

IMPLEMENTASI METODE PROTEKSI SITUS WEB DARI WEB SCRAPING

IMPLEMENTATION METHOD TO PROTECT WEBSITE FROM WEB SCRAPING

Dalvizar Kafilham Ristijana¹, Burhanuddin Dirgantoro², Muhammad Faris Ruriawan³

^{1,2,3} Universitas Telkom, Bandung

¹dalvizarkafilham@student.telkomuniversity.ac.id, ²burhanuddin@telkomuniversity.ac.id,
³muhammadfaris@telkomuniversity.ac.id

Abstrak

Web Scraping adalah proses pengambilan sebuah dokumen-semi terstruktur dari internet, umumnya berupa halaman-halaman Web dalam bahasa markup seperti HTML (Hypertext Markup Language) atau XHTML (Extensible HyperText Markup Language), data tersebut dianalisis untuk diambil data tertentu dari web tersebut untuk digunakan bagi kepentingan lain. Banyaknya pelaku web scraping dikarenakan kebutuhan informasi yang semakin tinggi dan keinginan untuk memperoleh informasi dengan cepat membuat banyak orang yang ingin menggunakan dan mempelajari teknik web scraping untuk dapat meraup informasi dari sebuah situs web penyedia informasi.

Penelitian ini bertujuan untuk mengumpulkan seluruh metode yang dapat digunakan untuk memproteksi data pada situs web dari teknik scraping data dan bot dan juga melakukan test secara langsung dari bot untuk pembuktian metode. Penelitian akan dilakukan pada sebuah situs web buatan tim penulis yang akan diimplementasikan beberapa metode untuk proteksi situs web seperti javascript checker, blokir inspect element, CAPTCHA, manajemen fitur javascript, manajemen ip session, manajemen trafik data, dan blokir IP yang mencurigakan.

Parameter yang akan dijadikan sebagai tolak ukur untuk tingkat keamanan yang baik atau tidaknya adalah bagaimana metode tersebut dapat menghalangi sebuah bot untuk dapat masuk kedalam sistem situs web penyedia informasi yang akan dibuat oleh penulis yang kemudian akan di lakukan testing menggunakan bot scraping.

Kata kunci : *Web Scraping, How to stop web scraping*

Abstract

Web Scraping is the process of retrieving a semi-structured document from the internet, generally in the form of Web pages in a markup language such as HTML (Hypertext Markup Language) or XHTML (Extensible HyperText Markup Language), the data is analyzed to retrieve certain data from the web for use. for other purposes. The number of web scraping actors due to the higher information needs and the desire to obtain information quickly makes many people who want to use and learn web scraping techniques to be able to collect information from an information provider website.

This study aims to collect all methods that can be used to protect data on websites from data scraping techniques and bots and also conduct direct tests from bots to prove the method. The research will be conducted on a website made by the author's team which will implement several methods for website protection such as javascript checker, inspect element block, CAPTCHA, javascript feature management, ip session management, data traffic management, and blocking suspicious IPs.

The parameter that will be used as a benchmark for the level of security that is good or not is how this method can prevent a bot from being able to enter the information provider website system that will be created by the author which will then be tested using bot scraping.

Keywords: Kata kunci : *Web Scraping, How to stop web scraping*

1. Pendahuluan

Meningkatnya kebutuhan akan informasi mendorong manusia untuk mengembangkan teknologi-teknologi baru agar pengolahan data dan informasi dapat dilakukan dengan mudah dan cepat. Salah satu teknologi yang sedang berkembang dengan pesat saat ini adalah teknologi informasi komputer[1]. Untuk mengolah data dan informasi tersebut dibutuhkan yang namanya teknologi internet, dengan adanya internet akan mempermudah seseorang untuk mendapatkan atau bertukar

informasi dan mempercepat proses pengolahan data agar dapat diolah menjadi sebuah informasi. Dengan meningkatnya kecanggihan teknologi internet maka kegiatan jual beli pun dapat dilakukan secara online, dan pada zaman sekarang sudah banyak toko dan para penjual yang sudah dapat memasarkan produknya secara online melalui e-commerce dan e-marketplace.

