Konfigurasi SSL Untuk Meningkatkan Keamanan *Web server* Pada Program Studi Teknik Informatika Universitas Muhammadiyah Surakarta

Ihsan Cahyo Utomo *[1], Siti Rokhmah*[2]
[1] Teknik Informatika, Universitas Muhammadiyah Surakarta,[2]Informatika, Institut Teknologi Bisnis AAS Indonesia Sukoharjo,
e-mail: ***[1]Ihsan.cahyo@ums.ac.id** , [2] sitirokhmah.itbaas@gmail.com

Abstrak

Keamanan sistem merupakan aspek terpenting dalam sebuah sistem informasi, lemahnya keamanan sistem dapat meningkatkan resiko serangan yang dapat menimbulkan kerusakan sistem. Salah satu serangan yang sering terjadi adalah serangan pada web server, jal tersebut dikarenakan web server terkoneksi dengan internet dan di akses oleh banyak user secara luas, sehingga resiko serangan menjadi meningkat. Sebagai contoh prodi Teknik Informatika UMS yang mengelola website jurusan dan diakses oleh ribuan mahasiswa, sehingga diperlukan sistem pengamanan web server untuk menhindari potensi serangan dan untuk meningkatkan keamanan data. Salah satu metode dalam mengamankan web server dengan menggunakan Secure Socket Layer (SSL). SSL sering digunakan untuk mengamankan komunikasi antara client dan server. Tahapan yang dilakukan dalam penelitian ini adalah analisis kebutuhan sistem, melakukan instalasi sistem operasi dan web server, konfigurasi sertifikat SSL dan analisis keamanan sistem setelah adanya konfigurasi SSL. Penelitian ini bertujuan untuk meningkatkan keamanan web server pada prodi informatika dan mengantisipasi berbagai potensi serangan pada web server prodi informatika.

Kata kunci*—3-5 web server, Secure Socket layer, keamanan jaringan.*

1. PENDAHULUAN

Salah satu aspek penting dalam sistem informasi dalam sebuah organisasi dan perusahaan adalah keamanan jaringan. Lemahnya keamanan jaringan pada sistem informasi dapat meningkatkan serangan *hacker* pada sistem tersebut. Serangan pada sistem yang dilakukan oleh *hacker* menyebabkan kerusakan pada system dan perubahan pada fungsi sistem [1]. Salah satu serangan yang sering terjadi adalah serangan pada *web server*. Hal tersebut dikarenakan *web server* tekoneksi dengan internet yang dapat diakses luas oleh user dari berbagai negara. Berbagai serangan ditujukan untuk merusak keamanan *web server* [2]. Beberapa jenis serangan sering terjadi pada *web server* adalah *Daniel of Services* (DOS), *SQL Injection*, *Cross Site Scripsting* (XSS). Data yang menjadi target serangan pada *web server* adalah data pada autentikasi *web server* [3] .

Program studi Teknik Informatika Universitas Muhammadiyah Surakarta (UMS) memiliki *web server* yang digunakan untuk mengelola website yang akan di akses oleh mahasiswa Teknik Informatika UMS. *Website* yang dikelola adalah *website* jurusan yang digunakan untuk memberikan informasi kepada mahasiswa tentang infomasi perkuliahan, informasi Tugas Akhir, informasi praktek kerja nyata, informasi akademik dan informasi *tracer studi* Alumni Teknik Informatika UMS serta semua informasi terbaru tentang kegiatan perkuliahan di teknik informatika UMS. Aktivitas yang tinggi dalam mengakses *web server* mengakibatkan meningkatnya potensi serangan terhadap *web server*[4]. Sehingga diperlukan upaya untuk mengamankan *web server* pada program studi Teknik Informatika UMS. Salah satu

metode dalam mengamankan *web server* dengan menggunakan *Secure Socket Layer* (SSL). SSL sering digunakan untuk mengamankan komunikasi antara *client* dan *server* [5].

SSL merupakan jalur khusus yang aman dalam melakukan transaksi pada *website*, hal tersebut dikarenakan transaksi data mengelami proses *enkripsi*. Konsep yang digunakan dalam SSL adalah kosep kriptografi dengan kosep kunci publik dimana kedua pihak yang saling berkomunikasi dengan data yang disamarkan dan untuk membacanya menggunakan sandi yang hanya dimiliki kedua pihak tersebut [6]. Salah satu alasan penggunaan SSL adalah dikarenakan SSL merupakan standar protocol yang umum diamankan dalam mengamankan komunikasi pada website, selain itu SSL merupakan protocol yang mudah digunakan karena cukup mengaktifkan layanan tanpa adanya software tambahan [6].

