Jurnal Sistim Informasi dan Teknologi
https://jsisfotek.org/index.php
2021 Vol. 3 No.1 Hal: 1-6 e-ISSN: 2686-3154

Meningkatkan Keamanan Webserver Aplikasi Pelaporan Pajak Daerah Menggunakan Metode Penetration Testing Execution Standar

Yosua Ade Pohan[1]✉, Yuhandri Yunus[2], Sumijan[3]
[1,2,3]Universitas Putra Indonesia YPTK Padang
joeadepohan@gmail.com

Abstract

Regional Tax Reporting Application Webserver is one of the public services for taxpayers to report their sales transactions. This application can be accessed on the domain http://sptpd.payakumbuhkota.go.id. This application is public, so the principles of information security must be applied to prevent cyber attacks. The principles of information security include confidentiality, integrity, and availability. To apply this information security principle, it is necessary to conduct vulnerability assesment of the application webserver. This study aims to improve the security of the application webserver so that the data and information in it is secure. The method used in this study is the Penetration Testing Execution Standard which is one of the methods developed by the Pentest Organization to become a standard in analyzing or auditing security systems. The results of vulnerability testing using software Acunetix, Nikto, BurpSuite and Owasp, there are seven types of vulnerabilities, namely: X-Frame Header Options is Missing, CSRF Attack, Cookie Without Only Flash, DNS Vulnerability, Ddos Attack, Bruteforce Page Login and Open Port. The vulnerability can be exploited, where the level of application vulnerability is in the medium category. The recommendations for fixing vulnerabilities can be applied by the developer, so that after repairs are made, the vulnerability level of the application webserver is in the low category and there is only one type of vulnerability, namely Brute Force Page Login.

Keywords: Pentest, Webserver, Vulnerability Assesment, Security, Attack.

Abstrak

Webserver Aplikasi Pelaporan Pajak Daerah merupakan salah satu layanan publik bagi Wajib Pajak untuk melaporkan transaksi penjualannya secara online. Aplikasi ini dapat dikases pada domain http://sptpd.payakumbuhkota.go.id. Aplikasi ini bersifat publik sehingga prinsip keamanan informasi harus diterapkan agar tidak mendapat serangan cyber. Prinsip keamanan informasi itu sendiri mencakup *confidentiality*, *integrity*, dan *availability*. Untuk menerapkan prinsip keamanan informasi tersebut maka perlu dilakukan pengujian kerentanan terhadap webserver aplikasi. Penelitian ini bertujuan untuk meningkatkan keamanan webserver aplikasi sehingga data dan informasi didalamnya terjaga kemanannya. Metode yang digunakan pada penelitian ini adalah *Penetration Testing Execution Standar* yang merupakan salah satu metode yang dikembangkan Pentest Organisation untuk menjadi standar dalam menganalisa atau mengaudit sistem kemanan Hasil dari pengujian kerentanan menggunakan software Acunetix, Nikto, BurpSuite dan Owasp terdapat tujuh buah jenis kerentanan yaitu: *X-Frame Header Options is Missing*, *CSRF Attack*, *Cookie Without OnlyFlag*, *DNS Vulnerability*, *Ddos Attack*, *Bruteforce Page Login* dan *Open Port*. Kerentanan dapat dilakukan exploitasi, dimana level kerentanan aplikasi adalah pada kategori medium. Rekomendasi perbaikan kerentanan dapat diterapkan oleh pengembang, sehingga setelah dilakukan perbaikan maka level kerentanan webserver aplikasi pada kategori low dan hanya terdapat satu jenis kerentanan yaitu Brute Force Page Login.

Kata kunci: Pentest, Webserver, Penilaian Kerentanan, Keamanan, Serangan.

