

LAPORAN PENGUJIAN KEAMANAN WEBSITE / APLIKASI

Nama Aplikasi	https://dev-siakri.gresikkab.go.id/
Pemilik Aplikasi	Bagian Organisasi
Kode Servicedesk	SD-69EEC08FA33B20.55544390
Tanggal Pengajuan	27 Apr 2026 08:49:03
Tanggal Pengujian	12 Mei - 29 Mei 2026 (hari kerja)

I. INFORMASI TARGET

Main URL	https://dev-siakri.gresikkab.go.id/
Target URL	https://dev-siakri.gresikkab.go.id/
OS	Ubuntu 24.04
Bahasa Pemrograman	PHP 8.3
Framework	Laravel 9
Web Server	Apache 2.4.58
Pengembang	Dinotech

II. HASIL PENGUJIAN KEAMANAN APLIKASI :

(sebagaimana terlampir)

III. KESIMPULAN DAN SARAN :

- a) Kesimpulan :
1. Terdapat celah "Content Security Policy (CSP) Header Not Set" terjadi saat situs web tidak mengatur header keamanan konten yang mencegah serangan seperti XSS, sehingga menjadi rentan terhadap penyusupan skrip berbahaya.
 2. Phpmyadmin terekspos ke publik Celah keamanan akibat phpMyAdmin terekspos ke publik terjadi ketika antarmuka manajemen database MySQL/MariaDB ini dapat diakses langsung melalui internet tanpa pembatasan. Kondisi tersebut sangat berbahaya karena phpMyAdmin menyediakan fitur administrasi penuh, termasuk eksekusi query, pengelolaan user, dan

konfigurasi database. Jika tidak dilindungi dengan autentikasi yang kuat, pembatasan IP, atau firewall, penyerang dapat mencoba brute force login, mengeksploitasi kerentanan versi phpMyAdmin yang tidak diperbarui, atau bahkan langsung mengakses dan memodifikasi data sensitif. Akibatnya, risiko kebocoran data, manipulasi sistem, hingga pengambilalihan penuh database menjadi sangat tinggi.

3. Celah cookie tanpa flag keamanan terjadi ketika sebuah aplikasi web tidak menetapkan atribut penting seperti HttpOnly, Secure, atau SameSite pada cookie yang digunakan untuk menyimpan data sesi. Tanpa HttpOnly, cookie dapat diakses oleh JavaScript sehingga rawan dicuri melalui serangan XSS; tanpa Secure, cookie bisa terkirim melalui koneksi HTTP biasa dan berisiko disadap di jaringan tidak terenkripsi; sedangkan tanpa SameSite, cookie dapat dipicu lintas situs sehingga rentan terhadap serangan CSRF. Kondisi ini membuat data autentikasi dan sesi pengguna lebih mudah dieksploitasi oleh penyerang, yang dapat berujung pada pencurian identitas, pengambilalihan akun, hingga manipulasi transaksi sensitif.
4. muncul ketika file robots.txt yang seharusnya hanya berfungsi sebagai panduan bagi crawler mesin pencari dapat diakses publik tanpa pembatasan. Masalahnya, isi file ini sering berisi direktori atau path sensitif yang sengaja diblokir dari indeksasi, misalnya /admin/ atau /private/. Alih-alih melindungi, justru informasi tersebut bisa dimanfaatkan penyerang sebagai “peta” untuk menemukan area tersembunyi yang berpotensi berisi data penting atau fungsi administratif. Karena robots.txt bukan mekanisme keamanan, siapa pun yang mengetahui URL dapat mengakses direktori tersebut secara langsung, sehingga risiko eksploitasi, kebocoran data, dan serangan terhadap sistem meningkat signifikan.
5. Pengecekan VA ini bersifat sementara, tidak menutup kemungkinan dapat ditemukan celah-celah baru dengan teknik yang baru pula seiring dengan kemajuan teknologi yang cepat.
6. Pengecekan VA ini tidak menjamin aplikasi tidak terkena serangan/insiden siber.

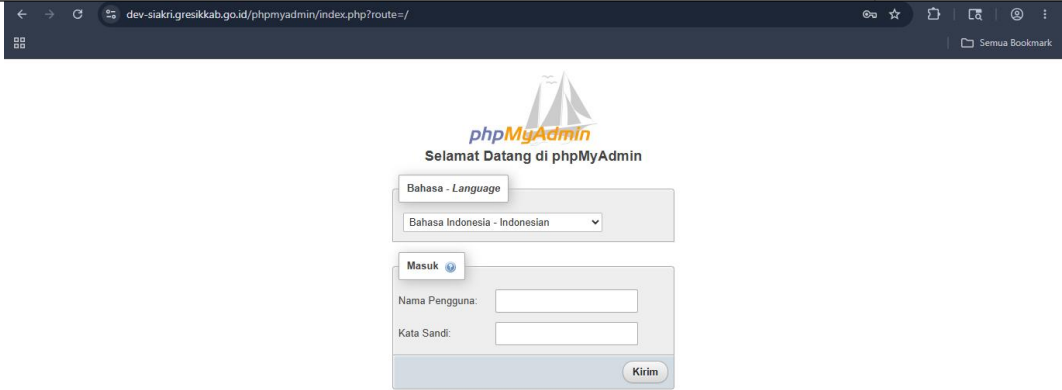
b) Saran :

