Burp Scanner

An essential feature of web proxy tools is their web scanners. Burp Suite comes with `Burp Scanner`, a powerful scanner for various types of web vulnerabilities, using a `Crawler` for building the website structure, and `Scanner` for passive and active scanning.

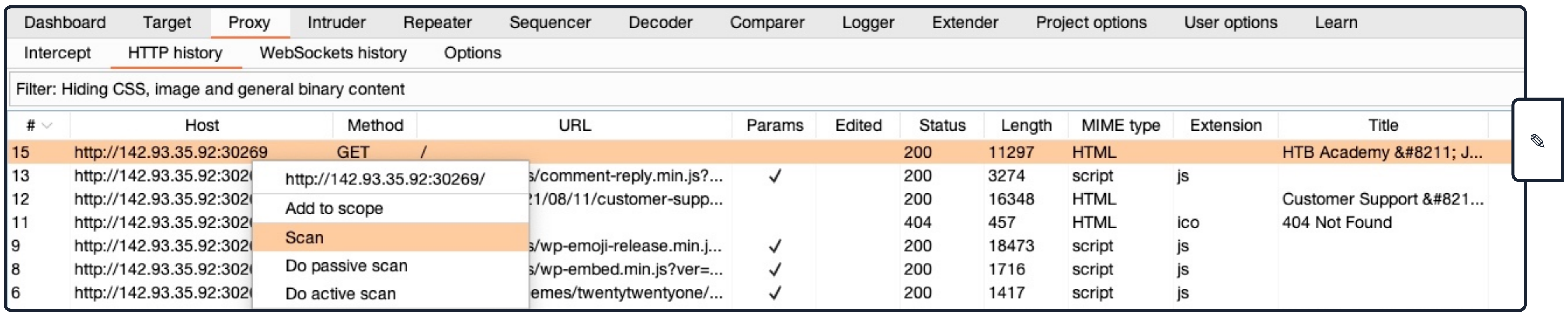
Burp Scanner is a Pro-Only feature, and it is not available in the free Community version of Burp Suite. However, given the wide scope that Burp Scanner covers and the advanced features it includes, it makes it an enterprise-level tool, and as such, it is expected to be a paid feature.

Target Scope

To start a scan in Burp Suite, we have the following options:

```
1. Start scan on a specific request from Proxy History
2. Start a new scan on a set of targets
3. Start a scan on items in-scope
```

To start a scan on a specific request from Proxy History, we can right-click on it once we locate it in the history, and then select `Scan` to be able to configure the scan before we run it, or select `Passive/Active Scan` to quickly start a scan with the default configurations:



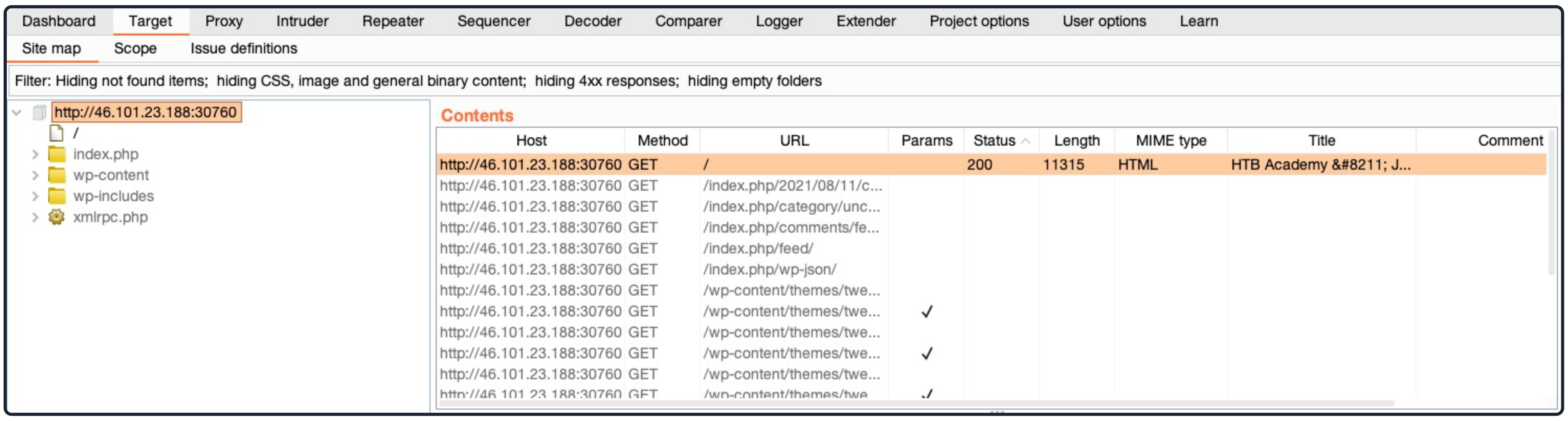
Filter: Hiding CSS, image and general binary content