1. Pendahuluan

Badan Keuangan Daerah Kota Payakumbuh telah mengembangkan layanan publik untuk pelaporan pajak daerah secara online. Manfaat layanan publik bagi pemerintahan adalah dapat mengurangi human error dan meningkatkan efektivitas kerja, meningkatkan dalam pengambilan keputusan serta memudahkan masyarakat tanpa harus mendatangi kantor [1]. Berdasarkan data Badan Siber dan Sandi Negara Tahun 2019, situs milik Pemerintah cukup banyak terjadi serangan seperti: *Web Defacement* 34%, *Phising* 9%, *Malware* 13% , Kerentanan lain 37% [2]. Berdasarkan data tersebut maka aplikasi layanan publik harus diterapkan prinsip dari keamanan informasi. Prinsip kemanan informasi terdiri dari *Confidentiality* yaitu kerahasiaan, *Integrity* yaitu data tidak berubah dari aslinya dan *Availability* yaitu ketersediaan [3]. Webserver aplikasi pelaporan pajak daerah dapat diakses di domain http://sptpd.payakumbuhkota.go.id. Aplikasi ini dapat diakses oleh publik sehingga terbuka untuk dilakukan serangan, data dan informasi didalamnya juga bersifat sensitive karena berhubungan dengan transaksi keuangan wajib pajak.

Webserver aplikasi pajak daerah menggunakan Bahasa Pemrograman PHP, dimana Engine PHP menerjemahkan PHP ke webserver yang ada, apa yang sebenarnya dilihat oleh pengguna di *browser* adalah bukan file PHP itu sendiri melainkan output dari semua perintah PHP yang dikrimkan kembali oleh webserver [4]. Berdasarkan data dan informasi tersebut maka

Diterima: 07-10-2020 | Revisi: 21-10-2020 | Diterbitkan: 31-03-2021 | DOI: 10.37034/jsisfotek.v3i1.36

webserver aplikasi perlu dilakukan pengujian kerentanan untuk mencari celah keamanan dan dilakukan perbaikannya. Penilaian kerentanan bisa mendeteksi hampir semua celah kerentanan yang biasanya terjadi pada sebuah sistem [5]. Metode yang digunakan adalah *Penetration Testing Execution Standar* yang dikembangkan *Pentest Organisation* untuk menjadi standar dalam menganalisa atau mengaudit sistem kemanan. Beberapa penelitian yang dilakukan dengan *Penetration Testing* diantaranya Penetration Testing dapat dilakukan pada *Internet of thing* didapatkan hasil kerentanan dari tiga lapisan layer yaitu *application layer, network layer* dan *perception layer*, terdapat kelemahan password yang tidak dienkripsi, kerentanan *sniffing* dan *spoofing* [6]. Penelitian berikutnya dilakukan dengan melakukan pengujian keamanan terhadap 3 buah aplikasi web dengan metode *Penetration Testing*. Hasil pengujian didapatkan kelamahan berupa *missing X-XSS header protection, insecure transportation security protocol (TLS 1.0), sourcode disclousure* dan beberapa file *jquery* yang sudah kadaluarsa [7].

Pengujian kerentanan juga dapat dilakukan pada website paud dikmas dengan menggunakan metode *Penetration Testing*, terdapat celah keamanan seperti *Port FTP* yang terbuka, web application information *disclosure*, dan lemahnya keamanan untuk autentikasi login pada website [8]. *Penetration Testing* juga dapat dilakukan terhadap IP server 192.168.197.130. Hasil pengujian terdapat beberapa kelemahan yaitu terbukanya port *FTP* dan *SSH* yang bisa dilakukan exploitasi kedalamnya [9]. Penetration Testing juga dapat dilakukan pada localhost dengan hasil kerentanan *X-Frame Option Header Missing, Cross site scripting (XSS)* dan *SQL Injection* [10]. Penelitian lainnya adalah dengan penetrasi aplikasi web didapat hasil *SQL Injection, Cross Site Scripting, Local File Inclusion (LFI)* dan *Parameter Tampering* [11]. Untuk meningkatkan keamanan *webserver* aplikasi pelaporan pajak daerah agar sesuai dengan prinsip keamanan informasi maka perlu dilakukan pengujian kerentanan aplikasi dengan menggunakan Metode *Penetration Testing Execution Standar*.

