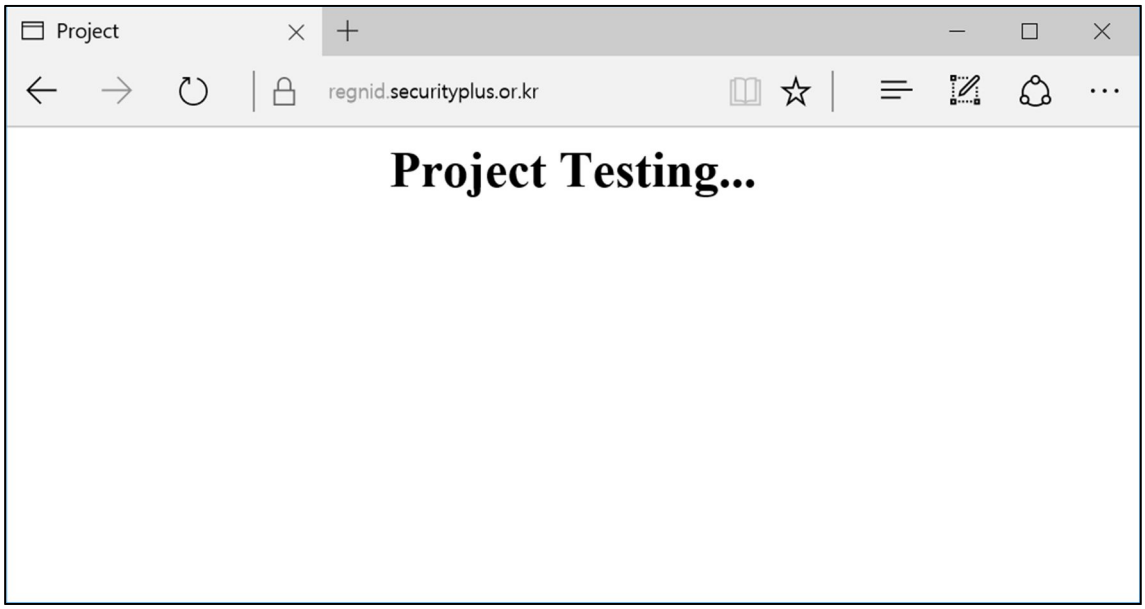


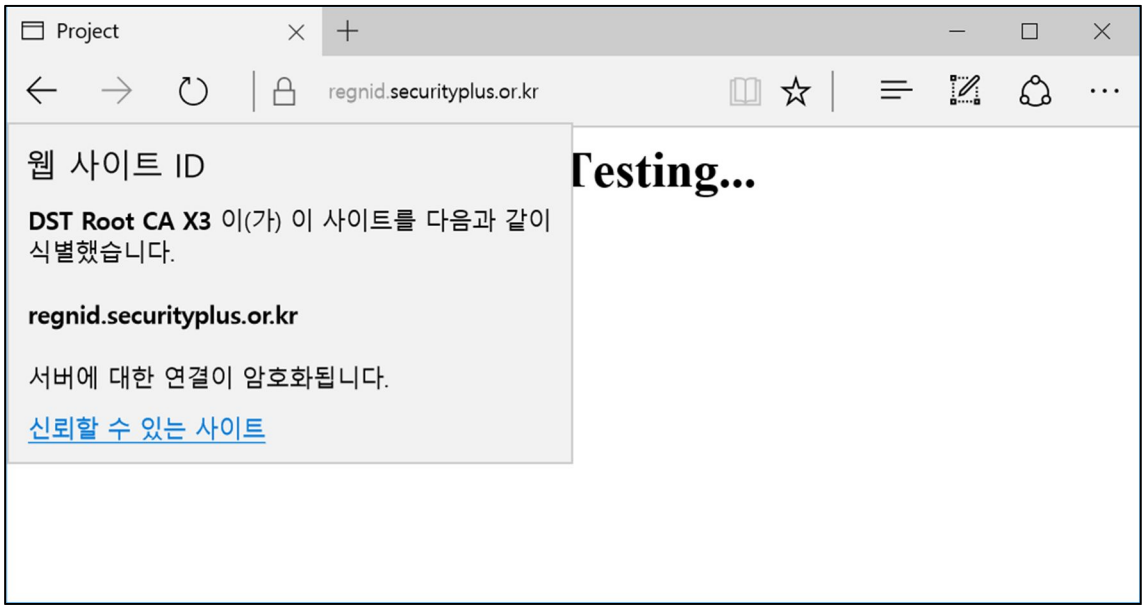




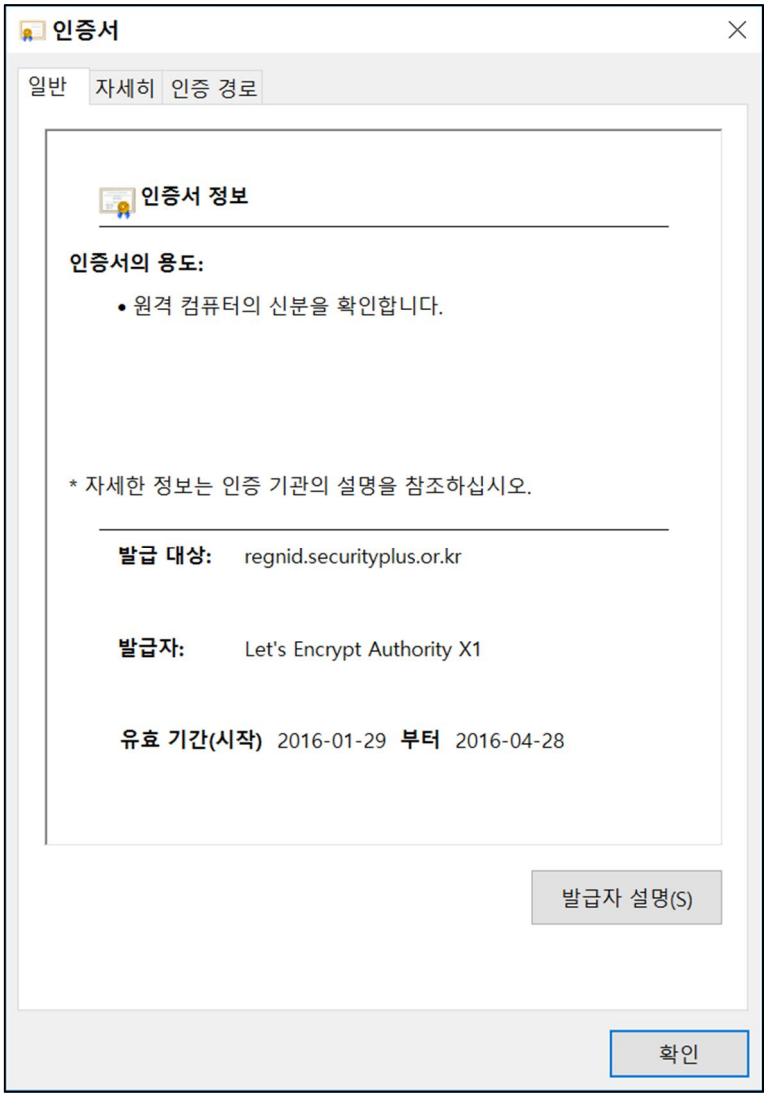
▣ 파이어폭스 브라우저