#	Host	Method	URL	Params	Edited	Status	Length	MIME type	Extension	Title
15	http://142.93.35.92:30269	GET	/			200	11297	HTML		HTB Academy &#8211; J...
13	http://142.93.35.92:302(		s/comment-reply.min.js?...	✓		200	3274	script	js	
12	http://142.93.35.92:302(		21/08/11/customer-supp...			200	16348	HTML		Customer Support &#821...
11	http://142.93.35.92:302(					404	457	HTML	ico	404 Not Found
9	http://142.93.35.92:302(		s/wp-emoji-release.min.j...	✓		200	18473	script	js	
8	http://142.93.35.92:302(		s/wp-embed.min.js?ver=...	✓		200	1716	script	js	
6	http://142.93.35.92:302(		emes/twentytwentyone/...	✓		200	1417	script	js	

We may also click on the `New Scan` button on the `Dashboard` tab, which would open the `New Scan` configuration window to configure a scan on a set of custom targets. Instead of creating a custom scan from scratch, let's see how we can utilize the scope to properly define what's included/excluded from our scans using the `Target Scope`. The `Target Scope` can be utilized with all Burp features to define a custom set of targets that will be processed. Burp also allows us to limit Burp to in-scope items to save resources by ignoring any out-of-scope URLs.

Note: We will be scanning the web application from the exercise found at the end of the next section. If you obtain a license to use Burp Pro, you may spawn the target at the end of the next section and follow along here.

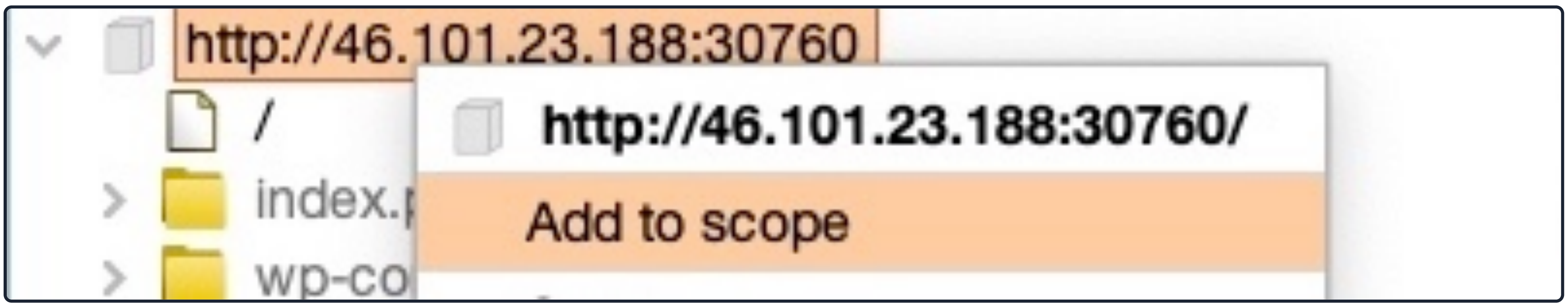
If we go to (`Target>Site map`), it will show a listing of all directories and files burp has detected in various requests that went through its proxy:



Filter: Hiding not found items; hiding CSS, image and general binary content; hiding 4xx responses; hiding empty folders

