STORAGE – Jurnal Ilmiah Teknik dan Ilmu Komputer, Vol. 5 No. 2, Mei 2026, 152 – 163
e-ISSN: 2828-5344 | DOI: 10.55123

ANALISIS KOMPARATIF AUTOMATED WEB RECONNAISSANCE MENGGUNAKAN NUCLEI, DIRSEARCH, DAN KATANA PADA WEBSITE BERBASIS NEXT.JS

[1)]Fenilinas Adi Artanto, [2)]Amrun Khakim
[1)]Informatika, Fakultas Teknik dan Ilmu Komputer, Universitas Muhammadiyah Pekajangan Pekalongan
[2)]Teknik Jaringan Komputer dan Telekomunikasi, SMK Nurul Ummah Paninggaran
[1)]fenilinasadi@gmail.com, [2)]amrunkhakim@gmail.com

INFO ARTIKEL

Riwayat Artikel :
Diterima : 30 April 2026
Disetujui : 26 Mei 2026

Kata Kunci :
Dirsearch, Katana, Nuclei, Reconnaissance

ABSTRAK

Perkembangan aplikasi web modern berbasis *JavaScript* dan *framework Next.js* menyebabkan meningkatnya kompleksitas struktur aplikasi serta luasnya *attack surface* yang dapat dimanfaatkan dalam serangan siber. Dalam proses pengujian keamanan aplikasi web, tahapan *automated web reconnaissance* menjadi penting untuk melakukan identifikasi *endpoint*, asset aplikasi, dan konfigurasi keamanan sebelum dilakukan pengujian lanjutan. Penelitian ini bertujuan untuk menganalisis keefektifan *tools reconnaissance* yaitu *Nuclei, Dirsearch,* dan *Katana* pada website mygarden.my.id berbasis *Next.js*. Penelitian menggunakan metode eksperimen komparatif dengan parameter evaluasi berupa jumlah *endpoint*, jenis HTTP *response, asset JavaScript* dan CSS, serta waktu eksekusi *scanning* dan *crawling*. Hasil pengujian menunjukkan bahwa *Nuclei* efektif dalam melakukan *fingerprinting* dan mendeteksi penggunaan *Web Application Firewall* (WAF) berbasis *Cloudflare*. *Dirsearch* memiliki kemampuan baik dalam menemukan direktori dan *endpoint* tersembunyi menggunakan metode *dictionary-based scanning*. Sementara itu, *Katana* menunjukkan performa terbaik dalam melakukan *crawling endpoint* dinamis serta identifikasi asset *JavaScript* dan CSS pada aplikasi berbasis *Next.js*. Penelitian ini menunjukkan bahwa setiap tools memiliki karakteristik *reconnaissance* yang berbeda dan saling melengkapi dalam proses identifikasi *attack surface* aplikasi web modern. Kontribusi utama penelitian ini adalah mengintegrasikan analisis *fingerprinting*, *directory enumeration*, dan *JavaScript-based crawling* dalam satu skenario pengujian *reconnaissance* pada aplikasi web modern berbasis *Next.js* secara sistematis.

ARTICLE INFO

Article History :
Received : Apr 30, 2026
Accepted : May 26, 2026

Keywords*:*
Dirsearch, Katana, Nuclei, Reconnaissance

ABSTRACT

The development of modern web applications based on JavaScript and the Next.js framework has led to increasing complexity in application structures and a wider attack surface that can be exploited in cyberattacks. In the process of testing web application security, the automated web reconnaissance stage is crucial for identifying endpoints, application assets, and security configurations before further testing. This study aims to analyze the effectiveness of reconnaissance tools, namely Nuclei, Dirsearch, and Katana on the Next.js-based mygarden.my.id website. The study used a comparative experiment method with evaluation parameters including the number of endpoints, HTTP

response types, JavaScript and CSS assets, and scanning and crawling execution times. The test results show that Nuclei is effective in fingerprinting and detecting the use of a Cloudflare-based Web Application Firewall (WAF). Dirsearch has good capabilities in discovering hidden directories and endpoints using dictionary-based scanning methods. Meanwhile, Katana shows the best performance in crawling dynamic endpoints and identifying JavaScript and CSS assets in Next.js-based applications. This study shows that each tool has different reconnaissance characteristics and complements each other in the process of identifying the attack surface of modern web applications. The main contribution of this research is to systematically integrate fingerprinting analysis, directory enumeration, and JavaScript-based crawling in one reconnaissance testing scenario on modern Next.js-based web applications.

1. PENDAHULUAN

Perkembangan Teknologi Informasi dan Komunikasi telah mendorong transformasi digital di berbagai sektor kehidupan, mulai dari pendidikan, bisnis, pemerintahan, layanan publik hingga media portofolio digital. Salah satu bentuk implementasi transformasi digital tersebut adalah meningkatnya penggunaan aplikasi berbasis web sebagai layanan informasi dan interaksi pengguna. Aplikasi web modern umunya dirancang agar diakses luas melalui jaringan internet dengan dukungan teknologi interaktif berbasis *Javascript* dan framework modern seperti *React* maupun *Next.js* (Artanto, 2023). Kemudahan akses dan kompleksitas fitur tersebut menjadikan aplikasi web memiliki permukaan serangan (*attack surface*) yang semakin luas sehingga membutuhkan perhatian khusus pada aspek keamanan sistem (Rosyadi dkk., 2025).

Berdasarkan berbagai laporan keamanan siber global, aplikasi web masih menjadi salah satu target utama serangan karena banyak organisasi belum menerapkan pengujian keamanan keamanan modern tidak hanya berfokus pada eksploitasi kerentanan, tetapi juga mencangkup proses tahapan *reconnaissance, asset discovery, endpoint enumeration,* dan konfigurasi keamanan aplikasi (Bardian & Sutanto, 2025). Tahapan tersebut bertujuan untuk memetakan struktur aplikasi, menemukan *endpoint* tersembunyi, mengidentifikasi aset yang dapat diakses publik, serta mengenali proteksi yang diterapkan pada sistem (Ramadhan dkk., 2025). Informasi hasil dari *reconnaissance* menjadi penting dalam proses *security assessment* karena dapat digunakan untuk memahami karakteristik aplikasi seelum dilakukan pengujian keamanan lanjutan.