Situs-situs e-commerce dan e-marketplace yang ada di dunia internet sekarang ini memberikan dan menawarkan setiap produk dengan harga dan deskripsi produk yang berbeda-beda. Jumlah deskripsi produk dan jenis produk ini tentunya jumlahnya sangat banyak dan beraneka ragam sehingga untuk mencari produk yang diinginkan menghabiskan waktu yang sangat lama, Untuk mengatasi masalah tersebut dalam mencari produk yang diinginkan secara detail dengan waktu yang efisien dapat diatasi dengan pengambilan informasi menggunakan web scraping.

Web Scraping adalah proses pengambilan sebuah dokumen-semi terstruktur dari internet, umumnya berupa halaman-halaman Web dalam bahasa markup seperti HTML (Hypertext Markup Language) atau XHTML (Extensible HyperText Markup Language), dan menganalisis dokumen tersebut untuk diambil data tertentu dari halaman tersebut untuk digunakan bagi kepentingan lain. Bahasa markup adalah bahasa komputer yang menggunakan tag untuk mendefinisikan elemen dalam dokumen. (Matthew Turland, 2010). Dilakukannya kegiatan Web Scraping biasanya adalah untuk Kepentingan pelaku seperti mencari sebuah produk secara detail dengan waktu yang cepat, menggali konten sebuah situs web secara cepat perhalaman.

Dengan adanya teknik web scraping ini tentu saja sudah banyak beberapa orang yang menggunakan teknik ini untuk mencari informasi dan detail yang lebih cepat. Tetapi ada juga yang berdampak buruk terhadap beberapa penyedia informasi tertentu yang tidak ingin informasi, data, dan kontennya yang diambil dengan cara scraping ini karena untuk kepentingan persaingan bisnis yang dimana data dapat diambil untuk dijadikan sebuah informasi yang memiliki kepentingan dari pelaku web scraping ini, dan sebagai contoh ada beberapa kasus yang melibatkan bot yang diprogram untuk kecepatan mengambil sebuah produk secara cepat di dunia e-commerce dan e-marketplace untuk mengambil beberapa produk yang memiliki promo diskon terutama promo flash sale yang dimana bot ini sudah disiapkan untuk mengambil produk tersebut ketika baru pertama kali rilis sehingga menyebabkan produk flash sale tersebut habis terjual secara cepat, bahkan pengguna biasa yang ingin membeli produk tersebut tidak memiliki kesempatan untuk membeli karena kalah cepat dengan bot-bot yang telah dirancang oleh pelaku web scraping ini. Karena kasus yang ada tersebut maka dibuatlah sebuah metode sistem proteksi situs web dari web scraping ini untuk dapat menghalangi sebuah robot untuk masuk kedalam sistem situs web.

CAPTCHA atau Captcha (Completely Automated Public Turing test to Tell computer and Human Apart) adalah sebuah sistem tes uji tanggapan (challenge-response test) yang biasanya digunakan di dunia perkomputeran untuk melakukan verifikasi manusia, apakah yang melakukan akses atau meminta request ke sebuah sistem adalah seorang manusia sungguhan atau hanya komputer(bot). Dengan adanya CAPTCHA ini, dapat melindungi sistem situs web penyedia informasi dari bot, crawler, scraper, spammer, bahkan melindungi konten web.

Dalam penelitian ini, penulis mengembangkan sebuah metode untuk mencegah dan mengatasi masalah tersebut, yaitu Metode bagaimana memproteksi situs web dari teknik web scraping ini. Penulis dapat menggunakan metode human verification untuk mencegah bot masuk kedalam sebuah situs web, kemudian metode manajemen IP address, proteksi javascript, blokir alamat IP yang mencurigakan, membatasi request untuk halaman web yang dilindungi, dan manajemen traffic data situs web.

Dengan adanya metode-metode tersebut mungkin dapat mengurangi bahkan mencegah bot untuk masuk kedalam sebuah situs web yang tidak ingin dimasuki oleh bot. Karena metode untuk melakukan verifikasi Manusia atau tidak biasanya hanya dapat dilakukan oleh manusia yang asli karena manusia memiliki otak untuk berfikir sedangkan bot hanya mengandalkan mesin untuk bekerja, dan juga melakukan manajemen web server juga sangatlah penting agar pemilik atau penyedia informasi dapat membatasi akses bot untuk masuk dan mengambil data-data yang ada.

