Laporan Resmi Praktikum Keamanan Data

Web Secure Access & SSL/TLS

Dimas Rizky H.P. – 2110141011 – 3D4 IT A

1. Web Secure Access
   1. Dasar Teori

Apache Web server memiliki beragam opsi konfigurasi bagi administrator. Masalahnya, pada saat anda hosting, anda tidak memiki akses pada konfigurasi apache. Akibatnya anda tidak dapat melakukan perubahan setting dan terpaksa menggunakan konfigurasi default. Salah satu alternative yang dapat dilakukan untuk melakukan overriding konfigurasi default adalah dengan melakukan setting lewat file .htacces.

File .htaccess berupa file ascii yang ditempatkan pada direktori www atau subdirektori www. Anda dapat membuat file ini lewat editor teks seperti notepad dan diupload dalam bentuk ASCII (bukan BINARY) di direktori mana yang anda ingin memodifikasi setting. Selain itu, hak akses file diset sebagai 644 (rw-r--r--). Fungsinya adalah memungkinkan server mengakses file, namun mencegah pengunjung untuk mengakses file lewat browser mereka karena alas an keamanan.

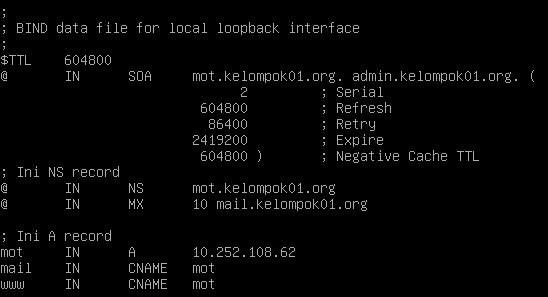
Perintah pada file .htaccess akan mempengaruhi semua direktori dan subdirektori dimana file tersebut diletakkan. Jika anda meletakkannya di sub direktori dari direktori www, maka file htaccess akan mempengaruhi direktori www dan semua direktori dibawahnya. Yang perlu diperhatikan adalah semua perintah pada file .htacess bersifat case sensitive dan didesain untuk diletakkan pada satu baris.

Beberapa fitur yang dapat didapatkan dari file .htaccess antara lain adalah seperti berikut ini :

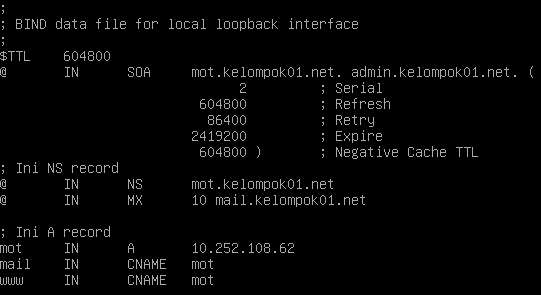
* Untuk memberikan pesan error
* Melakukan overriding setting SSI (Server-side includes)
* Mengganti file index.\* pada homepage
* Memungkinkan untuk melakukan browsing direktori
* Melakukan blok IP spesifik untuk mengakses website
* Redireksi pengunjung ke halaman web baru atau direktori
* Mencegah hot linking dan bandwith leeching
  1. Virtual hosting

Untuk melakukan percobaan web securing, akan dibuat dua virtual host, yaitu *kelompok01.org* & *kelompok01.net*. Web server yang digunakan adalah menggunakan apache, berikut adalah langkah-langkahnya,

* Sebelum mengkonfigurasikan virtual host dengan apache2, hal yang harus dilakukan adalah melakukan konfigurasi DNS server agar virtual host yang kita buat dapat diakses
* Yang akan digunakan adalah dua buah komputer dengan IP address *10.252.108.61 (DNS Server),* dan *IP address 10.252.108.62 (Web Server)*
* Pada komputer DNS Server, masuk ke direktori */etc/bind/zones* (Jika tidak ada maka silahkan dibuat). Lalu buat file zones untuk website *kelompok01.net* dan *kelompok01.org* serta reverse db-nya yaitu *108.252.10.in-addr.arpa*



