Procedia of Engineering and Life Science Conference Vol. 1. No. 1 March 2021
Seminar Nasional & Call Paper Fakultas Sains dan Teknologi (SENASAINS 1st)
Universitas Muhammadiyah Sidoarjo

Implementation of Regular Expressions for CLI and WEB-based Backdoor Scanners

Implementasi Regular Expression Untuk Backdoor Scanner Berbasis CLI dan WEB

Nikko Enggaliano Pratama[1], Arif Senja Fitrani[2]
{nikko@umsida.ac.id[1], asfim@umsida.ac.id[2]}

Program Studi Informatika, Fakultas Sains dan Teknologi Universitas Muhammadiyah Sidoarjo

Abstract. *The author realizes that in this era, information systems are increasingly developing, there are so many system developers who can develop great and useful applications. But not many of these developers care about data security issues or information systems. The more an application that is running and is also being developed will have an impact along with the increasing number of cybercrime or hacktivists attacking the application or a server. The actors not only carried out attacks and then ignored the servers, many of them left a backdoor, to facilitate future actions. Many of the information system developers do not understand what a backdoor is and what a backdoor looks like, because many of the backdoor is developed with complicated techniques or with common syntax that the developer considers reasonable. Here the author developed a backdoor scanner application that can run on any OS, Windows or Unix with the same results, and can run in CLI (Command Line Interface) or in Web mode. Testing of the backdoor scanner application has been carried out by various elements ranging from government, private sector and the Open Source community who can still detect all tested backdoors with 100% accuracy even though the results issued are still False Positive. The backdoor tested is a backdoor that has been developed and has been in circulation before or is called a well-know backdoor.*

Keywords – *Security; injection; Backdoor; regular expression*

Abstrak. *Penulis menyadari pada saat era ini, sistem informasi semakin berkembang, sangat banyak sekali para pengembang sistem yang dapat mengembangkan aplikasi yang hebat dan bermanfaat. Tapi tidak banyak dari para pengembang tersebut yang peduli dengan masalah keamanan data ataupun sistem informasinya. Semakin banyak sebuah aplikasi yang berjalan dan juga dikembangkan menimbulkan dampak yang beriringan dengan semakin banyak juga tindak kejahatan dunia maya atau hacktivist yang menyerang aplikasi ataupun sebuah server. Para aktor tidak hanya melakukan serangan lalu mengabaikan servernya, banyak dari aktor tersebut menitipkan sebuah backdoor, untuk memudahkan aksi-aksi yang akan dilakukan di kemudian hari. Banyak dari para pengembang sistem infromasi tidak mengerti apa itu backdoor dan bagaimana bentuk backdoor tersebut, dikarenakan banyak dari backdoor dikembangkan dengan teknik yang rumit atau dengan syntax-syntax umum yang dianggap wajar oleh para pengembang. Di sini penulis menggembangkan sebuah aplikasi backdoor scanner yang dapat berjalan di OS apapun, baik windows ataupun unix dengan hasil yang sama, serta dapat berjalan secara CLI (Command Line Interface) atau dengan mode Web. Pengujian aplikasi backdoor scanner ini telah dilakukan oleh berbagai unsur mulai dari pemerintahan, swasta dan komunitas Open Source yang tetap dapat mendeteksi semua backdoor yang diuji dengan akurasi 100% meskipun dengan catatan hasil yang dikeluarkan masih False Positive. Backdoor yang diuji adalah backdoor yang telah dikembangkan dan telah beredar sebelumnya atau disebut well-know backdoor.*

Kata kunci - *Keamanan; Deteksi; Regular Expression*

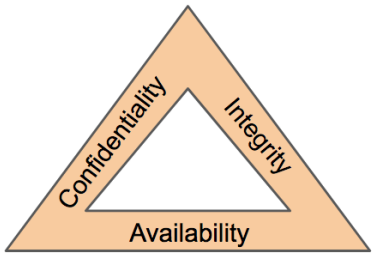
I. PENDAHULUAN

Semua bentuk infomasi menjadi kebutuhan sangat penting dan kebutuhan pokok pada era saat ini. Beberapa informasi juga dapat dikategorikan menjadi sesuatu hal yang sensitif yang berarti hanya orang-orang tertentu yang dapat menikmati ataupun mengakses informasi tersebut. Berdasarkan data yang diterbikan oleh Symantec (Majalah Cyber Security), terdapat pertumbuhan tindak hacktivist (Tindak Illegal Hacking) sebanyak 5% pertahun sejak tahun 2016. Indonesia juga menempati posisi 2 Dunia dalam sumber serangan tindak hacktivist menurut BSSN (Badan Siber Sandi Negara) dalam project Honeynet. Biasanya hal yang dilakukan oleh para aktor (Pelaku hacktivist) setelah sukses melakukan tahapan exploitasi adalah 'menanamkan / menyisipkan / mengunggah' Backdoor hal ini dilakukan agar para aktor tetap mendapatkan kontrol penuh atas target/server yang telah sukses di-exploitasi[5]. Backdoor yang dimaksud ini biasanya berupa sebuah file yang berisi kode untuk memberi akses ke pada aktor atas server tersebut. Pendeteksian Backdoor juga tidak dapat dilakukan dengan sederhana. Dengan inilah mengapa penulis menggembangkan Backdoor Scanner sebagai alat bantu pendeteksian backdoor itu dengan menggunakan Regular Expression[6].