Seiring berkembangnya teknologi aplikasi web modern, berbagai tools otomatis dikembangkan untuk membantu proses pengumpulan informasi dan analisis permukaan serangan aplikasi web (Agustina dkk., 2025). Salah satu tools yang banyak digunakan adalah *Nuclei* yaitu tools berbasis template yang mampu melakukan *fingerprinting* teknologi, deteksi konfigurasi keamanan, serta identifikasi potensi kerentanan pada target web (Rachmini dkk., 2025). Keunggulan utama *Nuclei* terletak pada kemampuannya melakukan otomatisasi *scanning* dengan cakupan berbagai jenis kerentanan web yang telah terdokumentasi dalam template komunitas maupun template resmi (Alallah dkk., 2025).

Selain *Nuclei* tools lain yang umum digunakan dalam proses *reconnaissance* adalah *Dirsearch* yang merupakan *tools content discovery* yang berfungsi menemukan direktori, file tersembunyi, halaman administrasi, maupun endpoint yang tidak terindeks secara langsung oleh mesin pencarian (Tarigan, 2026). Tool ini bekerja menggunakan metode *dictionary-based scanning* dengan melakukan pengujian terhadap sejumlah kata kunci (*wordlist*) untuk menemukan *resource* yang dapat diakses melalui protokol HTTP (Saputra dkk., 2023).

Pada aplikasi web modern berbasis *Single Page Application* (SPA) proses eksplorasi endpoint sering kali memerlukan *crawler* yang

mampu membaca assset *JavaScript* dan *dan* struktur navigasi dinamis. Untuk kebutuhan tersebut, *Katana* hadir sebagai *web crawler* modern yang dirancang untuk melakukan eksplorasi endpoint secara otomatis dan mendalam (Hmaidy, 2024). Dalam konteks penetration testing, keberadaan direktori tersembunyi sering kali menjadi celah keamanan karena dapat mengandung file sensitif, konfigurasi sistem, ataupun panel administratif yang tidak terlindungi dengan baik (Listartha & Saskara, 2024).

Meskipun berbagai *tools reconnaissance* dan *enumeration* telah banyak digunakan dalam praktik keaman siber, penelitian yang menggabungkan efektivitas tools tersebut pada satu objek pengujian yang sama masih relatif terbatas. Seperti pada penelitian Khosiri dkk., (2025) digunakan *Nuclei* untuk pada website Fakutas Teknik Universitas Islam Madura yang menghasilkan temuan website masih memiliki celah keamanan seperti open directory, informasi sensitif yang masih dapat diakses publik, serta potensi serangan XSS. Sedangkan pada penelitian Putri dkk., (2023) pada website pemerintahan Kabupaten Kediri digunakan *tools dirsearch* yang dijalankan di *kali linux* yang menghasilkan informasi direktori yang *accessable* yang artinya ada celah yang rentan untuk dimasuki. Sebagian besar penelitiansebelumnya lebih terfokus pada *vulnerability scanning* atau *penetration testing* secara umum tanpa melakukan analisis terhadap kemampuan masing-masing tools dalam melakukan *attack surface discovery*. Selain itu penelitian terkait *reconnaisaance* modern pada aplikasi web berbasis Next.js dan asset *JavaScript* dinamis masih belum banyak dibahas..

Berdasarkan penelusuran literatur pada berbagai publikasi ilmiah dan repositori penelitian, study terkait *vulnerability scanning* umumnya fokus pada akurasi deteksi kerentanan menggunakan OWASPZAP, Nikto atau Burp Suite. Penelitian lain umumnya membahas crawling ataupun *directory* enumeration secara terpisah tanpa mengintegrasikan analisis *fingerprinting, endpoint discovery* dan *asset* enumeration dalam satu kerangka evaluasi. Sehingga masih minim penelitian komparatif yang secara khusus mengevaluasi performa *Nuclei, Dirsearch,* dan *Katana* dalam satu skenario pengujian keamanan aplikasi web yang sama.

Penelitian ini fokus pada pendekatan pengujian keamanan web dengan *pendekatan automated web reconnaissance*, yaitu *fingerprinting* dan *security identification* menggunakan *Nuclei,* pada *directory enumeration* menggunakan *Dirsearch*, serta *endpoint crawling* menggunakan *Katana* pada web modern berbasis *Next.Js*.

Objek penelitian ini adalah website mygaden.my.id yaitu sebuah aplikasi web portofolio yang menampilkan proyek pengembangan perangkat lunak dan eksperimen botani yang dapat di akses oleh publik. Sebagai apllikasi yang tersedia online, website tersebut dibangun menggunakan arsitektur web modern dengan berbagai asset *JavaScript* dan *endpoint* dinamis sehingga relevan digunakan sebagai objek pengujian *reconnaissance* dan *endpoint discovery.*

Fokus utama dalam penelitian ini adalah bagaimana menentukan tools yang paling efektif dalam mendukung proses pengujian keamanan applikasi web. Setiap tools memiliki metode kerja, fokus pengujian, dan kemampuan deteksi yang berbeda sehingga hasil yang diperoleh juga berpotensi berbeda. *Nuclei* berfokus pada identifikasi kerentanan berbasis template (Singadji dkk., 2022). *Dirsearch* memiliki lebih menitik beratkan pada pencarian direktori dan file tersembunyi (Nelmiawati dkk., 2025). Sedangkan *Katana* berorientasi pada eksplorasi strktur *endpoint* aplikasi web (Cuncis, 2023). Perbedaan tersebut menimbulkan kebutuhan akan evaluasi yang komprehensif untuk mengetahui tools mana yang paling optimal berdasarkan kebutuhan pengujian tertentu.