*Gambar 1.0, zones file untuk kelompok01.org*

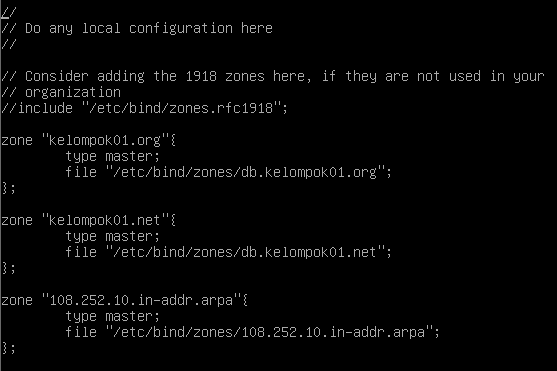
**

*Gambar 1.1, zones file untuk kelompok01.net*

**

*Gambar 1.2, reverse zones file untuk web kelompok01*

* Setelah itu beralih ke direktori */etc/bind/* lalu buka file *named.conf.local* lalu tambahkan zone file yang tadi ditambahkan



*Gambar 1.3, tambahkan definisi zones file untuk masing masing domain*

* Restart service bind9



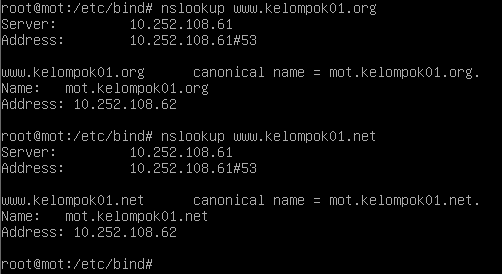
*Gambar 1.4, restart service bind9*

* Setting DNS-Server dari seluruh komputer ke IP *10.252.108.61* untuk melakukannya buka file */etc/resolv.conf* dan tambahkan/ganti nameservernya



*Gambar 1.5, alamat DNS server*

* Tes DNS-server dari komputer *10.252.108.62* dengan menggunakan perintah *$nslookup* [*www.kelompok01.org*](http://www.kelompok01.org) dan *$nslookup* [*www.kelompok01.net*](http://www.kelompok01.net)



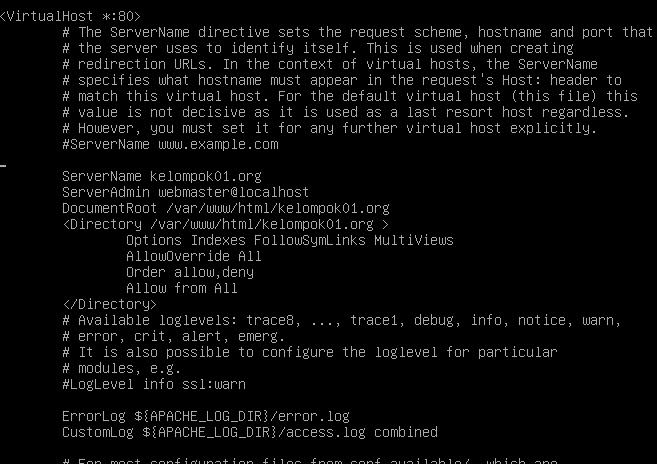
*Gambar 1.6, test menggunakan nslookup*

* Masuk ke direktori */etc/apache2/sites-available*
* Copy file 000-default.conf menggunakan perintah *$cp 000-default.conf <nama file hasil copy>*
* Pada nama file, beri nama *kelompok01.net.****conf*** dan *kelompok01.org.****conf.*** File ini akan menjadi file virtual host yang akan digunakan nantinya