Beberapa penelitian terkait kemanan *web server* dan SSL telah dilakukan, diantaranya adalah penelitian yang dilakukan oleh nazwita dan ramadhani. Dalam penelitan ini menganalisis keamanan pada *web server* dimana penyerang melakukan penyusupan dengan port scanning [2]. Penelitian lain terkait SSL adalah penelitian dalam mengimplementasikan SSL dalam membantu meningkatkan keamanan *web server* [7][8][9].dalam penelitian yang dilakukan oleh pranata, dkk dilakukan analisis kemanan SSL terhadap serangan s*niffing*, sehingga dapat mengetahui sejauh mana SSL dapat mengamankan data [10]. Sejalan dengan penelitian sebelumnya, penelitian ini memanfaatkan SSL untuk membantu meningkatkan keamanan *web server* pada program studi informatika UMS. Dengan penelitian ini diharapkan dapat meningkatakn keamanan *web server* pada prodi teknik informatika UMS. Sehingga dapat mengurangi terjadinya serangan pada *web server*.

2. METODE PENELITIAN

2.1 Analisis permasalahan

Langkah Analisa masalah merupakan Langkah awal untuk menentukan ruang lingkup penelitian yang akan dilaksanakan. Dengan menganalisa masalah yang terjadi, diharapakan mendapatkan solusi dari masalah tersebut.

2.2 Studi pustaka

Dalam penelitian ini terlebih dahulu dilakukan studi terhadap referensi baik berupa jurnal, buku maupun artikel lain yang relevan. Adapun jurnal yang dijadikan referensi diantaranya adalah jurnal tentang keamanan web server, implementasi SSL dalam keamanan web server dan analisis penerapan SSL.

a. Secure Socket Layer

Secure Socket Layer (SSL) adalah sebuah protokol 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. SSL adalah protocol yang diciptakan sebelum TLS yang mengaplikasikan hal ini. SSLbiasanya dioperasikan secara bersama-sama dengan HTTP, sehingga membentuk protocol baru yang disebut HTTPS, untuk mengamankan transaksi lewat *website* [9], [11].

b. HTTPS

Hipertext Transfer Protocol Secure (HTTPS) menggabungkan protokol HTTP dan SSL untuk menjamin keamanan komunikasi antara Web 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 kita tidak dapat melakukan otentikasi user selama koneksi. User harus melakukan sejumlah otentikasi tambahan seperti password, biometric atau metode otentikasi lain. Komunikasi SSL 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 [12].

2.3 Analisis Kebutuhan sistem

Pada tahapan ini dilakukan terhadap hardware dan software yang terkait dengan *web server*. Pada tahapan ini juga dilakukan analisis terhadap data – data yang rentan terhadap serangan dan port-port yang sering dijadikan celah dalam menyerang *web server*.

2.4 Perancangan software

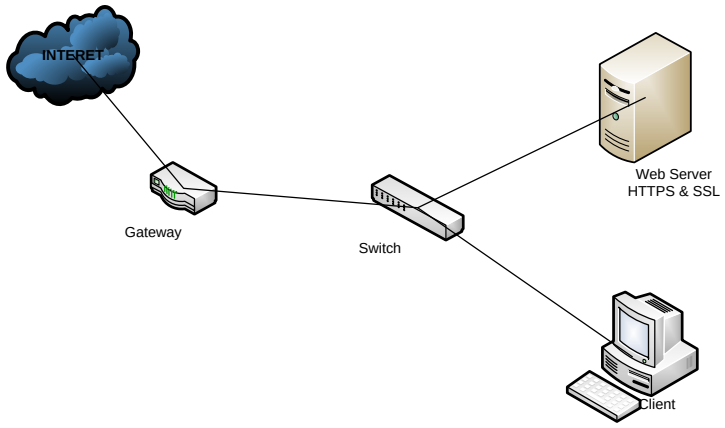
Pada perancangan software dilakukan beberapa tahapan, diantaranya