Berdasarkan kondisi tersebut diperlukan analsis komparatif untuk mengethaui efektvitas masing-masing tools dalam proses *reconnaissance* aplikasi web modern. Penelitian ini membandingkan kemampuan *Nuclei, Dirsearch* dan *Katana* berdasarkan parameter jumlah *endpoint* yang ditemukan, jenis respon HTTP, cakupan asset aplikasi serta efisiensi waktu eksekusi *scanning* dan *crawling*. Kontribusi utama penelitian ini adalah melakukan analisis komparatif terhadap tiga pendekatan *reconnaissance* yang berbeda, yaitu

template-based fingerprinting menggunakan Nuclei, *dictionary-based directory enumeration* menggunakan *Dirsearch*, dan *JavaScript-based crawling* menggunakan *Katana* untuk memetakan *attack surface* aplikasi web modern berbasis *Next.js* secara lebih sistematis. Dimana hasil penelitian ini dapat menjadi referensi bagi pengembangan aplikasi web, administrator sistem mamupun keamanan siber dalam memilih tools yang sesuai dengan kebutuha identifikasi aset, *enumerasi endpoint* dan pemetaan struktur aplikasi web.

2. METODE

Penelitian ini menggunakan metode eksperimen dengan pendekatan komparatif untuk menganalisis efektivitas *tools automated web reconnaissance* dalam melakukan identifikasi permukaan serangan (*attack surface discovery*) pada aplikasi web modern. Pendekatan komparatif dipilih karena penelitian berfokus pada perbandingan tiga metode *reconnaissance* yang berbeda, yaitu *template-based fingerprinting*, *dictionary-based directory enumeration*, dan *JavaScript-based crawling* pada target aplikasi web yang sama. Proses penelitian dimulai dari identifikasi kebutuhan pengujian hingga analisis hasil *scanning* dan *crawling* dari masing-masing tools.

Lingkungan pengujian menggunakan sistem operasi Windows 11 dengan *Command Line Interface* (CLI) berbasis *PowerShell* dan koneksi internet. Pengujian dilakukan menggunakan perangkat dengan prosesor AMD Ryzen 5, RAM 8 GB, dan jaringan internet fiber optik. Tools yang digunakan meliputi *Nuclei* versi terbaru dari *ProjectDiscovery*, *Dirsearch* versi *Python-based scanner*, dan *Katana* versi terbaru dari *ProjectDiscovery*.

2.1. Identifikasi Kebutuhan Pengujian

Tahapan ini dilakukan untuk menentukan ruang lingkup pengujian parameter evaluasi yang digunakan dalam penelitian. Fokus pengujian diarahkan pada proses automated web reconnaissance terhadap aplikasi web modern menggunakan tiga tolls berbeda yaitu Nuclei, Dirsearch dan Katana. Adapun ruang lingkup meliputi:

1. *Fingerprinting* dan identifikasi konfigurasi keamanan menggunakan *Nuclei*.
2. *Directory discovery* dan *endpoint enumeration* menggunakan *Dirsearch*.
3. *Endpoint crawling* dan *asset discovery* menggunakan *Katana.*

Selain itu, ditentukan parameter evaluasi yaitu:

- Jumlah *endpoint* dan direktori yang ditemukan.
- Jenis HTTP response.
- Assets *JavaScript* dan CSS yang berhasil diidentifikasi.
- Serta waktu ekseskusi masing-masing tools.

Pada penelitian ini, jumlah *endpoint* dihitung berdasarkan *endpoint* unik (*unique endpoint*) yang diperoleh dari hasil *scanning* dan *crawling* setelah dilakukan penghapusan data duplikat. HTTP *response* dikelompokkan berdasarkan kode status seperti 200 (*OK*), 403 (*Forbidden*), 404 (*Not Found*), dan 308 (*Permanent Redirect*). Sedangkan waktu eksekusi diukur menggunakan waktu total proses *scanning* yang ditampilkan pada terminal masing-masing tools.

Tabel 1. Parameter Evaluasi Tools

Tools	**Pengujian**	**Parameter**
Nuclei	*Fingerprinting* dan Indentifikasi Keamanan	Deteksi WAF, *Fingerprint* teknologi, identifikasi konfigurasi keamanan
Dirsearch	*Directory discovery* dan *endpoint enumeration*	Jumlah direktori, *endpoint*, dan variasi HTTP *response*
Katana	*Endpoint crawling* dan *asset discovery*	*Endpoint* dinamis, *asset JavaScript,*

		struktur *route* aplikasi

Kerangka evaluasi terpadu tersebut digunakan untuk membandingkan efektivitas tiga pendekatan reconnaissance yang berbeda pada target aplikasi web yang sama sehingga hasil analisis menjadi lebih sistematis dan komprehensif.

2.2. Instalasi Tools

Tahapan berikutnya adalah melakukan instalasi tools yang digunakan dalam penelitian pada lingkungan *Command Line Interface* (CLI). Ketiga tools dipilih karena mewakili pendekatan *reconnaissance* yang berbeda dalam proses pemetaan *attack surface* aplikasi web modern.

a. Instalasi *Nuclei*
   Instalasi *Nuclei* dilakukan menggunakan perintah
   ```
   go install -v github.com/projectdiscovery/nuclei/v2/cmd/nuclei@latest
   ```
b. Instasi *Dirsearch*
   Instalasi *Dirsearch* dilakukan menggunakan perintah
   ```
   git clone https://github.com/maurosoria/dirsearch.git
   ```
c. Instalasi *Katana*
   Instalasi *Katana* dilakukan menggunakan perintah
   ```
   go install github.com/projectdiscovery/katana/cmd/katana@latest
   ```
d. Konfigurasi PATH
   Agar seluruh tools dapat dijalankan melalui terminal, folder hasil instalasi dimasukan ke dalam variabel PATH sistem operasi dengan
   ```
   export PATH=$PATH:$GOPATH/bin
   ```

2.3. Konfigurasi Lingkungan Pengujian

Tahapan ini dilakukan untuk menentukan target pengujian dan parameter *scanning* yang digunakan pada masing-masing tools.