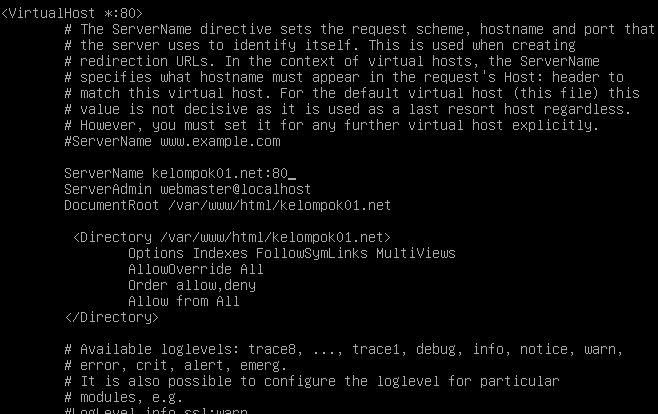


*Gambar 1.7, Direktori sites-available*

* Buka masing-masin file dan edit field nya

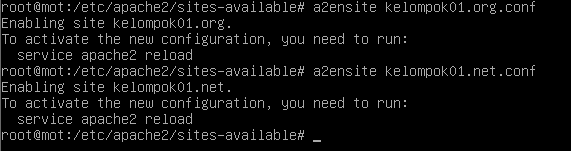


*Gambar 1.8, file kelompok01.org.conf*

**

*Gambar 1.9, file kelompok01.net.conf*

* Amati pada *gambar 1.8* dan *gambar 1.9* pada bagian *DocumentRoot* menunjukan ke direktori */var/www/html/kelompok01.(net/org)*. Direktori ini harus dibuat dan akan menjadi direktori penyimpanan file web yang ada
* Buat direktori dengan perintah *$mkdir /var/www/html/kelompok01.net* dan *$mkdir /var/www/html/kelompok01.org*
* Buat masing-masing file index.html yang berisi teks string yang dapat digunakan untuk membedakan antara web satu dan yang lainnya
* Masuk kembali ke direktori */etc/apache2/sites-available/* lalu jalankan perintah *$a2ensite kelompok01.net.conf* dan *$kelompok01.org.conf* untuk mengaktifkan kedua website tersebut



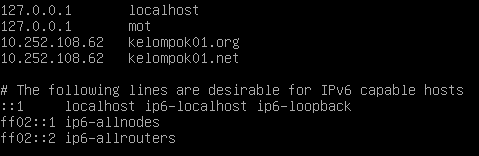
*Gambar 1.10, mengaktifkan website yang telah dibuat*

* Setelah itu lakukan restart terhadap service apache2 dengan perintah *$service apache2 restart*



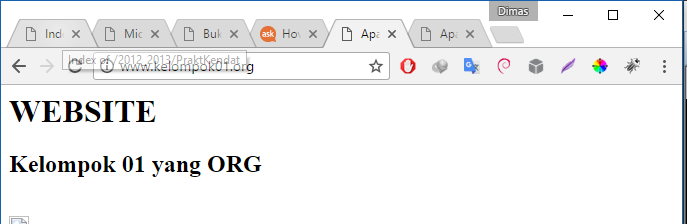
*Gambar 1.11, restart service apache2*

* Langkah selanjutnya adalah dengan menambahkan kedua domain ke file */etc/hosts*



*Gambar 1.12, tambahkan nama domain ke file /etc/hosts*

* Test dengan mengaksesnya menggunakan web browser, web browser akan diakses melalui komputer dengan IP address *10.252.108.118*



*Gambar 1.13 Akses kelompok01.org melalui website*

**

*Gambar 1.14 Akses kelompok01.net melalui website*

* 1. Web Redirection

Pada percobaan web redirection akan dilakukan redirect seluruh user yang mengakses website *kelompok01.net* agar ditujukan ke website *kelompok01.org.* Langkah-langkahnya adalah

* Masuk ke direktori website *kelompok01.net* di direktori */var/www/html/kelompok01.net* buatlah file .htaccess dengan cara menjalankan perintah *$mkdir .htaccess*
* Lalu isikan file .htaccess dengan ini



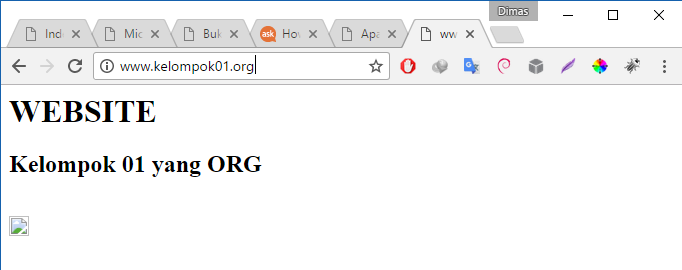
*Gambar 2.0 isi file .htaccess untuk redirect*

* Lalu masuk ke direktori */etc/apache2/mods-available* lalu jalankan perintah *$a2enmod rewrite* untuk mengaktifkan fitur rewrite