1. Penambahan Header Content Security Policy (CSP) wajib yang diperlukan seperti :
 - X-Frame-Options
 - X-Content-TypeOptions
 - CSRF TokenPenambahan header diperlukan untuk mencegah serangan seperti “Click Jacking”, “XSS” dan “Drive by download attacks” pada header dari website atau aplikasi.
2. Lakukan pembatasan IP atau VPN, menempatkannya di belakang firewall atau reverse proxy dengan autentikasi tambahan, serta memastikan penggunaan kredensial yang kuat dan koneksi aman dengan HTTPS. Selain itu, URL default sebaiknya diubah agar tidak mudah ditebak, versi phpMyAdmin harus selalu diperbarui untuk menutup celah keamanan yang diketahui, dan layanan ini sebaiknya dinonaktifkan jika tidak digunakan secara rutin. Dengan kombinasi langkah-langkah tersebut, risiko eksploitasi oleh pihak tidak
3. Memastikan setiap cookie yang digunakan aplikasi web diberi atribut keamanan yang sesuai, yaitu HttpOnly
4. Tidak menaruh direktori atau path sensitif di dalam file tersebut, melainkan mengatur akses melalui mekanisme keamanan lain seperti autentikasi, firewall, atau konfigurasi server. Robots.txt sebaiknya hanya digunakan untuk mengarahkan crawler mesin pencari, bukan sebagai alat proteksi. Jika ada direktori yang benar-benar harus disembunyikan, gunakan kontrol akses yang tepat (misalnya .htaccess atau pengaturan permission) agar tidak bisa diakses publik. Dengan begitu, file robots.txt tetap berfungsi sesuai tujuannya tanpa membuka informasi yang bisa dimanfaatkan penyerang.
5. Pastikan bahwa web server, application server, load balancer, dll. diatur untuk menetapkan header Content-Security-Policy.
6. Melakukan perbaikan pada semua kolom inputan dan URL untuk dapat mencegah serangan XSS contoh pada menu “cari” masih dapat dilakukan serangan XSS
7. Memastikan tidak ada file penting yang dapat diakses oleh publik.

8. Melakukan pengecekan terhadap CORS domain misconfiguration.
9. Menerapkan manajemen sesi dengan keacakan token sesi dan menyetting masa berlaku sesi 15 menit
10. Menambahkan fitur penggantian password yang harus diubah 3 bulan sekali
11. Melakukan enkripsi / hashing pada penyimpanan data penting (dibuktikan dengan screenshot kodingan/database) dan menerapkan masking file(dibuktikan dengan screenshot server yang menyimpan file upload/database/kodingan) dan masking data (dibuktikan dengan screenshot tampilan data yang telah dimasking)
12. Membuat log aktivitas user
13. Aplikasi layak / **tidak layak** untuk dipublikasikan, harus segera diperbaiki. perbaikan harus sudah selesai sampai dengan tanggal **12 Juni 2026**.
14. Hasil pengecekan secara detail dapat dilihat pada file terlampir.
15. Setelah dilakukan perbaikan, agar memberikan surat laporan hasil perbaikan.
16. Jika tidak ada anggaran untuk perbaikan, mohon agar dianggarkan di papbd atau tahun berikutnya

HASIL PENGUJIAN KERENTANAN KEAMANAN / VULNERABILITY ASSESSMENT

NO	KERENTANAN KEAMANAN	LEVEL	DESKRIPSI	BUKTI DUKUNG
1.	Phpmyadmin ter ekspos ke publik	Tinggi	Permasalahan utama dari phpMyAdmin yang terekspos ke publik adalah terbukanya pintu akses langsung ke sistem manajemen database melalui internet tanpa kontrol yang memadai, sehingga siapa pun dapat mencoba masuk atau mengeksploitasi celah yang ada. Karena phpMyAdmin memiliki hak penuh untuk mengelola database, kondisi ini memungkinkan penyerang melakukan brute force terhadap kredensial, memanfaatkan bug atau kerentanan pada versi yang tidak diperbarui, hingga mengeksekusi query berbahaya untuk mencuri, mengubah, atau menghapus data sensitif. Dampaknya bukan hanya kebocoran informasi rahasia, tetapi juga potensi sabotase sistem, hilangnya integritas data, dan pengambilalihan penuh database yang dapat melumpuhkan layanan organisasi. dapat diatasi dengan membatasi akses hanya untuk administrator melalui pembatasan IP atau VPN, menempatkannya di belakang firewall atau reverse proxy dengan autentikasi tambahan, serta memastikan penggunaan kredensial yang kuat dan koneksi aman dengan HTTPS. Selain itu, URL default sebaiknya diubah agar tidak mudah ditebak, versi phpMyAdmin harus selalu diperbarui untuk menutup celah keamanan yang diketahui, dan layanan ini sebaiknya dinonaktifkan jika tidak digunakan secara	https://dev-siakri.gresikkab.go.id/phpmyadmin/index.php?route=/

