**1. Menjelaskan Definisi Malware dan menyebutkan contoh sesuai dengan penyebarannya dan penyembunyiannya**

***Malware*** adalah singkatan dari malicious software, yaitu perangkat lunak yang dirancang untuk merusak atau mengakses sistem komputer tanpa izin. Tujuan utama malware dapat bervariasi, termasuk mencuri informasi pribadi, merusak sistem, menciptakan backdoor untuk akses ilegal, atau menyebabkan gangguan dalam operasi normal suatu sistem.

*- Virus*: Penyebaran: menyebar dengan cara menyisipkan dirinya pada program atau file yang sudah ada. Saat pengguna menjalankan program atau file tersebut, virus akan aktif dan dapat menyebar ke file lain di dalam sistem. Penyembunyian: Virus sering kali menyembunyikan dirinya di dalam file atau program yang tampaknya normal. Mereka dapat menggunakan teknik penyembunyian, seperti enkripsi atau polimorfisme, untuk menghindari deteksi oleh perangkat lunak keamanan.

*- Worm*: Penyebaran: Worm adalah jenis malware yang dapat menyebar sendiri tanpa bantuan pengguna atau host. Mereka menggunakan kelemahan dalam sistem atau protokol jaringan untuk menyebar ke komputer-komputer lain melalui jaringan. Penyembunyian: Worm cenderung menyembunyikan dirinya dengan menggandakan diri dan dapat menggunakan teknik enkripsi atau manipulasi file untuk mengelabui perangkat lunak keamanan.

*- Trojan Horse*: Penyebaran: Trojan horse menyembunyikan dirinya di dalam program atau file yang tampaknya berguna atau sah. Pengguna yang menjalankan program atau file tersebut tanpa menyadari akan membuka pintu bagi Trojan untuk menginfeksi sistem. Penyembunyian: TH dapat menggunakan teknik penyembunyian untuk menyamarkan dirinya sebagai aplikasi yang diinginkan oleh pengguna. Mereka tidak selalu menunjukkan tanda-tanda aktivitas berbahaya yang mencolok.

*- Ransomware*: Penyebaran: biasanya menyebar melalui email phishing, situs web yang disusupi, atau eksploitasi kelemahan keamanan. Setelah masuk ke sistem, ransomware mengenkripsi file pengguna dan menuntut pembayaran tebusan untuk mendapatkan kunci dekripsi. Penyembunyian: Ransomware dapat menyembunyikan aktivitasnya hingga saat enkripsi dimulai. Beberapa varian dapat menggunakan teknik enkripsi yang kuat untuk menyulitkan atau bahkan menghalangi upaya dekripsi tanpa kunci yang benar.

**2. Menjelaskan Layer Operating Sistem dan Sistem Jaringan**

***Layer Operating System*** adalah konsep lapisan keamanan dalam konteks sistem operasi. Ini mencakup berbagai tingkat proteksi dan kontrol yang disediakan oleh sistem operasi untuk melindungi integritas dan keamanan sistem. Beberapa komponen utama dalam Layer Operating System mencakup:

*- Kernel*: Fungsi: Menyediakan manajemen sumber daya hardware dan antarmuka antara hardware & software .Peran Keamanan: Memiliki kontrol penuh atas sumber daya sistem dan melindungi dari akses yang tidak sah atau modifikasi.

*- Driver*: Fungsi: Memungkinkan sistem operasi berkomunikasi dengan perangkat keras.

Peran Keamanan: Perlu dijamin keamanannya agar tidak dimanfaatkan oleh malware atau aplikasi yang tidak sah.

*- System Calls*: Fungsi: Antarmuka antara aplikasi dan kernel yang memungkinkan aplikasi meminta layanan dari kernel.

Peran Keamanan: Perlu diatur dengan ketat untuk mencegah akses yang tidak sah.

*- User Interface (UI):* Fungsi: Menyediakan antarmuka antara pengguna dan sistem operasi.

Peran Keamanan: Melibatkan kontrol akses, autentikasi pengguna, dan pencegahan terhadap serangan GUI.

***Layer Sistem Jaringan***: adalah konsep lapisan keamanan dalam konteks jaringan komputer. Ini melibatkan langkah-langkah dan teknologi untuk melindungi integritas dan keamanan komunikasi data di jaringan. Beberapa komponen utama dalam Layer Sistem Jaringan mencakup:

*- Perangkat Keras Jaringan*: Fungsi: Termasuk router, switch, firewall, dan perangkat keras jaringan lainnya.

Peran Keamanan: Mengontrol lalu lintas jaringan, memblokir akses yang tidak sah, dan mendeteksi ancaman keamanan.

*- Protokol Jaringan*: Fungsi: Menyediakan aturan dan standar untuk komunikasi di jaringan.