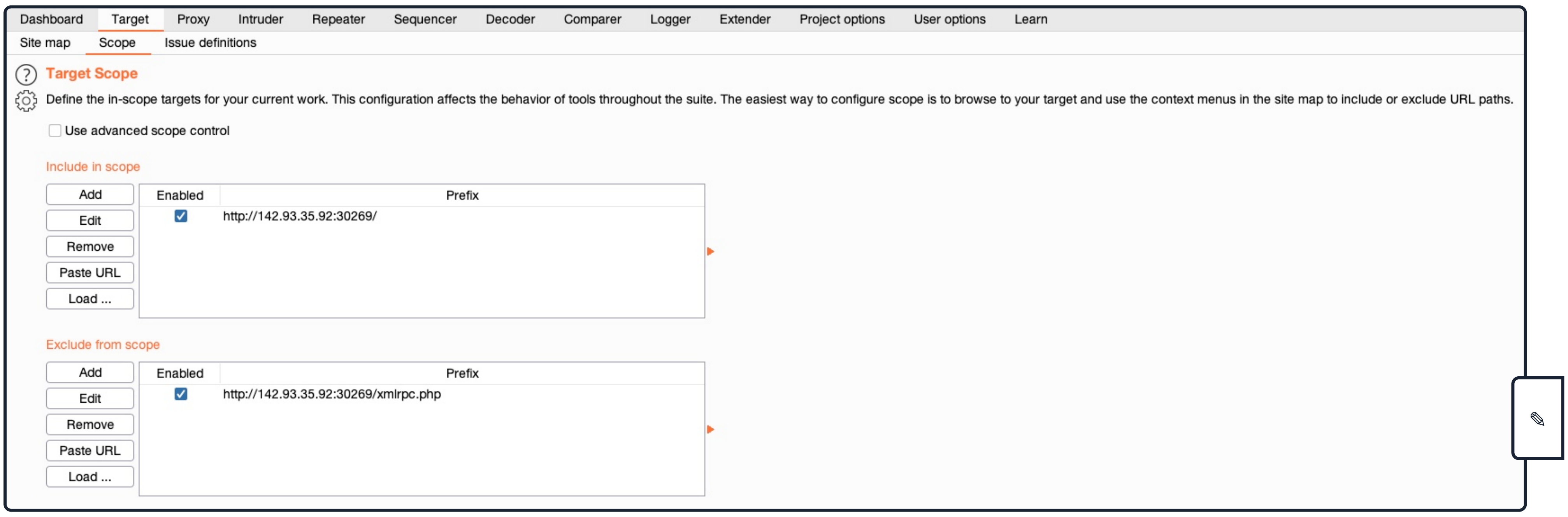
Host	Method	URL	Params	Status	Length	MIME type	Title	Comment
http://46.101.23.188:30760	GET	/		200	11315	HTML	HTB Academy &#8211; J...	
http://46.101.23.188:30760	GET	/index.php/2021/08/11/c...						
http://46.101.23.188:30760	GET	/index.php/category/unc...						
http://46.101.23.188:30760	GET	/index.php/comments/fe...						
http://46.101.23.188:30760	GET	/index.php/feed/						
http://46.101.23.188:30760	GET	/index.php/wp-json/						
http://46.101.23.188:30760	GET	/wp-content/themes/twe...						
http://46.101.23.188:30760	GET	/wp-content/themes/twe...	✓					
http://46.101.23.188:30760	GET	/wp-content/themes/twe...						
http://46.101.23.188:30760	GET	/wp-content/themes/twe...	✓					
http://46.101.23.188:30760	GET	/wp-content/themes/twe...						
http://46.101.23.188:30760	GET	/wp-content/themes/twe	✓					

To add an item to our scope, we can right-click on it and select `Add to scope`:



Note: When you add the first item to your scope, Burp will give you the option to restrict its features to in-scope items only, and ignore any out-of-scope items.

We may also need to exclude a few items from scope if scanning them may be dangerous or may end our session 'like a logout function'. To exclude an item from our scope, we can right-click on any in-scope item and select `Remove from scope`. Finally, we can go to (`Target>Scope`) to view the details of our scope. Here, we may also add/remove other items and use advanced scope control to specify regex patterns to be included/excluded.



