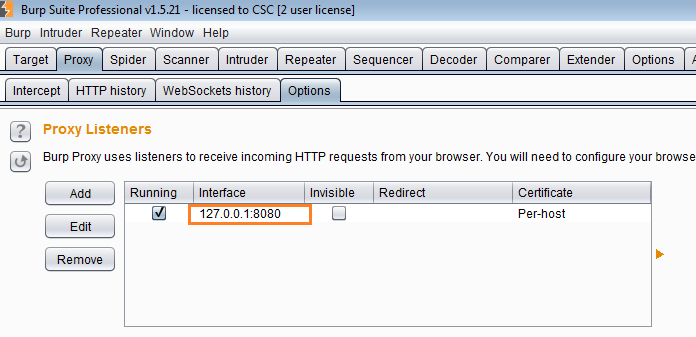
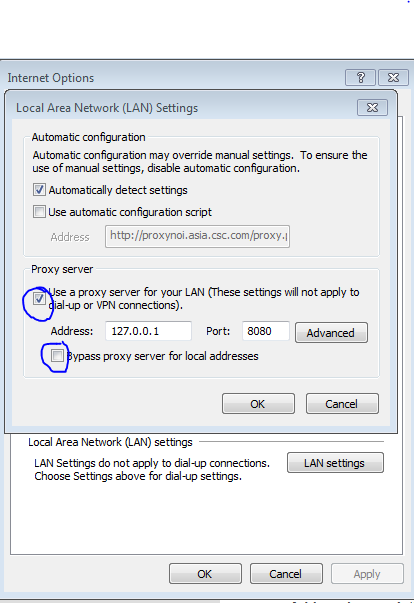
**1. Burp Proxy Setting:** When Burp Suite is launched, Burp Proxy is started by default on port 8080 of the local loopback interface. By setting a web browser to use this as its proxy server, all web traffic can be intercepted, inspected and modified



Navigate to

Proxy-> Intercept- Make sure ‘Intercept is OFF’

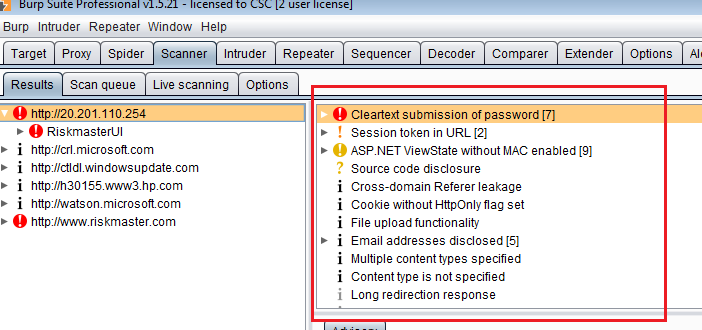
**2. IE Settings**

Launch application & browse the screens. Check if Burp suite is intercepting the web traffic & capturing issues as user browse, under Target-> Results.

Check HTTP request history in Proxy -> HTTP History

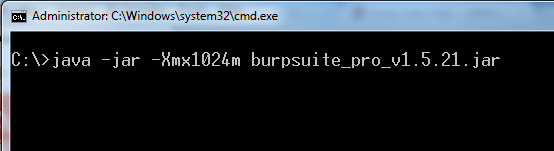
**Penetration testing results captured in Burp Suite:**

All web traffic passing through Burp Proxy is by default analyzed and incorporated into the target site map, to build up a picture of the content and functionality of the applications visited

****

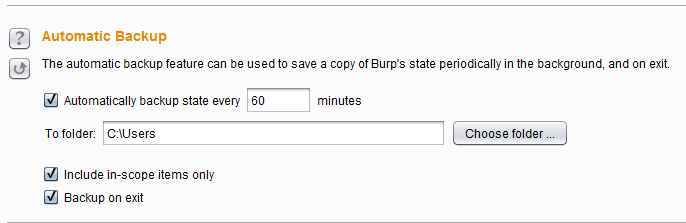
**3. Raising Burp’s Memory**

You can raise Burp’s default memory allocation manually to your desired value. Manually setting the memory will help optimize performance and prevent crashing. This is often used when you expect to test a large site.You can manually set the allocated memory for burp via the command line. Just open the command line, browse to where your burp is located and run the command:  
                Java –jar –Xmx<value in megabytes> <filename of burp>