a. Penentuan target pengujian
   Target pengujian kali ini adalah website dengan alamat:
   ```
   http://mygarden.my.id
   ```
b. Konfigurasi template *Nuclei*
   Nuclei menggunakan tempalte *scanning* berbasis YMAL untuk melakukan *fingerprinting* dan identifikasi konfigurasi keamanan.
   Perbaruan template *Nuclai* dapat menggunakan
   ```
   nuclei -update-templates
   ```
   penggunaan *Nuclei* dengan
   ```
   nuclei -u http://mygarden.my.id -t cves/
   ```
   digunakan template standart dari *ProjectDiscovery* yang berkaitan dengan: teknologi web, *fingerprinting* dan deteksi *Web Application Firewall* (WAF).
   Hasil *scanning* disimpan dalam file log menggunakan:
   ```
   nuclei -u http://mygarden.my.id -t cves/ -o nuclei_result.txt
   ```
c. Wordlist *Dirsearch*
   Dirsearch menggunakan metode *dictionary-based scanning* sehingga membutuhkan *wordlist* untuk melakukan *enumerasi* direktori dan file dengan isi *worlist* admin, *dashboard*, *login*, *backup, uploads, api, assets, config, images, js, css* dan disimpan dalam file *wordlist.txt* lalu selain digunakan *wordlist* manual digunakan *wordlist* umum dari *SecList,* dan *Dirb common wordlist*. Sehingga pada *Dirsearch* menggunakan:
   ```
   python3 dirsearch/dirsearch.py -u http://mygarden.my.id -w wordlist.txt
   ```
   lalu hasil *scanning* disimpang dengan
   ```
   python3 dirsearch/dirsearch.py -u http://mygarden.my.id -w wordlist.txt -o dirsearch_result.txt
   ```
d. Konfigurasi Pramater *Katana*
   Katana digunakan untuk melakukan *crawling endpoint* dan *asset* aplikasi web modern dengan konfigurasi
   ```
   katana -u http://mygarden.my.id
   ```
   beberapa parameter tambahan yang digunakan ada *dept crawling, passive crawling* dan *JavaScript parsing* sehingga menjadi

```
katana -u http://mygarden.my.id -js -d 3
```

dan hasil *crawling* disimpan dengan:

```
katana -u http://mygarden.my.id -js -o
katana_result.txt
```

2.4. Pengumpulan Data

Seluruh hasil *scanning* dan *crawling* dari masing-masing tools disimpan dalam bentuk file log untuk dianalisis lebih lanjut. Data yang dikumpulkan meliputi jumlah *endpoint*, direktori dan file, HTTP *response*, *asset JavaScript* dan CSS, *fingerprinting* keamanan, serta waktu eksekusi scanning.

Pada tahap ini dilakukan proses normalisasi data dengan menghapus *endpoint* duplikat, mengelompokkan jenis HTTP *response*, serta memisahkan asset *JavaScript* dan CSS dari hasil *crawling. Endpoint* yang memiliki path identik hanya dihitung satu kali sebagai *endpoint* unik.

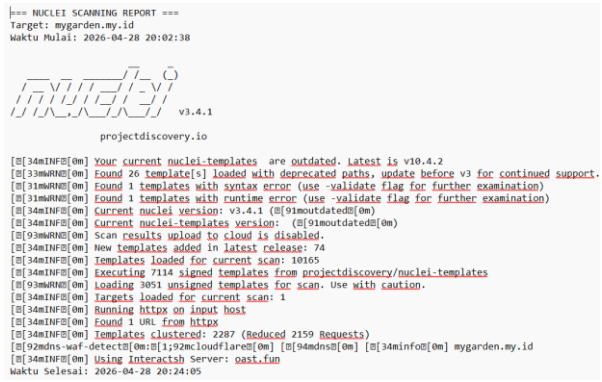
Selain berasal dari *output terminal tools*, beberapa *endpoint* dan asset hasil *reconnaissance* juga diverifikasi ulang menggunakan browser untuk memastikan *resource* dapat diakses dan sesuai dengan hasil *scanning.* Tahapan verifikasi dilakukan untuk mengurangi kemungkinan *false positive* pada hasil *reconnaissance*.

Hasil log yang telah dinormalisasi kemudian dianalisis secara komparatif untuk mengetahui efektivitas masing-masing tools dalam melakukan pemetaan *attack surface* aplikasi web modern berbasis Next.js.

3. HASIL DAN PEMBAHASAN

3.1. Hasil Pengujian *Nuclei*

Pengujian menggunakan *Nuclei* dilakukan untuk mengidetifikasi konfigurasi keamana, *fingerprinting* teknologi, serta deteksi proteksi keamanan pada website target. Proses *scanning* dilakukan dengan menggunakan *template* strandar dari *ProjectDicovery* dengan fokus pada deteksi teknologi web dan *Web Application Firewall* (WAF).



Gambar 1. *Output Nuclei*

Berdasarkan hasil *scanning*, *Nuclei* berhasil mengidentifikasi bahwa website mygarden.my.id menggunakan proteksi keamanan berbasis *Cloudflare*. Hasil tersebut ditunjukan melalui deteksi template dns-waf-detect:cloudflare yang muncul ada *output scanning*. Temuan ini menunjukan bahwa website telah menerapkan mekanisme perlindungan terhadap aktivitas *scanning* maupun *request* mencurigakan dari luar jaringan.

Selain deteksi WAF, hasil pengujian tidak menunjukan adanya kerentanan kritis seperti SQL *Injection*, *Remote Code Execution* (RCE), maupun *Cross-Site Scripting* (XSS). Namun demikian, tidak ditemukannya kerentanan kritis tidak dapat diartikan bahwa website sepenuhnya aman, melainkan hanya menunjukkan bahwa template dan konfigurasi scanning yang digunakan belum mendeteksi adanya kerentanan tersebut.

Tabel 2. Hasil Pengujian Nuclei

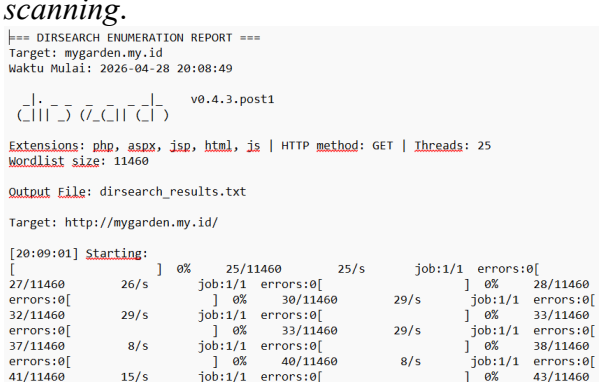
Parameter	**Hasil**
Fingerprinting Teknologi	Berhasil
Deteksi WAF	*Cloudflare*
Deteksi *Endpoint*	Terbatas
Deteksi Keamanan Kritis	Tidak Ditemukan
Output Utama	dns-waf-detect:cloudflare

Hasil tersebut menunjukan bahwa *Nuclei* lebih efektif digunakan untuk proses *fingerprinting* dan identifikasi konfigurasi keamanan dibandingkan eksplorasi *endpoint* aplikasi web. Dalam penelitian ini, *Nuclei* berperan sebagai *baseline reconnaissance* awal untuk mengetahui profil keamanan dan lapisan

proteksi target sebelum dilakukan proses enumeration menggunakan tools lain.