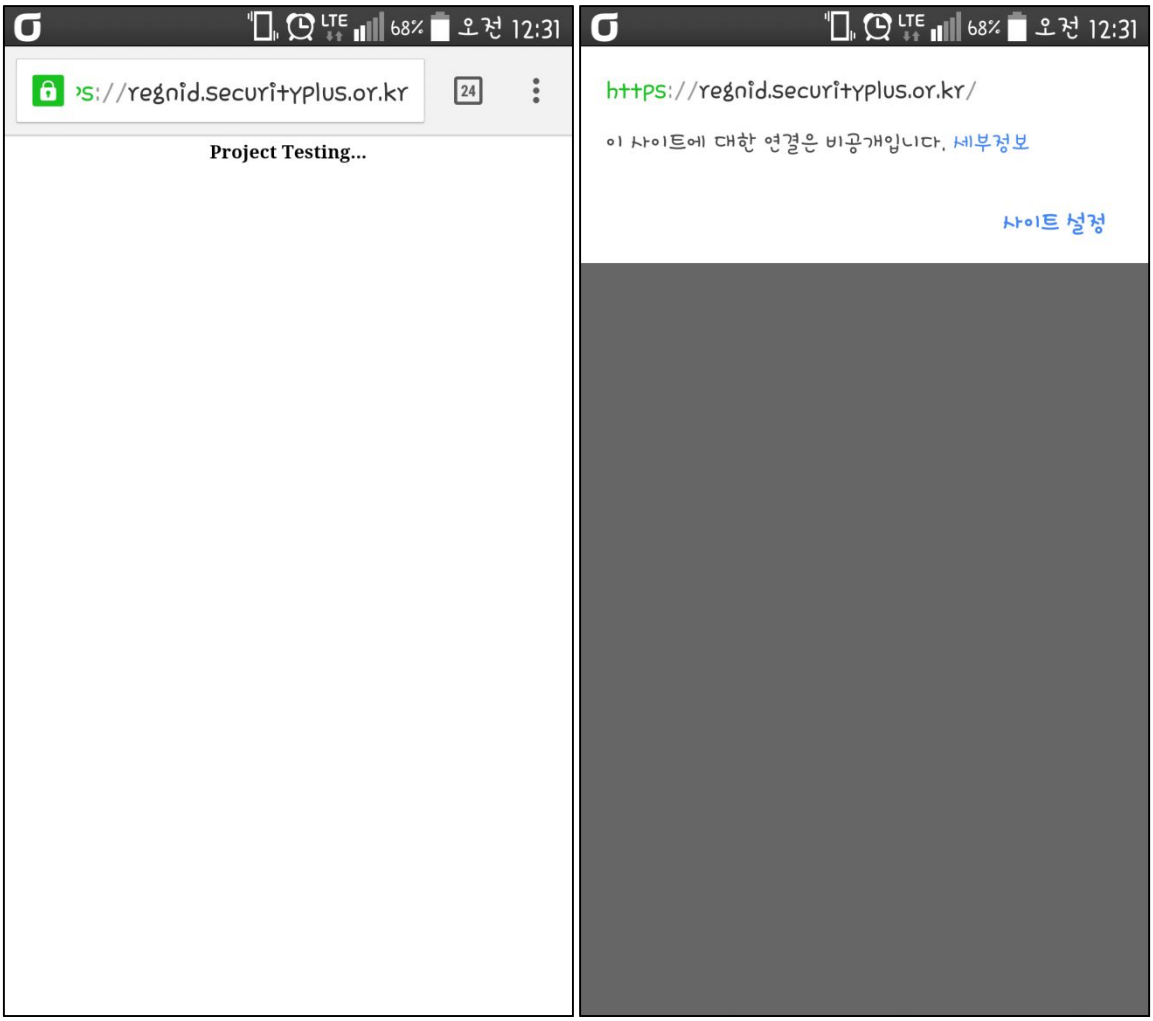


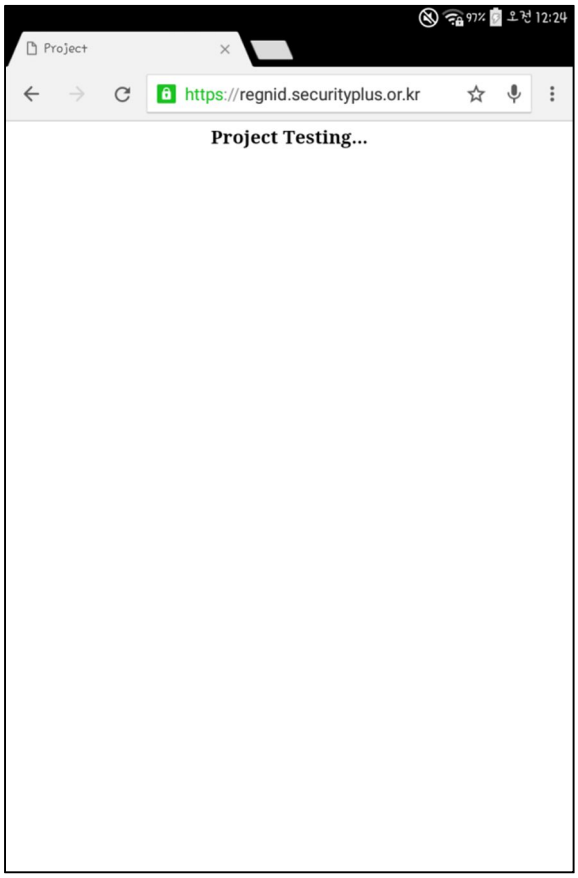
▣ 마이크로소프트 인터넷 익스플로러

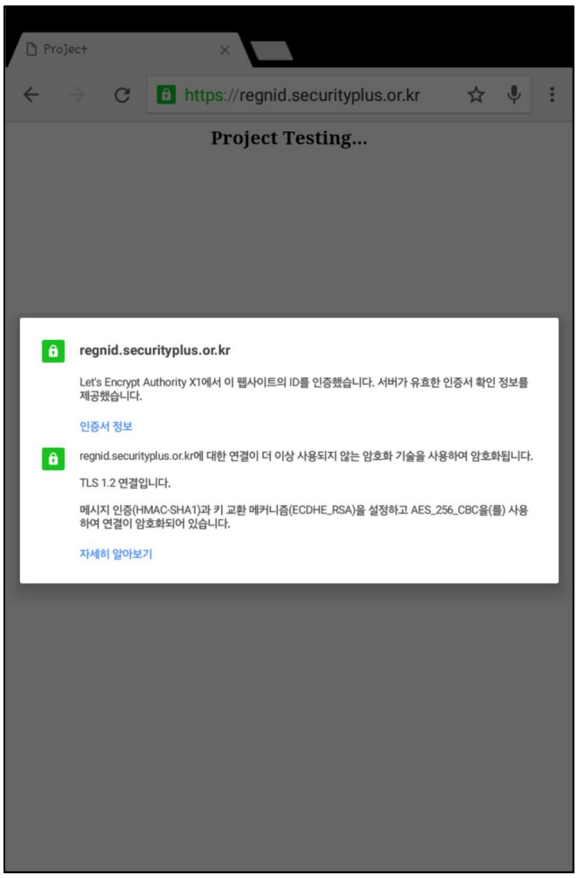


