IFS4103 Lab 3: Using Burp Suite's Repeater, Target, and Scanner

Notes:

- Once you have set up your Burp Proxy, you can utilize all Burp Suite's **other components**, such as **Burp Repeater**, **Target** (for target scoping), and **Scanner**.

Objectives:

For Lab 3, you want to perform the following tasks:

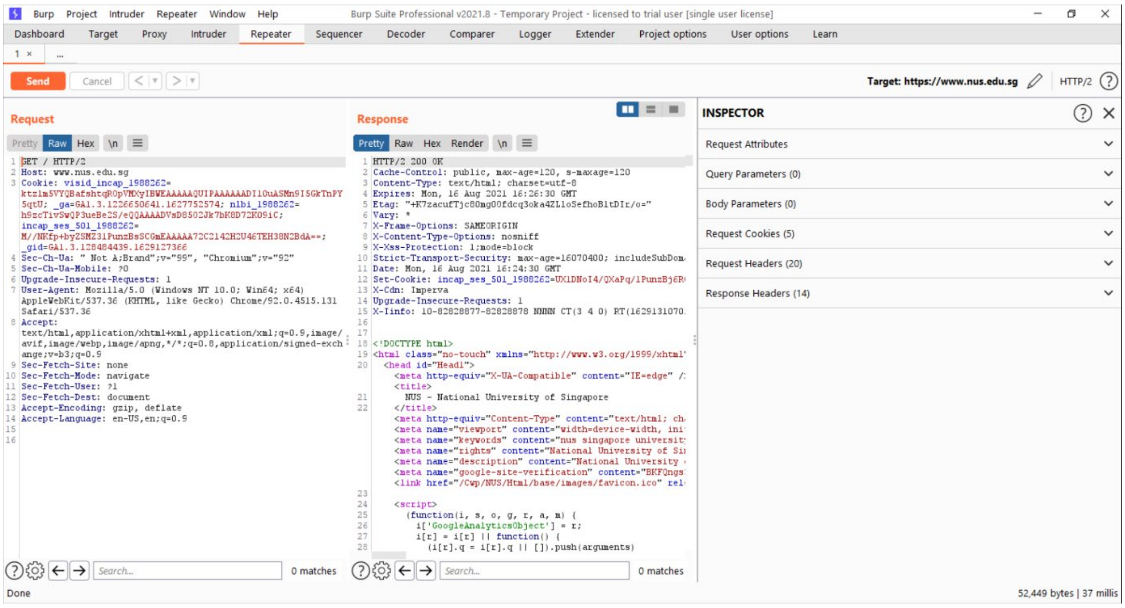
1. To resend (possibly modified) individual requests using **Burp Repeater**;
2. To specify your **target scope** in Burp Suite;
3. To scan a website for vulnerabilities using **Burp Scanner.**

Task 1: Resending Requests using Burp Repeater

Please refer to the following video from PortSwigger, which is viewable on YouTube:

- "How to resend individual requests with Burp Repeater": https://www.youtube.com/watch?v=_Wifm2g9ugg

From observing the video, you should be able to use Burp Suite's Repeater in **resending client requests** as part of the pen-testing of your target sites. For your reference, a sample screenshot of the **Repeater's UI** is shown below.


Burp Project Intruder Repeater Window Help
Burp Suite Professional v2021.8 - Temporary Project - licensed to trial user [single user license]
Dashboard Target Proxy Intruder Repeater Sequencer Decoder Comparer Logger Extender Project options User options Learn
1 × ...
Send Cancel
Target: https://www.nus.edu.sg HTTP/2
Request
Pretty Raw Hex \n
Response
Pretty Raw Hex Render \n
INSPECTOR
Request Attributes
Query Parameters (0)
Body Parameters (0)
Request Cookies (5)
Request Headers (20)
Response Headers (14)
Search...
0 matches
Search...
0 matches
Done
52,449 bytes | 37 millis

```
GET / HTTP/2
Host: www.nus.edu.sg
Cookie: visid_incap_1988262=
ktzlm5VYQBafshtqR0pVMXyIBWEAAAAAQUIPAAAAAADI10uASMn9I5GkTnPY
5qtU; _ga=GA1.3.1226650641.1627752574; nlbi_1988262=
h9zcTivSwQP3ueBe2S/eQQAAAADVsD8502Jk7bK8D72K09iC;
incap_ses_501_1988262=
M//NKfp+byZSMZ31PunzBsSCGmEAAAAA72C2142H2U46TEH38N2BdA==;
_gid=GA1.3.128484439.1629127366
Sec-Ch-Ua: " Not A;Brand";v="99", "Chromium";v="92"
Sec-Ch-Ua-Mobile: ?0
Upgrade-Insecure-Requests: 1
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64)
AppleWebKit/537.36 (KHTML, like Gecko) Chrome/92.0.4515.131
Safari/537.36
Accept:
text/html,application/xhtml+xml,application/xml;q=0.9,image/
avif,image/webp,image/apng,*/*;q=0.8,application/signed-exch
ange;v=b3;q=0.9
Sec-Fetch-Site: none
Sec-Fetch-Mode: navigate
Sec-Fetch-User: ?1
Sec-Fetch-Dest: document
Accept-Encoding: gzip, deflate
Accept-Language: en-US,en;q=0.9
```

