SKRIPSI

PEMBUATAN APLIKASI PENDETEKSI WEBSHELL BERDASARKAN HTTP ACCESS LOG

Friska Christiana

NPM: 2017730030

PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS TEKNOLOGI INFORMASI DAN SAINS
UNIVERSITAS KATOLIK PARAHYANGAN
2022

UNDERGRADUATE THESIS

WEBSHELL DETECTION APPLICATION DEVELOPMENT BASED ON HTTP ACCESS LOG

Friska Christiana

NPM: 2017730030

DEPARTMENT OF INFORMATICS
FACULTY OF INFORMATION TECHNOLOGY AND SCIENCES
PARAHYANGAN CATHOLIC UNIVERSITY
2022

LEMBAR PENGESAHAN

PEMBUATAN APLIKASI PENDETEKSI WEBSHELL BERDASARKAN HTTP ACCESS LOG

Friska Christiana

NPM: 2017730030

Bandung, 14 Januari 2022

Menyetujui,

Pembimbing

Digitally signed by Chandra Wijaya

Chandra Wijaya, M.T.

Ketua Tim Penguji

Digitally signed by Elisati Hulu

Elisati Hulu, M.T.

Anggota Tim Penguji

Digitally signed by Vania Natali

Vania Natali, M.T.

Mengetahui,

Ketua Program Studi

Digitally signed by Mariskha Tri Adithia

Mariskha Tri Adithia, P.D.Eng

PERNYATAAN

Dengan ini saya yang bertandatangan di bawah ini menyatakan bahwa skripsi dengan judul:

PEMBUATAN APLIKASI PENDETEKSI WEBSHELL BERDASARKAN HTTP ACCESS LOG

adalah benar-benar karya saya sendiri, dan saya tidak melakukan penjiplakan atau pengutipan dengan cara-cara yang tidak sesuai dengan etika keilmuan yang berlaku dalam masyarakat keilmuan.

Atas pernyataan ini, saya siap menanggung segala risiko dan sanksi yang dijatuhkan kepada saya, apabila di kemudian hari ditemukan adanya pelanggaran terhadap etika keilmuan dalam karya saya, atau jika ada tuntutan formal atau non-formal dari pihak lain berkaitan dengan keaslian karya saya ini.

Dinyatakan di Bandung,
Tanggal 14 Januari 2022

Friska Christiana
NPM: 2017730030

ABSTRAK

Situs web merupakan sebuah layanan aplikasi yang sudah banyak digunakan di masyarakat luas. Sudah hampir setiap orang menggunakan situs web untuk kebutuhan sehari-hari, seperti komunikasi, belajar, bekerja, serta melihat berita. Konten-konten pada situs web dibuat dalam sebuah *file* yang disimpan dalam server web. Salah satu server web yang paling umum digunakan adalah *Apache*. Dengan adanya layanan server web, situs web dapat menampilkan gambar, tulisan, mengakses halaman web lain, dan lain-lain. Salah satu protokol yang digunakan untuk mengakses halaman web dari server web adalah HTTP. Setiap aktivitas di situs web akan tercatat di *file log* pada server web yang disebut *file access log*. Semakin banyaknya situs web yang beredar, semakin besar peluang situs web dieksploitasi. Tujuan penyerang melakukan eksploitasi terhadap situs web adalah untuk mengambil data pribadi pengguna, mengubah hak akses pengguna, dan sebagainya. Salah satu cara agar penyerang dapat mengeksploitasi server web adalah dengan mengunggah skrip *webshell* melalui halaman di situs web.

Pada skripsi ini, dibuat sebuah aplikasi untuk mendeteksi adanya *webshell*. Terdapat lima jenis *webshell* yang akan dianalisis dalam skripsi, yaitu *webshell Webadmin, C99, B374K, R57,* dan *WSO*. Aplikasi digunakan untuk mendeteksi aktivitas *webshell* dan menampilkan laporan aktivitas dalam bentuk web. Pendeteksian ini digunakan dengan metode pembelajaran mesin, yaitu *Decision Tree*. Sebelum diolah dengan metode *Decision Tree*, *file log* akan dipreproses menjadi beberapa bagian dan empat atribut untuk fitur dalam *Decision Tree*. Agar dapat mendeteksi *webshell* dari suatu *log*, dibutuhkan data latih untuk pemberian label dan mengetahui karakteristik dari masing-masing *webshell* sebelum menguji data uji.