1. Instalasi Sistem Operasi.
   Sistem operasi yang digunakan dalam konfigurasi ini adalah Ubuntu. Salah satu kelebihan Ubuntu adalah memiliki tingkat keamanan yang baik.
2. *Instalasi software*
   Software yang diinstal untuk *web server* adalah LAMP, Apache, dan Mysql Server. Selain itu dilakukan instalasi aplikasi untuk SSL pada *web server*.
3. Konfigurasi SSL
   Sertifikat SSL merupakan protokol keamanan yang dirancang dengan algoritma *Rivest Shamir Addleman* (RSA) dan menggunakan *Advanced Encription Standart* (AES) yang merupakan standart enkripsi dengan kunci simetris. Standar ini memiliki 3 blok penyandian yaitu AES-128, AES-192 dan AES-256. Enkripsi AES menggabungkan beberapa *class* yang terdapat pada *OpenSSL* dan aplikasi *keytool* yang merupakan bagian dari *java SDK*.

3. HASIL DAN PEMBAHASAN

3.1. Instalasi server

Sebagai langkah dalam penelitian ini dilakukan instalasi server ubuntu/debian beserta *software* didalamnya. Desain topologi *web server* pada prodi Teknik Informatika UMS dapat dilihat pada Gambar 1. Tahap paling penting dari penelitian ini adalah membuat *public/private key*, *SSL sertificate*, *certificate signing request* (CSR).



Gambar 1 Topologi *web server* pada prodi informatika

Adapun langkah instalasi pada penelitian ini adalah

1. Masuk pada Ubuntu Server masuk ke Root Mode
2. Melakukan update repository software
3. Melakukan instalsai software *web server* yaitu LAMP Server (apache2, mysql-server, PHP5, PhpMyAdmin)
4. Melakukan instalasi packet software SSL Langkah instalasi dapat dilihat pada gambar 2. dan gambar 3.

```
Do you want to continue? [Y/n]
Get:1 http://kambing.ui.ac.id/ubuntu/ trusty-updates/main openssl i386 1.0.1f-1u
buntu2.16 [479 kB]
Fetched 479 kB in 15s (31.8 kB/s)
(Reading database ... 58965 files and directories currently installed.)
Preparing to unpack .../openssl_1.0.1f-1ubuntu2.16_i386.deb ...
Unpacking openssl (1.0.1f-1ubuntu2.16) over (1.0.1f-1ubuntu2) ...
Processing triggers for man-db (2.6.7.1-1) ...
Setting up openssl (1.0.1f-1ubuntu2.16) ...
root@ubuntuserver:/#
```

Gambar 2. Proses Instalasi Open SSL

3.2. Konfigurasi sertifikat SSL

Untuk menyimpan sertifikat SSL, dibuat sebuah folder yang ditempatkan di direktori "/etc/apache2/ssl" kemudian mengaktifkan mod SSL dan Restart service apache2 dan melakukan request SSl. Tahapan konfigurasi untuk request SSL dapat dilihat pada gambar 3.

```
root@ubuntuserver:/home/user1# cd /etc/apache2
root@ubuntuserver:/etc/apache2# cd ssl
root@ubuntuserver:/etc/apache2/ssl# ls
apache.crt  apache.key
root@ubuntuserver:/etc/apache2/ssl#
```

Gambar 3. request sertifikat

Untuk menghindari kesalahan konfigurasi, dilakukan backup file default-ssl.conf. tahapan backup file dapat dilhat pada gambar 5. Backup file dilakukan untuk menghindari kehilangan data jika terjadi kesalahan dalam melakukan konfigurasi.

```
root@ubuntuserver:/etc/apache2# cd sites-available/
root@ubuntuserver:/etc/apache2/sites-available# ls
000-default.conf  default-ssl.conf
root@ubuntuserver:/etc/apache2/sites-available# cp default-ssl.conf default-ssl-backup.conf
root@ubuntuserver:/etc/apache2/sites-available# ls
000-default.conf  default-ssl-backup.conf  default-ssl.conf
root@ubuntuserver:/etc/apache2/sites-available#
```