2. Dasar Teori /Material dan Metodologi/perancangan

2.1 Web Scraping

Web Scraping adalah proses pengambilan sebuah dokumen-semi terstruktur dari internet, umumnya berupa halaman-halaman Web dalam bahasa markup seperti HTML (Hypertext Markup Language) atau XHTML (Extensible HyperText Markup Language), dan menganalisis dokumen tersebut untuk diambil data tertentu dari halaman tersebut untuk digunakan bagi kepentingan lain.

Bahasa markup adalah bahasa komputer yang menggunakan tag untuk mendefinisikan elemen dalam dokumen[2].

Ketika pengguna web scraping sudah mendapatkan data atau dokumen yang diinginkan maka data tersebut akan disimpan melalui proses parsing, trimming, saving, dan compile yang kemudian dilakukan sebuah analisa untuk mengambil data tertentu dari halaman web tersebut yang diinginkan, sehingga pengguna web scraping dapat mencari data atau informasi yang diinginkan dengan cepat di berbagai situs e-commerce dan e-marketplace tanpa harus menghabiskan waktu yang sangat lama untuk mencari satu persatu di tiap halaman web. Bahkan ada juga pengguna web scraping yang tidak memerlukan login session untuk melakukan pencarian data pada sebuah halaman web jika situs web tersebut tidak memiliki sistem keamanan yang baik.

2.2 CAPTCHA

CAPTCHA atau Captcha (Completely Automated Public Turing test to Tell computer and Human Apart) adalah sebuah sistem tes uji tanggapan (challenge-response test) yang biasanya digunakan di dunia perkomputeran untuk melakukan verifikasi manusia, yang dapat membedakan apakah yang melakukan akses atau meminta request ke sebuah sistem adalah seorang manusia sungguhan atau hanya komputer(bot) yang menyamar menjadi manusia. CAPTCHA biasanya digunakan di beberapa sistem yang memang tidak ingin dimasuki oleh bot, karena biasanya bot dibuat oleh pembuatnya untuk melakukan pengambilan data dan juga beberapa hal yang mungkin dapat berdampak negatif bagi penyedia informasi. CAPTCHA biasanya berbentuk tes uji gambar (image based) untuk melakukan tes cara berfikir seorang manusia sungguhan, berbentuk barisan tulisan atau teks (text based) dimana pengguna diminta untuk melakukan inputan tulisan atau nomor yang benar, berbentuk suara(audio based) dimana pengguna diminta untuk berbicara melalui mikrofon untuk menginputkan suara yang diminta pada sistem CAPTCHA, dan ada juga yang akhir-akhir ini yang menjadi trending yaitu sistem CAPTCHA menggunakan video (Video based) dimana pengguna diminta untuk menonton sebuah video dan menginputkan sesuatu dari video tersebut untuk melakukan verifikasi manusia yang dimana akan sangat susah di prediksi oleh bot[3].

2.3 Javascript

Javascript adalah bahasa yang berbentuk kumpulan skrip yang pada fungsinya berjalan pada suatu dokumen HTML, sepanjang sejarah internet bahasa ini adalah bahasa skrip pertama untuk web. Bahasa ini adalah bahasa pemrograman untuk memberikan kemampuan tambahan terhadap bahasa HTML dengan mengizinkan pengekseskuan perintah perintah di sisi user, yang artinya di sisi browser bukan di sisi server web. (Andry Alamsyah, 2003).

2.3.1 Pemanfaatan Javascript Untuk CAPTCHA

Javascript dapat dimanfaatkan untuk membuat sebuah verifikasi manusia yaitu CAPTCHA. untuk menampilkan sebuah CAPTCHA, web browser harus dapat mengaktifkan dukungan javascript, maka dari itu banyak sekali pelaku web scraping yang mematikan fitur javascript untuk melakukan bypass terhadap sistem keamanan autentikasi manusia berbasis CAPTCHA ini. Karena hal tersebut dibuatlah sebuah sistem yang apabila browser yang digunakan tidak mengaktifkan fitur javascript, maka halaman web tidak akan termuat, dan akan ditampilkan pertanyaan berbentuk reCAPTCHA.