3.2. Hasil Pengujian *Dirsearch*

Pengujian menggunakan *Dirsearch* dilakukan untuk menemukan direktori, fiile, serta, *endpoint* tersembunyi pada website target menggunakan metode *dictionary-based scanning*.



Gambar 2. *Output Dirsearch*

Berdasarkan hasil *scanning*, *Dirsearch* berhasil menemukan beberapa direktori dan route yang memberikan berbagai jenis HTTP response seperti:

- 200 (OK),
- 403 (*Forbiden*),
- 404 (*Not Found*),
- 308 (*Permanent Redirect*)

HTTP response 200 menunjukan bahwa *endpoint* dapat diakses secara langsung oleh pengguna. Sementara *response* 403 menunjukan bahwa *endpoint* tersedia namun akses dibatasi oleh sistem keamanan website. *Response* 308 menunjukan adanya mekanisme *redirect permanent* yang digunakan pada struktur navigasi website modern.

Selain menemukan *endpoint* aktif, *Dirsearch* juga berhasil mengidentifikasi beberapa asset dan jalur akses yang berkaitan dengan struktur aplikasi web modern.

Tabel 3. Hasil Pengujian *Dirsearch*

Jenis Response	**Keterangan**
200 *OK*	*Endpoint* dapat diakses
403 *Forbidden*	*Endpoint* dibatasi sistem
404 *Not Found*	*Resource* tidak tersedia
308 *Redirect*	Pengalihan *endpoint*

Tabel 4. Karakteristik hasil *Dirsearch*

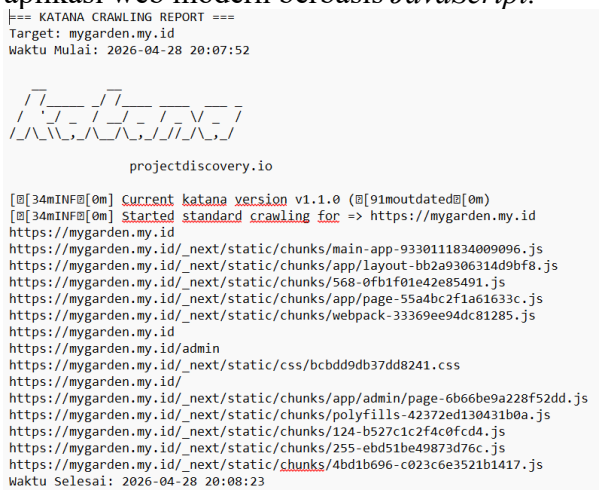
Parameter	**Hasil**
Directory Discovery	Tinggi
Endpoint Enumeration	Baik
Asset Discovery	Terbatas
Kecepatan *Scanning*	Sedang
Fokus Utama	Direktori dan endpoint

Kategori “tinggi”, “baik”, dan “terbatas” pada penelitian ini ditentukan berdasarkan jumlah endpoint unik, variasi HTTP *response* yang ditemukan, serta kemampuan tools dalam melakukan eksplorasi struktur aplikasi dibandingkan tools lainnya pada target yang sama.

Hasil pengujian menunjukan bahwa *Dirsearch* efektif digunakan untuk proses *enumerasi* direktori dan pencarian *endpoint* tersembunyi pada aplikasi web. Temuan ini memperlihatkan bahwa *Dirsearch* melengkapi fungsi *Nuclei* karena mampu memetakan struktur akses berbasis *path* yang bukan menjadi fokus utama *scanning* berbasis *template*.

3.3. Hasil Pengujian *Katana*

Pengujian menggunakan *Katana* dilakukan untuk *crawling endpoint* dan eksplorasi *asset* aplikasi web modern berbasis *JavaScript*.



Gambar 3. *Output Katana*

Berdasarkan hasil *crawling*, *Katana* berhasil menemukan berbagai *endpoint* dinamis, file *JavaScript*, file CSS, serta struktur asset yag digunakan oleh website berbasis *Next.js*. Penggunaan parameter *JavaScript crawling* (-js) memungkinkan *Katana* membaca struktur *endpoint* yang tidak dapat ditemukan melalui *scanning* direktori biasa.

Hasil *crawling* menunjukan bahwa Sebagian besar *asset* website tersimpan dalam struktur *static assets* dan *JavaScript bundle* yang umum digunakan pada framework modern seperti *Next.js.*

Tabel 5. Hasil Pengujian *Katana*

Parameter	**Hasil**
Endpoint Crawling	Berhasil
JavaScript Discovery	Tinggi
CSS *Discovery*	Berhasil
Dynamic Route Discovery	Berhasil
Fingerprinting	Terbatas

Katana menghasilkan jumlah *endpoint* dinamis dan *asset JavaScript* lebih banyak dibandingkan tools lain karena mendukung *crawling* berbasis *JavaScript parsing.* Selain itu, *Katana* mampu mengidentifikasi struktur *route* modern yang tidak dapat diperoleh melalui metode *brute-force wordlist* biasa.

Katana menunjukan kemampuan yang lebih baik dalam melakukan eksplorasi struktur aplikasi modern dibandingkan *scanning* berbasis *wordlist*. Hal ini menunjukkan bahwa *crawler modern* lebih efektif digunakan pada aplikasi berbasis *Single Page Application* (SPA) dan *framework Next.js* yang memiliki navigasi dinamis berbasis *JavaScript*.

3.4. Potensi Temuan Keamanan Website

Deteksi *fingerprinting* WAF menunjukan bahwa konfigurasi keamanan website masih dapat dikenali oleh tools *reconnaissance* yang menunjukan bahwa informasi tersebut berpotensi digunakan penyerang melakukan identifikasi teknologi dan menyusun strategi serangan yang lebih spesifik terhadap target. Namun dalam konteks *defensive security*, hasil ini dapat dimanfaatkan sebagai bahan evaluasi konfigurasi WAF agar lebih optimal dalam membatasi *fingerprinting* otomatis.