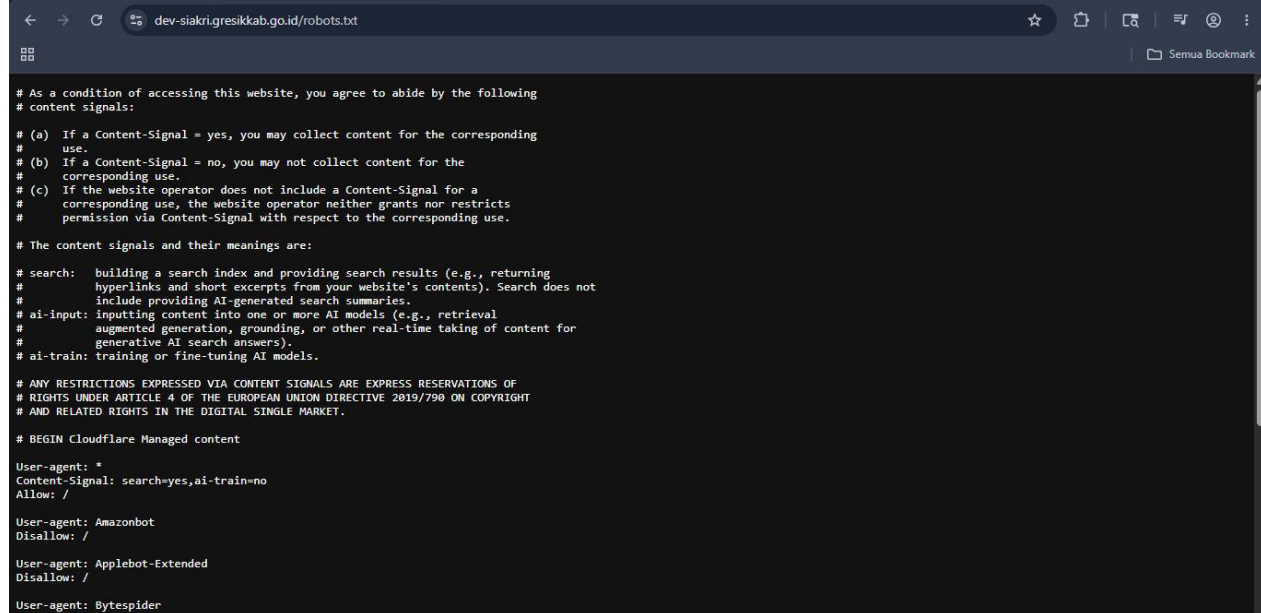
NO	KERENTANAN KEAMANAN	LEVEL	DESKRIPSI	BUKTI DUKUNG
			rutin. Dengan kombinasi langkah-langkah tersebut, risiko eksploitasi oleh pihak tidak berwenang dapat diminimalkan secara signifikan.	
				
2.	Content Security Policy (CSP) Header Not Set	Menengah	Celah Content Security Policy (CSP) Header Not Set terjadi ketika sebuah situs web tidak mengirimkan header Content-Security-Policy dalam respon HTTP, sehingga browser tidak memiliki aturan eksplisit untuk membatasi sumber konten yang boleh dimuat dan dijalankan. Kondisi ini membuat aplikasi rentan terhadap serangan seperti Cross-Site Scripting (XSS), data injection, atau pemanggilan skrip berbahaya dari domain tidak terpercaya, karena penyerang dapat menyisipkan kode berbahaya yang langsung dijalankan di sisi klien. Akibatnya, data sensitif pengguna dapat dicuri, integritas sistem terganggu, dan reputasi organisasi berisiko menurun. Untuk mencegah hal ini, administrator perlu mengonfigurasi	https://dev-siakri.gresikkab.go.id/

NO	KERENTANAN KEAMANAN	LEVEL	DESKRIPSI	BUKTI DUKUNG
			CSP dengan benar, misalnya menggunakan direktif default-src 'self' atau script-src yang hanya mengizinkan sumber terpercaya, sehingga aplikasi memiliki lapisan pertahanan tambahan terhadap eksploitasi berbasis konten. melakukan penambahan dan mengonfigurasi header CSP pada server agar browser hanya memuat konten dari sumber yang sah. Administrator dapat menetapkan aturan dasar seperti default-src 'self' untuk membatasi semua konten ke domain internal, serta menggunakan direktif spesifik seperti script-src, style-src, dan img-src untuk mengontrol sumber skrip, gaya, dan gambar. Selain itu, mode report-only dapat digunakan terlebih dahulu untuk memantau pelanggaran tanpa langsung memblokir, sehingga aturan bisa diuji dan disesuaikan sebelum diaktifkan penuh. Dengan konfigurasi CSP yang tepat dan pemeliharaan berkala, aplikasi web akan memiliki lapisan perlindungan tambahan terhadap serangan XSS, data injection, maupun eksekusi skrip berbahaya dari pihak tidak terpercaya.	

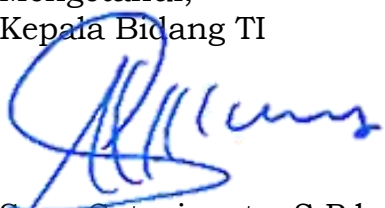
NO	KERENTANAN KEAMANAN	LEVEL	DESKRIPSI	BUKTI DUKUNG
			https://dev-siakri.gresikkab.go.id/ Scan ☐ Hide results ☒ Follow redirects Security Report Summary  Site: https://dev-siakri.gresikkab.go.id/ IP Address: 104.26.6.83 Report Time: 28 May 2026 22:42:27 UTC Headers: ✖ Strict-Transport-Security ✖ Content-Security-Policy ✖ X-Frame-Options ✖ X-Content-Type-Options ✖ Referrer-Policy ✖ Permissions-Policy Advanced: Ouch, you should work on your security posture immediately: Start Now	
3.	cookie tanpa flag keamanan	Menengah	Permasalahan cookie tanpa flag keamanan adalah bahwa data sesi pengguna yang tersimpan dalam cookie menjadi sangat rentan terhadap berbagai bentuk serangan karena tidak adanya pembatasan teknis yang melindungi cara cookie diakses dan dikirim. Tanpa flag HttpOnly, cookie dapat dibaca oleh JavaScript sehingga mudah dicuri melalui serangan XSS; tanpa flag Secure, cookie bisa terkirim melalui koneksi HTTP biasa yang tidak terenkripsi dan berisiko disadap oleh pihak ketiga; sedangkan tanpa flag SameSite, cookie dapat dipicu lintas situs sehingga rawan terhadap serangan CSRF. Kondisi ini membuka peluang besar bagi penyerang untuk mencuri identitas pengguna, mengakses akun tanpa izin, atau melakukan manipulasi transaksi sensitif, sehingga melemahkan integritas dan kerahasiaan sistem secara keseluruhan. memastikan setiap cookie yang digunakan aplikasi web diberi atribut keamanan yang sesuai, yaitu	https://dev-siakri.gresikkab.go.id/*

