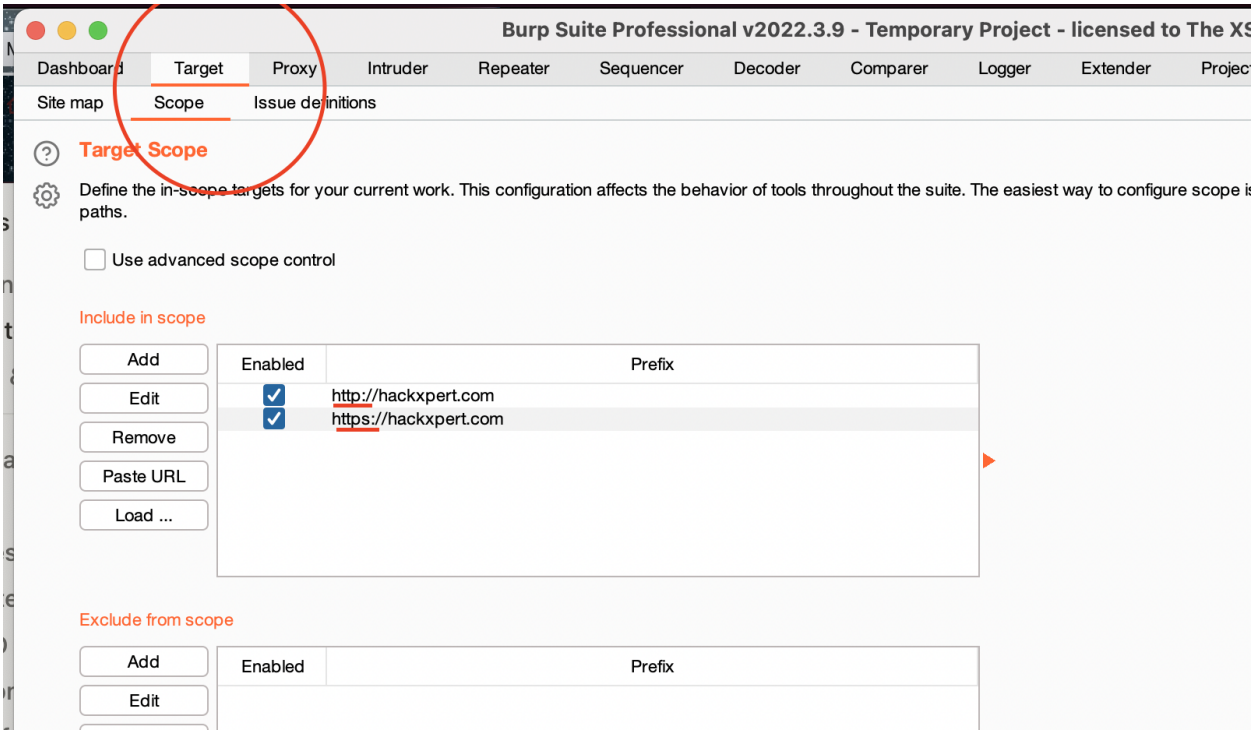
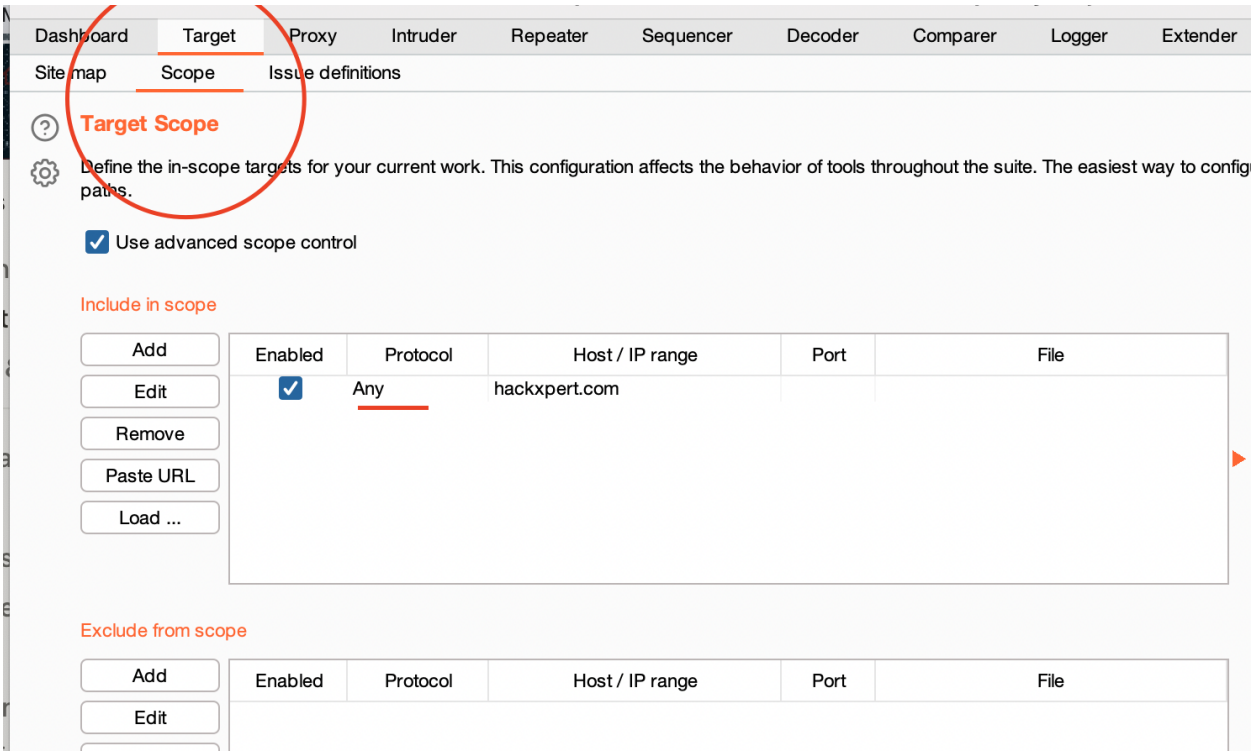
Setting Up Burp Suite