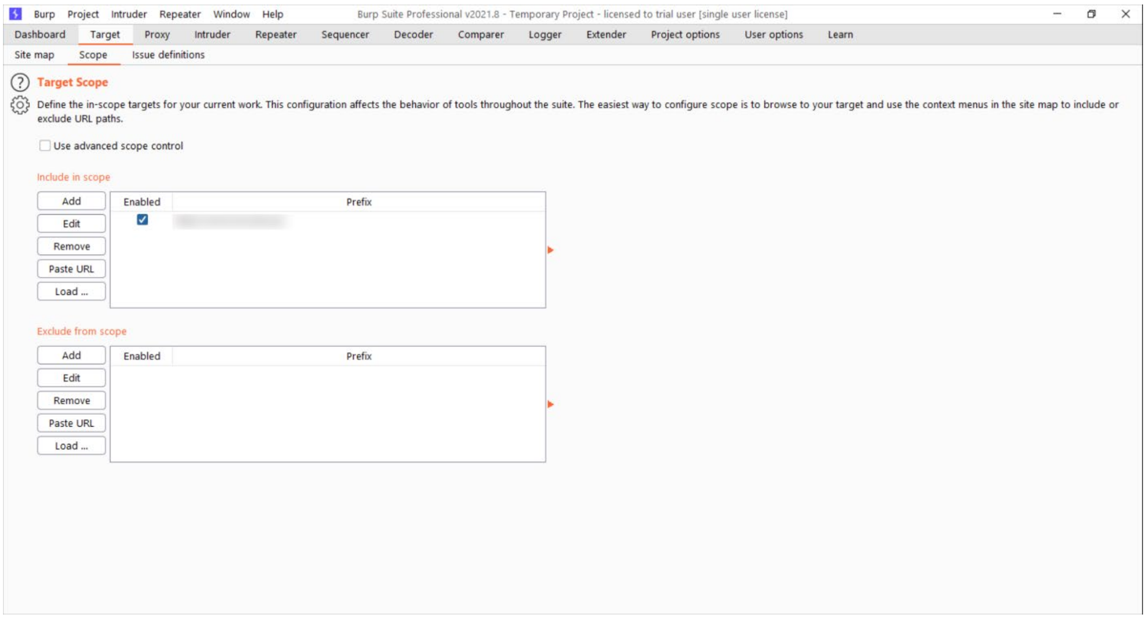
```
HTTP/2 200 OK
Cache-Control: public, max-age=120, s-maxage=120
Content-Type: text/html; charset=utf-8
Expires: Mon, 16 Aug 2021 16:26:30 GMT
Etag: "+K7zacufTjc80mg00fdcq3oka4ZLloSefhoBltDIr/o="
Vary: *
X-Frame-Options: SAMEORIGIN
X-Content-Type-Options: nosniff
X-Xss-Protection: 1;mode=block
Strict-Transport-Security: max-age=16070400; includeSubDom
Date: Mon, 16 Aug 2021 16:24:30 GMT
Set-Cookie: incap_ses_501_1988262=UX1DNoI4/QXaPq/1PunzBj6R
X-Cdn: Imperva
Upgrade-Insecure-Requests: 1
X-Iinfo: 10-82828877-82828878 NNNN CT(3 4 0) RT(1629131070


<!DOCTYPE html>
<html class="no-touch" xmlns="http://www.w3.org/1999/xhtml'
  <head id="Head1">
    <meta http-equiv="X-UA-Compatible" content="IE=edge" /
    <title>
      NUS - National University of Singapore
    </title>
    <meta http-equiv="Content-Type" content="text/html; ch
    <meta name="viewport" content="width=device-width, ini
    <meta name="keywords" content="nus singapore universit
    <meta name="rights" content="National University of Si
    <meta name="description" content="National University
    <meta name="google-site-verification" content="BKFQngs
    <link href="/Cwp/NUS/Html/base/images/favicon.ico" rel

    <script>
      (function(i, s, o, g, r, a, m) {
        i['GoogleAnalyticsObject'] = r;
        i[r] = i[r] || function() {
          (i[r].q = i[r].q || []).push(arguments)
```