2.3.2 Membatasi Akses Right Click dan Inspect Element

Memanfaatkan fitur javascript untuk membuat sistem dimana pengunjung situs web tidak dapat melakukan fitur right click atau klik kanan pada mouse dan juga tombol shortcut pada keyboard untuk fitur inspect element sehingga dapat membatasi akses pengunjung web untuk mengetahui struktur situs web pemilik.

2.3.3 Sistem Javascript Browser Checker

Membuat sistem situs web dimana sistem tersebut akan memuat javascript terlebih dahulu baru masuk ke HTML situs web, sehingga dapat dilakukan verifikasi apakah web browser yang digunakan oleh pengunjung web sudah mengaktifkan javascript.

2.4 Robot.txt

Sekumpulan kode atau file berbentuk text yang berfungsi sebagai pengatur dan pemberi instruksi pada robot search engine. File robot.txt ini akan digunakan untuk mengelola traffic bot

seperti bot scraping dan crawler yang biasanya akan bekejra menggunakan SEO (Search Engine Optimization) yaitu sebuah upaya untuk optimalisasi situs web agar muncul di paling atas hasil pencarian mesin pencari seperti google.

2.5 Blokir IP berdasarkan parameter

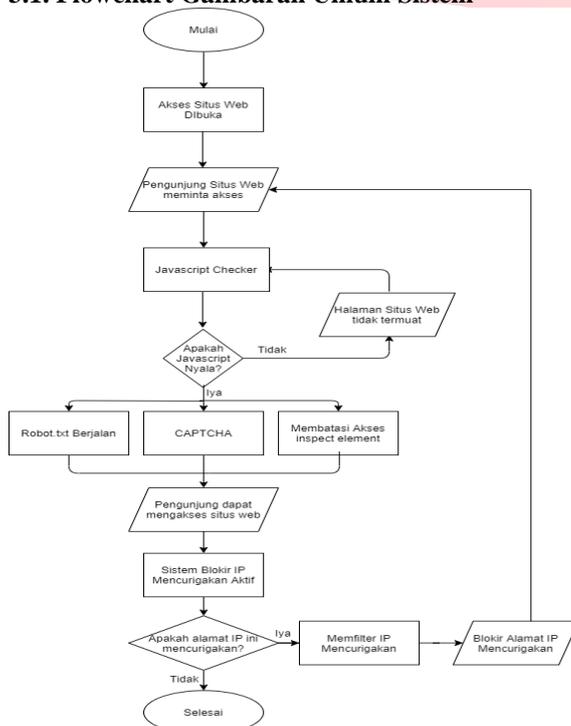
Setelah situs web berhasil dibangun, dibuatlah sebuah sistem yang dapat membatasi sesi untuk menghindari tindakan atau perilaku pengunjung situs web yang sebenarnya adalah robot yang menyamar menjadi manusia, karena terkadang ada beberapa pelaku web scraping yang dapat membuat bot super canggih yang dapat melakukan bypass keamanan dari sistem javascript browser checker dan CAPTCHA. Melihat hal itu maka dibuatlah sistem keamanan berlapis selanjutnya, dan manajemen traffic web juga dapat meminimalisir adanya bot yang masuk ke situs web karena pemilik web dapat memantau secara langsung request dari pengunjung, dan disini dapat dilihat dari HTTP header yang meminta request untuk membedakan apakah pengunjung tersebut adalah manusia atau robot.

2.5 Inspect Element

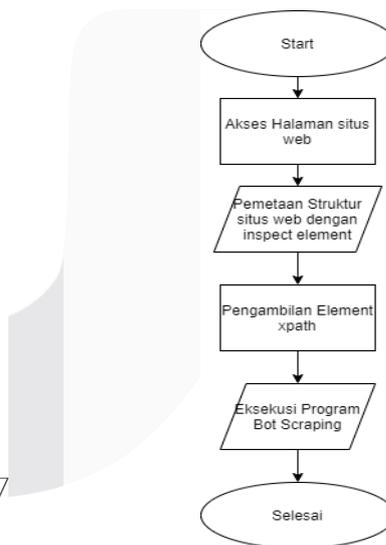
Inspect Element adalah sebuah tool yang biasanya adalah salah satu fitur dari alat developer pada web browser yang digunakan yang fungsinya adalah memudahkan kita untuk melihat struktur situs web dan juga mengubah tampilan situs secara simulasi dari web browser.