Day 0: Recon

Of course any good project starts with enumeration so you have to make sure well equipped for the task at and if your tool of choice burp suite or OWASP zap, you will find you have to set up very similar things to get started. Burp suite starts with setting up the scope section, ZAP also does the same but names it “contexts”. Make sure you configure this correctly or you might run out of scope without intending to do so which can have legal repercussions. Sometimes the programs determine the scope themselves while other times it might be governed by law for example. Whatever it is, in burp suite make sure to enable the advanced scope control as this will allow you to use parts of the URL instead of always having to use the first part of a URL.

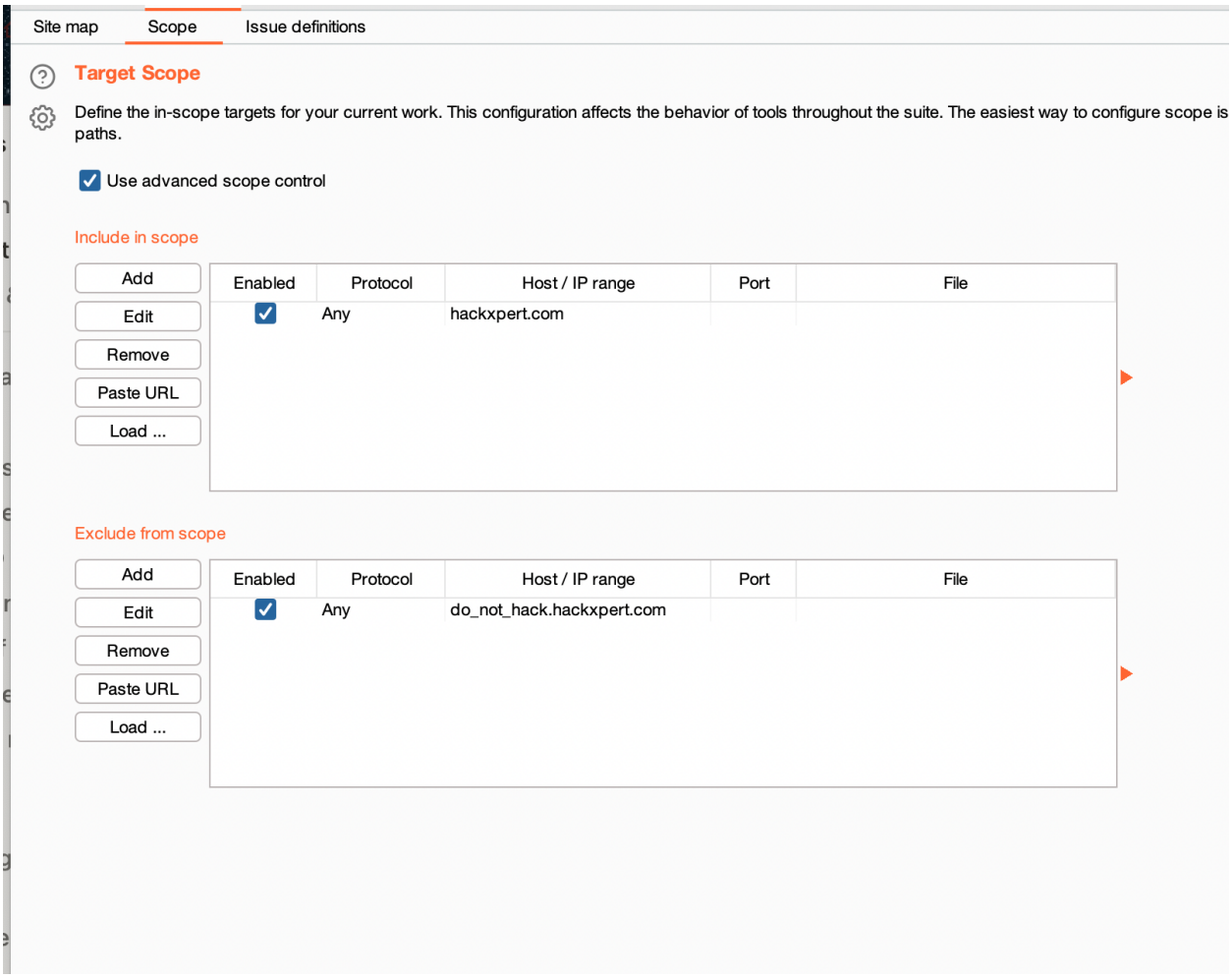


This can be done the same with advanced scope control:



Please note you can also load a list of URLs into burp suite or a list of regex in ZAP. For the sake of clarity, I will be making a separate ZAP article later on. You can potentially also paste a list from the clipboard.

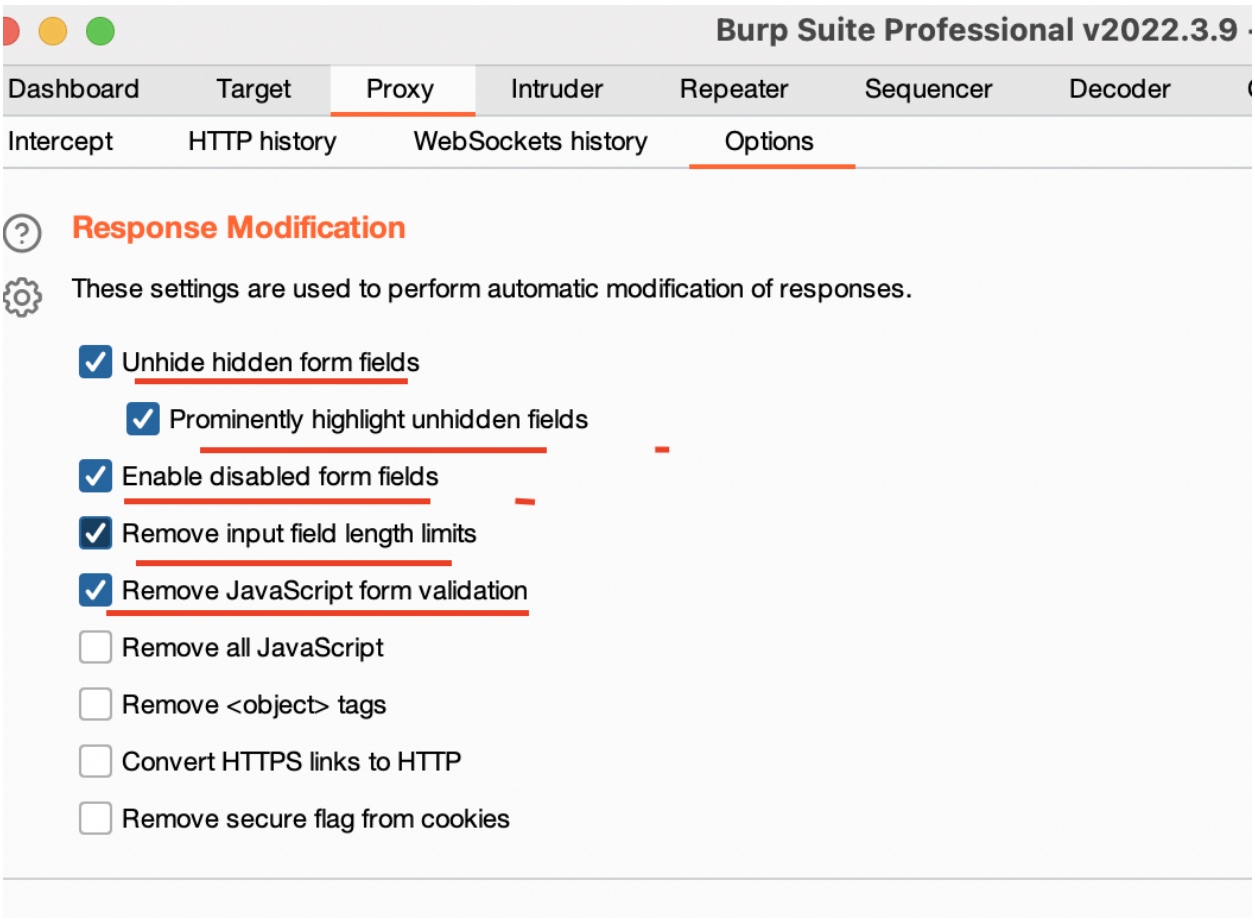
Make sure to also fill in domains you know that are out of scope in the appropriate section. It will save you a lot of headaches later on.