Task 2: Specifying Your Target Scope

Please refer to the following video from PortSwigger viewable on YouTube:

- "How to use target scope in Burp Suite": https://www.youtube.com/watch?v=0mTg2BsYVmg

From observing the video, you should be able to **set your target scope** in the pen-testing of your target sites. For your reference, some sample screenshots of **Burp Suite's Target tab** are shown below.


Burp Project Intruder Repeater Window Help
Burp Suite Professional v2021.8 - Temporary Project - licensed to trial user [single user license]
Dashboard Target Proxy Intruder Repeater Sequencer Decoder Comparer Logger Extender Project options User options Learn
Site map Scope Issue definitions
Add
Edit
Remove
Paste URL
Load ...
Add
Edit
Remove
Paste URL
Load ...

Target Scope

Define the in-scope targets for your current work. This configuration affects the behavior of tools throughout the suite. The easiest way to configure scope is to browse to your target and use the context menus in the site map to include or exclude URL paths.

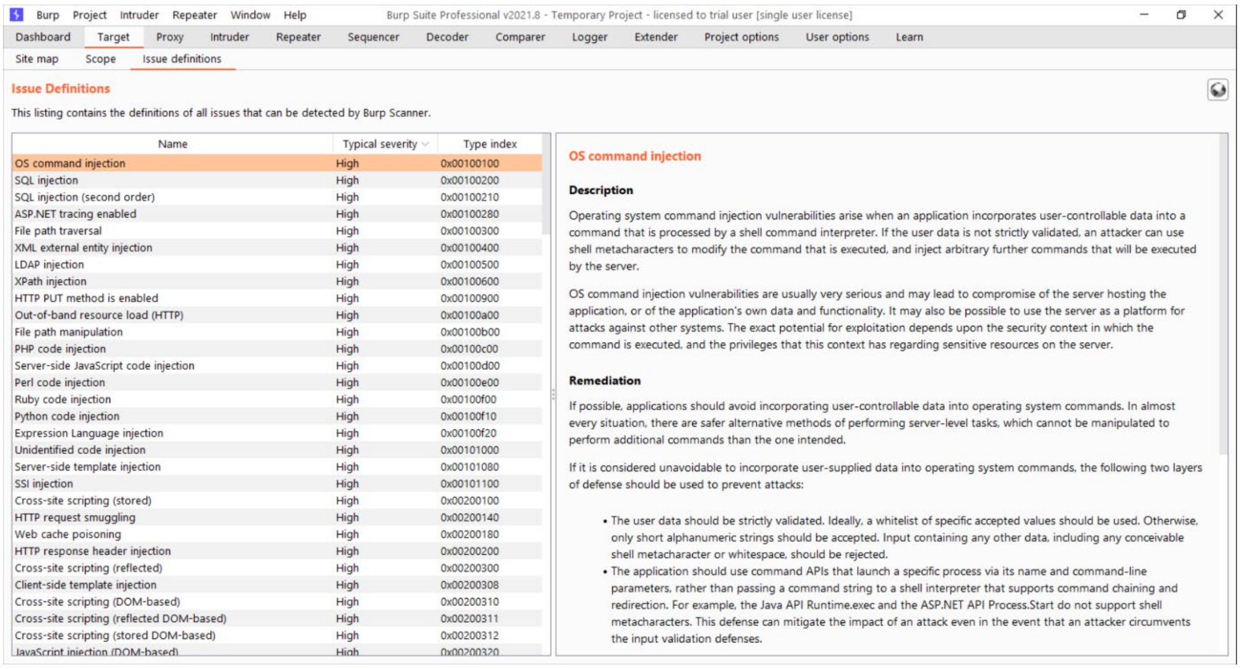
□ Use advanced scope control

Include in scope

Enabled	Prefix
☑	[redacted]

Exclude from scope

Enabled	Prefix


Burp Project Intruder Repeater Window Help
Burp Suite Professional v2021.8 - Temporary Project - licensed to trial user [single user license]
Dashboard Target Proxy Intruder Repeater Sequencer Decoder Comparer Logger Extender Project options User options Learn
Site map Scope Issue definitions

Issue Definitions

This listing contains the definitions of all issues that can be detected by Burp Scanner.