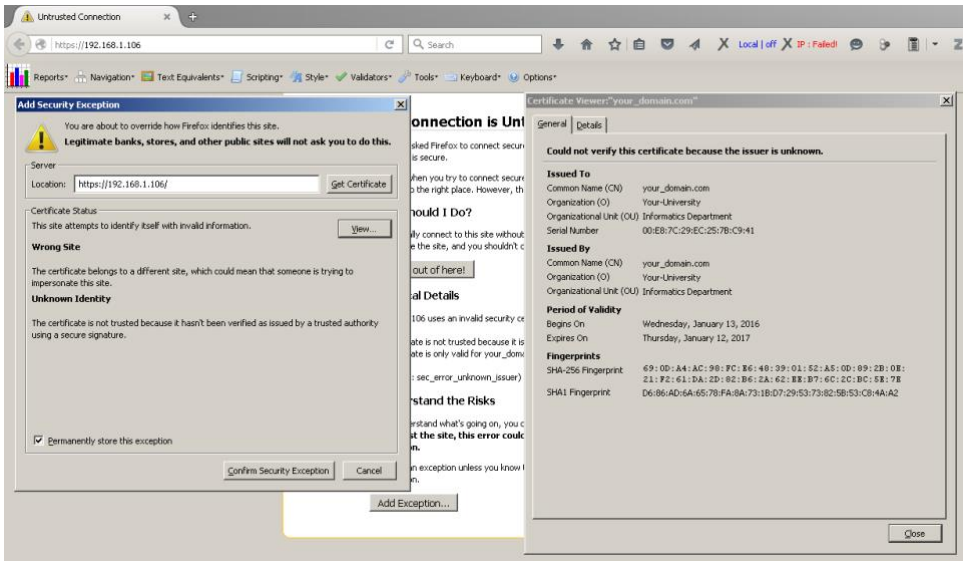
Gambar 5. Backup File default-ssl.conf

Setelah semua file ter-*backup* maka dilakukan aktivasi *virtualhost*. Aktivasi ini bertujuan untuk menambah banyak domain dalam satu server. Dengan aktifasi virtual host dimungkinkan adanya banyak website dengan domain yang berbeda pada server yang sama. Tahapan dalam aktivasi virtual host dapat dilihat pada gambar 6.

```
root@ubuntuserver:/etc/apache2/sites-available# a2ensite default-ssl.conf
Enabling site default-ssl.
To activate the new configuration, you need to run:
  service apache2 reload
root@ubuntuserver:/etc/apache2/sites-available# service apache2 reload
 * Reloading web server apache2
 *
root@ubuntuserver:/etc/apache2/sites-available#
```

Gambar 6. Aktifasi Virtualhost untuk SSL

Langkah berikutnya adalah memastikan apakah sertifikat telah terinstall pada webserver. Untuk menampilkan sertifikat SSL, masuk pada web browser dengan IP prodi Informatika dan memilih *confirm security menu* dan *get sertificate* SSL untuk memperoleh sertifikat SSL. tampilan sertifikat SSL dapat dilihat pada gambar 7.



