VIVEKANAND EDUCATION SOCIETY INSTITUTE OF TECHNOLOGY(VESIT)

SEMESTER [IV]

OPERATING SYSTEM SECURITY

A Study on Encryption, Identity Management, and Threats

Class:- D10C

Submitted by:

Vanshika Khatri (Roll No: 45)

Prem Narayani (Roll No: 53)

Hitesh Nihalani (Roll No: 54)

Drishti Ochani (Roll No: 55)

Date: [25/03/2025]

Summary Report: Operating System Security

1. Introduction

Operating system security (OS security) involves protecting data, ensuring only authorized users have access, and maintaining system functionality. It helps prevent cyber threats, maintain user trust, and comply with security regulations.

2. Encryption in Operating Systems

Definition:

Encryption transforms readable data (**plaintext**) into unreadable form (**ciphertext**) to prevent unauthorized access.

Types of Encryption:

- **Symmetric Encryption** (Uses a single key for encryption and decryption, e.g., AES, DES).
- **Asymmetric Encryption** (Uses a public key for encryption and a private key for decryption, e.g., RSA, ECC).

How OS Uses Encryption:

- **File System Encryption:** Encrypts specific files (**BitLocker, FileVault**).
- **Full Disk Encryption (FDE):** Encrypts the entire storage device (**LUKS, VeraCrypt**).
- **Network Encryption:** Protects data during transmission (**TLS, SSL, IPsec**).

Challenges:

- Performance slowdowns due to encryption/decryption processes.
- Key management complexity—losing encryption keys can cause data loss.

3. Identity Management in Operating Systems

Definition:

Identity Management ensures secure access by verifying users and controlling permissions.

Key Components:

- **Authentication:** Confirms a user's identity (**passwords, biometrics, MFA**).
- **Authorization:** Grants specific access based on roles (**RBAC, ABAC**).
- **Auditing:** Logs user activities for security monitoring.

Technologies Used:

- **Single Sign-On (SSO):** Users log in once to access multiple services (**OAuth, SAML**).
- **Directory Services:** Centralized user authentication management (**LDAP, Active Directory**).
- **Privileged Access Management (PAM):** Restricts admin-level access (**sudo, CyberArk**).

Challenges:

- Managing multiple user accounts in large organizations.
- Balancing security with user convenience.

4. Advanced Security Mechanisms in OS

Access Control Models:

- **Mandatory Access Control (MAC):** OS strictly controls access (**SELinux, AppArmor**).
- **Discretionary Access Control (DAC):** Users define access permissions (**Unix file permissions**).
- **Role-Based Access Control (RBAC):** Permissions are assigned based on job roles.
- **Sandboxing:** Isolates applications to prevent security risks (**Docker, Chrome Sandbox**).

5. Threats and Vulnerabilities in OS Security

Common Threats:

- **Malware:** Viruses, rootkits, and ransomware steal or damage data.
- **Privilege Escalation Attacks:** Hackers gain admin-level access through vulnerabilities.
- **Insider Threats:** Employees misusing their access.

Common Vulnerabilities:

- **Buffer Overflows:** Attackers exploit memory allocation errors.
- **Misconfigured Permissions:** Granting unnecessary access can lead to security risks.
- **Unpatched Software:** Outdated OS versions can have security weaknesses.

Mitigation Strategies:

- **Regular Updates & Patching** (Fixes security vulnerabilities).
- **Intrusion Detection Systems (IDS/IPS)** (Detects unauthorized access).
- **Secure Coding Practices** (Reduces software vulnerabilities).

6. Emerging Trends in OS Security

Key Trends:

- **Zero Trust Architecture:** No user is automatically trusted; continuous verification is required.
- **Hardware-Based Security:** Using hardware encryption tools like **TPMs and Secure Enclaves** (Intel SGX).
- **AI & Machine Learning in Security:** AI detects unusual behavior and predicts security threats.
- **Quantum-Resistant Encryption:** Developing encryption that protects against future quantum attacks.

7. Case Studies: Security in Different OS

- **Windows:** Uses **BitLocker** for encryption, **Windows Defender** for malware protection, and **Credential Guard** for login security.
- **Linux:** Implements **SELinux** for access control, **AppArmor** for restricting apps, and **grsecurity** for enhanced kernel security.
- **macOS:** Uses **FileVault** for encryption, **Gatekeeper** to block unauthorized apps, and **System Integrity Protection (SIP)** for system file security.

8. Best Practices for OS Security

- **Keep OS updated** with the latest security patches.
- **Use strong encryption** for stored and transmitted data.
- **Apply the least privilege principle** (only necessary access should be granted).
- **Conduct security audits** to find vulnerabilities.
- **Educate users** on security best practices.

9. Key Terms to Remember

- **Ciphertext:** Encrypted (unreadable) data.
- **Plaintext:** Original (readable) data.
- **Public Key Infrastructure (PKI):** Manages encryption keys and certificates.
- **Multi-Factor Authentication (MFA):** Requires multiple verification steps for security.
- **Rootkit:** Malware that hides itself while controlling a system.

11. Conclusion

Operating System Security is crucial for protecting data, preventing cyber threats, and ensuring smooth system operations. Implementing advanced security measures like encryption, identity management, and regular updates

can significantly reduce vulnerabilities. As cyber threats evolve, continuous education and updated security practices remain essential.

12. References

1. ResearchGate: **"Operating System Security"**
2. IEEE: **"Some Issues Regarding OS Security"**
3. IJCRT: **"Security of Windows OS"**
4. arXiv: **"Synergia: Hardening High-Assurance Security Systems"**
5. arXiv: **"Spectre Attacks: Exploiting Speculative Execution"**