PENGUJIAN KEAMANAN WEB TERHADAP SERANGAN SLOWHTTPTEST

Bambang Sugiarto[1], Arif Nursetyo[2], Kusnadi[3], Petrus Sokibi Sukanto[4]
Universitas Catur Insan Cendekia
Jl. Kesambi No. 202, Drajat, Kesambi, Cirebon City, West Java 45133
e-mail: bambang.sugiarto@cic.ac.id[1], arif.nursetyo@cic.ac.id[2], kusnadi@cic.ac.id[3], petrus.sokibi@cic.ac.id[4]

Abstrak

Penelitian ini menganalisis kerentanan ketersediaan layanan pada server web akademik https://www.cic.ac.id/ terhadap serangan Slow HTTP Denial of Service (DoS). Serangan Slow HTTP diklasifikasikan sebagai jenis serangan Lapisan Aplikasi (Layer 7) yang memanfaatkan keterbatasan thread koneksi server. Berbeda dengan serangan DoS tradisional yang berfokus pada saturasi jaringan, serangan ini mencapai resource exhaustion dengan menjaga koneksi tetap terbuka melalui transmisi data yang sangat lambat, sehingga sulit dideteksi karena menggunakan bandwidth minimal. Pengujian ini dilakukan menggunakan perangkat slowhttptest dalam mode Slow Headers, mensimulasikan mekanisme Slowloris. Target pengujian adalah meluncurkan 60.000 koneksi pada laju 200 koneksi per detik, dengan interval keep-alive 5 detik, selama 240 detik. Hasil pengujian pada detik ke-240 menunjukkan status koneksi aktif (Connected) sebanyak 247, disertai dengan 1.079 koneksi dalam status Pending. Jumlah koneksi aktif yang relatif kecil ini terbukti cukup untuk memicu kondisi kritis Service available: NO. Kegagalan ketersediaan layanan divalidasi secara eksternal melalui peramban web, yang menghasilkan pesan kesalahan Err_Connection_Timed_Out. Temuan ini mengindikasikan bahwa server target memiliki batas koneksi bersamaan (concurrent connection limit) yang rendah atau konfigurasi timeout yang tidak optimal (lebih dari 5 detik), menjadikannya sangat rentan terhadap serangan low and slow. Rekomendasi mitigasi mencakup implementasi modul manajemen timeout seperti mod reqtimeout dan penguatan Web Application Firewall (WAF) untuk membatasi laju koneksi parsial.

Kata kunci: DOS, DDOS, Kali Linux, Slowhttptest, HTTP

Abstract

This study analyzes the service availability vulnerability of the academic web server https://www.cic.ac.id/ to a Slow HTTP Denial of Service (DoS) attack. Slow HTTP attacks are classified as a type of Application Layer (Layer 7) attack that exploits limited server connection threads. Unlike traditional DoS attacks focused on network saturation, this attack achieves resource exhaustion by keeping connections open through extremely slow data transmission, making it difficult to detect due to minimal bandwidth usage. The test was executed using the slowhttptest tool in Slow Headers mode, simulating the Slowloris mechanism. The test targeted 60,000 connections at a rate of 200 connections per second, with a 5-second keep-alive interval, over a duration of 240 seconds. The test results at the 240th second showed 247 active connections (Connected), along with 1,079 connections in the Pending status. This relatively small number of active connections proved sufficient to trigger the critical status of Service available: NO. The service availability failure was externally validated via a web browser, resulting in an Err_Connection_Timed_Out error message. This finding suggests that the target server possesses either a very low concurrent connection limit or sub optimal timeout configurations (estimated to be longer than 5 seconds), rendering it highly susceptible to low and slow attacks. Mitigation recommendations include implementing timeout management modules such as mod_reqtimeout and strengthening the Web Application Firewall (WAF) capabilities to limit the rate of partial connections.

Keywords*: DOS, DDOS, Kali Linux, Slowhttptest, HTTP*

1. PENDAHULUAN

Ketersediaan (*Availability*) merupakan salah satu pilar fundamental dalam trias keamanan informasi (Kerahasiaan, Integritas, Ketersediaan) CIA, khususnya bagi infrastruktur digital krusial seperti layanan web akademik[1], [2], [3]. Kegagalan dalam memastikan ketersediaan dapat mengganggu operasi

dan aksesibilitas sumber daya pendidikan. Dalam dekade terakhir, ancaman Denial of Service (DoS) telah mengalami evolusi signifikan. Fokus serangan bergeser dari metode volumetrik pada Lapisan 3 dan 4 (Lapisan Jaringan dan Transportasi) yang bertujuan membanjiri bandwidth, menuju serangan yang lebih terselubung dan efisien pada Lapisan 7 (Lapisan Aplikasi).