Name	Typical severity	Type index
OS command injection	High	0x00100100
SQL injection	High	0x00100200
SQL injection (second order)	High	0x00100210
ASP.NET tracing enabled	High	0x00100280
File path traversal	High	0x00100300
XML external entity injection	High	0x00100400
LDAP injection	High	0x00100500
XPath injection	High	0x00100600
HTTP PUT method is enabled	High	0x00100900
Out-of-band resource load (HTTP)	High	0x00100a00
File path manipulation	High	0x00100b00
PHP code injection	High	0x00100c00
Server-side JavaScript code injection	High	0x00100d00
Perl code injection	High	0x00100e00
Ruby code injection	High	0x00100f00
Python code injection	High	0x00100f10
Expression Language injection	High	0x00100f20
Unidentified code injection	High	0x00101000
Server-side template injection	High	0x00101080
SSI injection	High	0x00101100
Cross-site scripting (stored)	High	0x00200100
HTTP request smuggling	High	0x00200140
Web cache poisoning	High	0x00200180
HTTP response header injection	High	0x00200200
Cross-site scripting (reflected)	High	0x00200300
Client-side template injection	High	0x00200308
Cross-site scripting (DOM-based)	High	0x00200310
Cross-site scripting (reflected DOM-based)	High	0x00200311
Cross-site scripting (stored DOM-based)	High	0x00200312
JavaScript injection (DOM-based)	High	0x00200320

OS command injection

Description

Operating system command injection vulnerabilities arise when an application incorporates user-controllable data into a command that is processed by a shell command interpreter. If the user data is not strictly validated, an attacker can use shell metacharacters to modify the command that is executed, and inject arbitrary further commands that will be executed by the server.

OS command injection vulnerabilities are usually very serious and may lead to compromise of the server hosting the application, or of the application's own data and functionality. It may also be possible to use the server as a platform for attacks against other systems. The exact potential for exploitation depends upon the security context in which the command is executed, and the privileges that this context has regarding sensitive resources on the server.

Remediation

If possible, applications should avoid incorporating user-controllable data into operating system commands. In almost every situation, there are safer alternative methods of performing server-level tasks, which cannot be manipulated to perform additional commands than the one intended.

If it is considered unavoidable to incorporate user-supplied data into operating system commands, the following two layers of defense should be used to prevent attacks:

- The user data should be strictly validated. Ideally, a whitelist of specific accepted values should be used. Otherwise, only short alphanumeric strings should be accepted. Input containing any other data, including any conceivable shell metacharacter or whitespace, should be rejected.
- The application should use command APIs that launch a specific process via its name and command-line parameters, rather than passing a command string to a shell interpreter that supports command chaining and redirection. For example, the Java API Runtime.exec and the ASP.NET API Process.Start do not support shell metacharacters. This defense can mitigate the impact of an attack even in the event that an attacker circumvents the input validation defenses.