Ditemukannya *endpoint* dengan response 403 menunjukan adanya *resource internal* yang masih dapat diidentifikasi melalui proses *enumeration.* Kondisi ini daoat meningkatkan *attack surface* aplikasi apabila konfigurasi kontrol akses tidak dikelola secara optimal.

Asset JavaScript yang dapat diakses publik berpotensi mengungkap struktur *endpoint* aplikasi dan jalur komunikasi *frontend-backend*. Kondisi tersebut dapat dimanfaatkan untuk proses *reconnaissance* lanjutan apabila tidak dilakukan pembatasan *exposure asset* scara optimal.

Variasi HTTP *response* yang teridentifikasi selama proses *scanning* menunjukan bahwa struktur aplikasi masih dapat dipetakan melalui teknik *enumeration* otomatis. Informasi tersebut berpotensi digunakan dalam tahapan *reconnaissance* lenjutan pada proses *penetration testing*.

Dari hasil pengujian *reconnaissance* yang dilakukan, terdapat beberapa rekomendasi keamanan yang dapat diterapkan pada website mygarden.my.id antara lain:

1. Membatasi *exposure endpoint* dan direktori yang tidak digunakan secara publik.
2. Melakukan *filtering* terhadap *response enumeration* untuk meminimalkan *fingerprinting* struktur aplikasi.
3. Mengurangi informasi sensitif pada asset *JavaScript* yang dapat diakses publik.
4. Mengoptimalkan konfigurasi *Web Application Firewall* (WAF) untuk membatasi *automated scanning* dan *crawling.*
5. Melakukan audit berkala terhadap *endpoint* dan asset aplikasi web modern berbasis *JavaScript*.

Penelitian ini tidak hanya berfokus pada jumlah temuan *reconnaissance*, tetapi juga menghubungkan hasil scanning dengan rekomendasi pengurangan *attack surface* pada aplikasi web modern berbasis *Next.js*.

3.5. Analisis Komparatif Tools

Analisis komparatif dilakukan untuk membandingkan efektivitas masing-masing tools berdasarkan parameter yang telah ditentukan sebelumnya

Tabel 6. Perbandingan Karakteristik Tools

Parameter	***Nuclei***	***Dirsearch***	***Katana***
Fingerprinting	Berhasil mendeteksi *cloudflare*	Tidak memiliki tools tersebut	Tidak memiliki tools tersebut

Drectory Discovery	Tidak dirancang untuk *enumeration* dierktori	Sangat Baik karena menjadi fokus utama tools	Menemukan route tetapi bukan *brute-force directory scanner*
Endpoint Crawling	Tidak memiliki tools *crawler*	Menemukan *endpoint statis*	Merupakan *crawler* modern berbasis JS
Javascript Discovery	Tidak memiliki tools *crawling* JS	Hanya *brute-force path*	Mampu *Parsing JS*
HTTP Response Analysiss	Hanya *template matching*	Menampilkan banyak variasi response	Ada response tetapi bukan fokus utama
Waktu Eksekusi	Cepat	Sedang	Sedang

Berdasarkan pengujian masing-masing tools memiliki karakteristik dan fokus *reconnaissance* yang berbeda.

Nuclei lebih optimal digunakan untuk proses *fingerprinting* dan identifikasi proteksi keamanan website. hal ini terlihat dari kemampuan *Nuclei* mendeteksi penggunaan *Cloudflare* sebagai *Web Application Firewall* (WAF). Keunggulan tersebut disebabkan karena *Nuclei* menggunakan pendekatan *template-based scanning* sehingga lebih efektif dalam mengenali konfigurasi keamanan dan teknologi target. Namun, kemampuan *Nuclei* dalam menemukan *enpoint* dan asset aplikasi relatif terbatas karena tool bekerja berdasarkan *template scanning* tertentu.

Dirsearch menunjukan performa baik dalam proses *directory discovery* dan *endpoint enumeration*. Penggunaan metode *directionary-based scanning* memungkinkan *Dirsearch* menemukan berbagai direktori dan *endpoint* yang memberikan respons HTTP berbeda. Efektivitas *Dirsearch* dipengaruhi oleh kualitas dan kelengkapan *wordlist* yang digunakan. Semakin lengkap *wordlist*, semakin besar kemungkinan *endpoint* tersembunyi dapat ditemukan. Akan tetapi, kemampuan *Dirsearch* dalam membaca asset *JavaScript* dinamis masih terbatas.

Sementara *Katana* memiliki kemampuan terbaik dalam melakukan *crawling* aplikasi web modern berbasis *JavaScript*. *Katana* mampu menemukan asset, route, dan struktur *endpoint* yang tidak terdeteksi oleh metode *scanning* berbasis *wordlist*. Hal ini menunjukan bahwa *Katana* lebih sesuai digunakan pada aplikasi web modern berbasis *Single Page Application* (SPA) dan *Next.js.*

Secara umum hasil penelitian menunjukan bahwa tidak terdapat satu tools yang unggul pada seluruh parameter pengujian. Setiap tool memiliki keunggulan berdasarkan pendekatan *reconnaissance* yang digunakan. *Nuclei* efektif pada *fingerprinting* keamanan, *Dirsearch* unggul pada *enumerasi* direktori, dan *Katana* paling efektif dalam *crawling endpoint* dan *asset discovery* aplikasi web modern.

Kontribusi utama penelitian ini adalah menunjukkan bahwa proses *reconnaissance* aplikasi web modern berbasis *Next.js* lebih optimal apabila menggunakan kombinasi beberapa pendekatan *reconnaissance* yang saling melengkapi, yaitu *template-based fingerprinting, dictionary-based enumeration*, dan *JavaScript-based crawling*.

3.6. Pembahasan

Hasil penelitian menunjukan bahwa pendekatan *automated web reconnaissance* memiliki peran penting dalam identifikasi permukaan serangan aplikasi web modern. Penggunaan kombinasi tools *reconnaissance* memungkinkan proses pemetaan struktur aplikasi dilakukan secara lebih komperhensif dibandingkan hanya menggunakan satu tools tertentu.