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

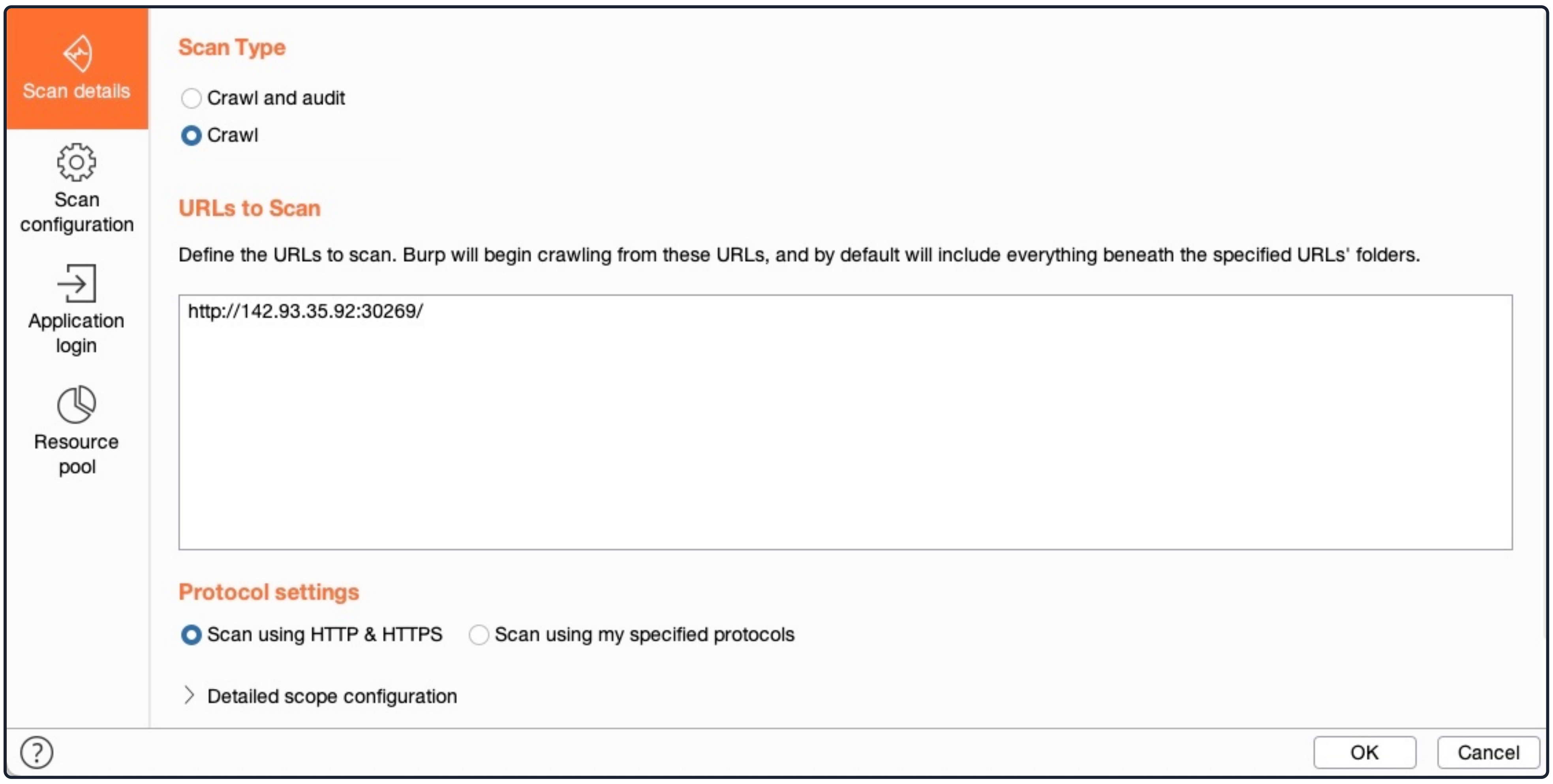
Enabled	Prefix
☑	http://142.93.35.92:30269/