*Gambar 2.1, aktifkan mod rewrite*

* Restart service apache2 dengan perintah *$service apache2 restart*
* Cek pada web browser, meskipun browser mengakses [*www.kelompok01.net*](http://www.kelompok01.net)akan di redirect ke [*www.kelompok01.org*](http://www.kelompok01.org)

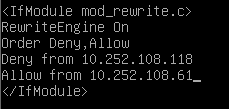


*Gambar 2.2, Redirect berhasil*

* 1. Blok Akses dengan .htaccess

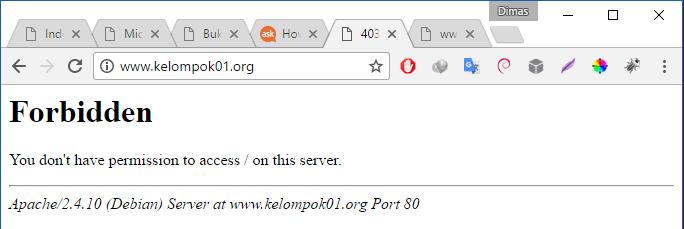
Fitur .htaccess yang lain adalah untuk melakukan blok akses dari IP address atau network address yang spesifik, langkah-langkahnya adalah

* Pada percobaan ini akan dilakukan blocking akses website dari komputer IP address 10.252.108.118 untuk kedua website [*www.kelompok01.org*](http://www.kelompok01.org) dan [*www.kelompok01.net*](http://www.kelompok01.net)namun dibolehkan ketika diakses dari IP 10.252.108.61
* Ubah file *.htaccess* pada masing-masing web directory, atau buat jika belum ada dan isikan seperti gambar 3.0 di bawah ini

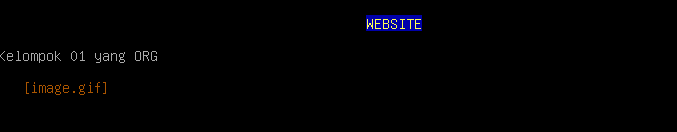


*Gambar 3.0, file .htaccess untuk blocking IP*

* Terapkan untuk kedua file .htaccess, lalu restart service apache2
* Cek website melalui komputer dari dua buah IP tadi



*Gambar 3.1, akses web dari IP 10.252.108.118*

**

*Gambar 3.2, akses web dari IP 10.252.108.61 via lynx*

* 1. Mencegah Hot Linking

Hot linking merupakan kasus dimana, website A memasang resource yang berasal dari website B, sehingga bandwith dari website B lah yang akan terpakai ketika seseorang mengakses website A. Salah satu fitur dari file .htaccess adalah mencegah adanya hot linking. Langkah-langkahnya adalah sebagai berikut,

* Pada percobaan ini, website [*www.kelompok01.org*](http://www.kelompok01.org) akan melakukan hot linking gambar yang ada pada website [*www.kelompok01.net*](http://www.kelompok01.net)*.* Pada file index.html milik [*www.kelompok01.org*](http://www.kelompok01.org) adalah sebagai berikut



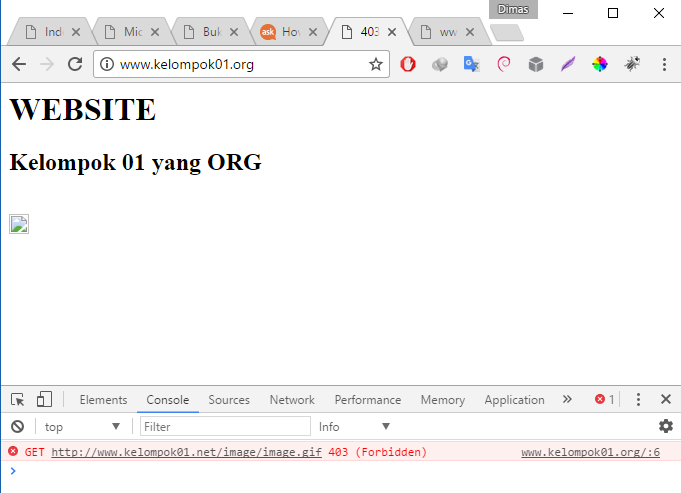
*Gambar 4.0, file index.html pada www.kelompok01.org*