3. Pembahasan

3.1. Flowchart Gambaran Umum Sistem



Gambar 3.1 Flowchart Gambaran umum sistem



Gambar 3.2 Gambaran umum pengujian scraping

Metode yang akan diimplementasikan ke situs web yang telah dikembangkan untuk situs web pengujian adalah javascript checker, CAPTCHA, Blokir Inspect element, robot.txt, blokir IP berdasarkan parameter. Setelah metode di implementasikan akan dilakukan pengujian menggunakan robot web scraping dengan library selenium menggunakan metode xpath.

3.2. Pengumpulan Informasi dan Metode

Dengan memperoleh informasi tentang bagaimana web scraping bekerja, kita dapat mengetahui bagaimana bot web scraping itu bekerja, dan bagaimana proses pengambilan data dilakukan melalui analisa yang ada. Pengumpulan informasi ini akan berguna karena membantu

untuk mengetahui bagaimana scraper bekerja, dari ekstensi, dan dari apa saja yang dapat mencegah scraper untuk melakukan aksinya.

3.3. Perancangan Sistem

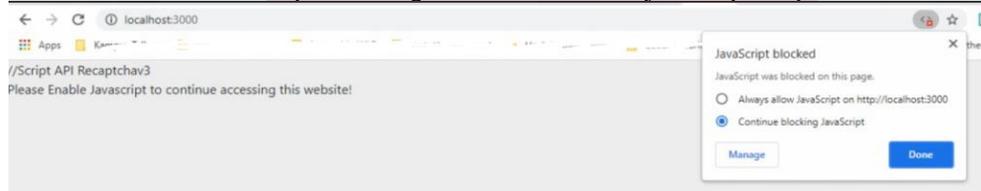
Sistem dirancang dengan menggunakan beberapa proses seperti proses pengembangan web menggunakan beberapa bahasa pemrograman dan runtime untuk membantu pengembangan situs web pengujian dengan cepat, dan juga proses tiap metode seperti implementasi sistem javascript checking, fitur CAPTCHA, robot.txt, dan pemblokiran IP yang dirancang dengan menggunakan sistem backend dan database agar dapat dilakukan blokir IP secara dinamis dan otomatis ketika pengunjung situs web melakukan pelanggaran.

4. Hasil dan Analisis

4.1. Pengujian Sistem dari Sisi Manusia Scraper

Pengujian ini dilakukan dari sisi manusia ketika pengguna tersebut mengakses halaman situs web, sistem untuk proteksi situs web akan dirangkai berdasarkan analisa bagaimana pengguna melakukan teknik awal web scraping pada sebuah situs web. Setiap metode yang sudah diimplementasikan ke dalam situs web akan di uji.

Pengujian Javascript berhasil menghalangi pengunjung Web ketika ingin melakukan akses ke situs web tanpa mengaktifkan fitur javascript pada web browser.

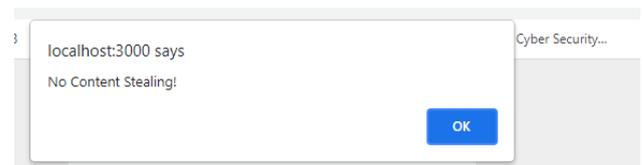


Gambar 4.1 Gambar Output Validasi Javascript

Dengan berhasilnya metode pertama, maka javascript akan mengeksekusi metode-metode lain yang membutuhkan bahasa pemrograman javascript. Setelah metode pertama berhasil metode kedua akan aktif yaitu script untuk melakukan blokir akses inspect element, robot.txt, serta CAPTCHA. Blokir akses terhadap klik kanan dan juga tombol shortcut keyboard untuk melakukan inspect element, karena tahap pertama untuk melakukan web scraping adalah melakukan inspect element terhadap situs web untuk mengetahui struktur web dan element *xpath* dari situs web. Dalam pengujian metode ini akan di buat sebuah program untuk mematikan fitur tombol key event menggunakan kode tombol keyboard event javascript.

