# SAST Exercise Student Worksheet

**Name:** **\_\_\_\_** **Date:** **\_\_\_\_**

**Lab Partner:** **\_\_\_\_** **Section:** **\_\_\_\_**

## 🎯 Learning Objectives

By completing this worksheet, I will be able to:

* ☐ Understand what Static Application Security Testing (SAST) is and when to use it
* ☐ Execute automated security analysis using industry-standard tools (Bandit, Semgrep, Safety)
* ☐ Interpret SAST tool outputs and prioritize security findings by severity
* ☐ Apply remediation techniques to fix common security vulnerabilities
* ☐ Understand OWASP Top 10 vulnerabilities through hands-on code analysis
* ☐ Evaluate the business impact and ethical implications of security findings

## 📋 Exercise 1: SAST Fundamentals

### 1.1 Tool Availability Check

# Command run:  
python -c "from src.analyzer.static\_analyzer import SecurityToolRunner; print(SecurityToolRunner().check\_available\_tools())"  
  
# Results:  
Available tools: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

### 1.2 Basic Command Understanding

Fill in what each command flag does:

| Command Flag | Purpose |
| --- | --- |
| --educational | **\_\_\_\_** |
| --verbose | **\_\_\_\_** |
| --output json | **\_\_\_\_** |

### 1.3 Reflection Questions

1. **What is the main difference between SAST and DAST?**
2. **When in the development process should SAST be performed?**
3. **What are two limitations of SAST tools?** a) **\_\_\_\_**\_**\_** b) **\_\_\_\_\_\_**

## 📋 Exercise 2: Flask Application Analysis

### 2.1 Initial Analysis Results

# Command run:  
python src/analyzer/analyze\_cli.py samples/vulnerable-flask-app --educational  
  
# Findings Summary:  
Total: \_\_\_\_\_ | Critical: \_\_\_\_\_ | High: \_\_\_\_\_ | Medium: \_\_\_\_\_ | Low: \_\_\_\_\_

### 2.2 Vulnerability Classification

Fill in the table with findings from your analysis:

| Vulnerability Type | Count | Highest Severity | Example Line Number |
| --- | --- | --- | --- |
| SQL Injection | **\_** | **\_\_\_\_**\_\_\_\_\*\* | **\_\_\_\_** |
| XSS | **\_** | **\_\_\_\_**\_\_\_\_\*\* | **\_\_\_\_** |
| Debug Mode | **\_** | **\_\_\_\_**\_\_\_\_\*\* | **\_\_\_\_** |
| Authentication Issues | **\_** | **\_\_\_\_**\_\_\_\_\*\* | **\_\_\_\_** |
| Other: **\_\_\_\_**\_**\_\_\_\_** | **\_** | **\_\_\_\_**\_\_\_\_\*\* | **\_\_\_\_** |

### 2.3 Deep Dive Analysis

Choose ONE SQL injection finding and analyze it:

**File:** **\_\_\_\_**\_**\_\*\*** Line:\*\* \*\*\_\_\*\*\*\*\_\_\*\*\_\_\_\_\*\*

**Vulnerable Code Snippet:**

# Copy the vulnerable code here:

**Why is this vulnerable?**

**How could an attacker exploit this?**

**How would you fix it?**

# Write your secure code here:

### 2.4 Risk Assessment

**What is the MOST CRITICAL vulnerability you found and why?**

## 📋 Exercise 3: PWA Application Analysis

### 3.1 Multi-File Analysis Results

# Command run:  
python src/analyzer/analyze\_cli.py samples/unsecure-pwa --educational  
  
# Results:  
Files Analyzed: \_\_\_\_\_\_\_\_\_ | Total Findings: \_\_\_\_\_\_\_\_\_ | Tools Used: \_\_\_\_\_\_\_\_\_

### 3.2 PWA-Specific Security Issues

**What PWA-specific security concerns did you identify?**

**How do PWA security considerations differ from traditional web apps?**

### 3.3 Cross-Application Comparison

| Application | Total Findings | Highest Risk Vulnerability | Overall Risk Level (1-10) |
| --- | --- | --- | --- |
| Flask App | **\_\_\_\_**\_**\_\_\_\_** | **\_\_\_\_**\_\_\_\_\*\* | **\_\_\_\_**\_\_\_\_\*\* |
| PWA App | **\_\_\_\_**\_**\_\_\_\_** | **\_\_\_\_**\_\_\_\_\*\* | **\_\_\_\_**\_\_\_\_\*\* |