Serangan Lapisan Aplikasi menargetkan sumber daya komputasi internal server seperti thread pool, memori, dan siklus CPU, dibandingkan hanya membanjiri jalur komunikasi. Pergeseran ini menghasilkan kelas serangan baru yang dikenal sebagai serangan low and slow, di mana penyerang hanya membutuhkan bandwidth minimal, sehingga sangat sulit dibedakan dari lalu lintas pengguna yang sah dengan koneksi internet yang lambat. Oleh karena itu, penelitian mengenai kerentanan server web terhadap jenis serangan Lapisan 7 ini menjadi penting untuk memperkuat postur keamanan siber institusi pendidikan. Serangan *Denial of Service* (DoS) didefinisikan secara akademik sebagai serangan siber yang dengan sengaja menghabiskan ketersediaan sistem, layanan, atau sumber daya jaringan, menjadikannya tidak tersedia bagi pengguna yang dimaksudkan[4], [5], [6].

Slow HTTP Attack adalah subtipe DoS yang mengeksploitasi cara server web menangani permintaan HTTP yang tidak lengkap. Serangan ini bertujuan untuk mengganggu kemampuan server dalam memproses permintaan yang sah dengan mengirimkan permintaan HTTP pada laju yang sangat lambat. Prinsip utama serangan ini adalah mempertahankan koneksi tetap terbuka selama mungkin, sehingga menghabiskan sumber daya server[7], [8], [9]. Karena serangan ini menggunakan aliran data yang kecil dan lambat, mekanisme keamanan tradisional yang berfokus pada volume trafik sering kali gagal mendeteksinya.

Jenis serangan yang disimulasikan dalam penelitian ini adalah Slow Headers, yang secara umum dikenal sebagai teknik Slowloris. Serangan ini bekerja dengan membuka sejumlah besar koneksi HTTP secara bersamaan ke server target[10], [11]. Setelah koneksi TCP/HTTP berhasil dibentuk, penyerang mengirimkan header HTTP secara parsial dan perlahan, tanpa pernah menyelesaikan permintaan tersebut dengan double Carriage Return Line Feed, yaitu urutan karakter yang secara standar menandai akhir dari bagian header dalam protokol HTTP

Server web berbasis thread mengalokasikan satu thread untuk setiap koneksi yang masuk, dan thread ini akan tetap sibuk menunggu sisa header permintaan[12], [13], [14]. Untuk mencegah timeout yang dapat menutup koneksi, penyerang secara periodik (misalnya, setiap 5 detik) mengirimkan header tambahan yang kecil dan parsial. Hal ini secara efektif mengikat thread server, sehingga ketika batas maksimum koneksi bersamaan server tercapai (thread pool habis), permintaan dari pengguna sah yang baru tidak dapat dilayani, yang berujung pada Denial of Service.

Penelitian ini memiliki tujuan yaitu melakukan uji coba langsung terhadap situs web akademik https://www.cic.ac.id guna membuktikan apakah situs tersebut rentan terhadap serangan Slow HTTP Headers, dengan menggunakan simulasi terkontrol melalui metode slowhttptest. Menganalisis hasil uji secara kuantitatif untuk mengetahui pengaturan server yang menyebabkan kehabisan sumber daya, terutama terkait jumlah koneksi bersamaan dan pengelolaan waktu tunggu (timeout).

Berdasarkan pengujian kerentanan menggunakan slowhttptest dalam mode Slow Headers, penelitian ini menyimpulkan bahwa server web akademik https://www.cic.ac.id terbukti rentan terhadap serangan *Denial of Service* (DoS) Lapisan 7 jenis Slow HTTP *Headers*. Kerentanan ini divalidasi oleh dua temuan utama: pertama, kondisi kelelahan sumber daya (resource exhaustion) tercapai hanya dengan mempertahankan 247 koneksi lambat aktif selama 240 detik, yang mengindikasikan batas koneksi bersamaan server sangat rendah (diperkirakan kurang dari 300 thread). Kedua, keberhasilan serangan dengan interval keep alive 5 detik menunjukkan bahwa konfigurasi timeout penerimaan header pada server terlalu permisif, sehingga memungkinkan koneksi parsial menguasai sumber daya. Kegagalan layanan yang terkonfirmasi melalui pesan Err_Connection_Timed_Out pada peramban eksternal menegaskan urgensi mitigasi, yang direkomendasikan melalui penyesuaian parameter timeout menggunakan modul seperti mod reqtimeout (Apache) dan penerapan pertahanan berlapis seperti Web Application Firewall (WAF) untuk membatasi laju dan durasi koneksi lambat.