**

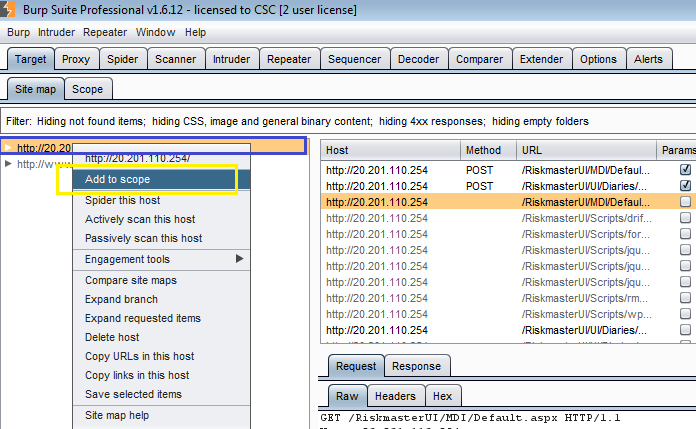
**4. Automatic backup**

Burp can be configured to automatically back up its data. This is important when you are testing a large site and you need to leave your scanner overnight. This way, if burp crashes, you still have a backup to return to. You can configure Burp’s backup settings by going to **Options** > **Misc** > **Automatic Backup**



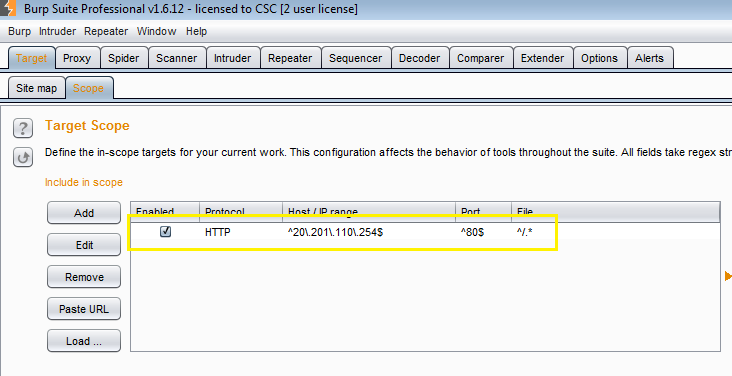
**5. Add to Scope**

1. Go to **Target** > Right Click the Application under test (AUT) **URL** > **Add to Scope**

****

1. Select **‘Scope’** Tab

Verify that Application URL is added to the scope.



**6. Scanner Options: PASSIVE & ACTIVE SCANNING**

**Use Passive scanning mode to check for below issues-**

* Clear-text submission of passwords.
* Insecure cookie attributes, like missing HttpOnly and secure flags.
* Liberal cookie scope.
* Cross-domain script includes and Refer leakage.
* Forms with autocomplete enabled.
* Caching of SSL-protected content.
* Directory listings.
* Submitted passwords returned in later responses.
* Insecure transmission of session tokens.
* Leakage of information like internal IP addresses, email addresses, stack traces, etc.
* Insecure ViewState configuration.
* Ambiguous, incomplete, incorrect or non-standard Content-type directives.

**Use Active scanning mode to check for below issues-**

1. Input-based vulnerabilities targeting the client side, such as cross-site scripting, HTTP header injection, and open redirection.
2. Input-based vulnerabilities targeting the server side, such as SQL injection, OS command injection, and file path traversal.

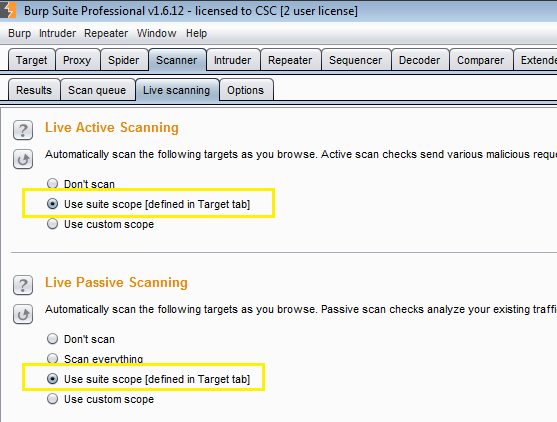
* **By Default Burps launch with Passive scanning mode**
* **To change the mode from Passive to active or to use both – Follow below steps**