Pengujian dilakukan dengan memeriksa kesesuaian hasil prediksi aktivitas *webshell* menggunakan model pembelajaran mesin dengan hasil preproses. Aplikasi yang dibangun dapat mendeteksi aktivitas *webshell* beserta jenisnya yang terjadi berdasarkan *file log* yang diuji. Perangkat lunak berhasil mendeteksi adanya aktivitas *webshell Webadmin, C99,* dan *B374K* dari *log* DVWA dan *Student Portal* dengan rata-rata ketepatan 96%. Perangkat lunak tidak berhasil mendeteksi aktivitas *webshell R57* dan *WSO* karena kedua *webshell* tersebut memiliki karakter yang sama dan tidak bisa diidentifikasi.

Kata-kata kunci: Server web, *Apache*, *file access log*, *webshell*, pembelajaran mesin

ABSTRACT

The website is an application service that has been widely used in the wider community. Almost everyone uses the website for their daily needs, such as communicating, studying, working, and viewing the news. The content on the website is created in a file that is stored on a web server. One of the most commonly used web server is Apache. With a web server service, websites can display images, text, access other web pages, and so on. One of the protocols used ti access web pages from a web server is HTTP. Every activity on the website will be recorded in a log file on the web server which is called an access log file. The more website ciculating, the greater the chances of the website being exploited. The purpose of an attacker to exploit a website is to retrieve user's personal data, change user access rights, and so on. One way that attackers can exploit a web server is by uploading a webshell script through a page on a website.

In this thesis, an application is made to detect the presence of a webshell. There are five types of webshells that will be analyzed in this thesis, namely Webadmin, C99, B374K, R57, and WSO webshells. The application is used to detect webshell activity and display activity reports in web form. This detection is used by machine learning method, namely Decision Tree. Before processing with the Decision Tree method, the log file will be preprocessed into several parts and four attributes for the features in the Decision Tree. In order to detect a webshell from a log, training data is needed for labeling and knowing the characteristics of each webshell before testing the test data.

The test is carried out by checking the suitability of the predicted results of webshell activity using machine learning models with preprocessed results. The application that is built can detect webshell activities and their types that occur based on the log files under test. The application successfully detects Webadmin, C99, and B374K webshell activity from DVWA and Student Portal logs with an average accuracy of 96%. The application was unable to detect the activity of the R57 and WSO webshells because the two webshells have the same character and cannot be identified.

Keywords: Web server, Apache, access log files, webshell, machine learning

KATA PENGANTAR

Puji syukur penulis panjatkan kepada Tuhan Yang Maha Esa karena atas karunia-Nya, penulis dapat menyelesaikan penyusunan skripsi yang berjudul "Pembuatan Aplikasi Pendeteksi Webshell berdasarkan HTTP Access Log". Selama penyusunan skripsi ini, penulis menghadapi banyak kendala dan berbagai masalah. Penulis menyadari bahwa penyusunan skripsi ini juga tidak terlepas dari bantuan berbagai pihak, baik langsung maupun tidak langsung. Secara khusus, penulis ingin berterima kasih kepada:

1. Tuhan Yang Maha Esa atas Berkat dan Rahmat-Nya.
2. Orang tua beserta kakak laki-laki penulis yang selalu memberikan dukungan kepada penulis baik berupa doa atau dukungan mental serta materil.
3. Bapak Chandra Wijaya, M.T. selaku dosen pembimbing yang telah membimbing penulis dan memberikan dukungan maupun bantuan kepada penulis dalam proses penyusunan skripsi ini.
4. Bapak Elisati Hulu, M.T. dan Ibu Vania Natali, M.T. selaku dosen penguji yang telah memberikan kritik dan saran yang membangun sehingga penelitian ini menjadi lebih baik.
5. Michael Liondy sebagai pacar penulis yang selalu memberikan dukungan secara mental, motivasi, memberikan asupan yang penulis sukai sehingga penulis bisa terus semangat dalam penyusunan skripsi ini.
6. Sahabat perempuan informatika angkatan 17 seperti Cristine Artanty, Linna Trisnawati, Melody Victorian, Denise Stevani, Hendrika Valeska, dan Rachel Florencia yang sudah memberikan dukungan, bantuan, dan semangat kepada penulis.
7. Sahabat laki-laki informatika angkatan 17 seperti Juan Nandrissa dan Enrico Wibawa yang selalu ikut grup cewek-cewek. Kemudian Henrico Leodra, Kevin Draven, Kelvin Dravin, Richard Morris, Yovananta Jong, Jodi Tanato, Indra Permana dan Harry Senjaya yang menjadi dekat karena salah satu mata kuliah dan telah memberikan dukungan kepada penulis.
8. *Youtube* yang telah memberikan musik untuk menemani penulis dalam penyusunan skripsi.

Penulis menyadari bahwa penelitian ini masih jauh dari kata sempurna. Oleh karena itu, penulis memohon maaf jika terdapat kesalahan. Penulis juga mengharapkan kritik dan saran yang membangun untuk menyempurnakan penelitian ini. Semoga penelitian ini dapat memberi informasi yang bermanfaat dan menjadi inspirasi untuk penelitian-penelitian berikutnya.

Bandung, Januari 2022

Penulis

DAFTAR ISI

DAFTAR GAMBAR

BAB 1

PENDAHULUAN

1.1 Latar Belakang

Situs web merupakan layanan aplikasi yang sudah menarik banyak perhatian masyarakat umum. Situs web terus berkembang sehingga orang dapat saling berkomunikasi di dalam maupun di luar lingkungan kerja mereka. Halaman-halaman web berisi banyak informasi dan saling berhubungan yang dapat diakses oleh browser melalui suatu alamat URL. Saat ini, situs web sudah banyak digunakan untuk banyak keperluan seperti komunikasi, membaca berita, menonton, bermain, berjualan, dan menampilkan profil perusahaan. Dengan adanya situs web, orang-orang dapat dengan mudah membuat dan memperoleh informasi bahkan dari tempat yang jauh sekalipun.

Situs web memiliki konten tersendiri berdasarkan tujuannya. Konten pada situs web dibuat dalam sebuah *file* yang disimpan dalam sebuah *web server*. *Web server* memberikan layanan yang diminta oleh *web browser*, seperti halaman web, gambar, *file*, dan lain-lain. *Web browser* atau browser merupakan sebuah perangkat lunak yang menampilkan situs web dan membantu pengguna untuk mengakses halaman-halaman lainnya. Ketika pengguna meminta halaman web pada suatu situs web, *web browser* akan menerima konten dari *web server* dan menampilkannya kepada pengguna. Terdapat dua buah protokol yang dapat digunakan untuk mengakses halaman web dari *web server*, yaitu HTTP (*Hypertext Transfer Protocol*) dan HTTPS (*HyperText Transfer Protocol Secure*) [5]. Sampai saat ini, tidak semua koneksi situs web sudah menggunakan HTTPS. Saat pengguna meminta halaman web, browser mengirimkan pesan permintaan HTTP untuk konten di halaman tersebut ke server. Server menerima permintaan tersebut dan merespon dengan pesan respon HTTP yang berisi konten halaman tersebut. Dengan demikian, pengguna dapat melihat suatu konten dari halaman web yang diaksesnya.

```
1 192.168.100.252 - - [31/Jul/2014:09:30:23 +0700] "GET /p4p/report_detail_edit.php?name_t=3 HTTP/1.1" 200 8013
2 110.168.229.112 - - [31/Jul/2014:09:30:25 +0700] "GET /p4p/detail.php HTTP/1.1" 200 5433
3 110.168.229.112 - - [31/Jul/2014:09:30:25 +0700] "GET /p4p/jquery/jquery.calendars.persian.js HTTP/1.1" 404 324
```