Peran Keamanan: Penggunaan protokol keamanan seperti SSL/TLS untuk melindungi data saat transit.

*- Firewall*: Fungsi: Mengontrol lalu lintas jaringan berdasarkan aturan keamanan.

Peran Keamanan: Memberikan perlindungan terhadap serangan jaringan dan mengamankan lalu lintas.

*- Enkripsi Data*: Fungsi: Melibatkan pengkodean informasi untuk melindungi kerahasiaan data.

Peran Keamanan: Melindungi data dari pencurian atau manipulasi selama transit.

**Perbedaan**:

*-* Layer Operating System fokus pada keamanan di tingkat sistem operasi dan interaksi dengan perangkat keras, sementara Layer Sistem Jaringan fokus pada keamanan di tingkat jaringan dan komunikasi data.

*-* Layer Operating System melibatkan konsep keamanan dalam menjalankan aplikasi dan proses pada suatu sistem, sedangkan Layer Sistem Jaringan berkaitan dengan keamanan data saat berpindah di antara perangkat dalam jaringan.

*-* Layer Operating System melibatkan kontrol akses terhadap sumber daya sistem, sedangkan Layer Sistem Jaringan melibatkan perlindungan lalu lintas dan komunikasi di tingkat jaringan.

**3. Menyebutkan Contoh Serangan Denial of Service** adalah jenis serangan yang bertujuan membuat sumber daya atau layanan tidak tersedia untuk pengguna yang sah. Berikut adalah beberapa contoh serangan Denial of Service:

- *Flood Attack:* Penyerang mengirimkan sejumlah besar permintaan ke server atau jaringan dengan tujuan melebihi kapasitas normalnya, menyebabkan penolakan layanan kepada pengguna yang sah. Contoh: Serangan Ping Flood, di mana penyerang mengirimkan sejumlah besar permintaan ping ke server untuk menghabiskan bandwidth dan sumber daya.

*- Amplification Attacks*: Penyerang menggunakan layanan atau protokol yang memungkinkan respon yang jauh lebih besar daripada permintaan yang dikirimkan, untuk memperbesar dampak serangan. Contoh: Serangan NTP Amplification, di mana penyerang mengirimkan permintaan ke server NTP (Network Time Protocol), dan server merespon dengan paket yang lebih besar.

*- Slowloris Attack*: Penyerang mencoba memanfaatkan server dengan mempertahankan banyak koneksi ke server tanpa pernah menyelesaikan koneksi atau mengirim data. Contoh: Serangan Slowloris membuat server sibuk menangani banyak koneksi yang terbuka, menghabiskan sumber daya server.

**4. Menjelaskan cara kerja serangan-serangan pada Web Security**

*- Injection Attacks*: Cara kerja: Penyerang menyisipkan data yang tidak valid atau berbahaya ke dalam input yang diharapkan oleh aplikasi web. Ini dapat menyebabkan eksekusi perintah atau query yang tidak sah. Contoh: SQL Injection, di mana penyerang menyisipkan perintah SQL yang tidak sah ke dalam formulir atau parameter URL untuk merusak atau mendapatkan akses ke basis data.

*- Cross-Site Scripting (XSS)*: Cara kerja: Penyerang menyisipkan skrip atau kode berbahaya ke dalam halaman web yang kemudian dijalankan oleh browser pengguna yang mengakses halaman tersebut. Contoh: Stored XSS, di mana skrip berbahaya disimpan di server dan dieksekusi setiap kali halaman tersebut dimuat oleh pengguna yang sah.

*- Cross-Site Request Forgery (CSRF)*: Cara kerja: Penyerang membuat pengguna yang sah melakukan tindakan yang tidak disengaja pada aplikasi web tanpa sepengetahuan atau izin mereka. Contoh: Penyerang mengirimkan tautan yang mengandung permintaan HTTP yang otomatis dijalankan oleh browser pengguna yang sudah login ke aplikasi target.

*- Security Misconfigurations*: Cara kerja: Penyerang memanfaatkan pengaturan atau konfigurasi yang salah pada server atau aplikasi web yang dapat memberikan akses tidak sah atau informasi sensitif. Contoh: Mencoba mengakses berkas atau direktori yang tidak diatur dengan benar, atau memanfaatkan pengaturan default yang tidak aman.

*- Session Hijacking:* Cara kerja: Penyerang mencoba mengakses atau mencuri sesi pengguna yang sudah terotentikasi untuk mendapatkan akses tanpa izin. Contoh: Menggunakan teknik seperti sniffing jaringan atau mencuri cookie sesi untuk mendapatkan akses ke akun pengguna.

*- Brute Force Attacks*: Cara kerja: Penyerang mencoba masuk ke akun dengan mencoba kombinasi username dan password secara berulang-ulang hingga berhasil. Contoh: Mengeksploitasi kelemahan kata sandi yang lemah dengan mencoba semua kemungkinan kombinasi atau menggunakan daftar kata sandi yang umum.