NO	KERENTANAN KEAMANAN	LEVEL	DESKRIPSI	BUKTI DUKUNG
			HttpOnly untuk mencegah akses JavaScript dan mengurangi risiko pencurian melalui XSS, Secure agar cookie hanya dikirim melalui koneksi HTTPS sehingga tidak bocor di jaringan tidak terenkripsi, serta SameSite untuk membatasi pengiriman cookie lintas situs dan mencegah serangan CSRF. Selain itu, penting juga membatasi masa berlaku cookie agar tidak terlalu lama, serta hanya menyimpan informasi yang benar-benar diperlukan dalam cookie. Dengan penerapan kombinasi langkah tersebut, risiko pencurian sesi dan pengambilalihan akun dapat diminimalkan secara signifikan.	
			Bukti (Evidence) <pre>Set-Cookie: XSRF-TOKEN=eyJpdil6lnhSnFTMnRld2Q3bU...; expires=Fri, 29 May 2026 03:08:21 GMT; Max-Age=7200; path=/; samesite=lax [] MISSING: Secure flag [] MISSING: HttpOnly flag</pre> Proof of Concept (PoC) <pre># Langkah 1: Cek header Set-Cookie curl -sI https://dev-siakri.gresikkab.go.id/ grep -i set-cookie # Output aktual: # Set-Cookie: XSRF-TOKEN=eyJpdil6lmw45XF4... # Max-Age=7200; path=/; samesite=lax # [] Tidak ada 'Secure' -> dikirim via HTTP plaintext # [] Tidak ada 'HttpOnly' -> document.cookie bisa baca token # Langkah 2: Simulasi XSS theft di DevTools Console document.cookie # -> 'XSRF-TOKEN=eyJpdil6...' <- cookie terbaca dari JS!</pre>	
4.	robots.txt terekspos	Menengah	terjadi ketika file robots.txt yang seharusnya hanya berfungsi sebagai panduan bagi crawler mesin pencari dapat diakses publik dan berisi daftar direktori atau path yang sengaja diblokir dari indeksasi, seperti /admin/, /private/, atau /config/.	https://dev-siakri.gresikkab.go.id/robots.txt

NO	KERENTANAN KEAMANAN	LEVEL	DESKRIPSI	BUKTI DUKUNG
			Alih-alih melindungi, informasi ini justru memberi petunjuk kepada penyerang tentang lokasi sensitif yang mungkin menyimpan data penting atau fungsi administratif. Karena robots.txt bukan mekanisme keamanan, direktori yang tercantum tetap dapat diakses langsung melalui URL, sehingga membuka peluang eksploitasi, pencurian data, atau serangan terhadap area tersembunyi aplikasi. Kondisi ini memperbesar risiko kebocoran informasi dan melemahkan keamanan sistem secara keseluruhan. Pastikan tidak menaruh direktori atau path sensitif di dalam file tersebut, melainkan mengatur akses melalui mekanisme keamanan yang tepat seperti autentikasi, firewall, atau konfigurasi server agar direktori penting tidak bisa diakses publik. Robots.txt sebaiknya hanya digunakan untuk mengarahkan crawler mesin pencari, bukan sebagai alat proteksi. Jika ada direktori yang harus benar-benar disembunyikan, gunakan kontrol akses seperti .htaccess, pengaturan permission, atau sistem otorisasi aplikasi. Selain itu, lakukan audit berkala terhadap isi robots.txt untuk memastikan tidak ada informasi yang dapat dimanfaatkan penyerang. Dengan langkah-langkah ini, robots.txt tetap berfungsi sesuai tujuannya tanpa membuka celah keamanan yang melemahkan sistem.	

