SCENARIO

The application is vulnerable to server-side template injection due to the way it unsafely uses a Tornado template. We will try to figure out the way to execute arbitrary code on the backend server.

**PROCEDURE**

1. Open the application and log in using the provided credentials to act as a target.
2. Make a request to the server in order to change what the site displays in place of your name and study the request in BurpSuite’s Repeater.
3. We see that in the parameter **blog-post-author-display,** a parameter is sent.
4. Now, to identify the template we will try to execute the Payload 1 into the request within the **blog-post-author-display** parameter.
5. We see that instead of getting the supplied input reflected on the screen we get 49 as response which confirms that the server is using Tornado template engine.
6. We know that we can execute arbitrary python in Tornado template.
7. So, we will try to inject Payload 2 into the URL in that parameter in order to do Remote Code Execution by going through the documentation of Tornado template engine.

**PAYLOAD**

1. user.name}}{{7\*7}}
2. user.name}}{%25+import+os+%25}{{os.system('rm%20/home/carlos/morale.txt')

**REMEDIATION**

1. **Avoid User-Controlled Templates:** Refrain from allowing user input to dictate or influence templates in any capacity. When users can control templates or parts thereof, the application becomes susceptible to server-side template injection attacks.
2. **Safe API Methods:** Opt for methods or functions that are explicitly designed for constructing templates securely. For instance, in Tornado, avoid features or methods that allow for execution of arbitrary Python code.
3. **Sanitize and Validate Input:** Prior to processing, always sanitize and validate user inputs. Confirm that the information passed to templates meets the expected criteria in terms of type and format. Inputs with unexpected characters or those that seem harmful should be outrightly rejected.
4. **Disable Unnecessary Template Features:** If certain features in the template engine are not necessary, disable them. By turning off certain features in Tornado, you can reduce the exploitable surface area.
5. **Use Least Privilege Principle:** It's important to run the application with the least privileges necessary. This minimizes potential damage if an attacker does execute arbitrary commands.
6. **Input Escaping:** It's pivotal to escape user input to ensure it’s recognized as data and isn’t executed as code, especially in scenarios where user input is added to templates.
7. **Web Application Firewalls (WAFs):** Using a WAF can help identify and block suspicious requests. Contemporary WAFs are designed to detect and halt many attempts at template injection.