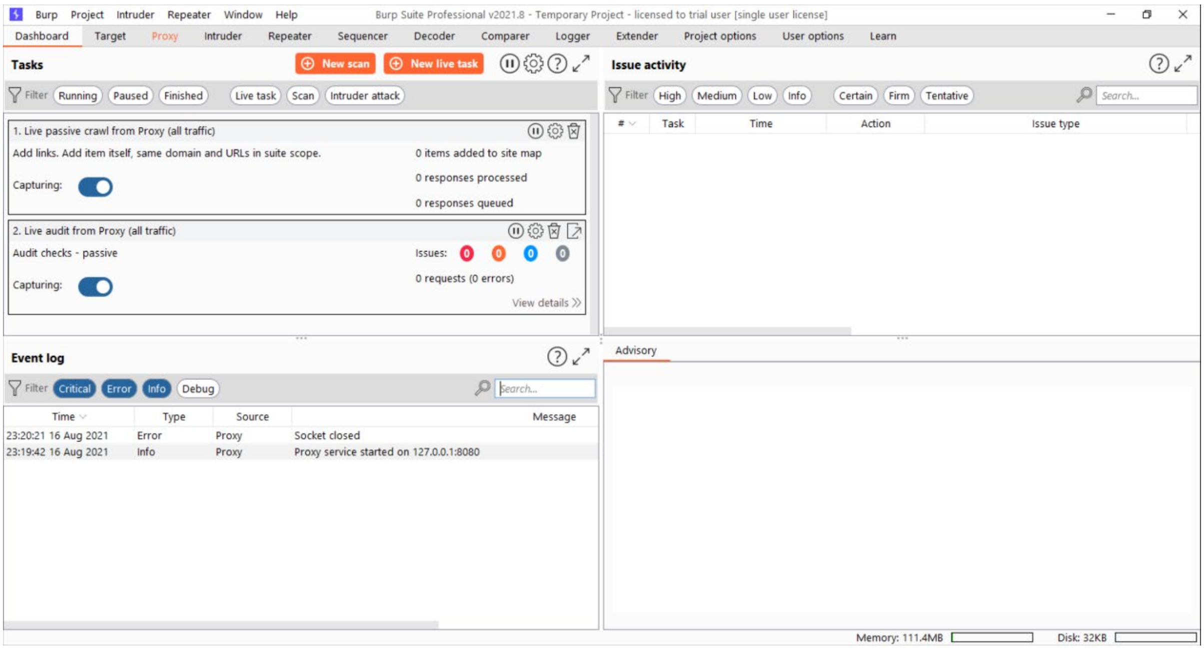
Task 3: Scanning a Target Website using Burp Scanner

Please refer to the following video from PortSwigger, which is viewable on YouTube:

- "How to scan a website for vulnerabilities using Burp Scanner": https://www.youtube.com/watch?v=VP9eQhUASYQ

From observing the video, you should be able to **scan your target site for vulnerabilities** in your pen-testing work. Note that this step is ***offensive*** in nature, and should be done only on **target sites *approved*** for your pen-testing. For your own practice, you can target a **vulnerable web application** (see Lab 2) running on a VM reachable from your Burp Suite.

A screenshot of **Burp Suite's Dashboard tab**, where the Scanner component can be invoked, is shown below for your reference.


Burp Project Intruder Repeater Window Help
Burp Suite Professional v2021.8 - Temporary Project - licensed to trial user [single user license]
Dashboard Target Proxy Intruder Repeater Sequencer Decoder Comparer Logger Extender Project options User options Learn
Tasks
New scan
New live task
Filter Running Paused Finished Live task Scan Intruder attack
Issue activity
Filter High Medium Low Info Certain Firm Tentative
Search...
# Task Time Action Issue type
Event log
Filter Critical Error Info Debug
Search...
Advisory
Memory: 111.4MB
Disk: 32KB

1. Live passive crawl from Proxy (all traffic)

Add links. Add item itself, same domain and URLs in suite scope.

0 items added to site map

0 responses processed

0 responses queued

Capturing: ●

2. Live audit from Proxy (all traffic)

Audit checks - passive

Issues: 0 0 0 0

0 requests (0 errors)

Capturing: ●

View details »

Time	Type	Source	Message
23:20:21 16 Aug 2021	Error	Proxy	Socket closed
23:19:42 16 Aug 2021	Info	Proxy	Proxy service started on 127.0.0.1:8080

For additional information on Burp's **automated scanning** feature, you can refer to the following relevant documentation pages:

- Launching scans – Running a ***full*** **crawl & audit**: https://portswigger.net/burp/documentation/desktop/automated-scanning/launching-scans/full-crawl-and-audit

- Launching scans – Scanning ***specific*** **HTTP messages**: https://portswigger.net/burp/documentation/desktop/automated-scanning/launching-scans/scanning-specific-http-messages

- Setting the **scan *scope***: https://portswigger.net/burp/documentation/desktop/automated-scanning/setting-pro-scope

- Configuring application logins – Adding **usernames & passwords**: https://portswigger.net/burp/documentation/desktop/automated-scanning/configuring-app-logins/adding-usernames-passwords