| Tombol Shortcut Keyboard Inspect Element | |
|--|---------------|
| Nama Tombol | Kode Keyboard |
| F12 | 123 |
| Ctrl + U | 85 |
| Ctrl + Shift + U | 85 |
| Ctrl + Shift + J | 74 |
| Ctrl + Shift + I | 73 |
| Ctrl + Shift + C | 67 |
| Ctrl + S | 83 |

Tabel 4.1 Tabel tombol keyboard shortcut inspect element



Gambar 4.2 Pengujian Membatasi hak akses inspect element

Dengan berhasilnya metode tersebut maka ketika web scraper tidak bisa melakukan inspect element pada situs web tersebut maka metode web scraping tidak akan pernah bisa dilakukan sampai

web scraper berhasil mendapatkan element yang diperlukan dari situs web tersebut untuk melakukan web scraping.

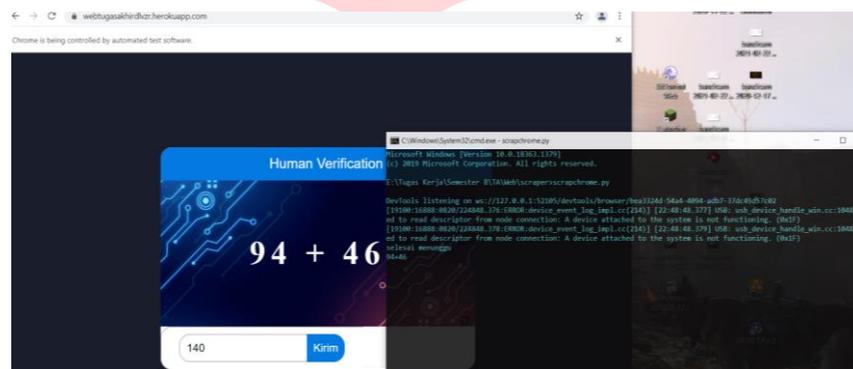
4.2. Pengujian Sistem dari Sisi Robot

Pengujian ini diibaratkan jika seorang scraper berhasil melakukan inspect element pada situs web yang telah dibuat oleh penulis, disini penulis akan berperan sebagai pelaku web scraping. Pengujian ini dilakukan menggunakan aplikasi web scraping dengan tool otomasi web browser selenium menggunakan metode xpath guna mendapatkan data dari situs web, untuk melakukan tahap ini pelaku scraper harus dapat melakukan inspect element terhadap situs web terlebih dahulu untuk dapat mengidentifikasi id dari setiap struktur situs web yang ada, jika berhasil diidentifikasi id tersebut maka dapat dimasukkan kedalam baris program yang ada yang kemudian akan dieksekusi untuk melakukan pekerjaan secara otomatis

Metode pertama yang diuji adalah CAPTCHA untuk melakukan verifikasi apakah pengunjung web adalah seorang manusia atau robot. Pada tahap ini diasumsikan manusia berhasil melakukan inspect element terhadap halaman CAPTCHA, dan berhasil mendapatkan *xpath*

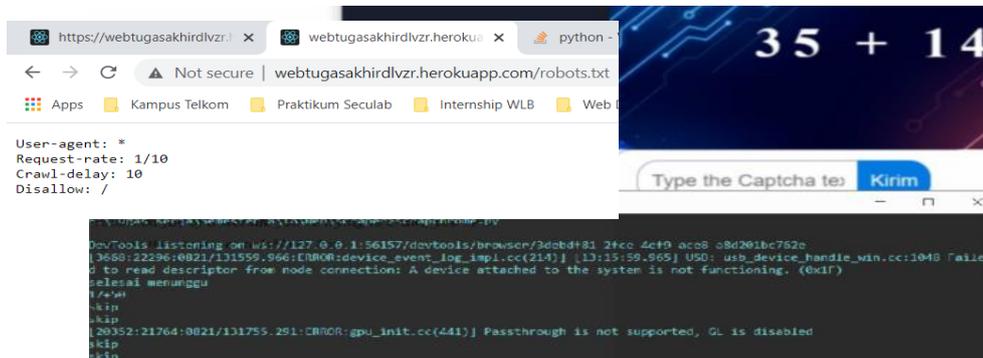
| Pengujian CAPTCHA Sisi Bot | | | |
|----------------------------|------------|---------------|---------------------------------------|
| No | Nama | HTML Element | Xpath |
| 1 | Code | div container | (//*[@id="code"]) |
| 2 | Text Input | input type | //*[@id="root"]/div/div/div[3]/input |
| 3 | Kirim | Submit button | //*[@id="root"]/div/div/div[3]/button |