II. METODE

A. System Security

System Security adalah sebuah aturan di mana developer diwajibakan untuk membuat sebuah sistem yang aman sesuai dengan prinsip dasar kemanan sebuah data dan sistemnya. System Security dibagi menjadi 3 bagian besar yaitu:



Gambar 1. CIA

1. Confidentiality yang artinya adalah kerahasiaan maksudnya adalah kerahasiaan data yang dimiliki oleh perusahaan atau instansi apapun yang berada pada Sistem yang berjalan maupun Basis Data dapat diakses oleh orang yang berwenang saja.
2. Integrity yang artinya adalah keaslian maksudanya adalah data yang dimiliki oleh sebuah instansi apapun tidak boleh dirubah oleh pihak manapun atauapun aplikasi apapun.
3. Availability yang artinya adalah ketersediaan maksudanya adalah sistem ataupun data yang sudah berbentuk informasi, jika pengguna ataupun user yang ingin mengakses sistem yang menyediakan data harus selalu tersedia dan tidak boleh ada cacat pemuatan data.

B. Server security

Server Security adalah konsep yang diterapkan pada sebuah Server, Server sendiri adalah tempat berjalannya sebuah aplikasi, berjalannya sebuah service atau layanan pendukung aplikasi yang sedang berjalan. Server juga dapat dikatakan sebagai tempat penyimpanan berbagai file ataupun data. Umumnya Server menggunakan Sistem Operasi berbasis Unix meskipun tidak jarang juga yang menggunakan Sitem Operasi Windows[11]. Sebanyak 32% Sistem Operasi Ubuntu (Sebuah Sistem Operasi berbasis Unix) menguasai Market Share atau pengguna aktif, dan sebanyak 19% menggunakan Sistem Operasi berbasis Unix (datanyze.com) Sebuah situs yang menyajikan Market Share data banyak Vendor Teknologi[7]. Di dalam server, developer meletakkan file aplikasi yang siap dijalankan. Dikarenakan sebuah server dapat menyimpan apapun, hal inilah yang dimanfaatkan para aktor untuk menitipkan / mengunggah / menyisipkan sebuah Backdoor. Cara mengamankan dari sisi server [1]. Pengamanan server dapat difokuskan di pembatasan hak akses setiap user yang artinya membatasi tiap aplikasi dapat melakukan aksi apa di dalam server itu sendiri, berfokuskan pada Access Permission pada setiap file atau folder yang ada di dalam server[8].

C. Backdoor

Backdoor atau yang biasa juga dikenal dengan istilah web shell merupakan salah satu kode jahat yang digunakan hacker untuk maintaining access sistem yang pernah dimasukinya [2]. Backdoor juga dapat diartikan secara umum sebagai jalan masuk yang sengaja dibuat oleh para aktor hacktivist setelah proses explotasinya selesai, untuk memudahkan aksi-aksi lain yang akan dilakukan di kemudian hari. Dalam proses penelitian ini penulis menghimpun berbagai Backdoor yang sudah beredar di Internet dan Backdoor yang dibuat dengan tools generate Backdoor[12]. Berikut nama-nama Backdoor yang dihimpun dari Internet dan Backdoor yang dibuat menggunakan Msfvenom Backdoor Generator:

1. WSO
2. IndoXploit
3. B374k
4. C99
5. Msfvenom Reverse Shell

D. Regular expression

Ekspresi regular (regular Expression) adalah sebuah urutan karakter yang dapat mencari sebuah pola. Pola pencarian kata atau teks memerlukan sebuah algoritma yang diatur dalam regular expression itu sendiri. Yang membedakan regex (regular expression) dengan string biasa yaitu metacharacters atau token di mana terdapat karakter-karakter khusus yang memiliki sebuah arti atau maksud tersendiri dalam pola pencarian, karakter-karakter ini tidak akan dicocokan secara literal dengan karakter itu sendiri, tapi mewakili sekelompok karakter lain atau pola khusus tertentu[9]. Reguler expression biasa digunakan untuk mencocokkan dan mencari pola dalam text atau kata, mulai dari pola sederhana hingga yang sangat kompleks. Terdapat beberapa contoh karakter-karakter regular expression yang digunakan menurut Michael Fitzgerald [3]. Reguler Expression sudah dapat diimplementasikan dalam Bahasa pemrogramman apapun. Alternatif Reguler Expression dapat dianalogikan seperti fungsi-fungsi atau API yang sudah disediakan oleh tiap bahasa pemrogramman. Contohnya pada PHP ada fungsi yaitu strcmp atau di dalam bahasa pemrogramman Python yaitu syntax ini[10]. Di bahasa pemrogramman Node JS menggunakan syntax localeCompare. Yang semuanya berfungsi sebagai pencocokan dua string namun string harus memperhatikan besar dan kecilnya (case-sensitif) yang akan menghasilkan output 0 dan 1. Namun pencocokan string dengan Reguler Expression jauh lebih ampuh. Selain untuk menguji kecocokan substring dan string, regular expression juga dapat dipakai untuk membelah dan mensubstitusi atau bahkan mengapus sebuah string [4]. Dikarenakan Regular Expression sudah bisa diimplementasikan disetiap Bahasa pemrogramman, berikut ini syntax-syntax atau API yang digunakan untuk menjalankan Reguler Expression sebagai berikut.

Tabel 1. Contoh Regular Expression API

No	Bahasa Pemrograman	Internal API
1.	PHP	preg_match preg_match_all
2.	Python	import re; re.findall import re; re.search
3.	NodeJS	Obj.search Obj.match