Gambar 1.1: Contoh *Common Log Format* pada *access log* [1]

```
185.10.104.132 - - [01/Feb/2015:00:00:02 -0800] "GET
/datasetlist?ids=Collections:Platform:AccessType:Measurement:Availability&values=CCMP:AQUA:OPEN:
Ocean%20Winds:HISTORICAL&view=list HTTP/1.1" 200 85283 "-" "Mozilla/5.0 (Windows NT 6.1;
WOW64; rv:22.0) Gecko/20100101 Firefox/22.0“

185.10.104.196 - - [01/Feb/2015:00:00:08 -0800] "GET
/datasetlist?ids=Collections:TimeSpan:Platform:GridSpatialResolution:Availability&values=CCMP:[1826%
20TO%20*]:DMSP-F14:0.25:HISTORICAL&view=list HTTP/1.1" 200 84531 "-" "Mozilla/5.0 (Windows NT
6.1; WOW64; rv:22.0) Gecko/20100101 Firefox/22.0"
```

Gambar 1.2: Contoh *Combined Log Format* pada *access log* [2]

Ketika setiap pengunjung yang mengirimkan permintaan atau mengakses aplikasi berbasis web, permintaan tersebut akan dicatat pada suatu *file* yang dinamakan *log* dalam hal ini yang terdapat pada *web server* [6]. Terdapat beberapa cara penulisan *access log*, seperti penulisan biasa atau *Common Log Format* pada Gambar 1.1 dan *Combined Log Format* pada Gambar 1.2. Kedua cara penulisan tersebut hampir sama, hanya berbeda pada bagian referer dan browser klien pada *Combined Log Format*.

Data pengunjung yang ada pada *log web server* akan sangat bermanfaat apabila nantinya terdapat suatu permasalahan yang terjadi di *web server*. Seperti misalnya kasus peretasan aplikasi website. Data-data seorang peretas dapat diketahui dengan memeriksa setiap catatan yang tersimpan pada *log* secara rinci. Salah satu cara agar data peretas dapat diketahui adalah dengan melihat dari alamat IP yang dipakai untuk mengakses *web server*. Karena catatan yang tersimpan dalam *log* sangat banyak, akan menjadi tidak efisien apabila metode pencarian data diperiksa satu per satu dari banyaknya data yang tersimpan.

Salah satu cara agar aktor tetap memiliki akses terhadap server adalah dengan menanamkan *webshell*. *Webshell* merupakan sebuah *script* yang dapat diunggah ke dalam server web. *Webshell* digunakan agar seorang aktor dapat menjalankan perintah-perintah pada server melalui *script* yang dipasang di *web server*. *Script* tersebut biasanya dipasang pada server sebuah *website* yang sudah berhasil diretas atau sudah memanfaatkan *Remote File Inclusion* (RFI) atau *Local File Inclusion* (LFI) *vulnerability* dari *web server*. RFI merupakan celah keamanan yang memungkinkan penyerang untuk menyisipkan *file* berbahaya dari luar server dan mengeksekusinya [7]. LFI merupakan celah keamanan yang memungkinkan penyerang untuk bisa membaca dan melihat *file-file* yang ada di server, termasuk *file* yang sensitif. Terdapat beberapa contoh jenis *webshell* yang umum digunakan oleh kalangan peretas seperti *C99, WSO, Webadmin, B374K* dan *R57*.

Saat suatu *web server* sudah ditanamkan suatu *webshell*, seorang aktor dapat melihat seluruh aktivitas di *web server* yang tercatat pada *log file* dengan nama *access.log* pada *web server* tersebut. Seorang aktor tidak hanya dapat melihat *log file* pada *web server*, melainkan data-data lain yang disimpan pada *web server* tersebut. Data-data ini dapat digunakan oleh aktor untuk kepentingan pribadi. Oleh karena itu, keberadaan *webshell* pada suatu server sangat berbahaya bagi server tersebut, karena aktor dapat berkuasa penuh atas server tersebut tanpa terdeteksi.