Tabel 4.2 Pengujian CAPTCHA Sisi Bot



Gambar 4.3 Pengujian CAPTCHA

Pada gambar tersebut berhasil dilakukan web scraping terhadap halaman CAPTCHA dimana bot scraping dapat menyelesaikan pola CAPTCHA secara otomatis walaupun pola yang diberikan adalah acak. Di halaman CAPTCHA tersebut juga terdapat robot.txt yang dimana difungsikan untuk melarang robot crawler untuk masuk kedalam situs web untuk melakukan scraping. Robot.txt yang sudah terpasang berhasil ditembus oleh selenium karena masih dapat melakukan scraping padahal robot.txt sudah berjalan.



Gambar 4.3 Pengujian Robot.txt

Setelah metode sebelumnya berhasil tertembus, ketika sudah sampai halaman utama, sistem blokir IP akan berjalan. Pengunjung web diharuskan menunggu 5 detik ketika masuk pertama kali ke halaman CV, karena biasanya robot akan melakukan klik secara interval ketika berhasil masuk, sehingga parameter untuk blokir IP ini adalah jumlah klik yang dilakukan dalam satu waktu. Berikut *xpath* untuk halaman CV.

| Tabel <i>Xpath</i> halaman CV | | |
|-------------------------------|---------------|--|
| Nama | HTML Element | <i>xpath</i> |
| Show More | button | //*[@id="root"]/div/main/button |
| Input captcha | Input type | ./input[contains(@class,"MuiInputBase-input MuiInput-input |
| sumbit | Submit button | ./button[contains(@class,"MuiButtonBase-root MuiButton-root MuiButton-text |

Tabel 4.4 Tabel *Xpath* halaman CV

Pada pengujian blokir IP robot scraper berhasil dilakukan blokir ketika pertama kali mengakses karena robot melakukan interval klik (klik secara berkala) dan menyebabkan robot terblokir. Berikut adalah output dari command line robot scraping selenium ketika terblokir, disini robot tidak dapat menemukan *xpath* lain yang telah ditentukan di program scraping.

```

E:\Tugas Kerja\Semester 8\TA\Web\scraper\scrapchromenodelay.py
DevTools listening on ws://127.0.0.1:56959/devtools/browser/316821d7-42ea-4251-9bac-c6f9d1ce456b
[24368:15684:0821/115253.916:ERROR:device_event_log_impl.cc(214)] [11:52:53.916] USB: usb_device_handle_win.cc:1048 Failed to read descriptor from node connection: A device attached to the system is not functioning. (0x1F)
[24368:15684:0821/115253.919:ERROR:device_event_log_impl.cc(214)] [11:52:53.919] USB: usb_device_handle_win.cc:1048 Failed to read descriptor from node connection: A device attached to the system is not functioning. (0x1F)
selesai menunggu
xvfbwin
Traceback (most recent call last):
  File "E:\Tugas Kerja\Semester 8\TA\Web\scraper\scrapchromenodelay.py", line 69, in <module>
    captcha2 = ibu.find_element_by_xpath("./div[contains(@class,\"MuiTypography-root MuiTypography-body1\")]").text
  File "C:\Users\Dalvizar1\AppData\Local\Programs\Python\Python38-32\lib\site-packages\selenium\webdriver\remote\webdriver.py", line 351, in find_element_by_xpath
    return self.find_element(by=By.XPATH, value=xpath)
  File "C:\Users\Dalvizar1\AppData\Local\Programs\Python\Python38-32\lib\site-packages\selenium\webdriver\remote\webdriver.py", line 658, in find_element
    return self.execute(Command.FIND_CHILD_ELEMENT,
  File "C:\Users\Dalvizar1\AppData\Local\Programs\Python\Python38-32\lib\site-packages\selenium\webdriver\remote\webdriver.py", line 633, in execute
    return self.parent.execute(command, params)
  File "C:\Users\Dalvizar1\AppData\Local\Programs\Python\Python38-32\lib\site-packages\selenium\webdriver\remote\webdriver.py", line 321, in execute
    self.error_handler.check_response(response)
  File "C:\Users\Dalvizar1\AppData\Local\Programs\Python\Python38-32\lib\site-packages\selenium\webdriver\remote\errorhandler.py", line 242, in check_response
    raise exception_class(message, screen, stacktrace)
selenium.common.exceptions.NoSuchElementException: Message: no such element: Unable to locate element: {"method": "xpath", "selector": "//*[@id='root']/div/main/button"}
(Session info: chrome=92.0.4515.159)
    
```