2. Metodologi Penelitian

2.1. Subjek Penelitian

Subjek pada penelitian ini adalah pada *webserver* aplikasi pelaporan pajak daerah, dengan domain http://sptpd.payakumbuhkota.go.id. Penelitian dilakukan dengan melakukan pengujian terhadap aplikasi (*source code*) dan webserver sebagai tempat berjalanya aplikasi.

2.2. Metode *Penetration Testing Execution Standar*

Tahapan dalam metode Penetration Testing Execution Standar adalah [12]:

a. Pre-Engegment

Tahap ini dilakukan interaksi dengan Badan Keuangan Daerah, mengenai tujuan dari penelitian, menentukan scope dan pertanyaan mengenai gambaran umum *webserver* aplikasi.

b. Intelligence Gathering

Tahap ini peneliti mengumpulkan informasi mengenai webserver aplikasi, cara kerja sistem, *IP Address* dan lainnya.

c. *Threat Modelling*

Tahapan ini peneliti menggunakan informasi yang didapat untuk membuat gambaran model ancaman apa yang dapat terjadi pada aplikasi. Menentukan aset kritis dan dampak yang terjadi.

d. *Vulnerability Analysis*

Tahapan ini dilakukan pemindaian kerentanan dan analisis hasil kerentanan dan menentukan jenis serangan apa yang bisa dilakukan exploitasi.

e. *Exploitation*

Tahap ini dilakukan serangan terhadap kerentanan yang ditemukan sekaligus menguji apakah kerentanan tersebut memang benar bisa di *exploitasi*.

f. *Post Exploitation*

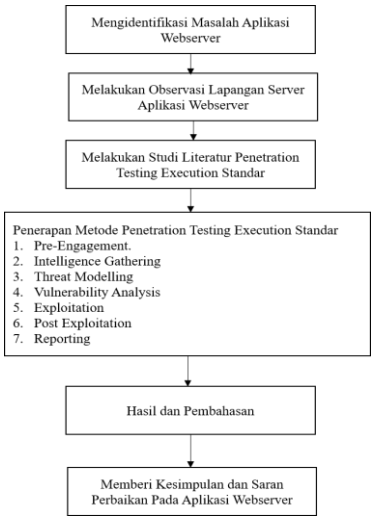
Tahap ini dilakukan perbaikan dengan menerapkan solusi yang tepat untuk mengatasi kerentanan dan dilakukan kembali pengujian kerentanan.

g. *Reporting*

Tahap ini peneliti membuat laporan berupa hasil pengujian kerentanan sebelum dilakukan perbaikan *bug* dan setelah dilakukan perbaikan *bug*.

2.3. Kerangka Kerja Penelitian

Kerangka kerja dibutuhkan agar peneliti memiliki pedoman dan arah yang jelas dalam melakukan penelitian [13].



Gambar 1. Kerangka Kerja Penelitian

Kerangka kerja penelitian tiap tahapnya memiliki keterkaitan dengan tahap berikutnya [14]. Penelitian dimulai dengan mengidentifikasi masalah pada webserver aplikasi, melakukan observasi langsung webserver aplikasi, melakukan studi *literatur* mengenai penetrasi testing, menerapkan metode yang digunakan, melakukan hasil dan pembahasan serta membuat kesimpulan dan saran [15].

3. Hasil dan Pembahasan

3.1. Analisa Data

Data yang digunakan dalam penelitian ini adalah *webserver* aplikasi dengan domain http://sptpd.payakumbuhkota.go.id. Data awal yang diperoleh oleh peneliti adalah

1. Adanya serangan *Brute Force SSH* pada *webserver*.
2. Response time yang lambat dari aplikasi pada saat tertentu.
3. Adanya serangan terhadap *Header HTTP* berdasarkan *log access* yang diterima.

