UJIAN AKHIR SEMESTER
SIMULASI KEAMANAN CYBER – DVWA ANALYSIS

Oleh:

ARDIANSHAH ANANDA JANUAL DIMA	2231740024
MAULIDINA AYU PUTRI AMEILIA	2231740039
MUHAMMAD SU'ADI	2231740040

PROGRAM STUDI DIII TEKNOLOGI INFORMASI

JURUSAN TEKNOLOGI INFORMASI

POLITEKNIK NEGERI MALANG

KAMPUS LUMAJANG

2025

1. **Identifikasi Layanan Web Server Tools yang Digunakan:**

   Kami mengandalkan dua alat utama untuk mengintip informasi server:

   A. Nmap: Ini kayak detektor canggih yang bisa kasih tahu kami port mana saja yang terbuka di server. Dari situ, kami bisa tahu ada layanan apa saja yang jalan, misalnya HTTP (untuk website), SSH (untuk akses jarak jauh), atau MySQL (untuk database)

   B. Nikto: Kalau Nmap buat jaringan, Nikto ini khusus buat ngecek website-nya. Kami suruh Nikto "ngenongkrong" di alamat web DVWA (nikto -h http://[IP_TARGET]) buat cari tahu ada konfigurasi yang salah enggak, ada file rahasia yang lupa dihapus, atau kode-kode yang gampang dijebol.

   Apa yang kami temukan:

   Hal yang di dapatkan pada nmap

   A. port Terbuka: Nmap bilang port 22 (SSH), 80 (HTTP), dan 3306 (MySQL) itu dalam status "open" (terbuka)
   B. Layanan dan Versi: Di port 22, ada layanan SSH dengan versi OpenSSH 7.2p2. Di port 80 ada Apache httpd 2.4.18. Dan di port 3306 ada MySQL 5.7.20.
   C. Sistem Operasi: Nmap nebak sistem operasinya adalah Linux 4.4 - 4.10.
   D. Alamat MAC: Ini identitas unik dari kartu jaringan (NIC).

   Hal yang di dapatkan pada nikto

   A. servernya adalah Apache/2.4.18 (Ubuntu).
   B. Nikto langsung nunjukkin kalau header X-Frame-Options dan X-XSS-Protection tidak ada.
   C. Ada file /phpinfo.php yang bisa bocorin banyak info sistem, dan /test.php yang seharusnya dihapus.
   D. Nikto menemukan potensi Local File Inclusion di /index.php?page=include. Ini artinya penyerang bisa jadi memasukkan file lokal dari server ke dalam halaman web.
   E. Ada direktori /admin/ yang ditemukan, menunjukkan kemungkinan adanya halaman login admin yang bisa jadi target brute force.
   F.

2. **Penanganan Kerentanan**

   Setelah tahu di mana saja celahnya, langkah berikutnya adalah nutupin celah-celah itu. Ini ibarat pasang tameng dan benteng pertama buat ngelindungi sistem dari serangan.

Cara kami menangani serangan

A. Pengerasan Server (Bikin Server Lebih Kuat)

   a. Kami atur Apache biar enggak nampilin versi software-nya (ServerTokens Prod, ServerSignature Off). Biar penyerang enggak gampang nebak kelemahannya.

   b. Kami pasang lagi header keamanan yang hilang. Misalnya, Header always set X-Frame-Options DENY biar website enggak bisa di-klik curang, dan Header always set X-XSS-Protection "1; mode=block" biar browser otomatis blokir skrip jahat.

   c. Semua file uji coba atau bawaan yang enggak kepakai harus dihapus dari folder website.

   d. Pastikan pesan error yang muncul enggak bocorin detail sensitif sistem ke pengguna.

B. Pengaman aplikasi

   a. Setiap kali ada masukan dari pengguna, harus disaring dan dicek ketat (disebut input validation dan sanitization). Ini penting banget buat nyegah serangan kayak SQL Injection (nyisipin perintah database jahat) atau XSS (nyisipin kode script jahat).

   b. Pasang Pelindung CSRF: Ini kayak token rahasia buat mastiin kalau permintaan yang masuk itu beneran dari pengguna yang sah, bukan dari penyerang.

   c. ara paling ampuh adalah pakai prepared statements saat berinteraksi dengan database. Ini kayak ngasih daftar menu yang sudah fix, jadi penyerang enggak bisa nyisipin pesanan aneh-aneh.

   d. Setiap data yang ditampilkan ke pengguna harus di-encode dulu. Biar kalau ada script jahat, dia cuma dianggap teks biasa, bukan kode yang bisa jalan.

3. Pemindaian Keamanan Jaringan Tools

Setelah selesai sama website-nya, kami lanjut ngecek keseluruhan jaringan. Ini penting biar kami punya gambaran lengkap soal seberapa aman sistem secara keseluruhan.