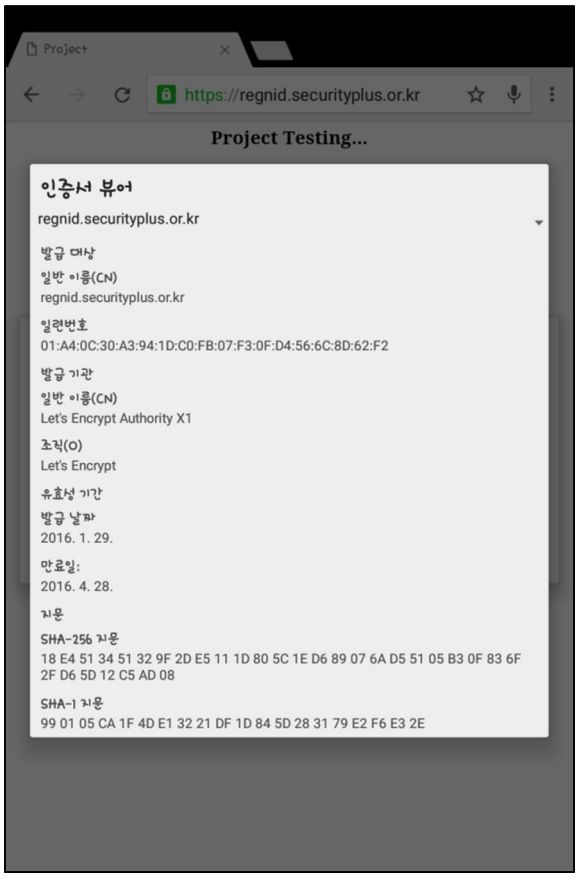
3.3.2 모바일 상에서의 HTTPS 연결 작동 테스트

다음 그림들은 모바일(Optimus G Pro, G Pad 2)에서 크롬 브라우저로 서버에 접속하여 인증서를 확인한 그림들이다.









끝.

PROJECT, STS

LET’S ENCRYPT USER GUIDE BOOK V1.0

그림 목차

표 목차

[i] 출처: 글로벌 웹 해킹 동향(2014) http://www.boan24.com/news/articleView.html?idxno=1321