NO	KERENTANAN KEAMANAN	LEVEL	DESKRIPSI	BUKTI DUKUNG
				

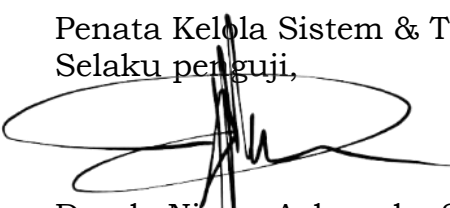
Mengetahui,
Kepala Bidang TI



Sony Caturiyanto, S.Pd

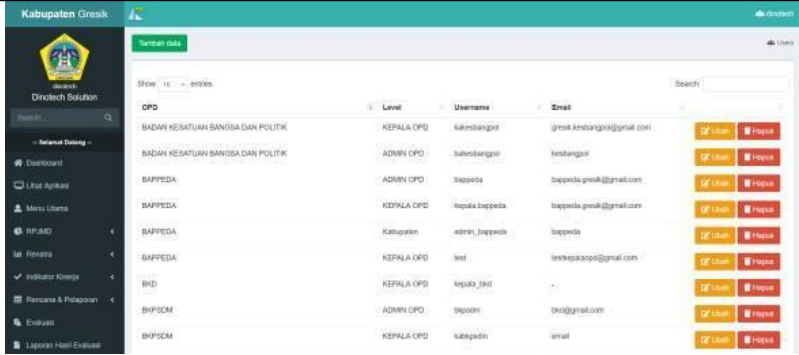
Gresik, 29 Mei 2026

Penata Kelola Sistem & TI
Selaku penguji,

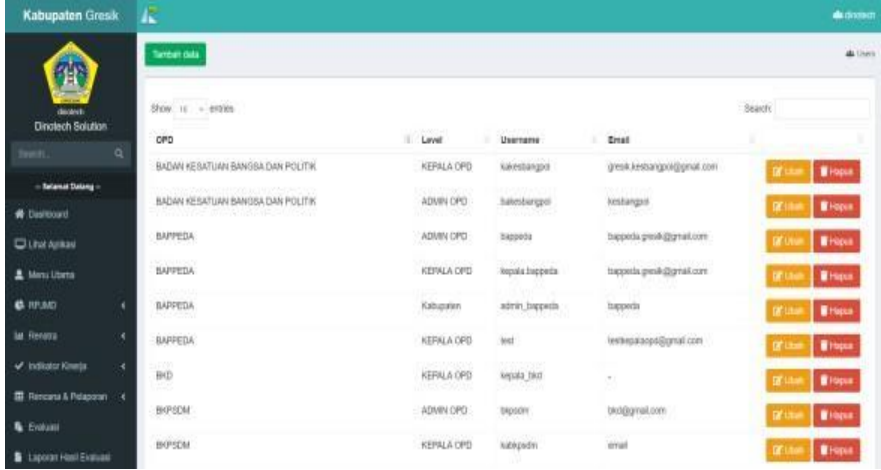


Dandy Nizam Achmady, S.Kom.

HASIL KESESUAIAN PENGUJIAN KEAMANAN APLIKASI DENGAN PROSEDUR KEAMANAN APLIKASI SPBE BERBASIS WEB

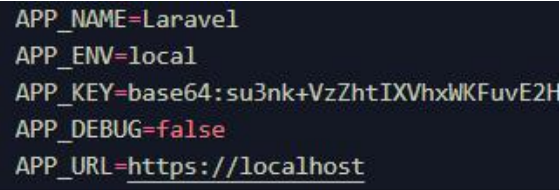
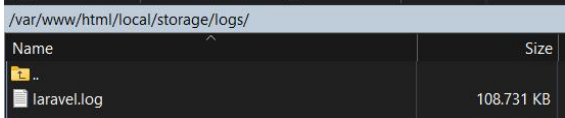
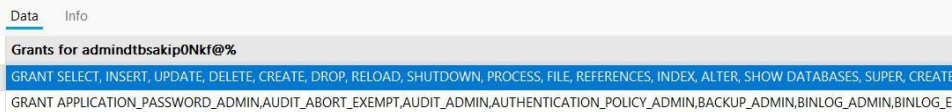
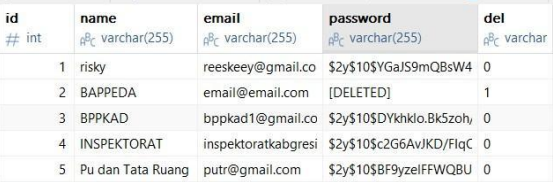
NO	PROSEDUR	KETERANGAN PROSEDUR	BUKTI DUKUNG
1	Pembuatan Akun Pengguna Aplikasi	Sistem menyediakan mekanisme pembuatan akun pengguna dengan password minimal 8 karakter terdiri dari huruf besar, huruf kecil, angka dan simbol.	
2	Pendaftaran Akun Pengguna	Fitur pendaftaran akun memvalidasi kekuatan password dan menerapkan captcha untuk mencegah pendaftaran otomatis oleh bot.	<pre>public function store(Request \$request) { \$validator = Validator::make(\$request->all(), ['username' => 'required max:255 unique:users', 'password' => ['required', Password::min(8) ->mixedCase() ->numbers() ->symbols(),],], ['password.confirmed' => 'Password dan konfirmasi password tidak sama.', 'password.min' => 'Password minimal 8 karakter.',]); if (\$validator->fails()) { return redirect('users/create') ->withErrors(\$validator) ->withInput(); } \$data = \$request->all(); \$data['password'] = bcrypt(\$data['password']); Users::create(\$data); return redirect('users'); }</pre>
3	Aktivasi Akun Pengguna	Sistem melakukan aktivasi akun melalui email atau OTP WhatsApp sebelum	Tambahkan Screenshot verifikasi link aktivasi akun melalui email/OTP