**Which application has the highest security risk and why?**

## 📋 Exercise 4: Advanced SAST Techniques

### 4.1 JSON Analysis Exercise

# Generate JSON reports  
python src/analyzer/analyze\_cli.py samples/vulnerable-flask-app --output flask\_report.json --format json  
  
# Count findings by severity  
grep -o '"severity": "high"' flask\_report.json | wc -l

**High severity findings count:** **\_\_\_\_**\_**\_\_\_\_**

**What are the advantages of JSON output for security teams?**

### 4.2 Automation Potential

**How could these SAST tools be integrated into a development workflow?**

**What would be the benefits of automated security scanning?**

## 📋 Exercise 5: Remediation Planning

### 5.1 Priority Matrix

Create a priority matrix for the Flask application vulnerabilities:

| Vulnerability | Severity | Ease of Exploitation | Business Impact | Priority (1-5) |
| --- | --- | --- | --- | --- |
| **\_\_\_\_**\_**\_\_\_\_** | **\_\_\_\_** | **\_\_\_\_**\_\_\_\_\*\* | **\_\_\_\_**\_**\_\_\_\_** | **\_\_\_\_**\_**\_\_\_\_** |
| **\_\_\_\_**\_**\_\_\_\_** | **\_\_\_\_** | **\_\_\_\_**\_\_\_\_\*\* | **\_\_\_\_**\_**\_\_\_\_** | **\_\_\_\_**\_**\_\_\_\_** |
| **\_\_\_\_**\_**\_\_\_\_** | **\_\_\_\_** | **\_\_\_\_**\_\_\_\_\*\* | **\_\_\_\_**\_**\_\_\_\_** | **\_\_\_\_**\_**\_\_\_\_** |
| **\_\_\_\_**\_**\_\_\_\_** | **\_\_\_\_** | **\_\_\_\_**\_\_\_\_\*\* | **\_\_\_\_**\_**\_\_\_\_** | **\_\_\_\_**\_**\_\_\_\_** |

### 5.2 Remediation Plan

**Top 3 vulnerabilities to fix first:**

1. **Vulnerability:** **\_\_\_\_**\_**\_\*\*** Why first: **\*\*\_\_\_\_\_\_** **How to fix:** **\_\_\_\_\_\*\*\_\_**\*\* **Estimated effort:** **\_\_\_\_**\_\_\_\_\*\*
2. **Vulnerability:** **\_\_\_\_**\_**\_\*\*** Why second: **\*\*\_\_\_\_\_**\_**\_\*\*** How to fix:\*\* \*\*\_\_\*\*\*\*\_\_\_**\_**\_\*\* **Estimated effort:** \*\*\_\_\*\*\*\*\_\_\_\_\*\*
3. **Vulnerability:** **\_\_\_\_**\_**\_\*\*** Why third: **\*\*\_\_\_\_\_\_** **How to fix:** **\_\_\_\_\_\*\*\_\_**\*\* **Estimated effort:** **\_\_\_\_**\_\_\_\_\*\*

### 5.3 Fix Implementation

**Choose ONE vulnerability to actually fix. Document your process:**

**Vulnerability chosen:** **\_\_\_\_**

**Original vulnerable code:**

# Paste original code here:

**Fixed secure code:**

# Paste your fix here:

**Re-run analysis results after fix:** Before: **\_** findings | After: **\_** findings | Improvement: **\_** fewer findings