Rule Regular Expression

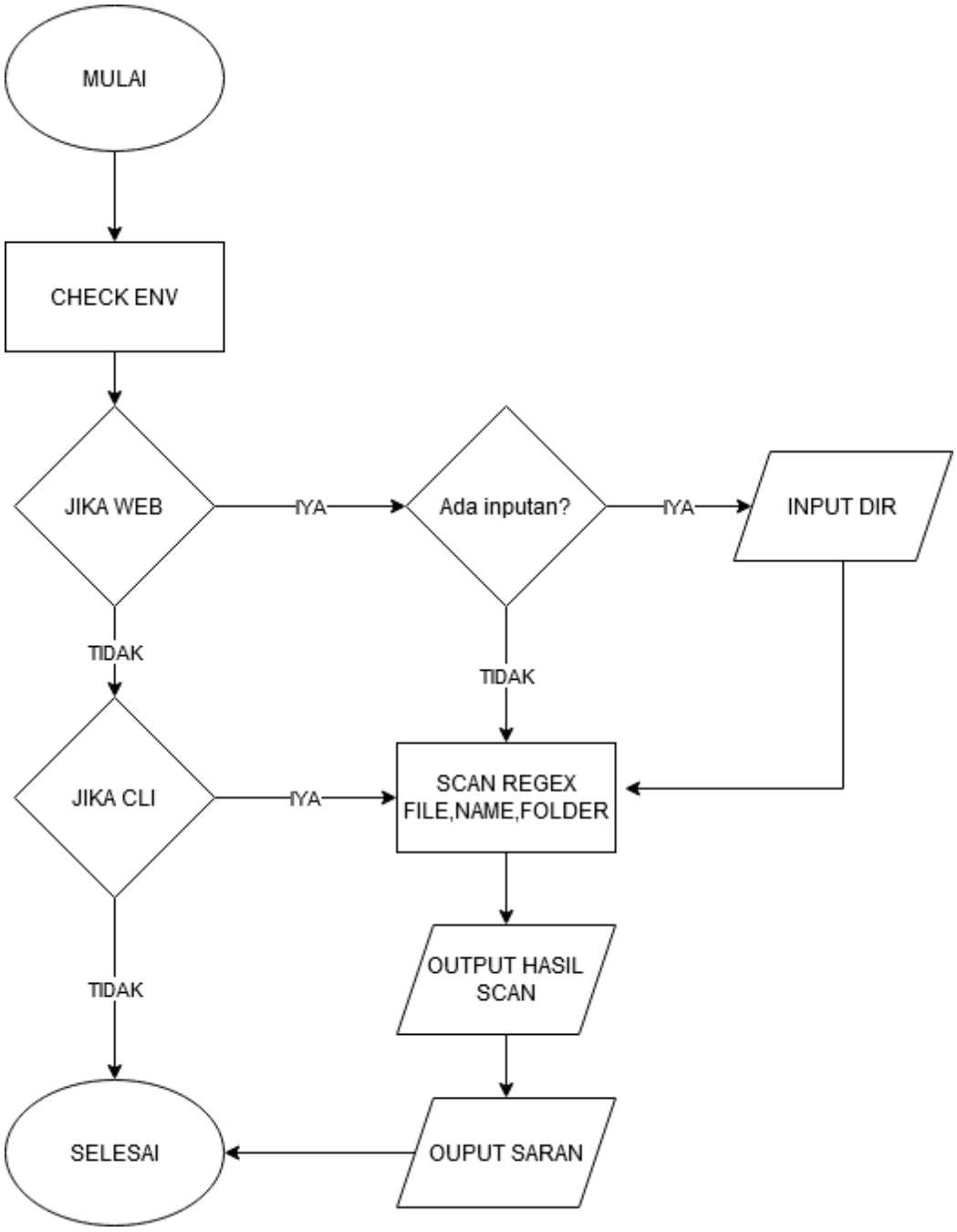
Tabel 2. Reguler Expression Rule

No	Regex	Keterangan
1.	/system/i	Difokuskan untuk mencari system call pada script
2.	/eval/mi	Difokuskan untuk mencari eval call pada semua Bahasa pemrograman
3.	/exec/mi	Mencari sebuah exec ataupun shell_exec pada sebuah file yang digunakan untuk system call

Regular Expression Rule di atas dibuat sesuai dengna karakteristik tiap-tiap backdoor yang telah tersebar di rana Internet. Hal yang paling penting adalah nama-nama dari tiap backdoor harus masuk dalam rule tersebut. Dan yang paling penting adalah eval yang mana sebagai jembatan utama sebuah backdoor.

