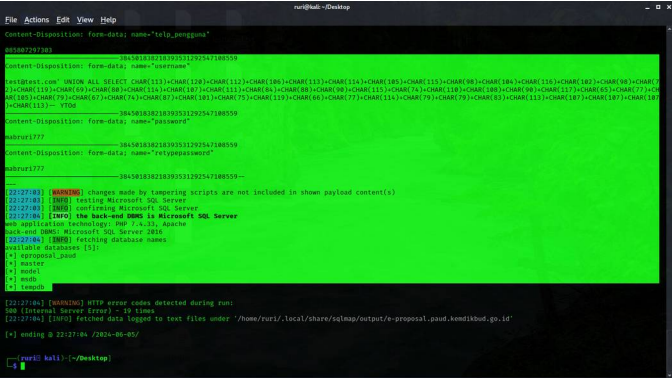


Laporan Pengujian atau Hasil Pencarian Celah Keamanan oleh Peserta Bug Bounty 2024

No.	Poin (Indikator)	Hasil
1.	Nama Target	PAUDPEDIA (aplikasi pendukung e-Proposal)
2.	Nama Kerentanan	SQL Injection
3.	Jenis Kerentanan	A03 Injetion
4.	Deskripsi Kerentanan	Saya ingin melaporkan penemuan saya terkait kerentanan SQL injection yang signifikan pada bagian Aplikasi pendukung PAUDPEDIA yaitu e-Proposal Paud. Setelah melakukan serangkaian pengujian, saya berhasil mengeksploitasi kerentanan ini dan mendapatkan akses tidak sah ke dalam database yang digunakan oleh Aplikasi pendukung tersebut. Ini adalah masalah yang sangat mengkhawatirkan dan membutuhkan perhatian segera dari tim pengembangan untuk memperbaikinya.
5.	Lokasi/URL	https://e-proposal.paud.kemdikbud.go.id/index.php/Clogin/tambahlembaga
6.	<i>IP Address Source</i> (IP Address Peserta)	125.166.118.135
7.	Dampak	Melalui penggunaan SQLmap, saya berhasil mengeksploitasi kerentanan SQL injection yang signifikan pada API aplikasi Pendukung PAUDPEDIA yaitu e-Proposal Paud kita. Ini memberikan akses tidak sah ke dalam

		<p>database, menghadirkan risiko serius terhadap keamanan sistem. Penyerang berpotensi untuk mengakses, mengubah, atau bahkan menghapus data sensitif yang disimpan dalam database kami. Tindakan pencegahan dan perbaikan segera harus diambil untuk memperkuat pertahanan sistem kami terhadap ancaman semacam ini.</p>
8.	<p>Langkah Penetrasi dan Tangkapan Layar (Screenshots) Temuan</p>	<ol style="list-style-type: none"> 1. Pada tahap awal, saya memulai upaya penetrasi pada sistem e-proposal dengan fokus pada bagian login. Saat melakukan uji coba pada parameter role, saya menemukan adanya indikasi kesalahan SQL, yang mengisyaratkan adanya potensi celah keamanan. Meskipun upaya SQL injection menggunakan tanda ' (kutip) belum berhasil, namun saya berhasil memperoleh error yang memberikan petunjuk mengenai jenis SQL server yang digunakan. Dari sini, saya memutuskan untuk kembali melakukan analisis dan pencarian potensi celah keamanan. 2. Setelah menemui kegagalan pada langkah sebelumnya, saya beralih untuk menguji parameter username dan password. Meskipun upaya injeksi pada parameter username tidak berhasil, saya berhasil menemukan celah pada parameter password yang memungkinkan injeksi SQL. Namun, perlu dicatat bahwa hasilnya adalah false-positive, yang menunjukkan perlunya pendekatan yang lebih cermat dalam proses ini. 3. Melihat adanya kegagalan dalam mencari celah pada proses login, saya memutuskan untuk mengalihkan fokus ke bagian form

		<p>tambah lembaga. Setelah melalui serangkaian uji coba terhadap berbagai parameter, akhirnya saya berhasil menemukan celah keamanan pada parameter username. Dengan menggunakan alat bantu sqlmap, saya berhasil mengeksploitasi celah tersebut dan mendapatkan akses yang diinginkan ke dalam sistem.</p> <p>Dengan begitu, langkah-langkah penetrasi yang saya jalankan mencakup eksplorasi menyeluruh terhadap berbagai aspek sistem e-proposal, dari proses login hingga form tambah lembaga. Tujuan akhirnya adalah memanfaatkan celah keamanan yang ditemukan untuk mendapatkan akses yang tidak sah ke dalam sistem.</p> 
9.	Rekomendasi	<ol style="list-style-type: none"> 1. Perbarui Sistem dan Perangkat Lunak Terkait: Pertama-tama, disarankan untuk memperbarui sistem e-proposal dan perangkat lunak terkait secara teratur. Ini termasuk pembaruan perangkat lunak

		<p>server database dan framework yang digunakan untuk mengurangi kemungkinan kerentanan yang terjadi karena kelemahan yang sudah diperbaiki dalam versi yang lebih baru.</p> <ol style="list-style-type: none"> 2. Implementasikan Filterisasi Input: Diperlukan implementasi filterisasi input yang ketat untuk mencegah serangan injeksi SQL di masa mendatang. Filterisasi input dapat dilakukan dengan memvalidasi dan membersihkan data masukan dari pengguna sebelum diizinkan masuk ke dalam perangkat lunak. 3. Penanganan Error yang Aman: Pastikan sistem memberikan pesan error yang tidak memberikan informasi yang sensitif kepada pengguna, seperti jenis database yang digunakan. Ini membantu mengurangi informasi yang dapat dieksploitasi oleh penyerang dalam proses penetrasi. 4. Pengujian Keamanan Rutin: Penting untuk menjalankan pengujian keamanan rutin secara berkala untuk mendeteksi dan mengatasi kerentanan baru yang mungkin muncul. Pengujian ini dapat mencakup pengujian penetrasi, pengujian kerentanan, dan audit kode secara menyeluruh. 5. Pelatihan Keamanan: Lakukan pelatihan keamanan kepada pengembang dan administrator sistem untuk meningkatkan pemahaman tentang praktik pengembangan perangkat lunak yang aman, termasuk tentang cara mencegah dan mengatasi serangan injeksi SQL. 6. Pemantauan Aktivitas Anomali: Implementasikan sistem pemantauan yang dapat mendeteksi aktivitas yang tidak biasa atau mencurigakan di dalam sistem, termasuk upaya serangan seperti injeksi SQL. Pemantauan ini dapat membantu dalam mendeteksi dan merespons serangan dengan cepat.
--	--	---

10.	Referensi	OWASP TOP TEN