Blind OS command injection with time delays(command injection)

Executive Summary

A **Blind OS Command Injection** vulnerability was discovered in the feedback submission feature of the application. This flaw allowed an attacker to execute arbitrary commands on the server. Although no output was returned in the application’s HTTP response, it was confirmed via time-based analysis, with a noticeable 10-second delay in response, verifying successful command execution. This is a **Critical** severity issue due to the potential for full system compromise.

Introduction

The purpose of this assessment was to evaluate the security posture of the application's feedback functionality. During the engagement, a command injection flaw was found, where user input was unsafely included in a shell command. This enabled remote command execution using a time delay technique to verify code execution in a blind environment.

Methodology

1. **Targeted Input Testing**: Used Burp Suite to intercept and modify the HTTP request for feedback submission.

2. **Payload Injection**: Crafted a payload to inject a `ping` command causing a known delay.

3. **Response Analysis**: Measured the time taken to receive the server’s response to determine successful command execution.

4. **Verification**: A consistent 10-second delay confirmed the presence of a blind command injection vulnerability.

Vulnerability Findings

- **Type**: Blind OS Command Injection (Time-Based)

- **Parameter**: `email`

- **Severity**: Critical

- **Location**: Feedback form (`POST /feedback`)

Description:

The application executed a shell command incorporating user-supplied data from the `email` parameter. Improper input sanitization allows the injection of shell command separators (`||`), enabling arbitrary command execution. A 10-second delay, caused by injecting a `ping` command to localhost, verified that the command was executed.

Proof of Concept (PoC)

1. Intercept the feedback form submission using Burp Suite.

2. Modify the `email` parameter as follows:

   ini

   CopyEdit

   `email=x||ping+-c+10+127.0.0.1||`

3. Send the modified request.

4. Observe that the server takes approximately 10 seconds to respond, confirming execution of the injected `ping` command.

Impact Assessment

Exploitation of this vulnerability allowed:

- Execution of arbitrary OS commands

- Potential full system compromise

- Lateral movement or privilege escalation

Given the critical nature of command injection vulnerabilities, immediate remediation is essential.

Recommendations

1. **Avoid concatenating shell commands** with user input. Use safer system interfaces or libraries.

2. **Strictly validate and sanitize** all user input using allowlists.

3. **Implement least privilege** at the OS level to reduce impact if command execution is achieved.

4. **Monitor and alert** on unusual command activity or extended response times.

Conclusion

A critical blind OS command injection flaw was confirmed via time delay analysis in the feedback feature. The vulnerability enables arbitrary command execution and represents a major risk. Immediate corrective measures are required to mitigate exploitation.