3.2. Penerapan *Penetration Testing Execution Standar*

3.2.1. Pre-Engagement

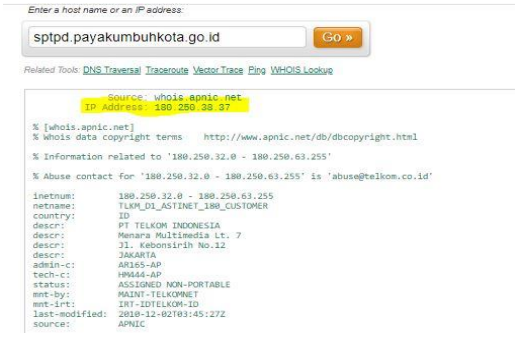
Tahap ini dilakukan interaksi dengan Badan Keuangan Daerah dan menentukan scope penelitian yaitu di domain http://sptpd.payakumbuhkota.go.id. Hasil dai tahapan ini sebagai berikut;

a. Waktu Penelitian disepakati 40 hari kerja.
b. Waktu untuk penetrasi testing dilakukan setelah jam kerja.
c. Backup aplikasi dilakukan rutin setiap hari.
d. *Username* dan *Password* aplikasi dan *webserver* dapat diberikan jika diperlukan.

3.2.2. Intelligence Gathering

Tahap ini peneliti mulai melakukan pencarian informasi terkait *webserver* aplikasi sebagai berikut:

a. IP Whois



Gambar 2. *Ip Whois*

Berdasarkan Gambar 2 IP Publik webserver aplikasi yang digunakan adalah 180.250.38.37 Astinet Telkom.

b. Scanning port dengan *Nmap*

Hasil pemindaian *port* dengan menggunakan tool nmap didapat hasil:

Tabel 1. Hasil Pemindaian *Port* dengan *Nmap*

No	Port	Status	Service
1	21	*Open*	*File Transfer Protocol (FTP)*
2	25	*Open*	*SMTP (Email)*
3	53	*Open*	*DNS Server*
4	80	*Open*	*Httpd (Apache)*
5	3306	*Open*	*MySQL*
6	10000	*Open*	*Webmin Panel*

3.2.3. Threat Modelling

Peneliti pada tahap ini membuat gambaran model ancaman yang terjadi aplikasi berdasarkan dari threat agents (penyerang) sebagai berikut:

Tabel 2. Threat Agents berdasarkan motivation

No	Motivasi	*Possibility*
1	Balas Dendam	*High*
2	Keingintahuan	*High*
3	Financial	*High*
4	Politik	*Medium*
5	Terorisme	*Low*
6	Agama	*Low*

Tabel 3. Threat Agents berdasarkan Capability

No	Capability	Analisa terhadap Aplikasi
1	Tool yang digunakan	Scanning Tools (Acunetix, Nmap) Exploitation tool: KaliLinux
2	Ketersediaan *Payload/exploits*	Mudah dan cukup banyak tersedia untuk membuat serangan ke aplikasi
3	Mekanisme Komunikasi	Adanya diskusi, lokakarya, seminar antara *threat agents* membuat serangan terhadap aplikasi meningkat,
4	Aksesibiltas	Ketersediaan internet, tool dan modul untuk serangan dapat meningkatkan ancaman terhadap *webserver.*

3.2.4. *Vulnerability Analysis*

Tahapan ini dilakukan untuk mencari kelemahan dengan melakukan pemindaian aplikasi dengan menggunakan *tool Acunetix* yang berfungsi untuk melakukan pengujian keamanan Web didapat hasil kerentanan sebagai berikut:

Tabel 4. Hasil Kerentanan *Webserver* Aplikasi

No	Indikasi File	Jenis Kerentanan	Kategori
1	/login.php	*X-Frame Header Options is Missing*	*Medium*
2	/login.php	*HTML Form without CSRF Protection*	*Medium*
3	/	*Cookie without HttpOnly flag set*	*Medium*
4	/login.php	*Login page-password-guessing attack*	*Low*
5	*Port 21, 25, 53, 10000*	*Open Port*	*Low*
6	*Bind DNS Server*	*DNS Vulnerability*	*Low*
7	*Httpd (Apache)*	*Ddos Vulnerability*	*Low*

Berdasarkan hasil analisa kerentanan maka didapat 7 jenis kerentanan yang dapat dilakukan exploitasi

terhadap webserver aplikasi yang dijelaskan sebagai berikut:

a. *Clickjacking*

Deskripsi: Kerentanan disebabkan *Header Options* tidak di konfigurasi *SAMEORIGIN*.