## 🎓 Final Reflection

### Knowledge Assessment

**1. In your own words, explain what Static Application Security Testing is:**

**2. What are the three most important things you learned from this exercise?** a) **\_\_\_\_**\_**\_** b) **\_\_\_\_\_\_** c) **\_\_\_\_**\_\_\_\_\*\*

**3. How has this exercise changed your perspective on software security?**

### Career Interest

**4. Which cybersecurity career path interests you most after this exercise?** □ Application Security Engineer □ DevSecOps Engineer  
□ Security Consultant □ Penetration Tester □ Other: **\_\_\_\_\_\*\*\_\_**\*\*

**Why?** **\_\_\_\_**\_\_\_\_\*\*

### Real-World Application

**5. How would you explain the importance of SAST to:**

**A software developer:**

**A business manager:**

**A friend who’s not in tech:**

### Future Learning

**6. What security topics would you like to learn about next?** □ Dynamic Application Security Testing (DAST) □ Penetration Testing □ Incident Response □ Network Security □ Cryptography □ Other: **\_\_\_\_\_\*\*\_\_**\*\*

## 📊 Self-Assessment Checklist

Rate yourself on each skill (1=Beginner, 5=Expert):

| Skill | Rating (1-5) |
| --- | --- |
| Understanding SAST concepts | **\_** |
| Using security analysis tools | **\_** |
| Interpreting tool outputs | **\_** |
| Identifying vulnerability types | **\_** |
| Planning remediation strategies | **\_** |
| Communicating security findings | **\_** |

**What skill do you most want to improve?**

**What was the most challenging part of this exercise?**

**What was the most interesting discovery you made?**

## ⚖️ Legal and Ethical Considerations

### Professional Responsibility in Code Analysis

**1. Employment Impact:** How could the vulnerabilities you found affect developers’ job security if exploited in production?

**2. Privacy Rights:** What types of personal data could be compromised through the vulnerabilities identified?

**3. Intellectual Property:** Could the security issues you found expose proprietary business logic or trade secrets?

### Regulatory Compliance

**4. Data Protection Laws:** How might the vulnerabilities you found violate regulations like Privacy Act,1988 (Privacy Act)?

**5. Industry Standards:** What compliance requirements (ISM, ISO9126, ISO14598) could be affected by these security issues?

### Ethical Security Testing

**6. Responsible Disclosure:** If you found these vulnerabilities in a real application, what would be the ethical way to report them?

**7. Professional Standards:** What responsibilities do cybersecurity professionals have when conducting security assessments?

## 🔐 Cryptography and Security Assessment

### Understanding Cryptographic Controls

**1. Encryption Assessment:** Did you find any issues with how the applications handle sensitive data encryption?

**2. Key Management:** What vulnerabilities were identified related to cryptographic key storage or management?

**3. Cryptographic Best Practices:** Based on your analysis, what cryptographic improvements would you recommend?

**4. Security by Design:** How does proper cryptography implementation contribute to ‘security by design’ principles?

## 💼 Business Impact Assessment

### Enterprise Impact Analysis

**1. Productivity Impact:** How would the vulnerabilities you found affect daily business operations if exploited?

**2. Financial Impact:** Estimate the potential cost to an organization if these vulnerabilities were exploited:

* **Direct Costs:** **\_\_\_\_**\_\_\_\_\*\*
* **Indirect Costs:** **\_\_\_\_**\_**\_\*\*\_\_**\*\*
* **Regulatory Fines:** **\_\_\_\_**\_\_\_\_\*\*

**3. Reputation Damage:** How could a security breach from these vulnerabilities affect an organization’s reputation?

**4. Business Continuity:** Which vulnerabilities pose the greatest risk to maintaining normal business operations?

**🎯 Congratulations on completing the SAST exercise! You’ve taken an important step toward understanding application security.**

**Teacher’s Comments:**

**Grade:** **\_\_\_\_**\_**\_\_\_\_** **Date Completed:** **\_\_\_\_**\_**\_\_\_\_**