III. HASIL DAN PEMBAHASAN

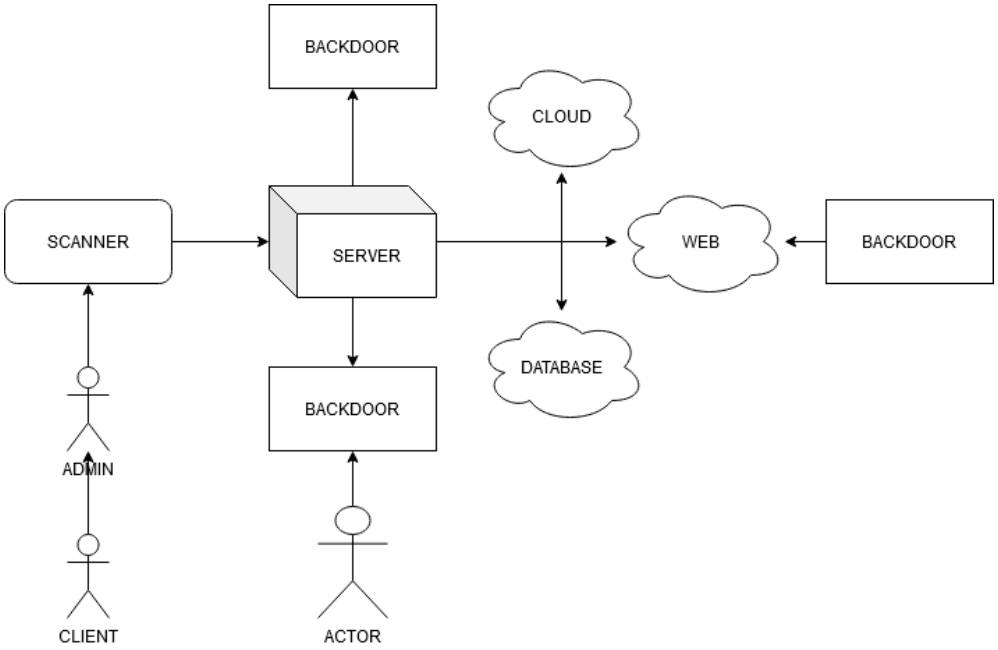
A. Flowchart



Gambar 2. Flowchart system

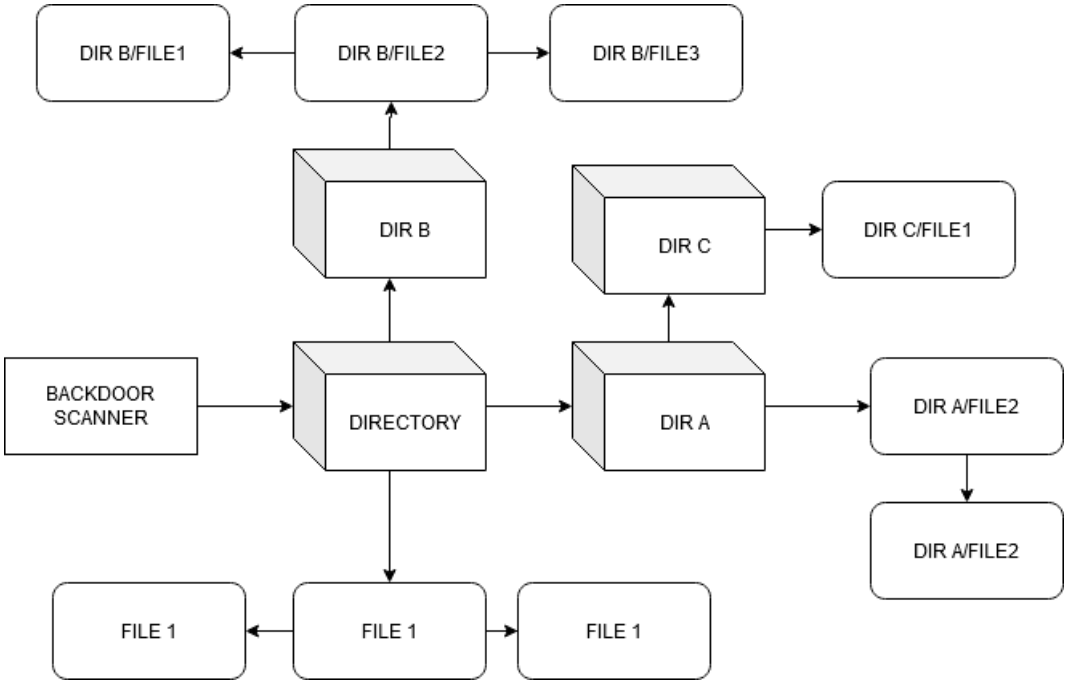
Pada gambar Flowchart yang dibuat penulis di atas. Menjelaskan bahwa akan ada pengecekan environtment atau tempat yang digunakan untuk menjalankan aplikasi scanner ini. Penentuan environtment atau tempat berjalannya aplikasi ditentukan dengan cara menggunakan fungsi atau API yang disediakan oleh PHP yaitu php_sapi_name(). Setelah dapat dibedakan program berjalan secara CLI atau Web maka program akan mulai melakukan pengecekan lagi jika aplikasi berjalan di browser, yaitu ada atau tidak sebuah inputan directory yang akan dilakukan scanning, jika tidak ada maka dilakukan scanning pada directory tempat ditaruhnya file. Jika ada inputan maka scanning berjalan pada direktori yang di tujuh. Lalu jika program berjalan secara CLI maka tidak ada inputan apapun, proses scanning

dilakukan pada direktori tempat file disimpan. Setelah proses scanning selesai maka hasil akan dikeluarkan oleh aplikasi baik di CLI ataupun WEB akan memiliki hasil output yang sama yang membedakan hanya bentuk tampilannya saja. Setelah output selesai ditampilkan, maka output saran akan mulai ditampilkan juga tidak ada perbedaan output saran hanya tampilannya saja yang berbeda antara CLI dan WEB. Setelah semua proses telah dilalui maka program akan berhenti.



Gambar 3. Alur System

Terlihat pada Gambar 3 atau gambar di atas. Menunjukan bahwa seseorang yang dapat menjalankan aplikasi scanner atau backdoor scanner adalah seorang admin dari pada server itu sendiri. Sebuah server umumnya memiliki lebih dari 2 aplikasi yang bernaung di sebuah server. Server juga memiliki banyak tempat atau directory penyimpanan. Seorang actor dapat saja menyisipkan pada level server itu sendiri ataupun pada directory kerja sebuah aplikasi yang telah ter-explotasi. Backdoor sendiri berjalan atau tersimpan pada sisi server, maka dari itu diperlukan seorang admin yang dapat menjalankan aplikasi Backdoor scanner ini. Sebuah Backdoor tidak hanya berupa sebuah file utuh dengan nama yang mirip, juga dapat disisipkan pada sebuah source code yang asli tanpa mempengaruhi jalannya sebuah aplikasi di dalam server. Berikut ini adalah gambaran bagaimana dan apa saja yang dilakukan oleh scanner di dalam sebuah server yang dilakukan scanning pada directory yang di tuju.



Gambar 4. Alur System

Terlihat pada Gambar 4 atau gambar di atas, scanner akan melakukan scanning pada directory yang dituju ataupun directory saat aplikasi ini dijalankan. Saat berjalan scanner akan mencari file yang ada pada directory inti terlebih dahulu. Saat udah memetahkan file-file pada directory dengan cara membaca tiap file dan mecocokan dengan syntax regular expression yang sudah ditentukan sesuai dengan karakteristik, maka scanner akan mulai masuk pada directory terdekat sesuai dengan urutan alfabet A-Z. Di dalam directory yang dituju setelah directory kerja scanner akan mulai memetahkan file-file di dalam directory yang dituju di dalam directory kerja. Jika di dalam directory itu masih ada folder lagi maka scanner akan terus mencari sampai tidak ada directory lagi di dalam directory yang sedang dikerjakan.

