BAB 5
KESIMPULAN DAN SARAN

5.1 Kesimpulan

Berdasarkan hasil pengujian dan analisis terhadap *Website Repository* UPN "Veteran" Jakarta yang dilakukan dengan mebandingkan hasil pemindaian menggunakan *Acunetix* dan *OWASP ZAP*, dapat ditarik beberapa kesimpulan sebagai berikut:

1. *Website Repository* UPN "Veteran" Jakarta masih berada pada tingkat keamanan *Medium* (menengah), yang ditunjukkan oleh temuan kerentanan dari hasil pemindaian *Acunetix* dan *OWASP ZAP* yang didominasi oleh risiko *Medium*. Meskipun demikian, tidak ditemukan kerentanan dengan tingkat risiko *high* maupun *critical*, sehingga tidak teridentifikasi ancaman langsung yang bersifat sangat kritis terhadap sistem.

2. Hasil pemindaian menggunakan *Acunetix* mengidentifikasi 15 kerentanan, dengan 12 di antaranya berada pada tingkat risiko *Medium*. Kerentanan yang paling dominan adalah *Insecure Transition from HTTP to HTTPS*, yang mengindikasikan bahwa penerapan mekanisme keamanan pada lapisan *transport layer* belum dilakukan secara konsisten pada seluruh halaman dan formulir. Sementara itu pemindaian menggunakan *OWASP ZAP* juga menemukan 15 kerentanan dengan variasi jenis yang lebih beragam antara lain *Missing Anti-Clickjacking Header, Content Security Policy (CSP) Not Set, Mixed Content, HTTP Strict Transport Security (HSTS) Not Set, serta Server Version Disclosure.*, mayoritas kerentanan yang ditemukan berada pada *OSI Layer 7* (*Application Layer*). Temuan ini menunjukkan bahwa konfigurasi *server* dan kebijakan keamanan *HTTP* yang diterapkan pada *Website Repository* UPN "Veteran" Jakarta belum sepenuhnya memenuhi standar keamanan web modern.

Dengan demikian, *Website Repository* UPNVJ memerlukan peningkatan keamanan secara komprehensif, yang mencakup perbaikan konfigurasi *server*, penerapan *header* keamanan modern, konsistensi penggunaan Protokol *HTTPS* pada seluruh layanan, serta

implementasi mekanisme proteksi formulir **y**ang memadai. Upaya tersebut diperlukan guna menjamin keamanan, integritas, dan keandalan sistem informasi akademik dalam menghadapi potensi ancaman keamanan siber yang terus berkembang.

3. Analisis dampak dan penilaian risiko terhadap *Repository* UPNVJ menunjukkan bahwa temuan utama diantaranya adalah transisi *HTTP to HTTPS* yang tidak konsisten, form tanpa proteksi *CSRF, header* keamanan yang hilang, dan *mixed content* yang menimbulkan risiko signifikan terhadap kerahasiaan, integritas, dan ketersediaan data akademik serta terhadap Sistem Informasi Akademik yang terintegrasi. Berdasarkan analisa, risiko tertinggi berkaitan dengan pencurian kredensial dan token SSO yang dapat menyebabkan manipulasi metadata, perubahan status akademik, atau akses tidak sah ke modul lain, sementara temuan pemindaian otomatis memberikan *baseline* penting untuk perbaikan teknis dan tata kelola namun tetap bersifat indikatif karena tidak disertai pengujian manual atau bukti eksploitasi terkontrol.

5.2 Saran

Berdasarkan hasil analisis kerentanan yang teridentifikasi pada situs *Repository* Universitas Pembangunan Nasional "Veteran" Jakarta, penelitian ini mengajukan beberapa rekomendasi strategis yang dapat diterapkan guna meningkatkan tingkat keamanan sistem secara menyeluruh, baik dari aspek konfigurasi aplikasi *web* maupun penguatan kebijakan keamanan, sebagaimana diuraikan sebagai berikut:

1. Penguatan keamanan pada lapisan transportasi data perlu menjadi prioritas utama. Seluruh komunikasi antara pengguna dan *server* harus dipastikan berlangsung melalui protokol *HTTPS* secara konsisten. Pengelola situs *Repository* disarankan untuk mengimplementasikan mekanisme *HTTP Strict Transport Security (HSTS)* guna memaksa *browser* klien menggunakan koneksi terenkripsi, sekaligus menutup seluruh kemungkinan akses melalui protokol *HTTP* yang masih terbuka.

2. Mekanisme proteksi formulir perlu diperkuat dengan penerapan token *Cross-Site Request Forgery (CSRF).* pada seluruh formulir yang mengubah status (*state*) aplikasi.

Token CSRF harus bersifat unik, acak, serta divalidasi pada sisi *server* untuk memastikan bahwa setiap permintaan yang diterima benar-benar berasal dari pengguna yang sah. Implementasi mekanisme ini dapat dilakukan secara manual maupun dengan memanfaatkan *library* keamanan seperti *OWASP CSRF Guard*, sehingga proses validasi dapat berjalan lebih stabil dan selaras dengan standar keamanan aplikasi *web* modern.

