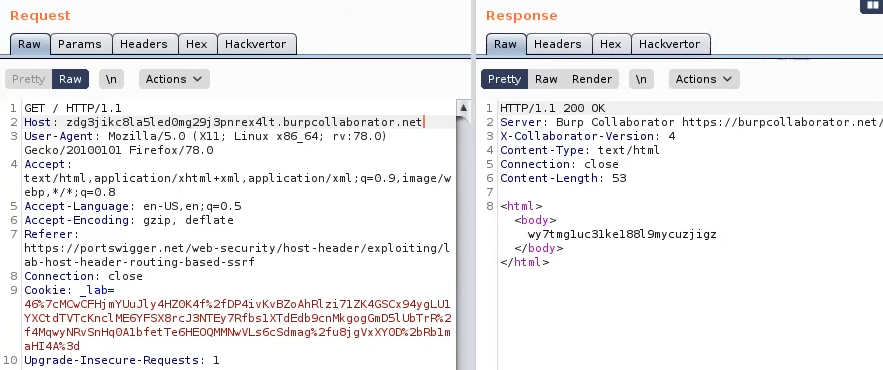
SCENARIO

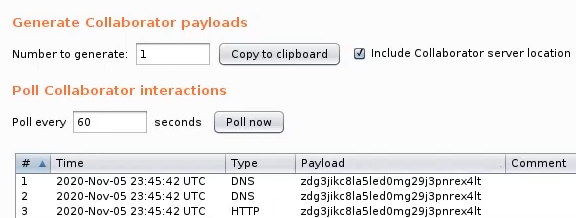
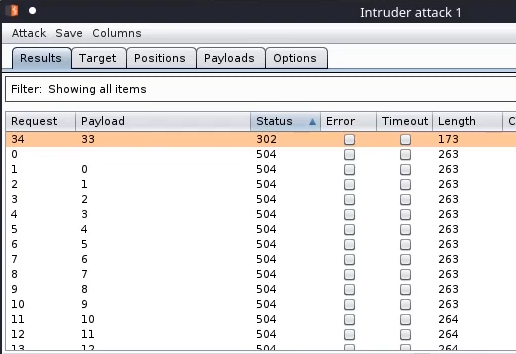
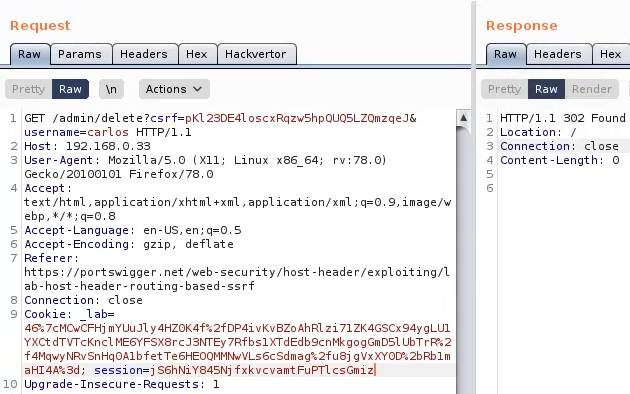
This application is vulnerable to routing-based SSRF via the Host header. You can exploit this to access an insecure intranet admin panel located on an internal IP address. We will try to get access to the admin panel.

**PROCEDURE**

1. Go to the web application and using BurpSuite send the request for homepage to the Repeater.
2. Go to the Collaborator Client and copy the URL from there and paste in place of the Host Header in the request then send the request, we see that after polling we successfully made some requests to an arbitrary server.
3. Send the GET request to BurpSuite’s Intruder and **untick the Update host header to match target option.**
4. Replace the Host header value with the provided IP Address which is 192.168.0.0 and add the last 0 as payload then brute force the address in order to look for a 302 response.
5. We see that we got a 302 response and now we were redirected to the admin panel.
6. Now send the request to access the admin page and we get access to it.

**PAYLOAD**

**PROOF OF CONCEPT**

****

**REMEDIATION**

1. **Whitelist Host Headers:** Only allow known good host headers to be processed by the application. Any unrecognized host header should be rejected or redirected to a default page. This reduces the chance of an attacker manipulating the host header for malicious purposes.
2. **Block Internal Network Access:** Implement network restrictions to prevent the server from making requests to internal resources. This might involve using a firewall, a web application firewall (WAF), or application logic that prevents connections to internal IP address ranges.
3. **Limit URL Redirection:** Ensure that any function which provides URL redirection is not based on user inputs without validation. If user input must be used, validate it rigorously against a whitelist.
4. **Disable Unused Protocols:** If the server supports protocols that are not needed for business functionality, such as "file://", "gopher://", "dict://", etc., they should be disabled.
5. **Secure Misconfigured DNS Servers:** Ensure that the DNS servers used by the application do not allow zone transfers or return information about internal network topology.