*- File Upload Exploits:* Cara kerja: Penyerang mencoba mengunggah file berbahaya ke server untuk mengeksploitasi kelemahan dalam penanganan file yang diunggah. Contoh: Mengunggah file eksekusi atau skrip berbahaya yang dapat digunakan untuk merusak server atau melakukan aksi tidak sah.

*- Clickjacking:* Cara kerja: Penyerang menutupi elemen halaman web dengan elemen transparan atau yang tidak terlihat sehingga pengguna tanpa sadar mengklik elemen yang tidak diinginkan. Contoh: Menyembunyikan tombol "Setuju" di bawah elemen halaman web lain sehingga pengguna tanpa sadar menyetujui sesuatu yang tidak diinginkan.

**5. Menjelaskan Proses Enkripsi dan Cara kerja Caesar Chiper**

***Proses Enkripsi***: Proses enkripsi melibatkan pengkodean atau transformasi data asli (plaintext) menjadi bentuk yang tidak dapat dibaca atau sulit diinterpretasikan tanpa pengetahuan kunci (ciphertext). Caesar Chipper adalah sala 1 algoritma enkripsi.

***Cara kerja Caesar chipper***: Setiap karakter dalam plaintext (teks/informasi yang belum dienkripsi/diubah.) digantikan oleh karakter yang terletak beberapa posisi di depannya dalam alfabet. Geseran ini ditentukan oleh kunci yang dipilih. Misalkan kita ingin menggunakan Caesar Cipher dengan kunci 3. Maka, setiap huruf dalam plaintext digeser ke depan sebanyak 3 langkah. Contoh: Plaintext: "HELLO", Enkripsi (geser 3): "KHOOR"

Dengan kunci yang berbeda, hasil enkripsinya akan berbeda pula. Kelemahan Caesar Cipher:

Caesar Cipher memiliki kelemahan utama, yaitu mudah diterobos dengan melakukan uji coba pada setiap kemungkinan geseran (brute-force). Karena hanya ada 25 kemungkinan geseran (kunci), serangan brute-force pada Caesar Cipher relatif mudah dilakukan.

Ciphertext: hasil dari proses enkripsi pada plaintext menggunakan suatu algoritma enkripsi & kunci tertentu.

**Contoh Serangan Denial of Service**

*Flood Attack*: Membanjiri sumber daya jaringan dengan lalu lintas yang tak terbatas.

*Ping of Death*: Mengirim paket ICMP yang berukuran abnormal untuk mengakibatkan crash pada sistem target.

*DDoS (Distributed DoS)*: Menggunakan banyak perangkat untuk melancarkan serangan DoS.

*- SYN/ACK Flood Attack*: Penyerang mengirimkan sejumlah besar permintaan koneksi (SYN) ke server tanpa pernah menyelesaikan tiga langkah koneksi TCP, yang menyebabkan penumpukan koneksi yang tidak terpakai di server. Contoh: Serangan SYN Flood, di mana penyerang mengirimkan banyak permintaan koneksi SYN tetapi tidak pernah menyelesaikan proses koneksi.

*- DNS Amplification Attack*: Penyerang memanfaatkan server DNS untuk mengirimkan permintaan ke server target dengan memanfaatkan mekanisme DNS yang memungkinkan permintaan kecil menghasilkan respon besar. Contoh: Penyerang mengirim permintaan DNS dengan alamat IP target yang dipalsukan, dan server DNS akan mengirimkan respon besar ke alamat target.

*- HTTP/S Flood Attack*: Penyerang mengirimkan sejumlah besar permintaan HTTP atau HTTPS ke server, mencoba menghabiskan bandwidth dan sumber daya server. Contoh: Serangan HTTP Flood, di mana penyerang menggunakan banyak komputer zombie atau bot untuk mengirim permintaan HTTP ke server target.

***Layer Operating System*** adalah konsep lapisan keamanan dalam konteks sistem operasi. Ini mencakup berbagai tingkat proteksi dan kontrol yang disediakan oleh sistem operasi untuk melindungi integritas dan keamanan sistem. Beberapa komponen utama dalam Layer Operating System mencakup:

*- Kernel*: Fungsi: Menyediakan manajemen sumber daya hardware dan antarmuka antara hardware & software .Peran Keamanan: Memiliki kontrol penuh atas sumber daya sistem dan melindungi dari akses yang tidak sah atau modifikasi.

*- Driver*: Fungsi: Memungkinkan sistem operasi berkomunikasi dengan perangkat keras.

Peran Keamanan: Perlu dijamin keamanannya agar tidak dimanfaatkan oleh malware / aplikasi yang tidak sah.