Dampak: *Frame* bisa di load dari URL selain URL Aplikasi sehingga korban bisa melakukan click pada aplikasi dimana informasi sensitive bisa dikirim ke penyerang.

b. *Cross Site Request Forgery Attack*

Deskripsi: Kerentanan disebabkan tidak menyertakan *token* dalam validasi data POST.

Dampak: Penyerang bisa melakukan *update* data pada aplikasi dengan memanfaatkan ketidaktahuan dari user. Penyerang memaksa user untuk melakukan aksi yang tidak diinginkan oleh user.

c. *Session Hijacking*

Deskripsi: Kerentanan disebabkan *cookies* saat *user login* tidak dikonfigurasi ke *secure* atau *HttpOnly*.

Dampak: Penyerang dapat melakukan *sniffing* dan mendapatkan *cookies user*, yang akan digunakan oleh peneyerang untuk *baypass login* aplikasi secara illegal, sehingga data bisa dimodifikasi.

d. *Brute Force Attack*

Deskripsi: Kerentanan disebabkan *page index* aplikasi adalah halaman *login* sehingga terbuka dan aplikasi tidak berjalan pada *port https.*

Dampak: Penyerang dapat masuk kedalam aplikasi dengan melakukan percobaan menebak kata sandi untuk *login* aplikasi sehingga data dapat dimodifikasi.

e. *Open Port Attack*

Deskripsi: Kerentanan disebabkan tidak ada *filter* terhadap port yang perlu dibuka dan ditutup.

Dampak: Penyerang dapat mengupload file *backdoor* pada aplikasi dan menghentikan *service* yang diberikan oleh port tersebut.

f. *DNS Attack*

Deskripsi: Kerentanan disebabkan DNS tidak diinstall *DNSSEC* dan mode recursion yes

Dampak: Penyerang dapat melakukan DNS *Spoofing* dan *DNS Amplification.*

g. *Ddos Attack*

Deskripsi: Kerentanan disebabkan tidak ada instalasi mod_security untuk melakukan filtering terhadap paket yang masuk.

Dampak: Traffic dari webserver aplikasi menjadi besar sehingga kehabisan resource dan membuat tidak dapat diakses.

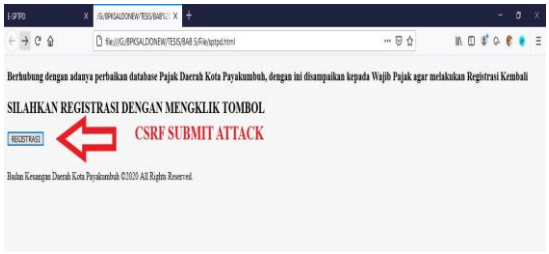
3.2.5. Exploitation

Tahapan ini peneliti melakukan pengujian terhadap hasil kerentanan dengan melakukan exploitasi. Tujuan dari tahapan ini adalah untuk membuktikan apakah hasil kerentanan dapat diexploitasi berdasarkan analisa yang dilakukan sebelumnya.