Day 1: Clicking through the application

Now it's our mission to fill up the site map and gather as many endpoints as possible. To make our task a bit easier on ourselves, we can enable a few options. These will allow us to more easily recognize hidden fields and automatically work around those pesky JS checks such as a disabled field.

Go to proxy > options and scroll down until you see the option to show hidden form fields, prominently highlight them, enable the disabled fields, remove the field length limits



All of these controls are usually ignored by hackers anyway so we will do the same. I don't like using the other options since that is kinda messing too much with the response. For example, the secure flag is an actual server setting.

Now you can start filling up that site map by clicking around and making sure you touch on every functionality with the basic CRUD actions. (Create Read Update Delete).

Day 2: Install your plugins

Of course, the vanilla burp suite experience is really nice but the beauty partially comes from its extensibility. We can write our own extension from scratch but that would usually be counterproductive as really extensions exist covering pretty much any requirement. I have a few basic installed before I dive into hacking the functionality.

- Authorize
- Add custom header

    - Sometimes your client or target requires you to add a specific header. This header needs to be added to every request coming from all the tools within burp suite. We can satisfy this requirement by using the “Add custom header”.
- Autorepeater
- Distribute damage
    - This instills a rate limit on all the requests from burp suite. Required in case of rate limiting requirements.
- Broken link hijacker
    - This will look for link takeover
- Bypass WAF
- CMS scanner
- Git bridge
    - Adding version control to your burp save files is always welcome IMO
- Headless burp
    - Essential if you want to run burp’s spider and scanner through CI/CD pipelines
- NoSQLi scanner
    - With the rise of NoSQL databases, of course, we need the ability to scan for vulnerabilities if we are allowed to.
- Param miner
    - This extension identifies hidden, unlinked parameters. It's particularly useful for finding web cache poisoning vulnerabilities.
- Reflected parameters
    - A reflection could be the start of an XSS attack.
- SQLipy
    - Add sqlmap
- Upload scanner
- WAF Detect

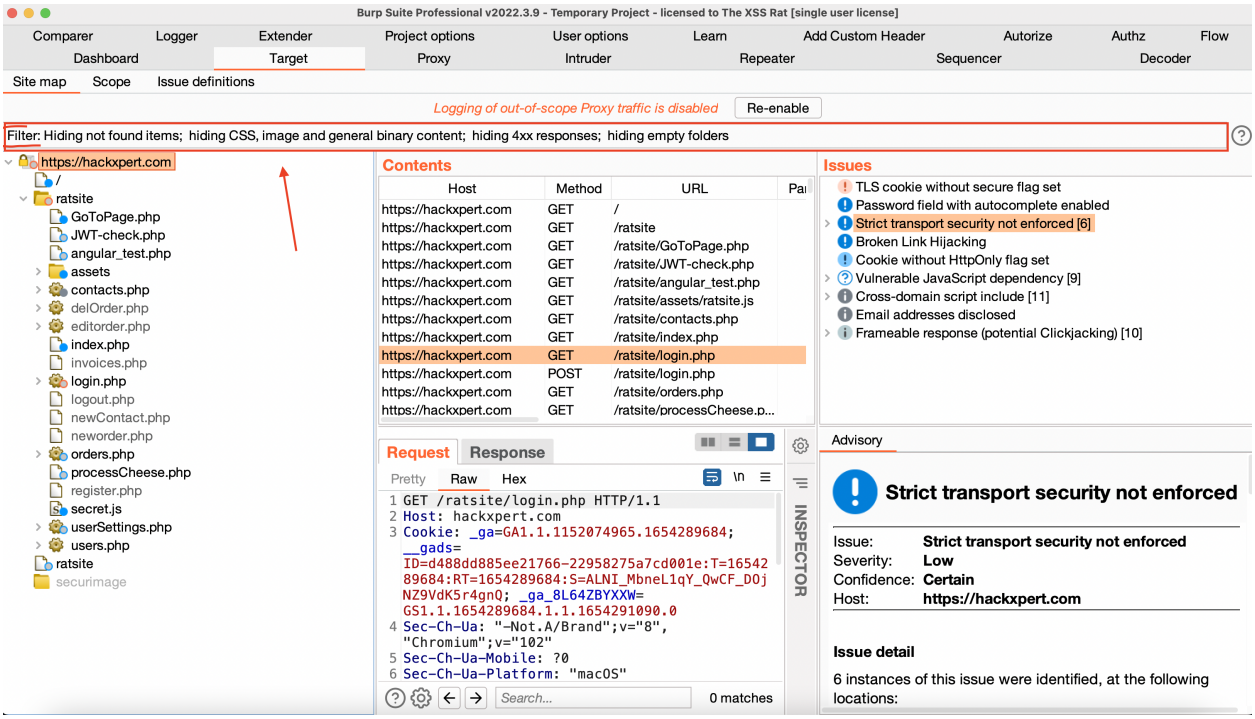
- Wordlist extractor
    - Scrapes all unique words and numbers for use with password cracking.