B. Pengujian Sistem

Langkah-langkah awal yang dilakukan dalam pengujian Backdoor Scanner ini adalah dengan mengumpulkan semua backdoor pada satu folder yang sama dengan backdoor scanner. Baik pada folder yang dijalankan oleh web server ataupun folder yang hanya akan dijalankan dengan menggunakan terminal / basis CLI cara-cara program membedakan CLI atau Web sebagai berikut ini.

Deteksi OS

Pertama penulis membuat satu konstanta bernama PHP_fOS yang mana PHP_OS berisi percabangan yang mana mengecek 3 Byte awal dari fungsi bawaan PHP_OS itu sendiri. Yang mana jika 3 Byte awal tersebut berisi kata ‘WIN’ maka sudah dipastikan, program sedang berjalan pada OS Windows. Jika kode di atas dijalankan pada OS yang berbeda maka akan menghasilkan output seperti yang di bawah ini, berikut ini screenshot jika program berjalan pada windows dan Linux.

Deteksi Platform

Deteksi Platform pada Backdoor Scanner berfungsi membedakan di mana program ini sedang berjalan, jika program berjalan pada CLI mode maka program tidak akan mengeluarkan output yang memiliki tag-tag HTML milik Browser. Cara mendeteksinya adalah penulis melakukan pengecekan terhadap php_sapi_name() yang mana jika fungsi php_sapi_name() tidak ada di dalam internal function maka kode di atas akan langsung menganggap bahwa program berjalan di CLI. Tapi php_sapi_name() sendiri memang mengembalikan string berupa CLI jika sedang berjalan pada shell / terminal.

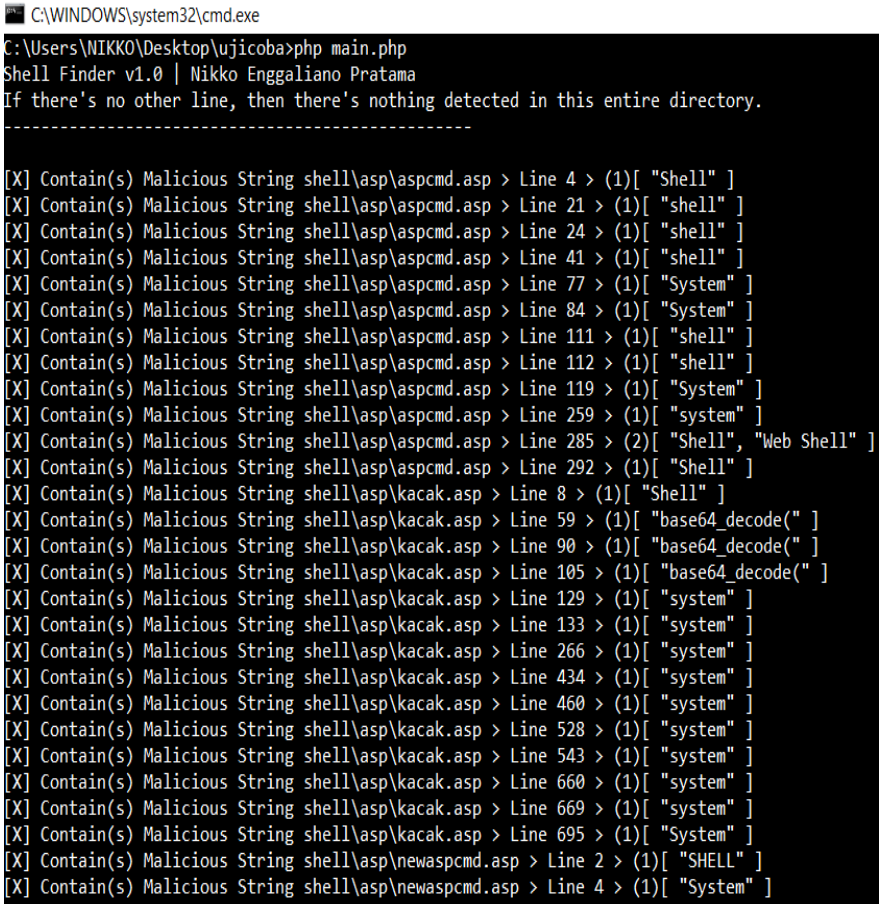
Line Breaking

Line Breaking berfungsi mendeteksi tiap-tiap line yang telah dibaca oleh program. Line yang terdeteksi berfungsi untuk memberitahu di line mana letak string yang berpontensi sebagai malicious object. Penulis membatasi hanya boleh file yang ukurannya di bawah 100Mb untuk dapat dilakukan pemecahan line. Pemecahan line dilakukan dengan

memecah new line atau \n tiap-tiap file, dengan begitu penulis dapat menghitung berapa banyak jumlah baris tiap filenya.