Webshell dapat dipantau dengan beberapa teknik. Salah satunya adalah dengan melakukan pemantauan berdasarkan *log file* pada *web server*. Banyak aspek yang dapat diperhatikan saat memantau terhadap *log file* tersebut. Pertama, pemantauan berdasarkan *IP* klien yang dicurigai sebagai aktor. Kedua, pemantauan agen user. Ketiga, pemantauan *referrer* dari akses yang berhasil. Biasanya, akses terhadap *webshell* tidak memiliki *referrer* karena aktor mengetahui langsung alamat *webshell*. Keempat, URL yang sedikit atau jarang diakses. Terakhir, pemantauan terhadap karakteristik klien. Karakteristik klien dapat dideteksi berdasarkan apa saja alamat yang diakses oleh klien tersebut.

Terdapat beberapa algoritma pembelajaran mesin yang dapat digunakan untuk mendeteksi adanya aktivitas *webshell*. Contoh beberapa algoritma tersebut adalah *Decision Tree, Naïve Bayes*, dan *K-Nearest Neighbor*. Ketiga algoritma tersebut termasuk dalam *supervised learning*, yang artinya algoritma ini mencari dan membuat hipotesis dengan data latih [8]. Hipotesis awal dilakukan terhadap data yang diberi label untuk memprediksi data baru yang belum diberi label.

Pada skripsi ini, akan dibangun sebuah sistem yang dapat mendeteksi *webshell* pada sebuah *website*. Pendeteksian *webshell* dapat dilakukan berdasarkan HTTP *access log* yang berada pada *log file* di *web server*. Pendeteksian adanya *webshell* pada suatu sistem akan dilakukan dengan algoritma pembelajaran mesin. Pada penelitian ini akan dilakukan pengujian untuk beberapa jenis *webshell*, beberapa masukkan *log file*, dan algoritma yang digunakan.

1.2 Rumusan Masalah

Berdasarkan latar belakang yang sudah dijelaskan, didapat beberapa rumusan masalah seperti berikut:

- Bagaimana struktur direktori pada *web server*?
- Bagaimana struktur *log file* dari *web server*?
- Bagaimana karakteristik dari masing-masing *webshell*?
- Metode apa yang digunakan pada perangkat lunak untuk mendeteksi *webshell* pada *web server*?
- Bagaimana memanfaatkan pembelajaran mesin untuk mendeteksi *webshell* pada *web server*?
- Bagaimana membangun perangkat lunak untuk mendeteksi *webshell*?

1.3 Tujuan

Dengan masalah yang sudah dirumuskan, berikut adalah tujuan dari skripsi ini, antara lain:

- Mengetahui struktur direktori pada *web server*
- Mengetahui struktur *log file* pada *web server*
- Mengetahui karakteristik dari masing-masing *webshell*
- Mengetahui metode pembelajaran mesin yang cocok pada perangkat lunak untuk mendeteksi *webshell* pada *web server*.
- Mengetahui cara memanfaatkan pembelajaran mesin untuk mendeteksi adanya *webshell* yang ditanam dalam sebuah *web server* berdasarkan masukkan *log file* dari *web server*.
- Membangun perangkat lunak untuk mendeteksi *webshell*

1.4 Batasan Masalah

Berdasarkan latar belakang, rumusan masalah dan tujuan penelitian yang sudah dijabarkan sebelumnya, batasan masalah yang digunakan dalam skripsi ini meliputi:

- *Log* yang akan dianalisis adalah *log* dengan format *Combined Log Format*
- *Web server* yang digunakan hanya menggunakan server *apache*
- Sistem operasi yang digunakan adalah sistem operasi *linux*
- *Webshell* yang digunakan dalam skripsi ini adalah *webshell C99, WSO, Webadmin, B374K* dan *R57*

1.5 Metodologi

Berikut merupakan langkah-langkah yang dilakukan pada penelitian ini:

- Melakukan studi literatur mengenai struktur direktori pada *web server*
- Melakukan studi literatur mengenai struktur *log file* yaitu *access log* dari *web server*
- Melakukan studi literatur mengenai jenis-jenis *webshell* seperti *C99, WSO, Webadmin, B374K* dan *R57*
- Mencoba dan menjalankan jenis-jenis *webshell*
- Menganalisis karakteristik pada masing-masing *web shell*
- Menganalisis perangkat lunak pendeteksi *webshell*
- Menganalisis algoritma untuk membangun perangkat lunak pendeteksi *webshell*
- Merancang perangkat lunak pendeteksian *webshell*
- Membangun perangkat lunak pendeteksi *webshell* menggunakan algoritma pembelajaran mesin
- Uji coba dengan beberapa masukan *log file* dan algoritma yang telah dibuat
- Menganalisis hasil uji coba dari perangkat lunak yang dibuat

1.6 Sistematika Pembahasan

Penulisan penelitian ini dibangun dalam enam bab dengan sistematika dan penjelasan sebagai berikut:

1. Bab 1. Pendahuluan
   Bab 1 berisi latar belakang, rumusan masalah, tujuan, metode penelitian, dan sistematika pembahasan yang digunakan untuk menyusun skripsi ini. Latar belakang menjelaskan masalah-masalah untuk mendeteksi adanya *webshell* pada *web server*. Pada bagian rumusan masalah menjelaskan masalah-masalah yang ingin diselesaikan dalam penelitian ini untuk mencapai tujuan penelitian. Pada bagian metodologi merupakan langkah-langkah penelitian yang dilakukan untuk mencapai tujuan penelitian. Pada bagian sistematika pembahasan merupakan gambaran singkat mengenai isi setiap bab pada buku skripsi ini.
2. Bab 2. Landasan Teori
   Bab 2 berisi tentang teori sebagai landasan utama dalam skripsi ini. Hal yang akan dibahas adalah *log* secara umum dan mengenai *access log* secara spesifik seperti format dan arti dari *log* tersebut. Selanjutnya akan dibahas secara umum mengenai jenis-jenis *webshell* untuk kebutuhan skripsi ini. Selanjutnya menjelaskan konsep dasar dari pembelajaran mesin dan beberapa contohnya yang akan digunakan dalam skripsi ini. Contoh yang akan dijelaskan antara lain *Decision Tree, Naïve Bayes*, dan *K-Nearest Neighbor*. Selanjutnya akan melakukan evaluasi dari model pembelajaran mesin yang dibuat.
3. Bab 3. Analisis
   Bab 3 berisi hasil analisis berdasarkan landasan teori yang digunakan. Hasil analisis tersebut antara lain untuk memilih model pembelajaran mesin yang digunakan untuk mencapai target yang diinginkan, perbedaan karakteristik masing-masing jenis *webshell*, serta analisis *dataset* yang digunakan.
4. Bab 4. Perancangan
   Bab 4 berisi penjelasan dari perancangan yang merupakan hasil dari analisis masalah di Bab 3. Pada bagian ini akan dijelaskan secara spesifik mengenai masukan dan keluarannya, bagaimana rancangan fungsi-fungsi dari programnya, serta tampilan antarmuka yang akan dibangun. Akan dijelaskan juga mengenai alur program bagaimana data *log* yang diolah dapat diproses untuk mencapai tujuan dari skripsi ini.
5. Bab 5. Pengujian dan Eksperimen
   Bab 5 berisi tentang pengujian dari perangkat lunak dalam mendeteksi *webshell* dengan berbagai macam skenario yang sudah dibuat. Akan dijelaskan hasil dari pengujian dari beberapa eksperimen yang dikerjakan dan juga hasil kesimpulan analisis dari eksperimen ini. Pada akhir dari bab ini akan dijelaskan hasil analisis dari keseluruhan pengujian dan eksperimen yang digunakan dari berbagai macam skenario yang ada.
6. Bab 6. Kesimpulan dan Saran
   Bab 6 akan berisi tentang kesimpulan dari hasil analisis secara keseluruhan saat melakukan pembangunan perangkat lunak ini. Pada bab ini juga akan berisi tentang saran untuk peneliti selanjutnya agar dapat mengembangkan perangkat lunak ini kedepannya.