# Sandbox Security Analysis - Student Worksheet

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

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

## 🔧 Pre-Exercise Setup Verification

**Before starting sandbox analysis, verify your environment is ready:**

### Step 1: Check Docker Environment

# Verify containers are running:  
cd docker && docker-compose ps

**Container Status Check:**

* ☐ cybersec\_sandbox - State: Up
* ☐ vulnerable\_web\_app - State: Up

### Step 2: Test Sandbox Tools

# Test system monitoring tools:  
strace --version  
netstat --version  
htop --version

**Sandbox Tools Check:**

* ☐ System call tracing (strace) available ✅
* ☐ Network monitoring (netstat) available ✅
* ☐ Resource monitoring (htop) available ✅

### Step 3: Verify Sample Applications

# Check suspicious applications:  
ls samples/suspicious-scripts/  
ls samples/backdoor-apps/  
ls samples/resource-abuse/

**Sample Applications Check:**

* ☐ Suspicious scripts accessible ✅
* ☐ Backdoor applications accessible ✅
* ☐ Resource abuse samples accessible ✅

### Step 4: Test Container Access

# Enter sandbox environment:  
docker exec -it cybersec\_sandbox bash

**Container Access Check:**

* ☐ Successfully entered sandbox environment ✅
* ☐ Can navigate to /workspace directory ✅

### Troubleshooting:

**If any verification fails:**

1. Restart containers: cd docker && docker-compose down && docker-compose up -d
2. Wait 30 seconds for services to initialize
3. Re-run verification commands
4. **Notify instructor if sandbox environment doesn’t work properly**

**⚠️ Complete ALL verification steps before proceeding with sandbox analysis exercises.**

## 🎯 Learning Objectives

By completing this worksheet, I will:

* ☐ Understand what sandbox security analysis is and how it differs from SAST and DAST
* ☐ Practice setting up secure analysis environments for testing suspicious applications
* ☐ Learn to monitor system calls, network activity, and resource usage
* ☐ Identify malicious behavior patterns in applications
* ☐ Document security findings with proper evidence

## 📚 Pre-Exercise Knowledge Check

### 1. Security Testing Methods Comparison

Fill in the comparison table:

| Method | What it analyzes | When it runs | Safety level |
| --- | --- | --- | --- |
| SAST | **\_\_\_\_**\_**\_\_\_\_** | **\_\_\_\_**\_**\_\_\_\_** | **\_\_\_\_**\_**\_\_\_\_** |
| DAST | **\_\_\_\_**\_**\_\_\_\_** | **\_\_\_\_**\_**\_\_\_\_** | **\_\_\_\_**\_**\_\_\_\_** |
| Sandbox | **\_\_\_\_**\_**\_\_\_\_** | **\_\_\_\_**\_**\_\_\_\_** | **\_\_\_\_**\_**\_\_\_\_** |

### 2. Risk Scenarios

**Scenario**: You receive an email with an attachment claiming to be a “System Performance Booster.exe”

**Question**: What are the potential risks of running this file directly on your computer?

**Question**: How could sandbox analysis help you safely determine if this file is malicious?

## 🧪 Exercise 1: Basic Behavioral Analysis

### Setup Phase

**Task**: Set up monitoring for the suspicious Python script

1. **Enter the sandbox environment**:

* Command to run: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

1. **Navigate to the analysis workspace**:

* Command to run: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

1. **Examine the script safely** (without running it):

* Command to run: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Initial Observations**: What does the script claim to do? **\_\_\_\_\_\*\*\_\_**\*\* Do you notice any suspicious imports or functions? **\_**\_\*\*\_\_\*\*\*\*

### Monitoring Setup

1. **Set up system call tracing**:

* Command to run: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

1. **Record baseline network state**:

* Command to run: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

### Analysis Phase

1. **Execute the script under monitoring** and let it run for 30 seconds
2. **Analyze system calls for file operations**:

* Command to run: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**File Access Findings**: What files did the script try to access? **\_\_\_\_\_\*\*\_\_**\*\*

Are any of these concerning? Why? **\_\_\_\_**\_\_\_\_\*\*

1. **Analyze system calls for network operations**:

* Command to run: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Network Activity Findings**: What network connections did the script attempt? **\_\_\_\_**\_\_\_\_\*\*

What data might have been transmitted? **\_\_\_\_**\_\_\_\_\*\*

### Risk Assessment

**Question**: Based on your analysis, is this script malicious? Explain your reasoning:

**Question**: What potential damage could this script cause if run on a real system?

## 🌐 Exercise 2: Web Application Backdoor Detection

### Discovery Phase

**Task**: Find hidden functionality in the web application

1. **Test normal web pages**:

* URLs tested:  
  - http://localhost:5000/ : \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  
  - http://localhost:5000/about : \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  
  - http://localhost:5000/contact : \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

1. **Use directory enumeration to find hidden endpoints**:

* Command to run: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Hidden Endpoints Discovered**: List any suspicious or hidden URLs you found:

### Backdoor Analysis

1. **Test the hidden endpoints**:

* Command 1: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  
  Result: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  
    
  Command 2: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  
  Result: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Backdoor Capabilities**: What can an attacker do through these hidden endpoints?

### Security Impact Assessment

**Risk Level** (Circle one): LOW MEDIUM HIGH CRITICAL

**Justification**: Why did you choose this risk level?

**Potential Impact**: What could an attacker accomplish with this backdoor?

## ⚡ Exercise 3: Resource Abuse Detection

### Baseline Measurement

**Task**: Detect applications that abuse system resources

1. **Record initial system state**:

* Commands to run:  
  - CPU usage: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  
  - Memory usage: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Baseline Measurements**:

* Current CPU usage: **\_\_**%
* Available memory: **\_\_\_\_**\_**\_\_\_\_**
* Number of running processes: **\_\_\_\_**\_**\_\_\_\_**

### Resource Monitoring

1. **Start the suspicious “maintenance tool”**:

* Command to run: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

1. **Monitor resource consumption for 60 seconds**:

* Commands to run every 10 seconds:  
  - \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  
  - \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Resource Consumption Observations**: Fill in the monitoring data:

| Time (seconds) | CPU Usage (%) | Memory Usage | Suspicious Activity |
| --- | --- | --- | --- |
| 0 (baseline) | **\_** | **\_** | **\_\_\_\_**\_**\_\_\_\_** |
| 10 | **\_** | **\_** | **\_\_\_\_**\_**\_\_\_\_** |
| 20 | **\_** | **\_** | **\_\_\_\_**\_**\_\_\_\_** |
| 30 | **\_** | **\_** | **\_\_\_\_**\_**\_\_\_\_** |
| 40 | **\_** | **\_** | **\_\_\_\_**\_**\_\_\_\_** |
| 50 | **\_** | **\_** | **\_\_\_\_**\_**\_\_\_\_** |
| 60 | **\_** | **\_** | **\_\_\_\_**\_**\_\_\_\_** |

### Network Activity Analysis

1. **Check for suspicious network connections**:

* Command to run: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Network Findings**: What external servers is the application trying to connect to?

Why might these connections be suspicious?

### Behavioral Pattern Analysis

**Question**: What type of malicious software does this behavior pattern suggest?

**Question**: How could you confirm your suspicion?

## 📊 Synthesis and Reflection

### Threat Summary

Complete the threat analysis table:

| Application | Threat Type | Risk Level | Key Evidence | Recommended Action |
| --- | --- | --- | --- | --- |
| Suspicious Script | **\_\_\_\_\*\*\_\_**\*\* | **\_\_\_\_\*\*\_\_**\*\* | **\_\_\_\_**\_\_\_\_\*\* | **\_\_\_\_**\_**\_\_\_\_** |
| Backdoor Web App | **\_\_\_\_\*\*\_\_**\*\* | **\_\_\_\_\*\*\_\_**\*\* | **\_\_\_\_**\_\_\_\_\*\* | **\_\_\_\_**\_**\_\_\_\_** |
| Resource Abuser | **\_\_\_\_\*\*\_\_**\*\* | **\_\_\_\_\*\*\_\_**\*\* | **\_\_\_\_**\_\_\_\_\*\* | **\_\_\_\_**\_**\_\_\_\_** |

### Critical Thinking Questions

1. **Detection Evasion**: How might malicious software try to avoid detection in sandbox environments?
2. **Real-World Application**: In what professional situations would sandbox analysis be most valuable?
3. **Limitations**: What are the limitations of sandbox analysis? What might it miss?
4. **Defense Strategy**: Based on today’s exercise, what defensive measures would you recommend for an organization?