C. Pengujian Berbasis CLI

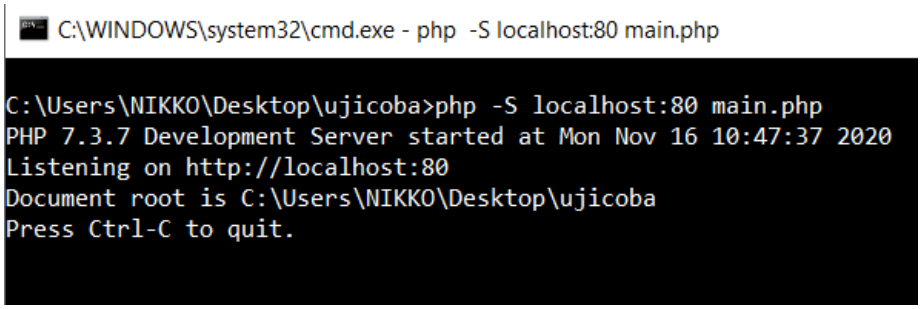
Pengujian ini memfokuskan aplikasi berjalan pada CLI yang mana CLI lebih banyak dan lebih mudah untuk dijalankan. Untuk menjalankan aplikasi di CLI dapat menjalankan dengan cara php main.php yang mana akan mengeluarkan *output* sebagai berikut ini.



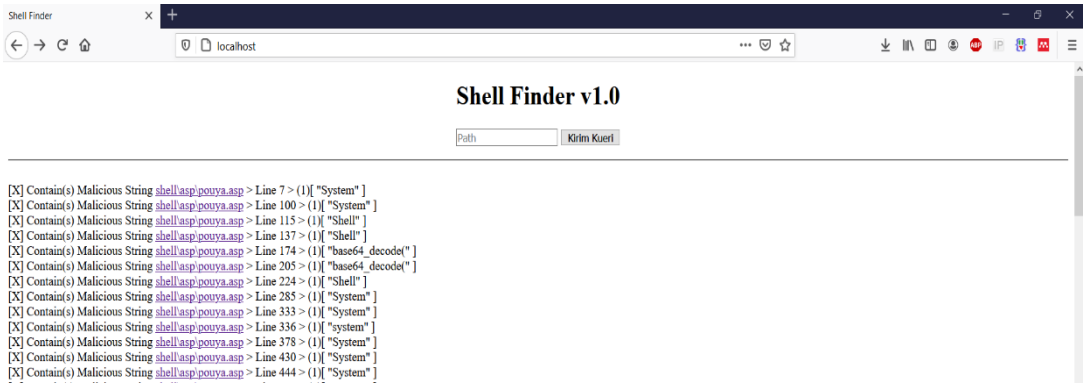
Gambar 5. CLI Result

D. Pengujian Berbasis Web

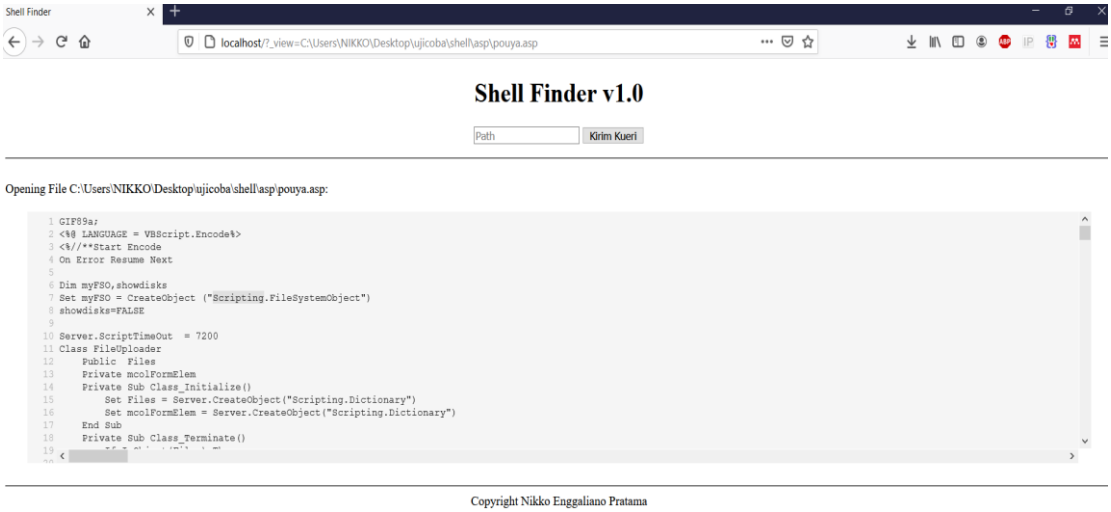
Pengujian Backdoor Scanner berbasis Web ini memfokuskan aplikasi berjalan di browser, yang mana akan ada tampilan yang akan menampilkan tiap-tiap file yang dicurigasi sebagai malicious script di dalam sebuah server. Karena aplikasi backdoor scanner dikembangkan dengan menggunakan Bahasa pemrograman PHP. Maka selama dalam device ter-install PHP maka menjalankan aplikasi dapat dengan cara **php -S localhost:80 main.php** yang mana tampilannya akan seperti berikut ini.