a. *Clickjacking*

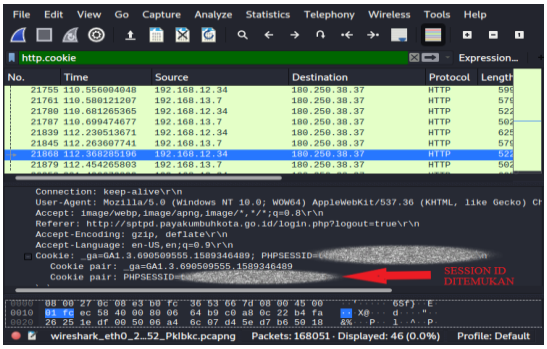
Gambar 3. *Clickjacking Attack*

b. *CSRF Attack*



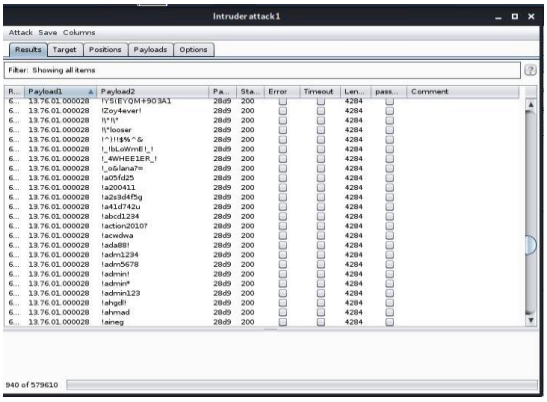
Gambar 4. *CSRF Attack*

c. *Session Hijacking*



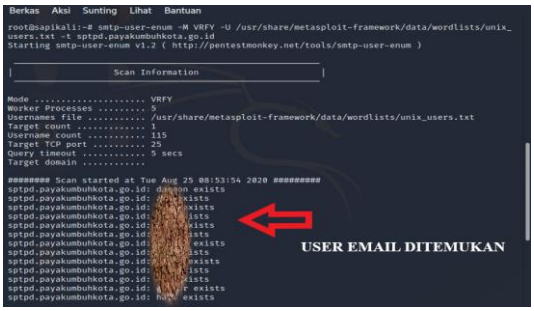
Gambar 5. *Session Hijacking*

d. *Brute Force Attack*



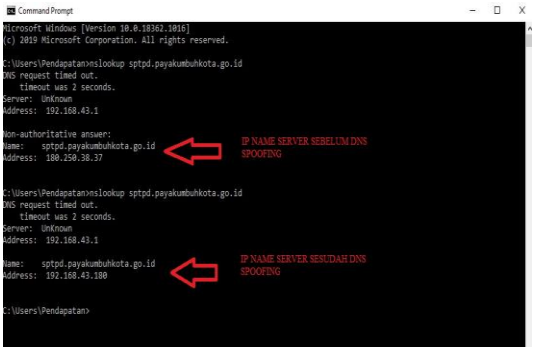
Gambar 6. *Brute Force Attack*

e. *Open Port Attack*



Gambar 7. *Port Attack*

f. *DNS Attack*



Gambar 8. *DNS Spoofing*

g. *DDos Attack*

Gambar 9. *DDos Attack*

Berdasarkan Hasil *exploitasi* yang dilakukan terhadap jenis kerentanan maka disimpulkan semua jenis kerentanan pada aplikasi dapat di lakukan *exploitasi*.

3.2.6. Post Exploitation

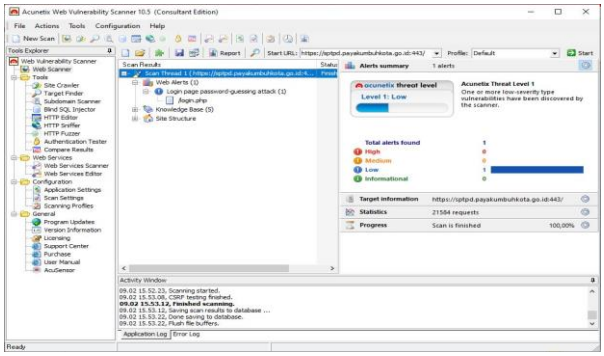
Sesudah tahapan *exploitation*, maka pada tahapan ini dilakukan perbaikan berdasarkan solusi yang tepat untuk mengatasi kerentanan.