Deteksi *Cloudflare* oleh *Nuclei* menunjukan bahwa website target telah menerapkan lapisan keamanan tambahan untuk membatasi aktivitas *scanning* dan *filtering request* mencurigakan. Kondisi tersebut mempengaruhi hasil *scanning* tools lain, khusunya response 403 dan *redirect* yang ditemukan selama proses *enumerasi.*

Dirsearch menghasilkan jumlah *endpoint* yang lebih banyak dibandingkan *Nuclei* karena menggunakan metode *brute-force* berbasis *wordlist*. Namun efektivitas *Dirsearch* sangat dipengaruhi oleh kualitas *wordlist* yang digunakan. Semakin lengkap *wordlist*, maka semakin besar kemungkinan *endpoint* tersembunyi ditemukan.

Sementara itu *Katana* menunjukan performa yang lebih relevan pada aplikasi web modern berbasis *Javascript* dan *Next.js*. kemampuan *Javascript parsing* memungkinkan *Katana* menemukan *endpoint* dinamis yang tidak tersedia secara langsung pada direktori website. Hal ini menunjukan bahwa *crawler* modern memiliki keunggulan pada aplikasi berbasis *Single Page Applicattion* (SPA).

Berdasarkan hasil tersebut dapat disimpulkan bahwa pemilihan tools *reconnaissance* perlu disesuaikan dengan kebutuhan pengujian dan karakteristik aplikasi web yang diuji. Kombinasi *Nuclei, Dirsearch,* dan *Katana* dapat menghasilkan proses *recconaissance* yang lebih optimal karena masing-masing tools memiliki fokus dan kemampuang yang saling melengkapi.

4. PENUTUP

4.1. Kesimpulan

Berdasarkan hasil penelitian yang telah dilakukan mengenai analisis *automatde web reconnaissance* menggunakan *Nuclei. Dirsearch,* dan *Katana* pada website mygarden.my.id diperoleh beberapa hasil pengujian.

Nuclei menunjukan kemampuan yang baik dalam proses *fingerprinting* dan identifikasi konfigurasi keamanan website. hasil *scanning* mendeteksi penggunaan proteksi keamanan berbasis *Cloudflare* melalui identifikasi *Web Application Firewall* (WAF). Namun, pada pengujian ini *Nuclei* tidak menemukan kerentanan kritis seperti *SQL Injection. Cross-Site Scripting* (XSS), maupun *Remote Code Execution* (RCE).

Dirsearch efektif digunakan dalam proses *dictionary discovery* dan *endpoint enumeration*. Tools ini berhasil menemukan berbagai direktori file serta *endpoint* yang memberikan beragam HTTP response seperti 200 (OK). 403 (*Forbidden*), 404 (*Not Found*), dan 308 (*Permanen Redicert*). Hasil tersebut menunjukan bahwa metode *dictionary-based scanning* cukup efektif dalam memetakan struktur akses aplikasi web.

Katana memiliki kemampuan optimal dalam melakukan *endpoint crawler* dan *asste dictionary* pada aplikasi web modern berbasis *JavaScript* dan *Next.js*. *Katana* berhasil menemukan berbagai file *JavaScript*, *asset CSS,* serta royte dinamis yang tidak ditemukan secara optimal oleh tools lain.

Hasil penelitian menunjukkan bahwa tidak terdapat satu tools yang unggul pada seluruh parameter pengujian. Setiap tools memiliki keunggulan berdasarkan pendekatan *reconnaissance* yang digunakan. Oleh karena itu, kombinasi *Nuclei*, *Dirsearch*, dan *Katana* menghasilkan proses *automated web reconnaissance* yang lebih komprehensif dibandingkan penggunaan satu tools tunggal.

Kontribusi utama penelitian ini adalah menunjukkan pentingnya pendekatan *multi-tools reconnaissance* dalam proses identifikasi *attack surface* aplikasi web modern, dimana *template-based fingerprinting*, *dictionary-based enumeration*, dan *JavaScript-based crawling* saling melengkapi dalam proses pemetaan struktur aplikasi web berbasis *Next.js.*

Berdasarkan hasil pengujian keamanan yang dilakukan, tidak ditemukan kerentanan kritis pada website mygarden.my.id. namun masih ditemukan beberapa *endpoint, asset JavaScript*, dan *response enumeration* yang berpotensi dimanfaatkan dalam proses *reconnaissance* lanjutan apabila tidak dilakukan pengelolaan keamanan secara optimal.

4.2. Saran

Pengelola website disarankan untuk melakukan audit keamanan secara berkala terhadap *endpoint*, direktori dan asset aplikasi web agar dapat meminimalkan *exposure* terhadap *attack surface* yang dapat diakses publik. Konfigurasi *Web Application Firewall* (WAF) perlu terus dioptimalkan untuk membatasi aktivitas *scanning* otomatis, *crawling* maupun *enumeration* yang dilakukan tools *reconnaissance*. Pengembangan aplikasi web disarankan untuk membatasi informasi sensitive pada asset *JavaScript* dan *endpoint* publik guna mengurangi pemetaan struktur aplikasi oleh pihak yang tidak bertanggung jawab.

Pada penelitian selanjutnya, analisis dapat dikembangkan dengan menambahkan tools *reconnaissance* maupun *vulnerability scanner* lain seperti OWASP ZAP, *Nikto*, atau *Burp Suite* sehingga cakupan evaluasi menjadi lebih luas. Selain itu, pengujian dapat dilakukan pada beberapa website dengan *framework* dan

karakteristik berbeda agar hasil komparatif menjadi lebih general dan representatif.

Penelitian berikutnya juga disarankan menambahkan parameter evaluasi kuantitatif seperti jumlah *endpoint* unik, *precision endpoint discovery*, *response time*, tingkat *false positive*, dan efektivitas *crawling JavaScript* dinamis. Pengembangan *framework reconnaissance* terpadu yang menggabungkan *fingerprinting, crawling, directory enumeration*, dan *vulnerability validation* dalam satu *pipeline* terstruktur juga dapat menjadi kontribusi lanjutan dalam penelitian keamanan aplikasi web modern.