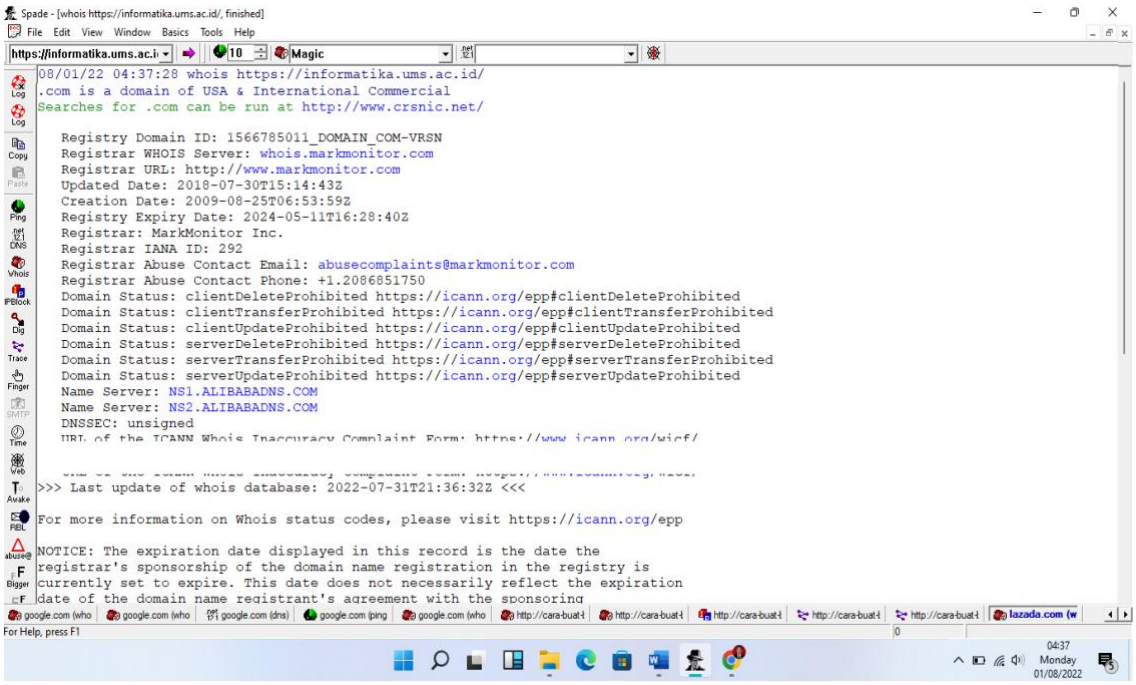
Gambar 7. Sertifikat SSL

3.3 Tahapan pengujian

Pada tahapan ini dilakukan pengujian dari konfogurasi SSL pada *web server* program studi Teknik Informatika. Dari hasil pengujian dilakukan proses *sniffing* untuk menguji *web server* pada program studi Teknik Informatika mengalami peningkatan dari segi keamanan. Proses *sniffing* pada tahap pengujian dilakukan dengan menggunakan aplikasi *sam speed*.

a. Tes awal (Pre-test)

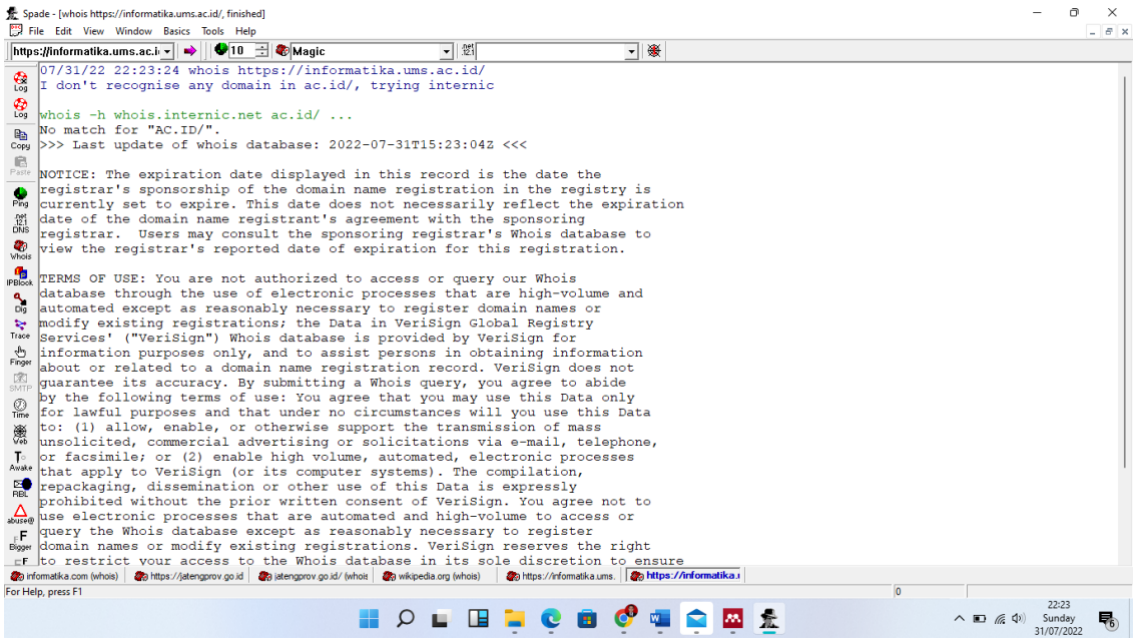
Sebelum menerapkan SSL, peneliti melakukan proses *sniffing* pada halaman *website* program studi teknik informatika, dan hasilnya muncul informasi tentang halaman website. Hasil proses *sniffing* dapat dilihat pada gambar 8.



Gambar 8. Proses Sniffing sebelum menerapkan SSl

b. Uji coba penerapan SSL

Setelah menerapkan SSL, pada saat orang melakukan *sniffing* pada halaman web tersebut, maka tidak akan muncul informasi tentang halaman tersebeut. Hasil sniffing setelah menerapkan SSL dapat dilihat pada gambar 10.



