Secure Software Design Project

Course Code: CY321

Submitted by: **• Zain Ali Memon. 2022630 • Mohsin Wazir. 2022309 • Ibrahim Imran 2022301 • Sagar Kumar. 2022522**

Faculty: **CyberSecurity**

Threat Modeling & Risk Assessment

1. Identifying Attack Vectors

Attack vectors are potential paths that attackers can exploit to compromise the system. Based on the project's features, here are key attack vectors:

Component	Potential Attack Vectors
User Authentication	- Credential stuffing - Phishing attacks - MFA bypass
End-to-End Encryption	- Man-in-the-middle (MITM) attacks - Key compromise - Weak encryption implementation
Access Control (RBAC)	- Privilege escalation - Broken access control
File Storage & Transmission	- Data leakage via metadata - Unauthorized access to stored files
File Integrity & Anti-Tampering	- Digital signature forgery - Checksum collision attacks
Secure File Deletion	- Data recovery from deleted files

2. Risk Levels & Security Mitigation Strategies

Attack Vector	Risk Level	Mitigation Strategies
Credential Stuffing & Phishing	High	- Enforce strong password policies - Implement account lockout - Use phishing-resistant MFA (FIDO2, WebAuthn)
MFA Bypass	Medium	- Implement device fingerprinting - Use time-based OTPs (TOTP) instead of SMS
MITM Attacks	High	- Enforce TLS 1.3 - Use certificate pinning in client applications
Key Compromise	High	- Implement HSM for key storage - Use periodic key rotation
Weak Encryption Implementation	Medium	- Use AES-256 encryption - Conduct security audits
Privilege Escalation	High	- Implement least privilege principle - Conduct access control audits
Broken Access Control	High	- Implement server-side

		access validation - Perform role-based penetration testing
Data Leakage via Metadata	Medium	- Encrypt metadata - Mask metadata from unauthorized users
Unauthorized File Access	High	- Apply zero-trust principles - Implement granular access control policies
Digital Signature Forgery	Medium	- Use SHA-3, BLAKE2 - Ensure private keys remain confidential
Checksum Collision Attacks	Low	- Use SHA-256 or SHA-3
Data Recovery from Deleted Files	Medium	- Use secure deletion algorithms (DoD 5220.22-M, Gutmann method)

3. Summary & Security Best Practices

1. Zero-Trust Security Model: Continuously verify all access requests.

2. Regular Security Audits & Penetration Testing: Identify vulnerabilities proactively.

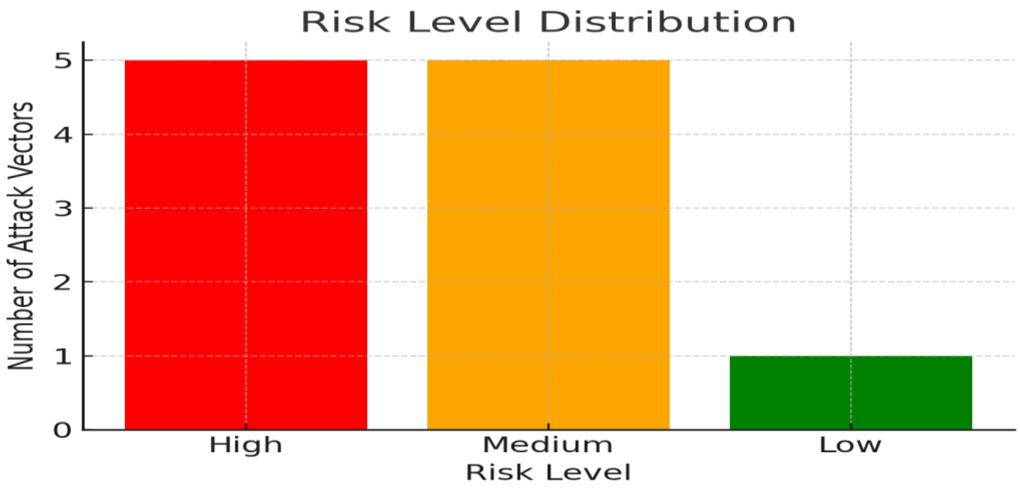
3. Strong Authentication & Authorization: Use MFA, OAuth 2.0, and strict RBAC policies.