Day 3: Let’s explore

First of all, it’s important to not directly want to dive in and start hacking. Our first mission will be to fill up the site map, and we can only do that by walking the site manually in the community edition of burp suite. In the pro version, we might be able to use “content discovery” but I’d also keep that for later.

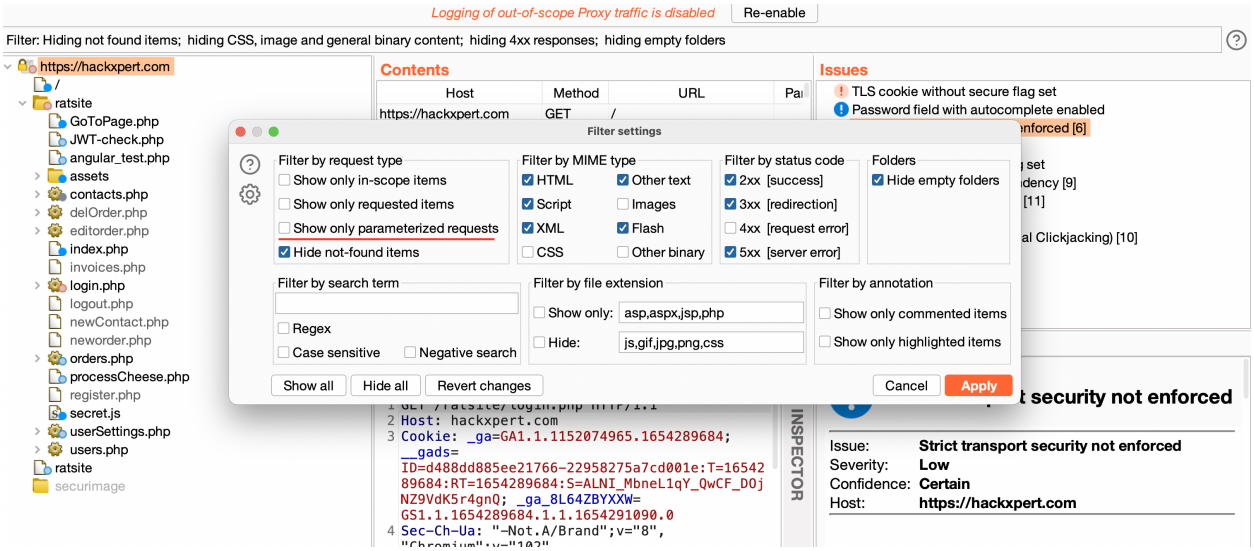
Click every link that you see, create accounts, log in and make sure you try everything. This will also help you get to know the functionality which will aid in better understanding exploits, their impact, and ways to find them. We can directly jump to business logic vulnerability testing but that is outside the scope of burp suite.

Day 4: Filtering out the good stuff



Now that our sitemap has been filled and we've explored the application, it is time to only show the requests that contain a parameter. This will help you focus on the more dynamic requests but don't forget to investigate the static requests as well for interaction with headers. No tunnel vision in our dear rat pack!

You can open the filters by clicking the "Filter:…". bar at the top of the screen.



We can activate several filters here which might be useful if you are looking for something specific. In the HTTP History, under the tab "Proxy" we have the same options.

We are now set to start exploring requests and using burp suite.