NO	PROSEDUR	KETERANGAN PROSEDUR	BUKTI DUKUNG
		akun dapat digunakan login.	
4	Validasi Akun Pengguna	Sistem menerapkan pembatasan percobaan login minimal 5 kali untuk mencegah brute force attack.	Belum ada pembatasan kesalahan login selama 3 x
5	Pemulihan Kata Sandi	Sistem menyediakan fitur lupa password dengan verifikasi melalui email atau nomor WhatsApp terdaftar.	Screenshoot bukti bahwa aplikasi sudah ada fitur lupa password melalui wa / email
6	Pengendalian Sesi	Sistem menerapkan manajemen sesi dengan masa berlaku maksimal 15 menit tanpa aktivitas (auto logout).	Di .env <pre>18 BROADCAST_DRIVER=log 19 CACHE_DRIVER=file 20 FILESYSTEM_DISK=local 21 QUEUE_CONNECTION=sync 22 SESSION_DRIVER=file 23 SESSION_LIFETIME=30</pre> Di config/session <pre>'lifetime' => env('SESSION_LIFETIME', 30), 'expire_on_close' => false,</pre>
7	Pemberian Hak Akses	Sistem menyediakan pengaturan hak akses berbasis role yang dikelola melalui halaman superadmin.	Disesuaikan per jenis pengguna

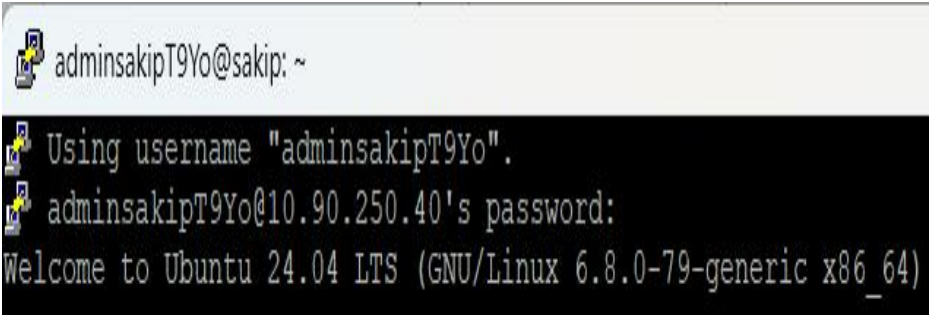
NO	PROSEDUR	KETERANGAN PROSEDUR	BUKTI DUKUNG
			
8	Pemeliharaan Akun Pengguna	Sistem melakukan review akun secara berkala termasuk penghapusan akun tidak aktif dan perubahan password minimal setiap 3 bulan.	Screenshoot bukti bahwa aplikasi sudah memperingatkan user untuk melakukan penggantian password minimal 3 bulan sekali ketika login
9	Penggunaan Captcha	Sistem menerapkan captcha pada form login, register atau input penting lainnya.	Screenshot penggunaan captcha pada form inputan penting
10	Manajemen Menu User	Superadmin dapat mengatur menu yang dapat diakses oleh masing-masing role pengguna.	Masih Di inject langsung di kode
11	Validasi Input	Sistem melakukan filtrasi dan sanitasi input untuk mencegah SQL Injection, XSS dan script berbahaya.	Memakai validasi input di setiap controller

NO	PROSEDUR	KETERANGAN PROSEDUR	BUKTI DUKUNG
			<pre><code>\$validated = \$request->validate(['opd_id' => 'required array', 'opd_id.*' => 'numeric', 'tahun' => 'required digits:4', 'name' => ['required', 'string', 'max:255', 'not_regex:/[<>()]/'], 'dataik' => ['required', 'string', 'not_regex:/<script\b[^\>]*(.*)<\script>/i'], 'jenis' => 'required numeric',], ['name.not_regex' => 'Nama tidak boleh mengandung karakter khusus seperti < > { }', 'dataik.not_regex' => 'Data IK tidak boleh mengandung script.',]);</code></pre>
12	Penggunaan Enkripsi	Sistem menggunakan algoritma enkripsi aman seperti SHA- 256 atau SHA- 512 dengan salt untuk melindungi data sensitif.	Pada input password password varchar(255) \$2y\$10\$YGaJS9mQBsW4/ejkioQSlmwkMY \$2y\$10\$yhjEwgwNCUKvi5ex2ICK9OIKLUmz \$2y\$10\$DYkhklo.Bk5zoh/eoJtRXOZvnsdq.H \$2y\$10\$c2G6AvJKD/FlqQZIH8UouHZviOP
13	Penggunaan Hashing	Sistem menggunakan hashing aman seperti Argon2 atau bcrypt untuk penyimpanan password.	Memakai hashing default dari Laravel