Alat yang Kami Gunakan:

A. Nmap: Sekali lagi, Nmap jadi alat utama kami buat "nyisir" jaringan. Kami pakai buat nemuin komputer atau perangkat lain yang aktif di jaringan, terus ngecek semua port-nya secara detail. Bahkan, kami pakai skrip Nmap khusus buat nyari kerentanan yang udah umum.

B. Masscan: Kalau jaringannya gede banget, Masscan ini jagoannya. Dia bisa scan port dengan super cepat. (Meski buat DVWA yang cuma satu komputer, Nmap aja udah cukup).
C. Nessus/OpenVAS: Ini kayak "dokter spesialis" buat ngecek kerentanan sistem. Mereka bisa ngasih laporan detail banget tentang celah keamanan dan setelan yang salah di server dan perangkat jaringan. Kami juga memakai skrip Nmap khusus kerentanan (nmap --script vuln [IP_TARGET]) untuk pemeriksaan cepat.

Apa yang Kami Temukan di Jaringan

A. Port Terbuka Pasti Ada: Keberadaan port 22 (SSH), 80 (HTTP), dan 3306 (MySQL) itu pasti ada. Ini jadi pintu masuk yang jelas buat penyerang.
B. Versi Layanan Teridentifikasi: Kami tahu versi Apache 2.4.x, OpenSSH 7.x, dan MySQL 5.7.x yang dipakai. Kalau versinya jadul, itu bisa jadi masalah besar karena mungkin ada celah yang sudah diketahui publik.
C. Kerentanan Spesifik: Skrip Nmap atau alat kayak Nessus bisa nemuin kelemahan seperti software yang sudah usang atau pengaturan keamanan SSL/TLS yang lemah di port HTTPS (kalau ada).
D. Jaringan Polosan: Di lingkungan latihan kayak DVWA, seringnya cuma ada satu server tanpa pembagian jaringan yang jelas. Ini artinya, kalau satu server jebol, dampaknya bisa nyebar ke mana-mana.

4. Pengujian SSH Remote Access Tools yang Digunakan:

Cara kami ngetesnya:

Kami coba "dobrak" SSH pakai metode brute force, yaitu nyoba kombinasi username dan password terus-menerus sampai nemu yang pas.

A. Hydra: Ini alat andalan kami buat "dobrak" password. Kami siapin daftar username (-L userlist.txt) dan daftar password (-P passlist.txt), terus kami suruh Hydra nyoba semua kombinasinya ke SSH: hydra -L userlist.txt -P passlist.txt ssh://[IP_TARGET]. Kami juga coba username umum kayak 'admin' pakai daftar password paling sering dipakai (hydra -l admin -P /usr/share/wordlists/rockyou.txt ssh://[IP_TARGET]). Kadang kami atur juga biar enggak terlalu cepat nyobanya (-t 4 -w 30) biar enggak langsung ketahuan.
B. Medusa & Ncrack: Ini adalah alternatif Hydra yang juga efektif untuk cracking otentikasi jaringan.
C. Nyari Nama Pengguna: Kami juga coba cari cara buat tahu nama-nama pengguna yang sah di server SSH (misalnya, kalau ada celah yang bikin kami bisa nebak nama pengguna).

Hasil uji coba ssh

A. Password Lemah Ketahuan!: Kami berhasil nemuin username dan password yang gampang ditebak, kayak admin/admin, root/password, atau user/123456. Ini bahaya banget!
B. Akun Bawaan Masih Aktif: Beberapa akun bawaan server (kayak root dengan password lemah) ternyata masih aktif dan bisa dipakai.
C. Enggak Ada Pembatasan Percobaan Login: Layanan SSH ini enggak punya sistem yang ngebatesin jumlah percobaan login. Artinya, penyerang bisa nyoba password berkali-kali sampai berhasil.
D. Aturan Password Loyo: Enggak ada aturan yang ketat soal bikin password, jadi pengguna bisa pakai password yang gampang ditebak.

Cara cepat nutupin celah SSH

A. Ganti Semua Password Default: Ini adalah langkah pertama dan terpenting.
B. Pakai Kunci Rahasia, Bukan Password Biasa: Nonaktifkan login pakai password biasa (PasswordAuthentication no). Ganti pakai otentikasi berbasis kunci (SSH Key) yang jauh lebih aman (PubkeyAuthentication yes). Ini kayak pakai kunci ganda yang super aman.
C. Atur SSH Jadi Lebih Ketat (/etc/ssh/sshd_config):
    a. PermitRootLogin no: Jangan izinkan login langsung sebagai root.
    b. MaxAuthTries 3: Batasi jumlah percobaan login yang salah, misalnya maksimal 3 kali.
    c. ClientAliveInterval 300: Jaga sesi tetap hidup atau putus setelah idle tertentu.
D. Tindakan Pengaman Tambahan:
    a. Fail2Ban: Pasang ini biar otomatis ngeblok IP yang lagi nyoba brute force.
    b. Port Knocking: Ini trik buat "menyembunyikan" port SSH. Port-nya baru kebuka kalau ada urutan "ketukan" (akses port) yang benar.
    c. Akses Lewat VPN: Kalau bisa, semua akses SSH harus lewat VPN dulu.
    d. Verifikasi Ganda (MFA): Tambahin lapisan keamanan kedua, kayak kode dari HP, buat login SSH.