Gambar 9. Hasil sniffing setelah menerapkan SSL

Pada proses pengujian tersebut, didapatkan hasil bahwa sebelum di terapkan SSL dilakukan proses sniffing. Sniffing merupakan kegiatan yang dilakukan untuk memantau kegiatan paket data yang melintasi jaringan. Sebelum diterapkan SSL user dapat melihat aktivitas dan informasi halaman *website*, dan setelah diterapkan SSL user yang melakukan sniffing tidak dapat melihat informasi dan aktifitas pada halaman *website*.

4. Kesimpulan

Dari Analisa keamanan *web server* pada program studi teknik informatika UMS menggunkan SSL, dapat diambil beberapa kesimpulan

- Untuk meningkatkan keamanan dan mengantisipasi potensi keamanan *web server* diperlukan upaya untuk mengamankan *web server*. Salah satunya adalah dengan menggunakan SSL.
- SSL merupakan protokol khusus untuk mengamankan data dengan teknik enkripsi. Pada penelitian ini dilakukan beberapa tahapan dalam konfigurasi SSL diantaranya instalasi sistem operasi, instalasi software *web server* dan konfigurasi sertifikat SSL.
- Melalui pengujian yang dilakukan dengan proses sniffing, didapatkan *web server* pada prodi Teknik Informatika UMS lebih aman setelah dilakukan konfigurasi SSL.

5. SARAN

Kami menyadari penelitian ini masih banyak kekurangan. Harapan untuk penelitian selanjutnya adalah megembangkan sistem keamanan *web server* dengan metode yang berbeda untuk lebih meningkatkan keamanan *web server* dari berbagai serangan.

UCAPAN TERIMA KASIH

Penulis mengucapkan terima kasih kepada semua pihak yang telah membantu dalam kegiatan penelitian ini.

DAFTAR PUSTAKA

[1] F. Fachri, A. Fadlil, and I. Riadi, “Analisis Keamanan Webserver menggunakan Penetration Test,” *J. Inform.*, vol. 8, no. 2, pp. 183–190, 2021, doi: 10.31294/ji.v8i2.10854.

[2] Nazwita and S. Ramadhani, “Analisis Sistem Keamanan *Web server* Dan Database Server Menggunakan Suricata,” *Semin. Nas. Teknol. Inf. Komun. dan Ind.*, vol. 0, no. 0, pp. 2579–5406, 2017.

[3] Handi Prasetyo, “ANALISA KEAMANAN *WEB SERVER* MENGGUNAKAN WEB APPLICATION FIREWALL MODSECURITY Artikel Ilmiah,” *Fak. Teknol. Inf. Univ. Kristen Satya Wacana*, pp. 1–20, 2016.

[4] M. Arman and N. Rachmat, “Implementasi Sistem Keamanan *Web server* Menggunakan Pfsense,” *Jusikom J. Sist. Komput. Musirawas*, vol. 5, no. 1, pp. 13–23, 2020, doi: 10.32767/jusikom.v5i1.752.

[5] E. Insanudin, “Implementasi Sistem Keamanan Berbasis Web dengan Protokol SSL Abstrak :,” no. May, 2016.

[6] Patel, “済無No Title No Title No Title,” pp. 9–25, 2019.

[7] E. W. Budihardjo *et al.*, “Pembuatan Konfigurasi SSL yang Aman untuk Diimplementasikan pada Apache dan Nginx,” pp. 1–6.

[8] H. Fernando, “Studi dan Implementasi Sistem Keamanan Berbasis Web dengan Protokol SSL di Server Students Informatika ITB,” 2010.

[9] N. Novi and Z. Zaini, “Secure Socket Layer untuk Keamanan Data Rekam Medis Tumor

Otak pada Health Information System," *J. Nas. Tek. Elektro*, vol. 6, no. 3, p. 137, 2017, doi: 10.25077/jnte.v6n3.405.2017.

[10] H. Pranata, L. A. Abdillah, and U. Ependi, "Analisis Keamanan Protokol Secure Socket Layer (SSL) Terhadap Proses Sniffing di Jaringan," pp. 21–22, 2015.

[11] W. Agustiara, A. Pratama, S. Junaidi, K. Padang, and S. Barat, "Analisis Keamanan Protokol Secure Socket Layer Terhadap Serangan Packet Sniffing Pada Website Portal," vol. 6, no. 1, 2022.

[12] W. S. Raharjo and A. A. Bajuadji, "Analisa Implementasi Protokol HTTPS pada Situs Web Perguruan Tinggi di Pulau Jawa," *J. Ultim.*, vol. 8, no. 2, 2017, doi: 10.31937/ti.v8i2.518.