Tabel 5. Solusi Terhadap Jenis Kerentanan

No	Jenis Kerentanan	Solusi
1	*X-Frame Header Options is Missing*	Setting Header Options ke SAMEORIGIN dan DENY Menambahkan SSL pada *webserver* aplikasi sehingga pertukaran data lebih secure karena menggunakan *https*.
2	*HTML Form without CSRF Protection*	Menambahkan token autentikasi pada saat pengiriman data dengan POST
3	*Cookie without HttpOnly flag set*	Melakukan konfigurasi *cookies flag* ke *Secure* atau *HttpOnly*
4	*Login page-password-guessing attack*	Menambahkan instalasi *fail2ban* pada *webserver* dan menambahkan *rule* untuk melakukan *filter* serangan *bruteforce* Mengganti port default webmin. Mengganti port ssh dari port *22* ke port lainnya
6	*DNS Vulnerability*	Melakukan konfigurasi *Bind DNS* dengan mengaktifkan *DNSSEC* dan menonaktifkan *mode recursive*
7	*Ddos Vulnerability*	Melakukan instalasi *mod_security* dan menambahkan rule untuk melakukan *block* serangan *DDos* terhadap *Port TCP* dan *UDP*.

3.2.7. Reporting

Tahap terakhir dari Metode Penetration Testing Execution Standar adalah membuat laporan pengujian kerentanan. Hasil penerapan solusi terhadap jenis kerentanan aplikasi dapat meningkatkan keamanan dari yang sebelumnya berkategori *Medium* menjadi Kategori *Low*,seperti pada Gambar berikut:



Gambar 10 Kategori Kerentanan Setelah Perbaikan

Laporan secara rinci Hasil pengujian kerentanan dapat disimpulkan sebagai berikut:

Tabel 6 Hasil Pengujian Kerentanan

No	Jenis Kerentanan	Hasil Exploitasi	Hasil Sesudah Perbaikan
1	*X-Frame Header Options is Missing*	Dapat dilakukan serangan *Clickjacking*	Aplikasi memblock Serangan *Clickjacking*
2	*HTML Form without CSRF Protection*	Dapat dilakukan update data melalui *form submit CSRF Attack*	*Webserver* menolak *request* tanpa adanya *token*
3	*Cookie without HttpOnly flag set*	*Session Cookies* dari *user login* dapat di *capture* dengan *sniffing wireshark*	*Session cookies* tidak terbaca di *wireshark*
4	*Login page-password-guessing attack*	Penyerang dapat mengirim *POST* data *username* dan *password* secara berulang-ulang.	*Webserver* menolak *POST* data login.php dengan status *403 Forbidden*
5	*Open Port*	*Port SMTP* dapat dieksplotasi dan mendapatkan *user email* aktif	Eksploitasi tidak berhasil mendapatkan *user email* aktif
6	*DNS Vulnerability*	Penyerang dapat melakukan *DNS Spoofing* dan *DNS Amplification* terhadap aplikasi	Serangan *DNS Spoofing* dan *DNS Amplification* dapat di block oleh aplikasi
7	*DDos Vulnerability*	Dapat dilakukan exploitasi sehingga aplikasi tidak dapat diakses	Aplikasi tetap dapat diakses dengan melakukan *filter* paket

4. Kesimpulan

Metode Penetration Testing Execution Standar dapat diterapkan pada *webserver* aplikasi pelaporan pajak daerah. Kategori keamanan aplikasi yang sebelumnya pada kategori *Medium* dengan 7 buah jenis kerentanan dapat diturunkan menjadi kategori *Low* dengan hanya 1 buah jenis kerentanan, sehingga keamanan webserver aplikasi pelaporan pajak daerah dapat ditingkatkan. Penelitian selanjutnya agar Metode Penetration Testing Execution Standar dapat dilakukan dalam melakukan analisa keamanan jaringan.

Daftar Rujukan

[1] Sugiartawan, P., Rustina, I. D. K. R., & Insani, R. W. S. (2018). E-Government Media Informasi Alat Kelengkapan Dewan Provinsi Bali dan Media Diskusi Berbasis Website. *Jurnal Sistem Informasi dan Komputer Terapan Indonesia (JSIKTI)*, *1*(2), 75-86. DOI: https://doi.org/10.33173/jsikti.17 .