* Untuk melakukan pencegahan hot-linking, buka file .htaccess yang ada pada */var/www/html/kelompok01.net* dan tuliskan



*Gambar 4.1, file .htaccess pada web direktori kelompok01.net*

* Coba akses website [*www.kelompok01.org*](http://www.kelompok01.org)dan liat gambar yang diambil dari hot-link dari website [*www.kelompok01.net*](http://www.kelompok01.net) tidak akan dapat di-load



*Gambar 4.2, gambar yang diakses mengembalikan code 403, atau forbiden*

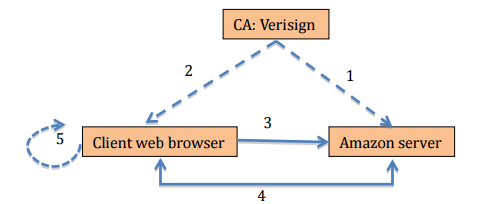
1. SSL/TLS
   1. Dasar Teori

Transport Layer Security (TLS) adalah protocol untuk mengamankan komunikasi antar aplikasi lewat internet. TLS mengamankan konten pada layer aplikasi, seperti halaman web dan diimplementasikan pada layer transport, yaitu TCP. Untuk menjamin keamanan. data yang dikirim dienkripsi dan diotentikasi pada sisi server dan client. Secure Socket Layer (SSL) adalah protocol yang diciptakan sebelum TLS yang mengaplikasikan hal ini.

SSL/TLS biasanya dioperasikan secara bersama-sama dengan HTTP, sehingga membentuk protocol baru yang disebut HTTPS, untuk mengamankan transaksi lewat web. Selain intu, protocol ini dapat digunakan untuk aplikasi-aplikasi lain seperti email, file transfer dan virtual private networks (VPN).

HTTPS menggabungkan protocol HTTP dan SSL/TLS untuk menjamin keamanan komunikasi antara eb server dan web browser. HTTPS beroperasi pada port 443 dan bukan pada port 80 seperti normalnya HTTP. HTTPS bekerja dengan menyediakan enkripsi untuk konten web dan otentikasi web server. HTTPS tidak melakukan otentikasi client sehingga web sita tidak dapat melakukan otentikasi user selama koneksi. User harus melakukan sejumlah otentikasi tambahan seperti password,biometric atau metode otentikasi lain.

Komunikasi SSL/TLS meliputi dua tahap yaitu handshaking dan data sending. Sebelum berkomunikasi, web site harus meminta certificate authority (CA) agar dapat menanda tangani (signing) digital certificate-nya yang berisi public key dari site. User yang menerima digital sertificate CA, segera memanggil sertifikat root, yang dimiliki ketika mereka menginstall web browser. Web browser seperti Internet Explorer atau Firefox sebelumnya telah dilengkapi dengan sejumlah sertifikat root dari bermacammacam perusahaan seperti VeriSign atau Entrust, yang memang menspesialiasisikan diri sebagai perusahaan yang bergerak di bidang sertifikasi.



1. Verisign menandatangani sertifikat Amazon dengan publik key-nya

2. Install CA sertikat dengan public key-nya pada browser client

3. Koneksi lewat https

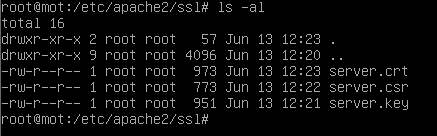
4. Saling menukar sertifikat digital, termasuk publik key

5. Client melakukan verifikasi sertifikat Amazon menggunakan public key dari CA

* 1. Menciptakan sertifikat SSL dengan OpenSSL

Percobaan ini akan dilakukan dengan tujuan dapat menginstall web server yang secure menggunakan SSL/TLS untuk transaksinya. Tools yang dibutuhkan anatara lain adalah apache2 dan openssl. Tahap dari percobaan ini adalah pembuatan public/private key, sertifikat SSL dan CSR. Langkah-langkahnya adalah sebagai berikut