2. METODE PENELITIAN

2.1. Teknik Pengumpulan Data

Berikut ini merupakan teknik pengumpulan data yang dilakukan pada penelitian ini:

1. Kali Linux

   Kali Linux digunakan sebagai sistem operasi utama dalam penelitian ini karena menyediakan lingkungan yang optimal untuk pengujian penetrasi dan keamanan siber. Sistem operasi ini dirancang khusus untuk kebutuhan profesional keamanan informasi, dengan koleksi alat yang sangat lengkap dan terintegrasi, seperti *SlowHTTPTest, Wireshark, Nmap, Burp Suite, dan*

Metasploit Framework. Kali Linux mendukung berbagai skenario pengujian, mulai dari simulasi serangan lapisan aplikasi hingga analisis lalu lintas jaringan secara real-time.
Dalam konteks penelitian ini, Kali Linux berperan sebagai platform yang stabil dan fleksibel untuk menjalankan simulasi serangan Slow HTTP DoS. Dengan dukungan terhadap parameter konfigurasi yang kompleks dan kemampuan logging yang detail, sistem ini memungkinkan peneliti untuk mengamati dampak serangan secara akurat, termasuk perubahan performa server, tingkat kegagalan koneksi, dan respons terhadap koneksi parsial. Selain itu, antarmuka terminal yang kuat dan kompatibilitas dengan berbagai protokol jaringan menjadikan Kali Linux pilihan ideal untuk eksperimen keamanan berbasis command-line dan scripting. Penggunaan Kali Linux juga mendukung prinsip etika dalam pengujian keamanan, karena sistem ini memungkinkan simulasi dilakukan secara lokal dan terkontrol tanpa melibatkan infrastruktur publik yang rentan. Dengan demikian, Kali Linux tidak hanya menjadi alat teknis, tetapi juga fondasi metodologis yang penting dalam pelaksanaan penelitian keamanan siber yang bertanggung jawab dan dapat direplikasi.

2. Slowhttptest
   Tools SlowHTTPTest digunakan untuk melakukan simulasi serangan Slow HTTP DoS, khususnya varian Slow Headers (seperti Slowloris) dengan cara membuka banyak koneksi HTTP secara bersamaan ke server target dan mengirimkan header HTTP secara perlahan tanpa menyelesaikannya dengan double Carriage Return Line Feed yang menandai akhir header. Teknik ini mengeksploitasi kelemahan manajemen koneksi server, karena koneksi tetap terbuka dalam waktu lama dan menghabiskan slot koneksi yang tersedia, sehingga permintaan sah dari pengguna lain tidak dapat diproses. Dengan fitur pengaturan parameter seperti jumlah koneksi, interval pengiriman, dan timeout, SlowHTTPTest memungkinkan pengujian ketahanan server secara empiris dan menghasilkan log serta grafik HTML untuk analisis dampak serangan terhadap performa dan ketersediaan layanan web.
3. Target Uji
   Pengujian dilakukan terhadap situs web akademik https://www.cic.ac.id melalui protokol HTTPS yang beroperasi pada TCP Port 443, yang berarti server tidak hanya harus menangani permintaan pada Lapisan HTTP/Aplikasi, tetapi juga memproses lapisan enkripsi TLS/SSL secara bersamaan. Hal ini menambah kompleksitas beban kerja server, karena setiap koneksi lambat yang dibuka oleh serangan Slow HTTP Headers juga memerlukan pemeliharaan, sehingga mempercepat terjadinya kelelahan sumber daya dan meningkatkan risiko gangguan layanan.

2.2. Konfigurasi Parameter Pengujian

Pengujian dilakukan dalam mode SLOW HEADERS (ditunjukkan oleh flag -H), yang meniru serangan Slowloris dengan mengirimkan header parsial secara lambat. Perintah eksekusi yang digunakan adalah: slowhttptest -c 60000 -H -g -o Slowhttp -i 5 -r 200 -t GET -u https://www.cic.ac.id/ -x 2 -p 3. Berikut ini adalah penjelasan dari parameter pengujian yang dirangkum pada Tabel 1.

Tabel 1. Parameter Pengujian

Parameter (Flag)	Deskripsi Parameter	Nilai Konfigurasi	Implikasi Serangan
-c	Jumlah total socket yang diminta.	60000	Eksploitasi ambang batas koneksi bersamaan yang tinggi.
-H	Jenis serangan HTTP lambat.	SLOW HEADERS (Slowloris)	Menjaga koneksi tetap terbuka dengan header parsial.
-i	Interval pengiriman header tambahan (keep-alive).	5 detik	Menguji timeout server. Jika > 5 detik, koneksi akan bertahan.
-g	Menghasilkan output grafik HTML untuk analisis	aktif	Memudahkan visualisasi dampak serangan terhadap performa server
-o	Prefix nama file output hasil pengujian	slowhttp	File log disimpan dengan nama Slohttp.html, berguna untuk dokumentasi

Parameter (Flag)	Deskripsi Parameter	Nilai Konfigurasi	Implikasi Serangan
-t	Metode HTTP yang digunakan	Get	Meniru permintaan sah; meningkatkan kesulitan deteksi serangan
-r	Laju pembukaan koneksi baru.	200 koneksi/detik	Mencapai batas thread server secara cepat.
-x	Durasi target pengujian.	240 detik (4 menit)	Waktu yang dialokasikan untuk memicu dan memvalidasi DoS.
-u	URL target yang diserang	https://www.cic.ac.id	Menentukan target pengujian dalam kasus ini situs akademik berbasis HTTPS
-p	Batas waktu tunggu awal koneksi.	3 detik	Memastikan koneksi TCP awal terbentuk dengan cepat.