*- System Calls*: Fungsi: Antarmuka antara aplikasi dan kernel yang memungkinkan aplikasi meminta layanan dari kernel.

Peran Keamanan: Perlu diatur dengan ketat untuk mencegah akses yang tidak sah.

*- User Interface (UI):* Fungsi: Menyediakan antarmuka antara pengguna dan sistem operasi.

Peran Keamanan: Melibatkan kontrol akses, autentikasi pengguna, dan pencegahan terhadap serangan GUI.

***Layer Sistem Jaringan***: adalah konsep lapisan keamanan dalam konteks jaringan komputer. Ini melibatkan langkah-langkah dan teknologi untuk melindungi integritas dan keamanan komunikasi data di jaringan. Beberapa komponen utama dalam Layer Sistem Jaringan mencakup:

*- Perangkat Keras Jaringan*: Fungsi: Termasuk router, switch, firewall, dan perangkat keras jaringan lainnya.

Peran Keamanan: Mengontrol lalu lintas jaringan, memblokir akses yang tidak sah, dan mendeteksi ancaman keamanan.

*- Protokol Jaringan*: Fungsi: Menyediakan aturan dan standar untuk komunikasi di jaringan.

Peran Keamanan: Penggunaan protokol keamanan seperti SSL/TLS untuk melindungi data saat transit.

*- Firewall*: Fungsi: Mengontrol lalu lintas jaringan berdasarkan aturan keamanan.

Peran Keamanan: Memberikan perlindungan terhadap serangan jaringan dan mengamankan lalu lintas.

*- Enkripsi Data*: Fungsi: Melibatkan pengkodean informasi untuk melindungi kerahasiaan data.

Peran Keamanan: Melindungi data dari pencurian atau manipulasi selama transit.

***- Injection Attacks***: Cara kerja: Penyerang menyisipkan data yang tidak valid atau berbahaya ke dalam input yang diharapkan oleh aplikasi web. Ini dapat menyebabkan eksekusi perintah atau query yang tidak sah. Contoh: SQL Injection, di mana penyerang menyisipkan perintah SQL yang tidak sah ke dalam formulir atau parameter URL untuk merusak atau mendapatkan akses ke basis data.

***- Cross-Site Scripting (XSS)*:** Cara kerja: Penyerang menyisipkan skrip atau kode berbahaya ke dalam halaman web yang kemudian dijalankan oleh browser pengguna yang mengakses halaman tersebut. Contoh: Stored XSS, di mana skrip berbahaya disimpan di server dan dieksekusi setiap kali halaman tersebut dimuat oleh pengguna yang sah.

***- Cross-Site Request Forgery (CSRF)*:** Cara kerja: Penyerang membuat pengguna yang sah melakukan tindakan yang tidak disengaja pada aplikasi web tanpa sepengetahuan atau izin mereka. Contoh: Penyerang mengirimkan tautan yang mengandung permintaan HTTP yang otomatis dijalankan oleh browser pengguna yang sudah login ke aplikasi target.

*- Security Misconfigurations*: Cara kerja: Penyerang memanfaatkan pengaturan atau konfigurasi yang salah pada server atau aplikasi web yang dapat memberikan akses tidak sah atau informasi sensitif. Contoh: Mencoba mengakses berkas atau direktori yang tidak diatur dengan benar, atau memanfaatkan pengaturan default yang tidak aman.

*- Session Hijacking:* Cara kerja: Penyerang mencoba mengakses atau mencuri sesi pengguna yang sudah terotentikasi untuk mendapatkan akses tanpa izin. Contoh: Menggunakan teknik seperti sniffing jaringan atau mencuri cookie sesi untuk mendapatkan akses ke akun pengguna.

*- Brute Force Attacks*: Cara kerja: Penyerang mencoba masuk ke akun dengan mencoba kombinasi username dan password secara berulang-ulang hingga berhasil. Contoh: Mengeksploitasi kelemahan kata sandi yang lemah dengan mencoba semua kemungkinan kombinasi atau menggunakan daftar kata sandi yang umum.

*- File Upload Exploits:* Cara kerja: Penyerang mencoba mengunggah file berbahaya ke server untuk mengeksploitasi kelemahan dalam penanganan file yang diunggah. Contoh: Mengunggah file eksekusi atau skrip berbahaya yang dapat digunakan untuk merusak server atau melakukan aksi tidak sah.

*- Clickjacking:* Cara kerja: Penyerang menutupi elemen halaman web dengan elemen transparan atau yang tidak terlihat sehingga pengguna tanpa sadar mengklik elemen yang tidak diinginkan. Contoh: Menyembunyikan tombol "Setuju" di bawah elemen halaman web lain sehingga pengguna tanpa sadar menyetujui sesuatu yang tidak diinginkan.