* Install apache2 dan openssl dengan perintah *$apt-get install apache2* dan *$apt-get install openssl*
* Untuk percobaan ini, akan digunakan website [*www.kelompok01.org*](http://www.kelompok01.org) untuk di secure kan. Langkah selanjutnya adalah dengan masuk ke direktori */etc/apache2/ssl* direktori ini merupakan tempat penyimpanan private keys dan certificate sifning request



*Gambar 5.1, isi direktori /etc/apache2/ssl*

* Untuk membuat CSR, sebelumnya harus dibuat dulu public/private key terlebih dahulu, untuk itu, pertama-tama kosongkan terlebih dahulu sertifikat yang telah di-generate sebelumnya dengan perintah *$rm \**



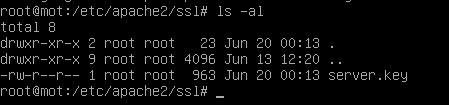
*Gambar 5.2, kosongkan direktori /etc/apache2/ssl*

* Kemudian dibuat key dengan nama ***server.key*** untuk private key nya dengan perintah *$openssl genrsa –des3 –out server.key 1024*



*Gambar 5.3, Buat private key*

* Pada saat pembuatan server.key, akan diminta memasukan passphrase. Pada percobaan ini digunakan passphrase ***student*.** Setelah private key tergenerate akan dibuat file pad direktori */etc/apache2/ssl* sebagai berikut

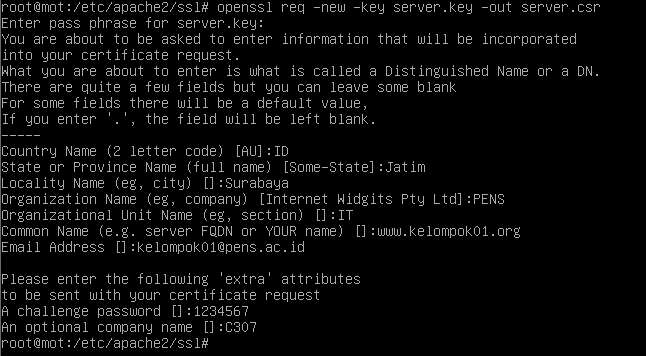


*Gambar 5.4, hasil key generate*

**

*Gambar 5.5, isi dari private.key*

* Setelah private key tergenerate, selanjutnya adalah pembuatan file *server.csr* menggunakan private key yang ada. Jalankan perintah *$openssl req –new –key server.key –out server.csr*



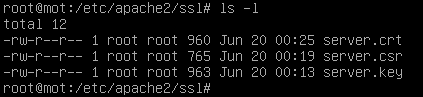
*Gambar 5.6 pembuatan file server.csr*

* Akan diminta informasi untuk informasi certificate yang akan digunakan. Pastikan pada Common Name nilainya sama dengan alamat web/IP address yang akan dipasang secure
* Selanjutnya adalah membuat file server.crt menggunakan server.key dan server.csr nya menggunakan perintah *$openssl x509 –req –days 365 –in server.csr –signkey server.key –out server.crt*



*Gambar 5.7, generate server.crt*

* Setelah itu cek direktori ssl, akan ada 3 file yaitu server.key, server.crt dan server.crs