2.3. Prosedur Pengujian

Pengujian kerentanan terhadap serangan Slow HTTP Headers dilakukan melalui tiga langkah antara lain sebagai berikut:

1. Eksekusi Serangan
   Simulasi serangan dilakukan menggunakan perangkat slowhttptest selama durasi 240 detik terhadap URL target https://www.cic.ac.id, dengan konfigurasi parameter agresif untuk memaksimalkan tekanan terhadap server.
2. Pemantauan Internal
   Status koneksi dan ketersediaan layanan dipantau secara real-time melalui output terminal dan log slowhttptest, khususnya pada detik ke-240 dan 241. Data ini mencakup jumlah koneksi aktif, koneksi tertutup, dan status layanan (Service Available), sebagaimana ditampilkan pada dokumentasi visual.
3. Validasi Eksternal
   Untuk memastikan dampak serangan terhadap pengguna akhir, ketersediaan layanan diuji secara independen menggunakan peramban web eksternal selama serangan berlangsung. Ketidakmampuan mengakses situs, yang ditandai dengan pesan Err_Connection_Timed_Out, menjadi indikator keberhasilan serangan dan dikonfirmasi melalui dokumentasi visual.

3. HASIL DAN PEMBAHASAN

Hasil pengujian yang dicatat pada terminal Kali Linux, pada detik ke-240 dan 241, menunjukkan status koneksi yang mencerminkan respons server terhadap simulasi serangan Slow HTTP Headers.

3.1. Hasil

1. Statistik Kuantitatif Akhir Pengujian
   Tabel 2 merangkum status koneksi pada saat pengujian berakhir, sesuai data yang dilaporkan oleh slowhttptest Analisis. hasil menegaskan bahwa serangan Slow HTTP Headers berhasil menyebabkan Denial of Service terhadap server target. Konfirmasi keberhasilan serangan ini didasarkan pada dua bukti kunci, internal dan eksternal.

Tabel 2. Status Koneksi Akhir pada Detik ke-240 (Slow HTTP Headers)

Kategori Status Koneksi	Jumlah Koneksi	Penjelasan Status
Initializing	0	Koneksi yang sedang dalam proses inisialisasi TCP.
Pending	1079	Koneksi yang telah diinisialisasi tetapi belum dihubungkan ke thread server (menunggu di antrian).
Connected (Aktif Lambat)	247	Koneksi yang berhasil mengikat thread server dan dijaga tetap hidup melalui transmisi lambat.
Error	0	Koneksi yang gagal karena kesalahan protokol atau jaringan.

Kategori Status Koneksi	Jumlah Koneksi	Penjelasan Status
Closed	11103	Koneksi yang ditutup oleh server (karena timeout awal atau batas koneksi terlampaui).
Service Available	NO	Indikasi ketersediaan layanan web pada server target.

2. Indikasi Internal (Status Layanan)
   Bukti paling langsung dari keberhasilan serangan adalah status kritis yang dilaporkan oleh alat uji pada detik ke-240 yaitu Service available: NO. Dalam konteks slowhttptest, status ini berarti bahwa server telah mencapai batas kapasitas koneksi bersamaan (concurrent connection pool) dan tidak mampu merespons atau menerima permintaan baru dari klien, yang merupakan definisi dari Denial of Service
3. Indikasi Eksternal (Validasi Browser)
   Untuk memvalidasi temuan internal, percobaan akses ke URL target, www.cic.ac.id, melalui peramban web eksternal pada saat serangan berlangsung menghasilkan pesan kesalahan ERR_CONNECTION_TIMED_OUT. Kesalahan connection timeout ini secara definitif menunjukkan bahwa meskipun server mungkin masih beroperasi, thread atau sumber daya yang diperlukan untuk membuat koneksi baru dan memproses permintaan yang sah telah habis dimonopoli oleh koneksi parsial yang lambat, sehingga secara fungsional server tersebut tidak dapat diakses.

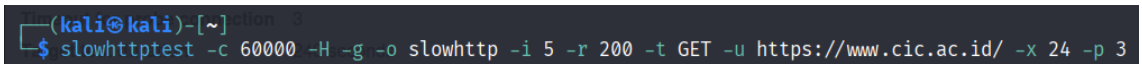
3.2. Pembahasan

Simulasi serangan dilakukan menggunakan perangkat lunak SlowHTTPTest yang dijalankan pada sistem operasi Kali Linux, sebuah distribusi Linux yang secara khusus dirancang untuk keperluan pengujian penetrasi dan keamanan siber. SlowHTTPTest dipilih karena kemampuannya dalam mensimulasikan serangan Slow HTTP DoS, khususnya varian Slow Headers, dengan parameter yang dapat dikonfigurasi secara fleksibel untuk menguji ketahanan server terhadap koneksi lambat dan parsial.

Dalam pelaksanaan pengujian, perintah serangan dieksekusi melalui terminal Kali Linux dengan menyesuaikan bagian URL target sesuai dengan situs web yang diuji. Setiap situs web yang menjadi objek pengujian dimasukkan secara eksplisit ke dalam parameter -u, yang menunjukkan alamat tujuan serangan. Parameter lainnya seperti jumlah koneksi simultan (-c), interval pengiriman data (-i), timeout koneksi (-p), dan metode HTTP (-t) disesuaikan untuk menciptakan skenario serangan yang agresif namun tetap terkendali.