Gambar 6. Start Web via PHP



Gambar 7. After Scanning



Gambar 8. Source Viewer

E. Tabel Hasil

Setelah melakukan pengujian dan berbagai uji coba backdoor scanner pada mode CLI dan pada mode Website, penulis mendapatkan hasil sebagai berikut ini yang telah diringkas menjadi sebuah table. Yang mana isi dari table ada ringkasan dari pengujian yang dilakukan terhadap backdoor yang sudah tersebar di rana internet / well-know backdoor.

Tabel 3. Hasil Pengujian

NO	*Nama Backdoor*	*String Terdeteksi*	*Status*
1	C99 (php)	*System,exec*	Terdeteksi
2	0byt3m1n1 (php)	*Eval*	Terdeteksi
3	Ak47shell (php)	*Exec*	Terdeteksi
4	Indoxploit (php)	*Eval*	Terdeteksi
5	Pouya (asp)	System,exec	Terdeteksi
6	Aspcmd (asp)	*Shell, system*	Terdeteksi
7	Kacak (asp)	*Shell, system*	Terdeteksi

IV. KESIMPULAN

Akhir dari pengujian dan analisa yang telah dilaksanakan pada bab sebelumnya dapat disimpulkan sebagai berikut: Perangkat lunak/tools yang dapat mendeteksi sebuah Backdoor yang disisipkan pada sebuah aplikasi ataupun server telah berhasil dikembangkan dengan tingkat akurasi 100%. Dengan memanfaatkan perangkat lunak yang telah dikembangkan, proses security audit dapat dilakukan secara otomatis. Selain itu, perangkat lunak dapat memberikan rekomendasi dari hasil security audit. Salah satu proses deteksi adanya backdoor adalah dengan cara menganalisis pattern atau dengan mengumpulkan banyak behaviour dari berbagai backdoor yang ada. Dengan reguler expression dapat dibuat pattern yang fleksibel untuk mencari sebuah backdoor di dalam file pada sebuah directory.

REFERENSI

[1] Enggaliano, N., 2020. Basic Secure Server Nikko Enggaliano. [online] Nikkoenggaliano.github.io. Available at: <https://nikkoenggaliano.github.io/asic-Secure-Server/> [Accessed 3 November 2020].
[2] Sopaheluwakan, C. R., & Chandra, D. W. (2020). Anti-WebShell PHP Backdoor Scanner pada Linux Server. ILKOM Jurnal Ilmiah, 12(2), 143–153. https://doi.org/10.33096/ilkom.v12i2.596.143-153
[3] Yogi, I. R. (2019). Analisa Log Web Server untuk Mengetahui Pola Perilaku Pengunjung Website Menggunakan Teknik Regular Expressions. Jurnal Komputer dan Aplikasi, 122-123.
[4] Jhon Nicolas Siahaan, Y. M. (2013). Aplikasi SMS Gateway Di PT. Mercava Globe Sphere. Jurnal Teknologi Informatika, 4-5.
[5] Oktaviani.J. (2018). 済無 No Title No Title. Sereal Untuk, 51(1), 51.
[6] Symantec. (2019). Internet Security Threat Report Volume 21, February 2019. Network Security, 21(February). https://doi.org/10.1016/S1353-4858(05)00194-7
[7] Russell L. Jones & Abhinav Rastogi (2004) Secure Coding: Building Security into the Software Development Life Cycle. Information Systems Security, 13:5, 29-39, DOI: 10.1201/1086/44797.13.5.20041101/84907.5
[8] Gandhi Pranoto, R. D. (2016). Rancang Bangun Aplikasi Terpadu. Jurnal Sistem dan Teknologi Informasi, 2.
[9] Novianty, C. (2017). Review Konsep Responsive Design dengan Framework. InfoTekJar (Jurnal Nasional Informatika dan Teknologi Jaringan) , 42.
[10] Syaifudin Ramadhani, Urifatun Anis, Siti Tazkiyatul Masruro. (2013). Rancang Bangun Sistem Informasi Geografis Layanan Kesehatan Di. Jurnal Teknika, 480.
[11] ElisaUsada, Y. R. (2012). Rancang Bangun Sistem Informasi. Jurnal Infotel, 43.
[12] Fauzan Masykur, F. P. (2016). Aplikasi Rumah Pintar (Smart Home) Pengendali Peralatan Elektronik Rumah Tangga Berbasis Web. Jurnal Sains, 95.