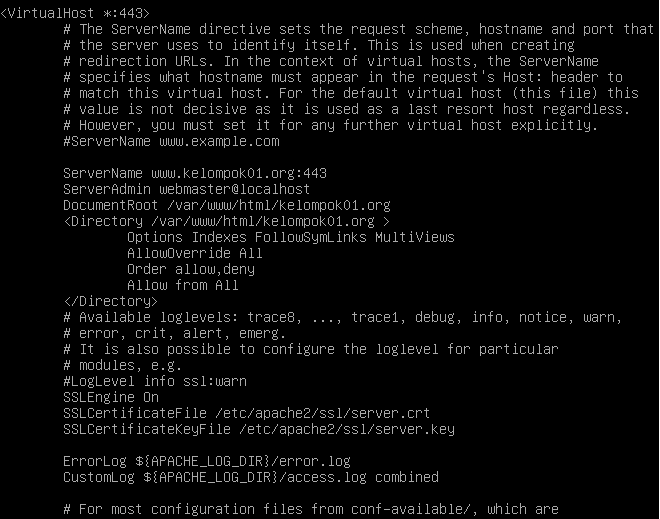


*Gambar 5.8, tiga file pada direktori ssl*

* 1. Konfigurasi Apache untuk secure site

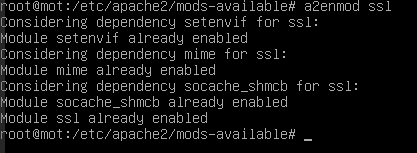
Terdapat beberapa perubahan untuk definisi virtual host pada website [*www.kelompok01.org*](http://www.kelompok01.org) jika ingin menggunakan https, yaitu perubahan pada file konfigurasi yang terletak di */etc/apache2/site -available/kelompok01.org.conf* edit file tersebut menjadi seperti di bawah ini

* Edit file /etc/apache2/site-available/kelompok01.org.conf



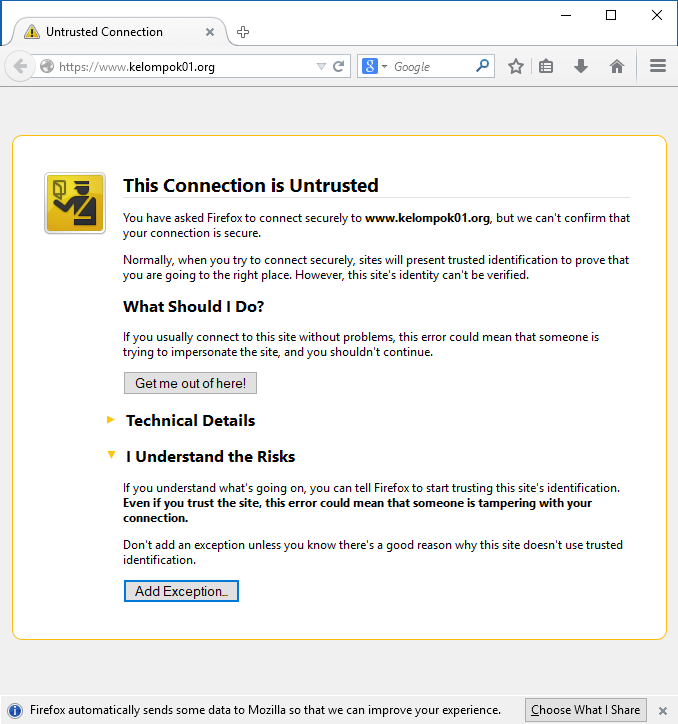
*Gambar 6.0, Edit konfigurasi website* [*www.kelompok01.org*](http://www.kelompok01.org)

* Setelah itu aktifkan mod ssl dengan berpindah ke direktori */etc/apache2/mods-available/* lalu menjalanakan perintah *$a2enmod ssl*



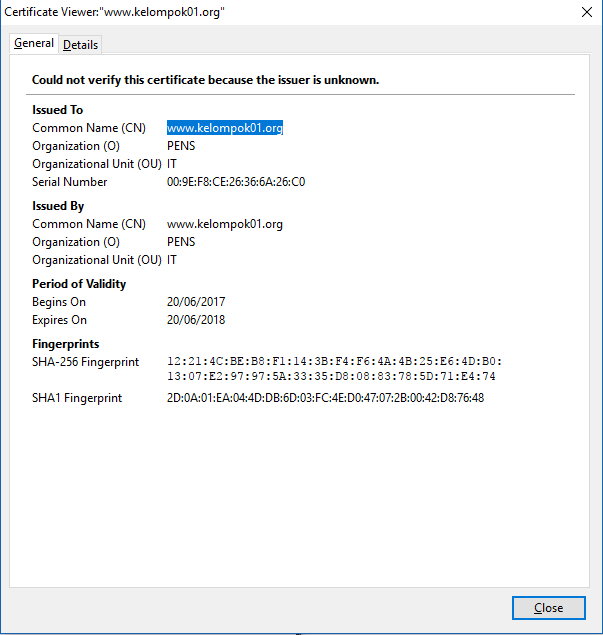
*Gambar 6.1, enable mod ssl*

* Restart service apache2, akan diminta passpahrase, masukan passphrase dari private.key
  1. Menjalankan HTTPS
* Tes https dengan mengakses website [*www.kelompok01.org*](http://www.kelompok01.org)dengan alamat [*https://www.kelompok01.org*](https://www.kelompok01.org)
* Akan muncul tampilan seperti gambar 7.0. Hal ini diakibatkan karena sertifikat yang dibangkitkan oleh [*www.kelompok01.org*](http://www.kelompok01.org) bersifat self signed (gunakan browser mozilla firefox untuk melihat certificate)