Perintah yang digunakan mencerminkan konfigurasi serangan yang bertujuan untuk membuka ribuan koneksi HTTP secara bersamaan, mengirimkan header secara perlahan, dan mempertahankan koneksi aktif selama durasi tertentu tanpa menyelesaikan permintaan. Hal ini dirancang untuk mengeksploitasi kelemahan manajemen koneksi pada server target, khususnya jika server tidak memiliki pengaturan timeout yang ketat atau mekanisme pembatasan koneksi per-IP.

Sebagai contoh, pada pengujian terhadap situs web akademik https://www.cic.ac.id, parameter serangan dikonfigurasi sebagai berikut:

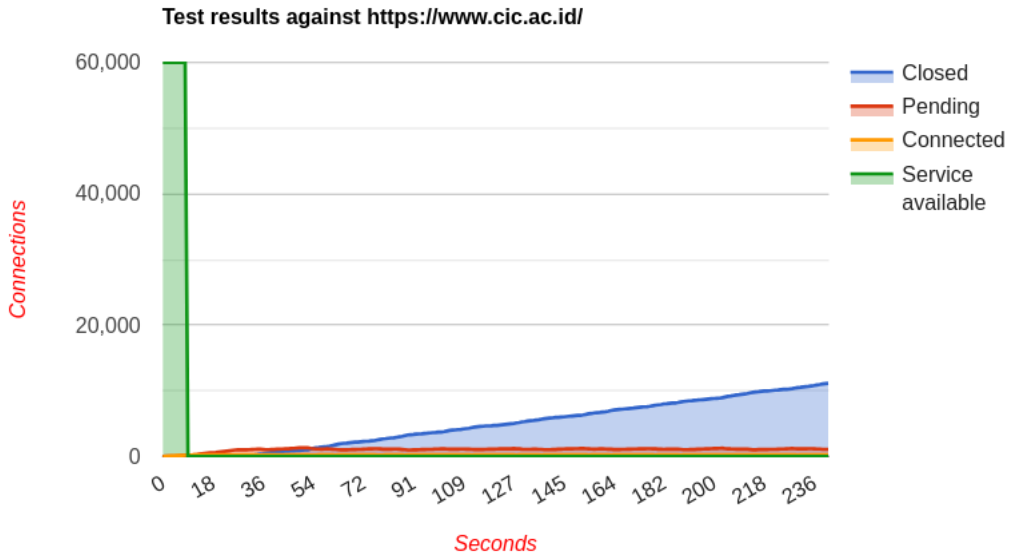


Gambar 1. Perintah Pengujian Slowhttptest Pada Kali Linux

Perintah tersebut membuka 60.000 koneksi simultan, mengirimkan header HTTP secara perlahan setiap 5 detik, dan mempertahankan koneksi dengan timeout 3 detik, sambil menghasilkan output grafik dan log hasil pengujian dalam format HTML. Dengan konfigurasi ini, peneliti dapat mengamati secara langsung bagaimana server merespons tekanan koneksi lambat, serta mengidentifikasi potensi kegagalan layanan akibat kelelahan sumber daya (*resource exhaustion*).

Gambar 2. Serangan Pada Website

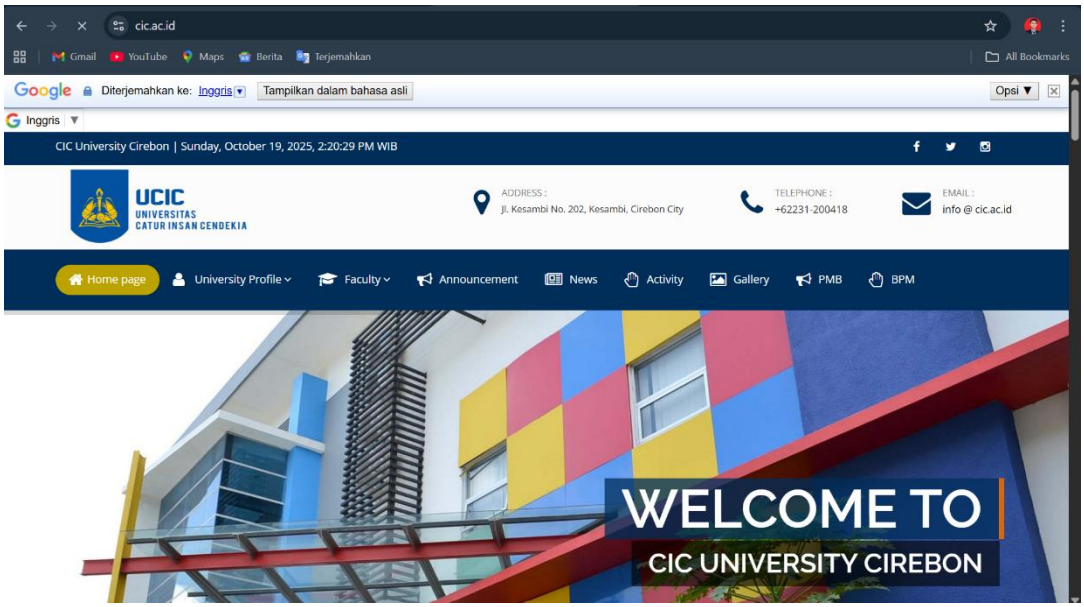
Pengujian menggunakan slowhttptest terhadap situs https://www.cic.ac.id/ menunjukkan bahwa server tersebut rentan terhadap serangan Slow HTTP DoS tipe SLOW HEADERS. Dalam uji ini, sebanyak 240 koneksi dikirim dengan laju 52 koneksi per detik, masing-masing memiliki timeout selama 300 detik, dan pengujian berlangsung selama 240 detik tanpa menggunakan proxy. Hasilnya menunjukkan bahwa meskipun tidak ada kesalahan koneksi (error = 0), sebagian besar koneksi ditutup (closed = 11.103), dan status layanan dinyatakan tidak tersedia (Service Available: NO), yang mengindikasikan bahwa server tidak mampu menangani beban dari koneksi lambat tersebut. Laporan hasil pengujian disimpan dalam format CSV, HTML, dan MHT untuk analisis lebih lanjut. Temuan ini penting bagi evaluasi ketahanan server terhadap serangan DoS berbasis aplikasi dan dapat menjadi dasar untuk rekomendasi peningkatan konfigurasi keamanan server.