DAFTAR PUSTAKA

Agustina, V. D., Ariyadi, T., Putra, T. S., & Lega, A. (2025). Teknik Pengujian Penetrasi HTTP Menggunakan Tools Burp Suite Pada Kali Linux. *STORAGE - Jurnal Ilmiah Teknik Dan Ilmu Komputer*, *4*(1), 16–21. https://doi.org/https://doi.org/10.55123/storage.v4i1.4770

Alallah, A. H. M., Nsrullah, M., & Alhari, M. I. (2025). Penetration Testing Pada Sebuah Website Perusahaan Education Development Dengan Framework Owasp Top-10. *Jurnal Sistem Informasi Dan Bisnis Cerdas*, *18*(2), 154–163. https://sibc.upnjatim.ac.id/index.php/sibc/article/download/418/53/706

Artanto, F. A. (2023). Perancangan sistem informasi perpustakaan negeri pelangi berbasis website. *Satesi: Jurnal Sains Teknologi Dan Sistem Informasi*, *3*(2), 52–57. https://doi.org/10.32672/jnkti.v6i2.6076

Bardian, H. A., & Sutanto, I. (2025). Pengembangan Aplikasi Vulnerability Scanner Untuk Mendeteksi Celah Keamanan Siber Pada Website. *JATI (Jurnal Mahasiswa Teknik Informatika)*, *9*(3), 4404–4411. https://ejournal.itn.ac.id/jati/article/download/13656/7589/

Cloramidine, F., & Badaruddin, M. (2023). Mengukur Keamanan Siber Indonesia Melalui Indikator Pilar Kerjasama Dalam Global CyberSecurity Index (CGI). *Populis: Jurnal Sosial Dan Humaniora*, *8*(1), 57–73. https://journal.unas.ac.id/populis/article/download/1957/1369/8477

Cuncis. (2023). *Katana: An Overview of the Powerful Web Application Security Scanner (Cheat Sheet)*. Medium. https://medium.com/@cuncis/katana-an-overview-of-the-powerful-web-application-security-scanner-cheat-sheet-6fc50236aff6

Hmaidy, A. H. (2024). *Hunting for Hidden API Endpoints Using Katana and Hakrawler*. Medium. https://anasbetis023.medium.com/hunting-for-hidden-api-endpoints-using-katana-and-hakraler-ba0bd6b9611f

Khosiri, M., Horiyah, & Anwari. (2025). Pengujian dan Analisis Kerentanan Keamanan Website Fakultas Teknik Universitas Islam Madura Menggunakan OWASP ZAP, Burp, Suite dan Nikto. *Seminar Nasional Humaniora Dan Aplikasi Teknologi Informasi(SEHATI)*, *11*(1), 10–16. https://prosiding.uim.ac.id/index.php/sehati/article/download/840/434/1249

Listartha, I. M. E., & Saskara, G. A. J. (2024). Security Testing With Penetration Testing Execution Standard (PTES) Methods to Find Misconfiguration Vulnerabilities in Network Devices. *Jurnal Elekreo Lucat*, *10*(2). https://jurnal.poltekstpaul.ac.id/index.php/jelekn/article/download/821/636

Nelmiawati, Kurniady, W., Arif, H., & Cahyono K., N. (2025). Performance Analysis of Heapdumpfinder and Dirsearch on Searching Vulnerability Heapdump Files in Spring Boot Web-Based Applications. *Proceeding of International Conference on Digital, Social, and Science*, *2*(01), 1494–1507. https://doi.org/10.62201/7kg1zw59

Putri, F. N. S., Utomo, Y. B., & Kurniadi, H. (2023). Analisa Celah Keamanan Pada Website Pemerintah Kabupaten Kediri Menggunakan Metode Penetration Testing Melalui Kali Linux. *Prosiding SEMNAS INOTEK (Seminar Nasional Inovasi Teknologi)*, *7*, 52–59. https://doi.org/10.29407/inotek.v7i1.3411

Rachmini, S. A., Indra, Zulkarnaim, N.,

Mukhram, M., & Rizaldi, A. (2025). Analisis Implementasi Nucklei Vulnerability dan Owasp-Zap Scanner Untuk Deteksi Kerentanan Keamanan (Secure System) Pada Platform Web Based. *JKT: Jurnal Komputer Terapan*, *11*(1), 10–15. https://doi.org/https://doi.org/10.35143/jkt.v11i1.6430

Ramadhan, F., Ruslianto, I., & Bahri, S. (2025). Klasifikasi Serangan SQL Injection Menggunakan Algoritma Support Vector Machine Pada HTTP Request. *Coding: Jurnal Komputer Dan Aplikasi*, *13*(3), 224–235. https://jurnal.untan.ac.id/index.php/jcskommipa/article/download/92215/756766076 74

Rosyadi, I., Artanto, F. A., & Nagara, Ikrar Stya Nafilaturrosyidah, F. (2025). Metode PSSUQ dan UMUX dalam Usability Testing pada Website 21-23 Garden. *JUMINTAL: Jurnal Manajemen Informatika Dan Bisnis Digital*, *4*(2), 245–254. https://doi.org/10.55123/jumintal.v4i2.6604

Saputra, D. R., Arizal, A., & Grinoto. (2023). Investigasi Insiden Kebocoran Data Menggunakan Integrasi Melalui Pendekatan Open Source Intelligence dan Detection Maturity Level Model. *Jurnal Info Kripto*, *17*(3), 101–108. https://infokripto.poltekssn.ac.id/index.php/infokripto/article/download/86/67

Singadji, M., Rayeb, A. El, & Azhari, M. D. (2022). Pemindaian Keamanan Web untuk Peningkatan Kesadaran Kerentanan Keamanan Web Menggunakan Nuclei. *JURNAL ADAT-Jurnal Seni, Desain & Budaya Dewan Kesenian Tangerang Selatan*, *4*(1), 91–98. https://jurnaladat.or.id/public/full_paper/Jurnal Adata HAL 91-98 PEMINDAIAN KEAMANAN WEB UNTUK PENINGKATAN KESADARAN KERENTANAN KEAMANAN WEB MENGGUNAKAN NUCLEI.pdf

Tarigan, M. (2026). Security Analysis Of Payroll System Using The Penetration Testing Execution Standard (PTES) and OWASP Top 10 2021. *PILAR Nusa Mandiri: Journal of Computing and Infromation System*, *22*(1), 96–101. https://doi.org/10.33480/pilar.v22i1.8267