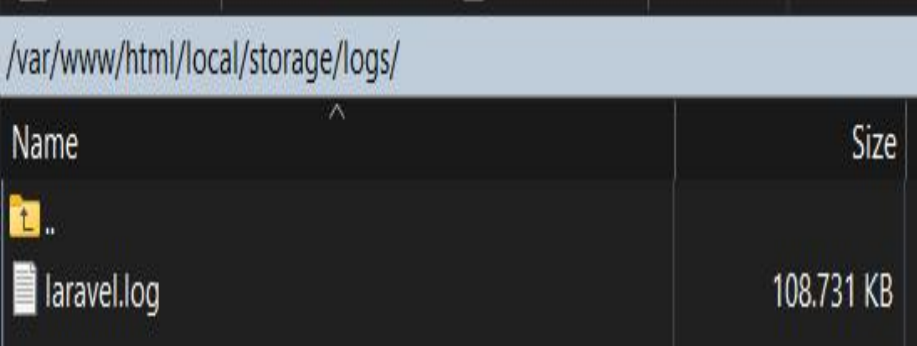
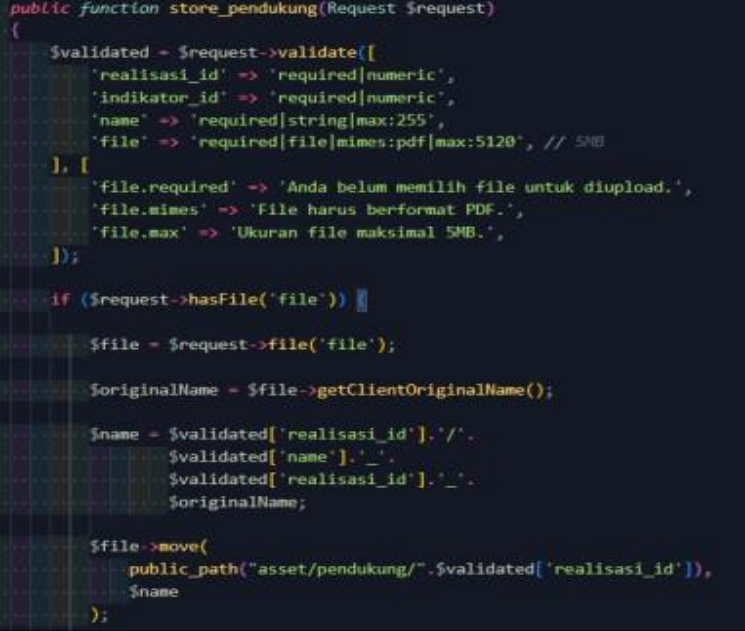
NO	PROSEDUR	KETERANGAN PROSEDUR	BUKTI DUKUNG
			password varchar(255) \$2y\$10\$YGaJS9mQBsW4/ejkioQSlemwkMY \$2y\$10\$yhjEwgwNCUKvi5ex2ICK9OIKLUmz \$2y\$10\$DYkhklo.Bk5zoh/eoJtRXOZvnsdq.H \$2y\$10\$c2G6AvJKD/FlqQZIH8UouHZviOP
14	Penggunaan Encoding	Sistem menerapkan encoding seperti Base64 atau HTML encoding untuk mencegah eksploitasi karakter khusus.	Ditambah validasi pada saat input data, dan penggunaan blade {{ }} untuk output data <pre>\$validated = \$request->validate(['opd_id' => 'required array', 'opd_id.*' => 'numeric', 'tahun' => 'required digits:4', 'name' => ['required', 'string', 'max:255', 'not_regex:/[<>{}]/'], 'dataik' => ['required', 'string', 'not_regex:/<script\b[^\>]*>(.*?)<\script>/i'], 'jenis' => 'required numeric',], ['name.not_regex' => 'Nama tidak boleh mengandung karakter khusus seperti < > { }', 'dataik.not_regex' => 'Data IK tidak boleh mengandung script.',]);</pre>

NO	PROSEDUR	KETERANGAN PROSEDUR	BUKTI DUKUNG
15	Penanganan Error	Sistem menerapkan error handling yang aman tanpa menampilkan informasi teknis server atau database.	Dimatikannya fitur debug di laravel 
16	Pencatatan Log	Sistem mencatat aktivitas penting seperti login, perubahan data dan akses informasi sensitif.	Ada log bawaan dari Laravel 
17	Pengelolaan Database	Sistem menerapkan database terstruktur, normalisasi data, dan pembatasan hak akses database.	Password disimpan sebagai hash Normalisasi menyesuaikan relasi tabel Hak akses database grant all 
18	Pemeliharaan Database	Sistem melakukan backup database berkala dan pengujian restore untuk mitigasi bencana.	Penghapusan dibuat 2 sistem, hapus temporary dan hapus permanen, hapus temp di tempatkan di kolom del = 1 agar otomatis tidak terlihat di aplikasi 

NO	PROSEDUR	KETERANGAN PROSEDUR	BUKTI DUKUNG
19	Penyimpanan File Upload	Sistem menerapkan pembatasan ekstensi file, ukuran file dan penyimpanan file di direktori khusus.	<pre>public function store_pendukung(Request \$request) { \$validated = \$request->validate(['realisasi_id' => 'required numeric', 'indikator_id' => 'required numeric', 'name' => 'required string max:255', 'file' => 'required file mimes:pdf max:5120', // 5MB], ['file.required' => 'Anda belum memilih file untuk diupload.', 'file.mimes' => 'File harus berformat PDF.', 'file.max' => 'Ukuran file maksimal 5MB.',]); if (\$request->hasFile('file')) { \$file = \$request->file('file'); \$originalName = \$file->getClientOriginalName(); \$name = \$validated['realisasi_id'].'/' . \$validated['name'].'_' . \$validated['realisasi_id'].'_' . \$originalName; \$file->move(public_path("asset/pendukung/".\$validated['realisasi_id']), \$name); } }</pre>
20	Pengaturan HTTP Header	Server menerapkan security header seperti CSP, HSTS, X- Frame- Options dan X- Content- Type- Options.	Belum menerapkan konfigurasi header