[2] Badan Siber Sandi Negara. (2019). Laporan Tahunan Gov-CSIRT 2019 .

[3] Azis, H., & Fattah, F. (2019). Analisis Layanan Keamanan Sistem Kartu Transaksi Elektronik Menggunakan Metode Penetration Testing. *Ilkom Jurnal Ilmiah*, *11*(2), 167-174. DOI: https://doi.org/10.33096/ilkom.v11i2.447.167-174 .

[4] Karayiannis, C. (2019). Web-Based Projects that Rock the Class. *Build Fully-Functional Web Apps and Learn Through Doing. Apress*. DOI: https://doi.org/10.1007/978-1-4842-4463-0 .

[5] Goel, J. N., Mehtre, B. M. (2015). Vulnerability Assessment & Penetration Testing as Cyber Defence Technology. *Procedia Computer Science*, *57*, 710-715. DOI: https://doi.org/10.1016/j.procs.2015.07.458 .

[6] Chu, G., & Lisitsa, A. (2018). Penetration Testing for Internet of Things and Its Automation. *In 2018 IEEE 20th International Conference on High Performance Computing and Communications; IEEE 16th International Conference on Smart City; IEEE 4th International Conference on Data Science and Systems (HPCC/SmartCity/DSS)*, 1479-1484. DOI: https://doi.org/10.1109/HPCC/SmartCity/DSS.2018.00244 .

[7] Krasniqi, G., & Bejtullahu, V. (2018). Vulnerability Assessment & Penetration Testing: Case Study On Web Application Security. *UBT Knowlegde Center – Making Local Knowlegde Visible*. DOI: https://doi.org/10.33107/ubt-ic.2018.213 .

[8] Syarifudin, I. (2018). Pentesting dan Analisis Keamanan Web Paud Dikmas. *Zenodo.org*. DOI: https://doi.org/10.5281/zenodo.1211847 .

[9] Khera, Y., Kumar, D., Sujay, S., & Garg, N. (2019). Analysis and Impact of Vulnerability Assessment and Penetration Testing. *In Proceedings of the International Conference on Machine Learning, Big Data, Cloud and Parallel Computing: Trends, Prespectives and Prospects, COMITCon*, 525–530. DOI: https://doi.org/10.1109/COMITCon.2019.8862224 .

[10] Goutam, A., & Tiwari, V. (2019). Vulnerability Assessment and Penetration Testing to Enhance the Security of Web Application. *In 2019 4th International Conference on Information Systems and Computer Networks, ISCON*, 601–605. DOI: https://doi.org/10.1109/ISCON47742.2019.9036175 .

[11] Gupta, U., Raina, S., Verma, P., Singh, P., & Aggarwal, M. (2020). Web Penetration Testing. *International Journal for Research in Applied Science and Engineering Technology*, *8*(5), 56–60. DOI: https://doi.org/10.22214/ijraset.2020.5011 .

[12] Abu-Dabaseh, F., & Alshammari, E. (2018). Automated Penetration Testing : An Overview. *Academy and Industry Research Collaboration Center (AIRCC)*, 121–129. DOI: https://doi.org/10.5121/csit.2018.80610 .

[13] Ula, M. (2019). Evaluasi Kinerja Software Web Penetration Testing. *TECHSI - Jurnal Teknik Informatika*, *11*(3). DOI: https://doi.org/10.29103/techsi.v11i3.1996 .

[14] M, S. P., & Lobo, S. J. (2019). A Study on Advanced Cross Site Request Forgery Attacks and its Prevention. *Journal of Web Development and Web Designing*, *4*(2), 31–35. DOI: http://doi.org/10.5281/zenodo.3346240 .

[15] James, L., & D, D. E. (2020). Technique to Thwart Brute-Force Attack : A Survey. *International Journal of Scientific Research in Science, Engineering and Technology*, *7*(1), 235–237. DOI: https://doi.org/10.32628/ijsrset207139 .