Gambar 4.4 Ouput Program Selenium (IP Terblokir)

Berikut adalah tampilan ketika sudah terblokir :



Gambar 4.5 IP Terblokir

Walaupun robot terblokir tapi sistem ini memiliki celah dimana ketika dilakukan perubahan kode pada program bot scraping yaitu *time.sleep(20)* maka sistem ini dapat ditembus karena berhasil melewati sistem blokir ip dengan parameter klik secara berkala.

REFERENSI

- [1] Abdillah L. A., & Emigawaty. "Analisis laporan tugasakhir mahasiswa Diploma I dari sudut pandang karya ilmiah dan penggunaan teknologi informasi", Jurnal Ilmiah Matrik, 2009.
- [2] Dana Febri Setiawan, Tristiyanto, Astria Hijriani, "Aplikasi Web Scraping Deskripsi Produk", Jurnal TEKNOINFO, 2020.
- [3] Eko Budi Setiawan, "Optimalisasi Keamanan Website Menggunakan CAPTCHA – AD VIDEO", Jurnal Komputer dan Informatika (KOMPUTA), 2012.
- [4] MattMatthew Turland. "Architect ' s Guide to Web Scraping defined. Los Angeles.hew Turland", Los Angeles Hew Turland, 2010.
- [5] Andry Alamsyah, "Pengantar Javascript,", Ilmu Komputer, 2003.
- [6] Analia Lourenco, Orlando Belo, "Catching Web Crawlers in the Act," ICWE, 2006.
- [7] Baljrit Singh Saini, Anju Bala, "A Review of Bot Protection using CAPTCHA for Web Security," iosjournals, 2013.
- [8] Nicol as Pogigi, Toni Moreno, Jordi Torres, Josep Lluís Berral, Ricard Gavald's, "Automatic Detection and Banning of Content Stealing Bots for E-commerce" Universitat Polit' enca de Catalunya, 2007.
- [9] Zachary Gold, Mark Latonero, "Robots Welcome? Ethical and Legal Considerations for Web Crawling and Scraping", Washington Journal of Law, Technology & Arts, 2018.
- [10] Ling He, "System and Method for Preventing Web Crawler Access", US Patent, 2014.
- [11] Damon Laython Petra, Bradley Keith Mobs, "Systems And Methods To Control WEB Crawler", US Patent Petta et al, 2013.
- [12] Paul Sweatte, "How to Stop Scraping", <https://stackoverflow.com/questions/3161548/how-do-i-prevent-site-scraping>.
- [13] By0 , "Script to block acces on right click and keyboard using javascript" <https://qastack.id/programming/737022/how-do-i-disable-right-click-on-my-web-page>
- [14] Fabrizzo "Disable Ctrl + U, Ctrl + C, ctrl + V, ctrl + S using some various javascript", <https://stackoverflow.com/questions/20372394/how-to-disable-ctrlu-using-javascript>
- [15] Glitz, "Pengertian, fungsi, dan cara kerja robot.txt", <https://glints.com/id/lowongan/robots-txt-adalah/>