NO	PROSEDUR	KETERANGAN PROSEDUR	BUKTI DUKUNG
			https://sinopel.gresikkab.go.id/login Scan ☐ Hide results ☒ Follow redirects Security Report Summary F Site: https://sinopel.gresikkab.go.id/login IP Address: 172.67.75.136 Report Time: 12 May 2026 21:36:35 UTC Headers: ✖ Strict-Transport-Security ✖ Content-Security-Policy ✖ X-Frame-Options ✖ X-Content-Type-Options ✖ Referrer-Policy ✖ Permissions-Policy Advanced: Ouch, you should work on your security posture immediately. Start Now
21	Pemasangan SSL/TLS	Website menggunakan sertifikat SSL/TLS untuk mengenkripsi komunikasi antara server dan pengguna.	Sudah dilakukan pemasangan sertifikat SSL
22	Penggunaan HTTPS	Seluruh akses dipaksa menggunakan HTTPS dengan redirect dari HTTP ke HTTPS.	Sudah menggunakan HTTPS https://dev-siakri.gresikkab.go.id/
23	Pembaruan Perangkat Lunak	Aplikasi menggunakan versi terbaru bahasa pemrograman, framework, database dan web server.	Ubuntu 24.04  PHP 8.3

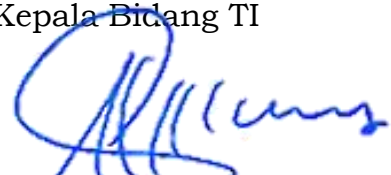
NO	PROSEDUR	KETERANGAN PROSEDUR	BUKTI DUKUNG
			<pre>adminsakipT9Yo@sakip:~\$ php -v PHP 8.3.6 (cli) (built: Jan 7 2026 08:40:32) (NTS) Copyright (c) The PHP Group Zend Engine v4.3.6, Copyright (c) Zend Technologies with Zend OPcache v8.3.6, Copyright (c), by Zend Technologies</pre> Laravel dan Library <pre>{ "name": "laravel/laravel", "type": "project", "description": "The Laravel Framework.", "keywords": ["framework", "laravel"], "license": "MIT", "require": { "php": "^8.0.2", "barryvdh/laravel-dompdf": "^3.1", "guzzlehttp/guzzle": "^7.2", "laravel/framework": "^9.19", "laravel/sanctum": "^3.0", "laravel/tinker": "^2.7", "laravel/ui": "^4.6", "laravelcollective/html": "^6.4", "maatwebsite/excel": "^3.1", "rorecek/laravel-ulid": "^2.0" },</pre>

NO	PROSEDUR	KETERANGAN PROSEDUR	BUKTI DUKUNG
24	Pengelolaan Log	Sistem melakukan penghapusan log lama secara otomatis dengan retensi minimal 3 bulan.	
25	Unggah File	Sistem melakukan validasi terhadap file yang diunggah meliputi ukuran, ekstensi dan metadata.	
26	Keamanan API	Sistem menerapkan keamanan API menggunakan API Key, JWT	Tidak Menggunakan API

NO	PROSEDUR	KETERANGAN PROSEDUR	BUKTI DUKUNG
		atau metode autentikasi lainnya.	
27	Konfigurasi Keamanan Server	Server menutup port yang tidak digunakan dan hanya membuka port standar seperti 80 dan 443.	<pre> adminsakipT9Yo@sakip:~\$ sudo ufw status [sudo] password for adminsakipT9Yo: Status: active To Action From -- Apache Full ALLOW Anywhere 19808 ALLOW Anywhere 80 ALLOW Anywhere 443 ALLOW Anywhere 3306 ALLOW Anywhere 5666 ALLOW Anywhere 9100 ALLOW Anywhere Apache Full (v6) ALLOW Anywhere (v6) 19808 (v6) ALLOW Anywhere (v6) 80 (v6) ALLOW Anywhere (v6) 443 (v6) ALLOW Anywhere (v6) 3306 (v6) ALLOW Anywhere (v6) 5666 (v6) ALLOW Anywhere (v6) 9100 (v6) ALLOW Anywhere (v6) adminsakipT9Yo@sakip:~\$ </pre> Port 5666 dan port 9100 masih terbuka, port terbuka hanya untuk yang digunakan saja
28	Penghapusan File/Fitur	Sistem melakukan penghapusan fitur atau file yang tidak digunakan melalui proses backup dan verifikasi.	Sudah dihapus untuk beberapa fitur yang dimatikan/tidak dipakai, dengan penghapusan source code, route url, serta library yang sudah usang/tidak dipakai
29	Perlindungan Data Pribadi	Sistem menerapkan data masking pada data pribadi seperti NIK, nomor telepon dan email.	Belum menerapkan masking data

NO	PROSEDUR	KETERANGAN PROSEDUR	BUKTI DUKUNG
30	Pemasangan Wazuh Agent	Server aplikasi telah dipasang agent Wazuh untuk monitoring keamanan.	instalasi agent wazuh akan dibantu oleh Kominfo
31	Integrasi TTE	Aplikasi yang menggunakan TTE telah memenuhi ketentuan integrasi BSrE seperti penggunaan SSL, passphrase dan log transaksi.	Aplikasi Tidak Menerapkan TTE

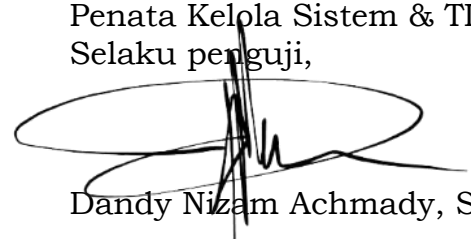
Mengetahui,
Kepala Bidang TI



Sony Caturiyanto, S.Pd

Gresik, 29 Mei 2026

Penata Kelola Sistem & TI
Selaku penguji,



Dandy Nizam Achmady, S.Kom.