Gambar 3. Grafik Hasil Serangan DOS Pada Website

Selama simulasi serangan berlangsung, grafik pada Gambar menunjukkan tren peningkatan jumlah koneksi terbuka secara signifikan. Fenomena ini mengindikasikan bahwa server web target mulai mengalami tekanan akibat tingginya volume koneksi parsial yang dikirimkan secara bertubi-tubi dan lambat. Lonjakan koneksi aktif mencerminkan ketidakmampuan server dalam mengelola permintaan yang tidak lengkap, sehingga slot koneksi yang tersedia secara bertahap terisi penuh. Kondisi ini menjadi indikator awal dari kelelahan sumber daya (resource exhaustion), yang berpotensi menyebabkan gangguan layanan atau downtime jika tidak segera dimitigasi melalui pengaturan koneksi dan timeout yang lebih ketat.

1. Tampilan Web Sebelum Serangan
   Sebelum simulasi serangan dilakukan, situs web akademik https://www.cic.ac.id menunjukkan performa yang stabil dan responsif. Halaman utama dapat diakses dengan cepat melalui peramban web, tanpa adanya gangguan atau keterlambatan dalam pemuatan konten. Server merespons permintaan HTTP dengan efisien, menandakan bahwa sistem berada dalam kondisi optimal dan siap melayani pengguna secara normal. Setelah perintah serangan dijalankan menggunakan konfigurasi yang ditampilkan pada Gambar 3, terjadi perubahan signifikan pada performa dan tampilan web target. Serangan ini melibatkan pembukaan hingga 60.000 koneksi simultan ke situs cic.ac.id, dengan interval pengiriman data setiap 5 detik dan laju pembukaan 200 koneksi baru per detik. Strategi ini dirancang untuk membebani server secara bertahap dengan koneksi lambat yang tidak pernah diselesaikan, sehingga sumber daya server terkunci untuk melayani permintaan parsial yang terus-menerus. Karena protokol yang digunakan adalah HTTPS (TCP port 443), setiap koneksi juga memerlukan proses enkripsi TLS/SSL, yang semakin memperberat beban komputasi server.