5. Kebijakan dan Prosedur Kebijakan yang Diperlukan:

Keamanan siber itu bukan cuma soal alat canggih atau settingan rumit, tapi juga soal aturan main dan kebiasaan orang-orangnya. Kebijakan dan prosedur yang jelas itu ibarat pondasi yang bikin bangunan keamanan jadi kokoh.

Kebijakan yang diperlukan

A. Aturan Password: Harus jelas! Minimal 12 karakter, harus campur huruf besar-kecil, angka, dan simbol. Ganti password setiap 90 hari, dan jangan pakai password yang sama dalam 12 bulan terakhir.
B. Aturan Akses: Orang cuma boleh akses apa yang dia butuhkan (prinsip least privilege). Hak akses harus dicek rutin (setiap 3 bulan). Kalau ada karyawan yang keluar, aksesnya harus langsung dicabut. Akun-akun penting harus pakai verifikasi ganda (MFA).
C. Manajemen Celah Keamanan: Harus ada jadwal pemindaian celah keamanan setiap bulan. Kalau ada celah yang bahaya banget (critical), harus diperbaiki dalam 48 jam. Yang bahaya tapi enggak sampai critical (high severity), perbaiki dalam 7 hari. Dan tes peretasan (penetration testing) harus dilakukan setahun sekali.
D. Respons Insiden: Harus ada tim dan prosedur jelas kalau terjadi serangan. Ada yang monitor keamanan 24/7, tahu gimana cara escalate kalau ada insiden, tahu cara nyimpen bukti digital, dan setelah insiden selesai, harus ada evaluasi buat belajar dari kesalahan.

SOP (Standart Operating Procedures)

A. Pengecekan Keamanan Harian: Rutinitas buat cek log atau peringatan keamanan.
B. Laporan Celah Mingguan: Rekap status celah dan progres perbaikannya.
C. Edukasi Keamanan Bulanan: Ngajarin karyawan tentang ancaman terbaru dan tips biar enggak kena tipu hacker.
D. Penilaian Keamanan Kuartalan: Penilaian lebih mendalam buat nemuin area mana lagi yang perlu diperbaiki.

6. Rekomendasi Peningkatan Keamanan Teknologi dan Tools:

kami mengusulkan peningkatan berkelanjutan dalam teknologi, arsitektur, dan kerangka kerja untuk membangun pertahanan siber yang tangguh dan bisa beradaptasi dengan ancaman yang terus berubah

teknologi dan alat yang lebih canggih

1. SIEM (Security Information and Event Management): Ini kayak pusat komando keamanan. Pakai Splunk atau ELK Stack buat ngumpulin semua log dan peringatan keamanan dari berbagai sumber, terus dianalisis biar bisa deteksi ancaman real-time yang lebih canggih.
2. Pengaman Jaringan Tingkat Tinggi: Menggunakan Next-Generation Firewall (NGFW) yang bisa "ngintip" isi data lebih dalam, dan Intrusion Detection/Prevention System (IDS/IPS) seperti Snort atau Suricata buat nangkal serangan langsung di jaringan.
3. Perlindungan Komputer Pengguna Komprehensif: Mengadopsi solusi EDR (Endpoint Detection and Response) seperti CrowdStrike atau SentinelOne buat ngelindungin komputer karyawan lebih canggih, dilengkapi dengan antivirus kelas atas dan enkripsi data (BitLocker/FileVault).

Mengintegrasikan standar keamanan global membantu menyelaraskan upaya keamanan:

1. NIST Cybersecurity Framework: Panduan komplit buat ngatur risiko keamanan (Identify, Protect, Detect, Respond, Recover).
2. ISO 27001: Standar internasional buat sistem manajemen keamanan informasi.
3. CIS Controls: Daftar kontrol keamanan yang sudah diprioritaskan buat panduan praktis.

Rencana Perjalanan Implementasi (Contoh Tahapan):

1. Fase 1 (Segera - 30 hari): Fokus utama adalah nutupin celah yang bahaya banget, bikin keamanan dasar, dan pasang alat monitoring yang penting.
2. Fase 2 (Jangka Pendek - 90 hari): Mulai pasang SIEM, mulai pisahin jaringan (segmentasi), dan kencengin lagi aturan akses.
3. Fase 3 (Jangka Panjang - 1 tahun): Bergerak ke arsitektur Zero Trust penuh, ningkatin kemampuan nyari ancaman yang tersembunyi, dan bikin proses keamanan otomatis.