

BAB V

PENUTUP

5.1 Kesimpulan

Berdasarkan hasil pengujian keamanan terhadap *website* web.bnpb.go.id/ppid menggunakan pendekatan OWASP Top 10 dan metodologi NIST SP 800-115, diperoleh beberapa temuan penting sebagai berikut:

1. Tingkat Kerentanan Sedang hingga Tinggi

Hasil pemindaian menunjukkan bahwa direktori web.bnpb.go.id/ppid memiliki skor rata-rata CVSS sebesar 6.6, yang dikategorikan dalam tingkat risiko *Medium to High*. Aspek *Integrity* menjadi yang paling rentan dengan nilai 6.9, disusul *Confidentiality* (6.5) dan *Availability* (6.4).

2. Kerentanan Utama yang Terdeteksi

- a. Beberapa celah keamanan yang berhasil diidentifikasi meliputi: *SQL Injection* (*Medium to High*): Ditemukan pada form input dengan potensi manipulasi basis data, meskipun dibatasi oleh proteksi WAF.
- b. *Directory Listing*: Beberapa path direktori dapat diakses publik tanpa autentikasi, membuka kemungkinan pengungkapan struktur sistem *file*.
- c. *Sensitive Data Exposure*: Terjadi akibat tidak adanya mekanisme CAPTCHA, rate-limiting, dan 2FA, memungkinkan serangan *brute-force login* berhasil dilakukan.

3. Konfigurasi Keamanan Web yang Lemah

Pengujian menunjukkan tidak adanya beberapa header keamanan penting seperti *Strict-Transport-Security*, *X-Frame-Options*, dan *X-Content-Type-Options*. Selain itu, *cookie* tidak dikonfigurasi dengan atribut *Secure* dan *HttpOnly*, meningkatkan risiko pencurian sesi.

4. Kelemahan dalam Infrastruktur SSL/TLS dan HTTP

Masih diaktifkannya protokol TLS 1.0 dan 1.1 serta *cipher* 64-bit membuka peluang terhadap serangan *downgrade* dan *cipher-based attacks*. Beberapa konfigurasi HTTP *server* juga mengungkapkan informasi versi perangkat lunak.

5. Absennya Mekanisme Pertahanan Tambahan

Website target belum menerapkan proteksi login tambahan seperti CAPTCHA, *lockout policy*, atau WAF rules yang lebih ketat. Hal ini membuka celah pada aspek *Availability* dan memperbesar potensi eksploitasi otomatis.

5.2 Saran

Untuk meningkatkan keamanan *website* web.bnpb.go.id/ppid dan memitigasi risiko dari kerentanan yang telah diidentifikasi, penulis menyarankan beberapa tindakan sebagai berikut:

1. Perbaikan Terhadap Celah Kerentanan Utama

- a. Terapkan validasi *input* dan penggunaan *prepared statements* pada seluruh form untuk mencegah *SQL Injection*.
- b. Nonaktifkan *directory listing* melalui konfigurasi *.htaccess* atau pengaturan *server* web (*Options -Indexes*).

- c. Pastikan setiap permintaan autentikasi memiliki CSRF Token untuk mencegah serangan *Cross-Site Request Forgery*.
2. Peningkatan Konfigurasi Keamanan *Server* dan Aplikasi
 - a. Nonaktifkan protokol TLS 1.0 dan TLS 1.1 serta *cipher suite* lemah dari konfigurasi SSL/TLS.
 - b. Terapkan *header* keamanan HTTP seperti *Strict-Transport-Security*, *X-Frame-Options*, dan *X-Content-Type-Options*.
 - c. Sembunyikan informasi versi perangkat lunak *server* melalui *Server Tokens Prod* dan *Server Signature Off*.
3. Penguatan Mekanisme *Autentikasi*
 - a. Implementasikan CAPTCHA, rate limiting, dan 2FA untuk memperkuat sistem login.
 - b. Lakukan audit berkala terhadap akun pengguna dan log aktivitas sistem untuk mendeteksi potensi serangan. Penggunaan *Firewall* dan Monitoring Keamanan
4. Penggunaan *Firewall* dan Monitoring Keamanan
 - a. Perkuat konfigurasi *Web Application Firewall* (WAF) untuk mendeteksi dan memblokir serangan berbasis *signature* maupun anomali.
 - b. Gunakan IDS/IPS serta sistem monitoring trafik guna mendeteksi pola serangan dini seperti *brute-force* dan DoS.
5. Evaluasi dan Uji Keamanan Berkala
 - a. Lakukan uji penetrasi secara periodik berdasarkan OWASP Top 10 terbaru.

- b. Gunakan pendekatan *defense in depth* dengan melibatkan tim keamanan internal dan pihak ketiga independen untuk meninjau ulang konfigurasi dan arsitektur sistem.