### Tool Mastery Checklist

Check off the tools and techniques you’ve successfully used:

**System Monitoring**:

* ☐ strace - System call tracing
* ☐ netstat - Network connection monitoring
* ☐ top/htop - Resource usage monitoring
* ☐ lsof - File access monitoring

**Analysis Techniques**:

* ☐ Baseline establishment
* ☐ Behavioral pattern recognition
* ☐ Evidence collection and documentation
* ☐ Risk assessment and classification

**Security Concepts**:

* ☐ Sandbox isolation principles
* ☐ Malicious behavior indicators
* ☐ Network traffic analysis
* ☐ Resource abuse detection

## 🏆 Challenge Questions (Optional)

### Advanced Analysis

**Challenge 1**: Research a real-world malware family (e.g., Zeus, Conficker, WannaCry). How would sandbox analysis help identify its malicious behavior?

**Challenge 2**: Design a theoretical malware that could evade the detection techniques we used today. What would it do differently?

### Career Connection

**Challenge 3**: Research the role of a “Malware Analyst” or “Incident Response Specialist.” How do they use sandbox analysis in their daily work?

## 📝 Self-Assessment

Rate your confidence level (1-5, where 5 = very confident):

* Understanding sandbox analysis concepts: **\_**/5
* Setting up monitoring environments: **\_**/5
* Identifying malicious behavior patterns: **\_**/5
* Using command-line analysis tools: **\_**/5
* Documenting security findings: **\_**/5

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

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

**What would you like to learn more about?**

## ⚖️ Legal and Ethical Considerations

### Professional Responsibility in Malware Analysis

**1. Employment Impact:** How do malware incidents affect IT security teams and organizational employment?

**2. Privacy Rights:** What privacy concerns arise when analyzing applications that may access personal data?

**3. Intellectual Property:** How could malware expose or steal proprietary software and trade secrets?

### Legal Framework for Security Analysis

**4. Authorized Analysis:** Why is it critical to only analyze suspicious software in controlled, authorized environments?

**5. Evidence Handling:** What legal requirements apply to documenting and preserving malware analysis evidence?

### Ethical Malware Research

**6. Responsible Disclosure:** How should security researchers ethically handle discovery of new malware families?

**7. Professional Standards:** What ethical obligations do cybersecurity analysts have when conducting malware analysis?

## 🔐 Cryptography and Sandbox Security

### Cryptographic Analysis in Sandboxing

**1. Encryption Assessment:** Did any analyzed applications use encryption to hide malicious activities?

**2. Communication Security:** How did malicious applications handle cryptographic protection of network communications?

**3. Key Management:** What cryptographic vulnerabilities were exposed in the malicious applications?

**4. Sandbox Cryptography:** How does cryptography contribute to secure sandbox design and operation?

## 💼 Business Impact Assessment

### Enterprise Malware Impact Analysis

**1. Operational Disruption:** How would the malware behaviors you observed affect business operations?

**2. Financial Impact:** Estimate the potential business costs of the threats you analyzed:

* **Data Loss Costs:** **\_\_\_\_\_\*\*\_\_**\*\*
* **System Recovery Costs:** **\_\_\_\_**\_\_\_\_\*\*
* **Regulatory Penalties:** **\_\_\_\_**\_\_\_\_\*\*
* **Business Interruption:** **\_\_\_\_**

**3. Reputation Damage:** How could malware incidents affect organizational reputation and customer trust?

**4. Incident Response:** What would be the cost and complexity of responding to the threats you analyzed?

**Completed by**: **\_\_\_\_**\_**\_\*\*** Date**: \*\*\_\_\_\_\_\_**  
**Instructor Review**: **\_\_\_\_**\_**\_\*\*** Grade**: \*\*\_\_**\*\*