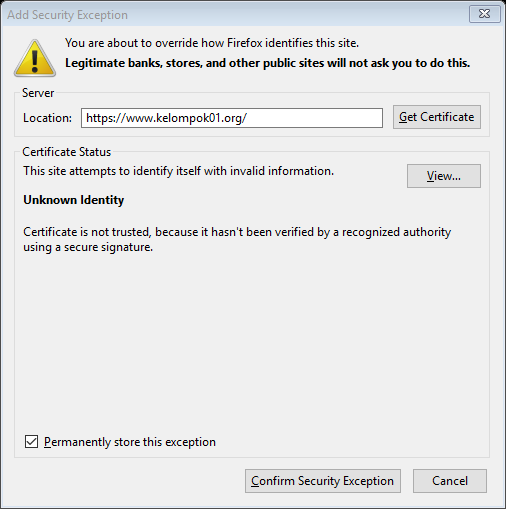


*Gambar 7.0 pesan untrusted connection pada mozilla*

* Untuk mengaktifkannya, kita akan membuat exception. Exception dapat dibuat dengan cara membuka menu add exception 🡪 get certificate 🡪 view (akan diperlihatkan certificate yang tadi dibuat) 🡪 checkbox **Permanently store this exception**🡪 Klik button **Confirm Security Exception**

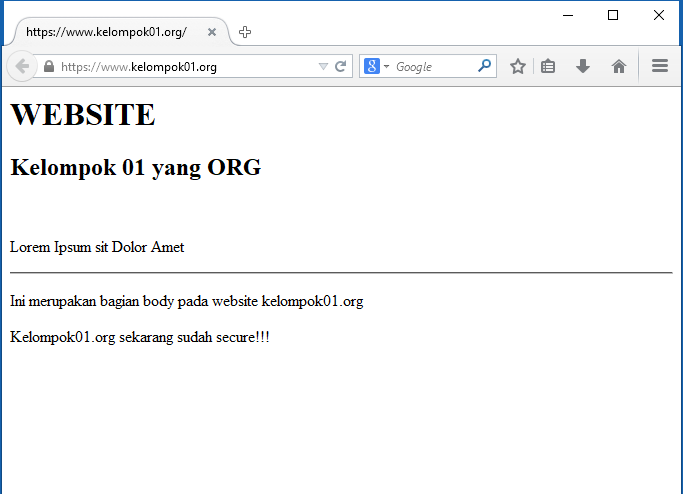


*Gambar 7.1, Certificate yang dibuat untuk* [*www.kelompok01.org*](http://www.kelompok01.org)

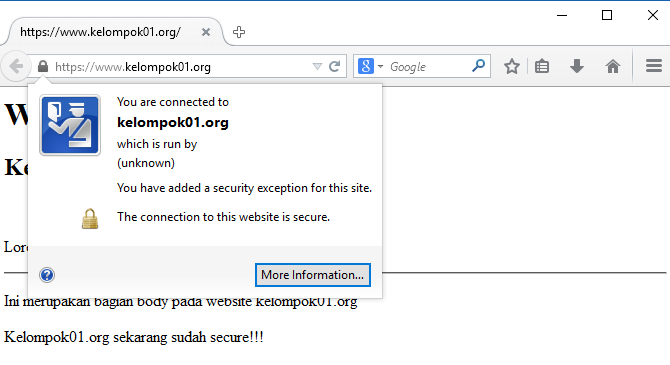
**

*Gambar 7.2, Exception untuk certificate*

* Setelah website terbuka, akan muncul lambang gembok di samping url yang menandakan bahwa koneksi sudah secur



*Gambar 7.3, Website* [*www.kelompok01.org*](http://www.kelompok01.org) *sudah secure*

**

*Gambar 7.4, website* [*www.kelompok01.org*](http://www.kelompok01.org) *secured*