4. Secure File Handling: Encrypt files at rest and in transit with AES-256 and TLS 1.3.

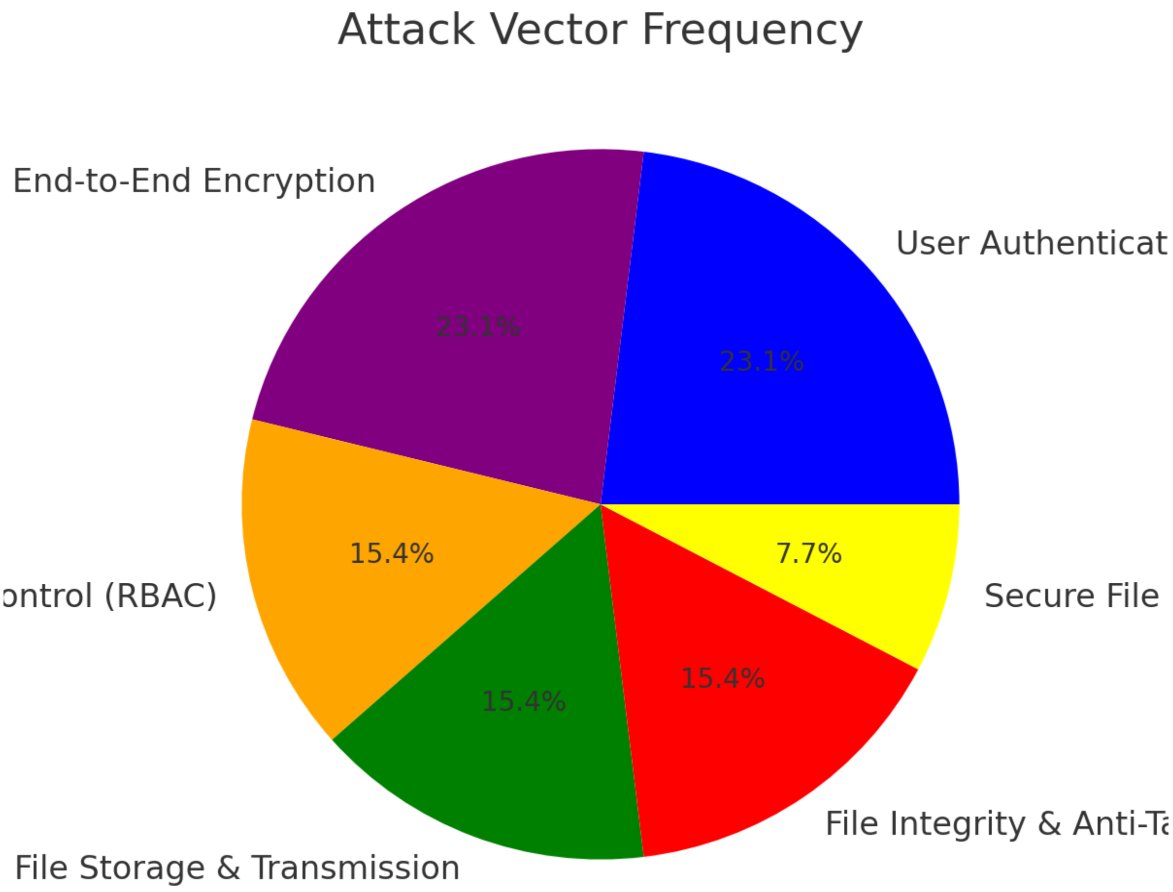
5. Activity Monitoring & Incident Response: Implement real-time anomaly detection.

4. Data Visualization

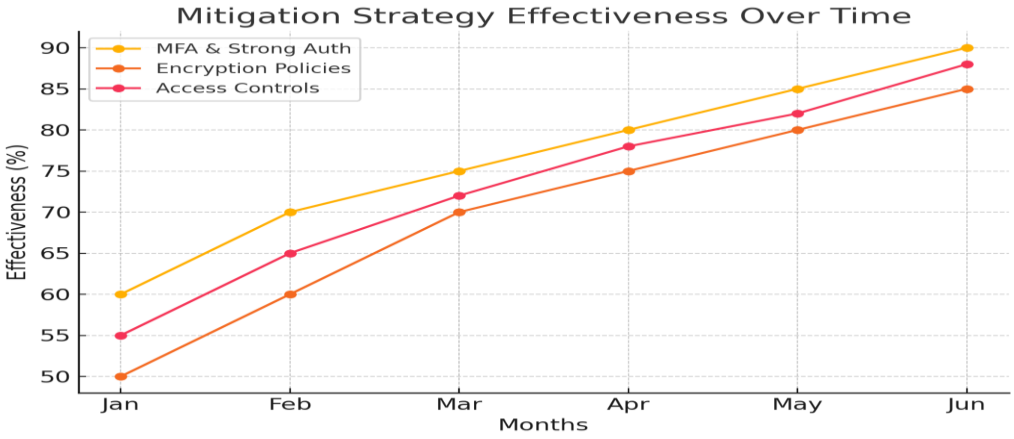
- **Risk Level Distribution:**



- **Attack Vector Frequency**



- **Mitigation Strategy Effectiveness:**



5. References

- Risk Level Distribution: Displays the number of attack vectors categorized as High, Medium, or Low risk.

- Attack Vector Frequency: Illustrates the proportion of different attack vectors in the system.

- Mitigation Strategy Effectiveness: Tracks the effectiveness of implemented security measures over a six-month period, modeled on estimated security improvements based on:

- NIST (National Institute of Standards and Technology) cybersecurity framework
- OWASP (Open Web Application Security Project) guidelines