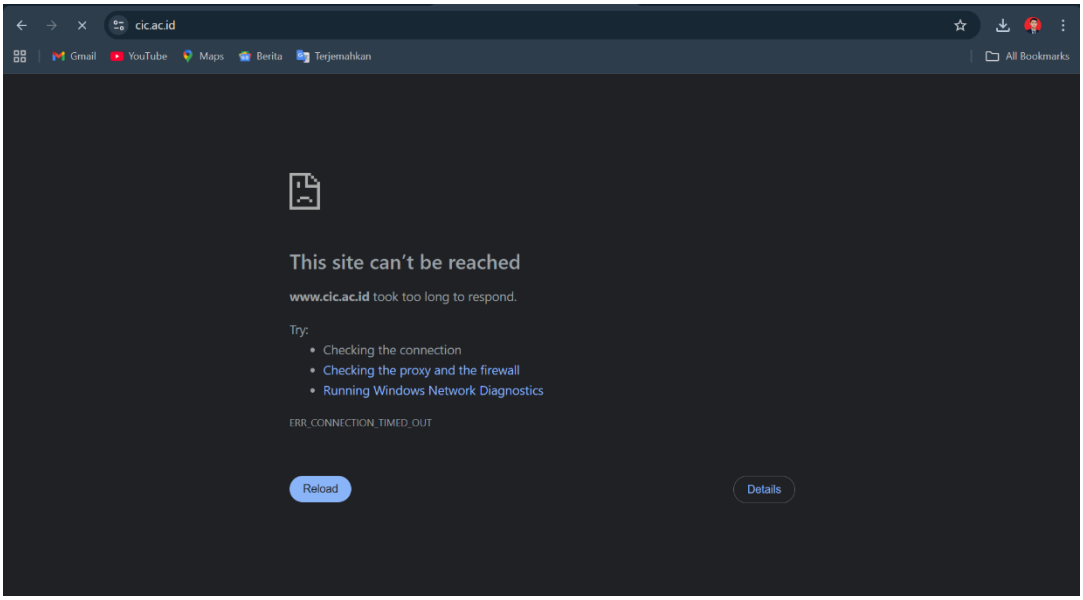


Gambar 4. Tampilan Website Sebelum Serangan

2. Tampilan Web Setelah Serangan

Setelah simulasi berjalan selama beberapa menit, situs web yang diuji menunjukkan tanda-tanda kegagalan layanan. Tampilan web berubah menjadi tidak dapat diakses, mengalami waktu muat yang sangat lama, atau menampilkan pesan kesalahan seperti ERR_CONNECTION_TIMED_OUT (Gambar 5). Hal ini menunjukkan bahwa server tidak mampu menangani beban koneksi lambat yang dikirimkan oleh SlowHTTPTest, sehingga layanan menjadi tidak tersedia bagi pengguna sah.

Temuan ini sejalan dengan hasil penelitian sebelumnya[13], yang menunjukkan bahwa peningkatan lalu lintas jaringan akibat serangan DoS dapat menyebabkan gangguan layanan dalam waktu singkat. Meskipun terdapat sedikit peningkatan waktu respons, situs tersebut tidak mengalami downtime atau kesalahan sistem. Hal ini mengindikasikan adanya perlindungan seperti pembatasan koneksi per-IP, pengaturan timeout yang ketat, serta penerapan firewall dan CDN yang efektif dalam memutus koneksi mencurigakan sebelum membebani server.



Gambar 5. Tampilan Website Setelah Serangan

Hasil pengujian ini menunjukkan bahwa serangan Slow HTTP DoS sangat efektif dalam menurunkan performa dan ketersediaan layanan web yang tidak memiliki konfigurasi pertahanan yang memadai.

Server yang rentan tidak mampu mengelola koneksi lambat dalam jumlah besar, sehingga permintaan sah dari pengguna lain gagal diproses. Fenomena ini memperkuat temuan [13], yang menyatakan bahwa serangan DoS dapat menyebabkan penurunan performa layanan secara drastis, memperlambat respons, dan meningkatkan risiko kesalahan sistem. Pembatasan koneksi per-IP dapat mencegah serangan dari satu sumber, namun serangan terdistribusi tetap menjadi tantangan yang memerlukan solusi lebih kompleks seperti firewall adaptif dan Content Delivery Network (CDN). Dokumentasi visual sebelum dan setelah serangan memperjelas dampak nyata dari SlowHTTPTest terhadap server web. Gambar-gambar yang disajikan menunjukkan perbedaan mencolok antara situs yang rentan dan yang memiliki ketahanan, baik dari sisi tampilan maupun respons sistem. Hal ini menegaskan pentingnya kesiapsiagaan dan penerapan strategi mitigasi yang tepat untuk menjaga ketersediaan layanan web di tengah ancaman serangan DoS yang semakin kompleks dan canggih.

4. KESIMPULAN

Pengujian kerentanan menggunakan slowhttptest dalam mode Slow Headers menunjukkan bahwa server web akademik https://www.cic.ac.id/ rentan terhadap serangan Denial of Service (DoS) Lapisan 7 jenis Slow HTTP Headers. Kerentanan ini divalidasi oleh dua temuan utama: pertama, kondisi kegagalan layanan berhasil dipertahankan selama 240 detik hanya dengan 247 koneksi lambat aktif, mengindikasikan batas koneksi bersamaan yang rendah; kedua, keberhasilan serangan dengan interval keep-alive 5 detik menunjukkan bahwa konfigurasi timeout penerimaan header terlalu permisif, memungkinkan koneksi parsial menghabiskan sumber daya. Kegagalan layanan yang terkonfirmasi melalui kesalahan ERR_CONNECTION_TIMED_OUT pada peramban eksternal menegaskan perlunya mitigasi segera. Rekomendasi mitigasi mencakup penyesuaian parameter waktu tunggu melalui modul khusus seperti mod_reqtimeout pada Apache, serta penerapan pertahanan berlapis seperti Web Application Firewall (WAF) untuk membatasi laju dan durasi koneksi lambat.

DAFTAR PUSTAKA

[1] E. Fatoyinbo and S. Shrestha, “ENHANCING IOT SECURITY IN SMART HOMES USING KALI LINUX FOR PENETRATION TESTING.”