1. Go to **Scanner** > **Live Scanning** tab

Make sure that-

**‘Use Suite Scope’** is selected under **Live Active Scanning**

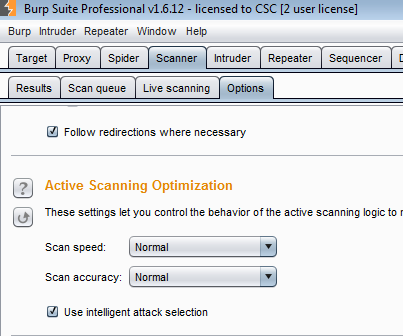
**‘Use Suite Scope’** is selected under **Live Passive Scanning**

****

1. Go to **Scanner** > **Options** tab

Make sure that-

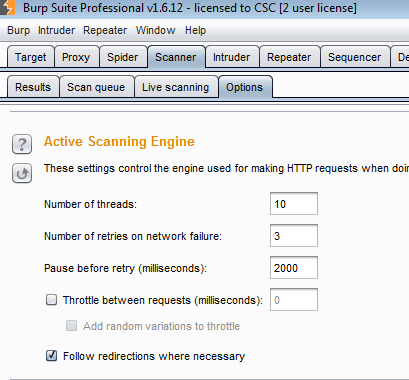
**Active Scanning Optimization** options selected as below



1. Go to **Scanner** > **Options** tab

Make sure that-

**Active Scanning Engine** options selected as below by default



Careful use of these options lets user fine tune the scanning engine.

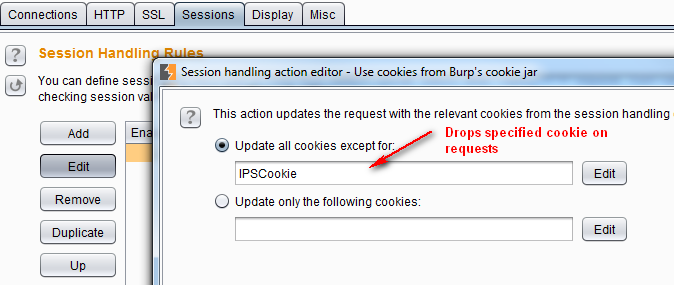
These settings control the engine used for making HTTP requests when doing active scanning. The following options are available:

* **Number of threads** - This option controls the number of concurrent requests the Scanner is able to make.
* **Number of retries on network failure** - If a connection error or other network problem occurs, Burp will retry the request the specified number of times before giving up and moving on. Intermittent network failures are common when testing, so it is best to retry the request several times when a failure occurs.
* **Pause before retry** - When retrying a failed request, Burp will wait the specified time (in milliseconds) following the failure before retrying. If the server is being overwhelmed with traffic, or an intermittent problem is occurring, it is best to wait a short time before retrying.
* **Throttle between requests** - Optionally, Burp can wait a specified delay (in milliseconds) before every request. This option is useful to avoid overloading the application, or to be more stealthy. *This setting should be enabled when we run the regression suite scripts for penTest. Put the value as 1500ms.*
* **Add random variations to throttle** - This option can further increase stealth by reducing patterns in the timing of your requests.
* **Follow redirections where necessary** - Some vulnerabilities can only be detected by following redirections (for example, cross-site scripting in an error message which is only returned after following a redirection). Because some applications issue redirections to third-party URLs that include parameter values that you have submitted, Burp protects you against inadvertently attacking third-party applications, by not following just any redirection which is received. If the request being scanned is within the defined [target scope](https://portswigger.net/burp/help/target_scope.html) (i.e. you are using target scope to control what gets scanned), then Burp will only follow redirections that are within that same scope. If the request being scanned is not in scope (i.e. you have manually initiated a scan of an out-of-scope request), Burp will only follow redirections which (a) are to the same host/port as the request being scanned; and (b) are not explicitly covered by a scope exclusion rule (e.g. "logout.aspx").

**7. Session handling rules (IPS evasion)-- OPTIONAL**

You can configure Burp to drop or only retain specific cookies while you are testing. This is usually done to test session cookies. But this feature can also be used to evade some IPS which use cookies to track sessions. You can do through the following steps:

1. Go to **Options** > **Sessions** > **Session Handling Rules**
2. **Add** a rule.
3. Under **Rule Action,** select **Add**
4. Select **Use cookies from the session handling cookie jar.**
5. In the “Session handling action editor”, add the IPS cookie on the ‘**Update all cookies except for’** option.



By default, this should drop the IPS cookie sent via Burp Spider and Scanner. You can also apply this rule to the other burp modules by changing the **Tools Scope** on the “Session handling rule editor.”

**8. Manual testing simulator -- OPTIONAL**

As the name suggests, this feature sends common test payloads to random URLs and parameters at irregular intervals to generate traffic that is similar to manual penetration testing.

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*