3. Pengelola *Repository* disarankan untuk mengaktifkan dan mengkonfigurasi berbagai *header* keamanan modern yang belum sepenuhnya diterapkan. *Header* keamanan seperti *X-Frame-Options* atau direktif *frame-ancestors* pada *Content Security Policy* (*CSP*) berperan penting dalam mencegah serangan *clickjacking*. Selain itu, penerapan *Content Security Policy* secara komprehensif diperlukan untuk membatasi sumber konten yang diizinkan dimuat oleh perangkat, sehingga potensi terjadinya serangan *Cross-Site Scripting* (*XSS*) dapat diminimalkan.

4. Perbaikan pada konfigurasi *server* perlu dilakukan untuk meminimalisir risiko eksploitasi akibat terungkapnya informasi yang tidak diperlukan. Informasi versi perangkat lunak *server*, seperti *Apache* atau sistem operasi yang digunakan, sebaiknya disembunyikan guna mengurangi kemungkinan *fingerprinting* oleh pihak penyerang. Selain itu, setiap respons *server* harus dilengkapi dengan *header Content-Type* yang sesuai untuk mencegah praktik *MIME-sniffing* oleh peramban yang berpotensi mengeksekusi konten berbahaya. Komentar *HTML* yang memuat informasi internal perlu dihapus sebelum aplikasi dipublikasikan, serta konsistensi kode karakter antara *header* dan isi respons harus dijaga dengan menerapkan standar *UTF-8* secara menyeluruh.

5. Struktur dokumen *HTML* perlu disesuaikan dengan standar pengembangan web modern. Penggunaan *DOCTYPE HTML5* direkomendasi untuk mencegah terjadinya *relative path confusion* serta menghindari aktivasi *quirks mode* pada perangkat, yang dapat menyebabkan interpretasi *path* maupun konten secara tidak konsisten. Penerapan standar ini tidak hanya meningkatkan konsistensi dan kompatibilitas lintas peramban, tetapi juga berkontribusi terhadap peningkatan keamanan dalam proses perenderan halaman *web*.

6. Pengujian keamanan perlu dilakukan secara berkala untuk memastikan *Repository* tetap berada dalam kondisi aman seiring dengan dinamika dan perkembangan ancaman siber. Pemindaian kerentanan secara rutin menggunakan perangkat seperti *Acunetix*, *OWASP ZAP*, atau alat sejenis perlu dijadwalkan secara sistematis. Selain itu, pelaksanaan *penetration testing* secara berkala, misal tahunan, sangat dianjurkan untuk mengidentifikasi potensi kerentanan baru yang muncul akibat pembaruan sistem atau perubahan konfigurasi. Pengelola juga perlu menyusun dan menerapkan standar operasional prosedur (SOP) keamanan internal yang mencakup pemantauan *log* sistem, pembaruan perangkat lunak, serta mekanisme respons insiden, sehingga pengelolaan keamanan *Repository* dapat dilakukan secara proaktif dan berkelanjutan.
7. Penelitian *Repository* UPNVJ dengan menggunakan *Acunetix* dan *OWASP ZAP* masih memiliki sejumlah keterbatasan, seperti ketergantungan pada pemindaian otomatis yang hanya mampu mengidentifikasi kerentanan umum dan belum dapat menjangkau celah keamanan yang bersifat kompleks atau berkaitan dengan logika aplikasi. Ketiadaan pengujian *penetration testing* secara manual dapat berpotensi menemukan kerentanan lanjutan yang tidak terdeteksi oleh *scanner* otomatis, serta tidak disertakannya *proof of concept* (*PoC*) membuat temuan yang dihasilkan masih berupa indikasi potensi risiko dan belum dapat dibuktikan melalui eksploitasi nyata. Oleh sebab itu, penelitian ke depan disarankan untuk mengombinasikan pemindaian otomatis dengan pengujian manual agar hasil analisis keamanan menjadi lebih menyeluruh dan akurat.
8. Disarankan untuk memprioritaskan perbaikan teknis segera seperti memaksa seluruh lalu lintas ke *HTTPS* dan mengaktifkan *HSTS*, mengimplementasikan token *CSRF* dan validasi *server-side* pada semua form, serta menerapkan *header* keamanan modern (*CSP, X-Frame-Options, X-Content-Type-Options*) dan memperbaiki *mixed content* serta *SRI* untuk skrip eksternal; lengkapi langkah teknis dengan integrasi temuan ke dalam *ISMS* sesuai *ISO 27001* (*risk register*, *Risk Treatment Plan*), penerapan monitoring dan *logging* terpusat, *patch management*, serta program pelatihan keamanan, dan lakukan *penetration testing manual* serta *PoC* terkontrol untuk memverifikasi mitigasi dan menilai risiko residual sebelum menyatakan kepatuhan penuh.