[2] A. Afifah Rodhiyatun Nisa, Ananditto Daffa Wijayanto, Arya Prabudi Jaya Priana, and A. Setiawan, “Analisis Log Server untuk mendeteksi Serang DDoS pada Keamaan Jaringan di Website,” *Journal of Internet and Software Engineering*, vol. 1, no. 3, p. 17, Jun. 2024, doi: 10.47134/pjise.v1i3.2612.

[3] I. Rahmadaniar, D. A. A. Tondang, B. S. Fernando, and A. Setiawan, “Implementasi Firewall Menggunakan Iptables untuk Melindungi Server dari Serangan DDoS,” *Journal of Internet and Software Engineering*, vol. 1, no. 3, p. 10, Jun. 2024, doi: 10.47134/pjise.v1i3.2564.

[4] I. Nedyalkov and G. Georgiev, “Kali Linux – a simple and effective way to study the level of cyber security and penetration testing of power electronic devices,” *International Journal on Information Technologies and Security*, vol. 16, no. 2, pp. 103–114, Jun. 2024, doi: 10.59035/JMFY4876.

[5] Z. I. Sumayyah, S. D. S. Permana, M. Tsabit, and A. Setiawan, “Penerapan dan Mitigasi Teknik Slowloris dalam Serangan Distributed Denial-of-Service (DDos) terhadap Website Ilegal dengan Kali Linux,” *Journal of Internet and Software Engineering*, vol. 1, no. 2, p. 14, Jun. 2024, doi: 10.47134/pjise.v1i2.2694.

[6] S. A. Hameed Alazawi, A. A. Abdulhameed, G. M. Hassan, and M. Hassooni, “Comparative Study on Applications of Cybersecurity Tools for Kali Linux Operating System,” in *AIP Conference Proceedings*, American Institute of Physics, Sep. 2024. doi: 10.1063/5.0234136.

[7] M. R. Sumar, A. Wahid, and J. M. Parenreng, “conditions of the Creative Commons Attribution 4.0 (CC BY) International License. (http://creativecommons.org/licenses/by/4.0/). Sistem Keamanan Jaringan Terhadap Serangan DOS (Denial Of Service) Menggunakan Snort Dan Firewall Berbasis Linux OS (Network Security System Against DOS (Denial Of Service) Attacks Using Snort and Firewall Based on Linux OS).”

[8] H. Gunawan, A. Handijono, A. Putra, and A. Zein, “Spectrum: Multidisciplinary Journal SISTEM MONITORING SERANGAN DOS DENGAN METODE INTRUSION DETECTION SYSTEM (IDS) SNORT MENGGUNAKAN APLIKASI BERBASIS PYTHON PADA SISTEM OPERASI LINUX,” vol. 2, no. 3, 2025.

[9] Della Yunika Zebua, Carolina Sayangi Cahaya Waruwu, Kesadaran Zebua, Mardin Zai, Agusdamai Lase, and O. L. Ofel, “Simulasi Serangan DOS Menggunakan SLOWHTTPTEST,”

Jurnal Komputer Teknologi Informasi Sistem Informasi (JUKTISI), vol. 4, no. 2, pp. 409–419, Jul. 2025, doi: 10.62712/juktisi.v4i2.402.

[10] K. Ruswandi, M. R. Z. Pohan, K. V. Halim, and S. N. Neyman, "Strategi Pencegahan Efektif terhadap Serangan DDoS Slowloris menggunakan Kali Linux dan Linux Mint," *Journal of Technology and System Information*, vol. 1, no. 4, p. 11, Jun. 2024, doi: 10.47134/jtsi.v1i4.2645.

[11] A. Prana Walidin *et al.*, "KALI LINUX SEBAGAI ALAT ANALISIS KEAMANAN JARINGAN MELALUI PENGGUNAAN NMAP, WIRESHARK, DAN METASPLOIT," 2025.

[12] A. Noor, A. S. Aldafian, M. Tahir, M. Ersa, N. Firmansyah, and S. N. Khofifah, "Jurnal Restikom : Riset Teknik Informatika dan Komputer Implementasi IDS Wireshark untuk Deteksi Serangan DDoS SLOW HTTPS di Kali Linux," vol. 7, no. 2, pp. 148–161, 2025, [Online]. Available: https://restikom.nusaputra.ac.id

[13] Della Yunika Zebua, Carolina Sayangi Cahaya Waruwu, Kesadaran Zebua, Mardin Zai, Agusdamai Lase, and O. L. Ofel, "Simulasi Serangan DOS Menggunakan SLOWHTTPTEST," *Jurnal Komputer Teknologi Informasi Sistem Informasi (JUKTISI)*, vol. 4, no. 2, pp. 409–419, Jul. 2025, doi: 10.62712/juktisi.v4i2.402.

[14] W. Haniyah, M. C. Hidayat, Z. F. I. Putra, V. A. Pertama, and A. Setiawan, "Simulasi Serangan Denial of Service (DoS) menggunakan Hping3 melalui Kali Linux," *Journal of Internet and Software Engineering*, vol. 1, no. 2, p. 8, Jun. 2024, doi: 10.47134/pjise.v1i2.2654.