Exclude from scope

Enabled	Prefix
☑	http://142.93.35.92:30269/xmlrpc.php

Crawler

Once we have our scope ready, we can go to the `Dashboard` tab and click on `New Scan` to configure our scan, which would be automatically populated with our in-scope items:



Scan Type

○ Crawl and audit

● Crawl

URLs to Scan

Define the URLs to scan. Burp will begin crawling from these URLs, and by default will include everything beneath the specified URLs' folders.

http://142.93.35.92:30269/

Protocol settings

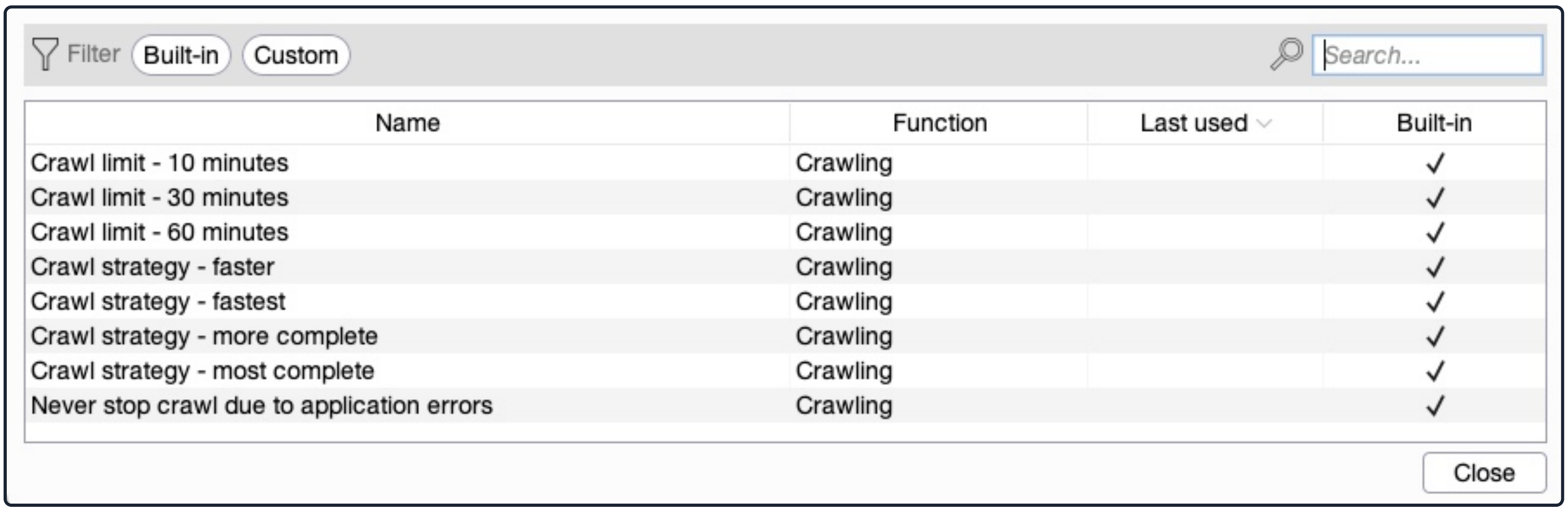
● Scan using HTTP & HTTPS ○ Scan using my specified protocols

> Detailed scope configuration

We see that Burp gives us two scanning options: `Crawl and Audit` and `Crawl`. A Web Crawler navigates a website by accessing any links found in its pages, accessing any forms, and examining any requests it makes to build a comprehensive map of the website. In the end, Burp Scanner presents us with a map of the target, showing all publicly accessible data in a single place. If we select `Crawl and Audit`, Burp will run its scanner after its Crawler (as we will see later).

Note: A Crawl scan only follows and maps links found in the page we specified, and any pages found on it. It does not perform a fuzzing scan to identify pages that are never referenced, like what dirbuster or ffuf would do. This can be done with Burp Intruder or Content Discovery, and then added to scope, if needed.

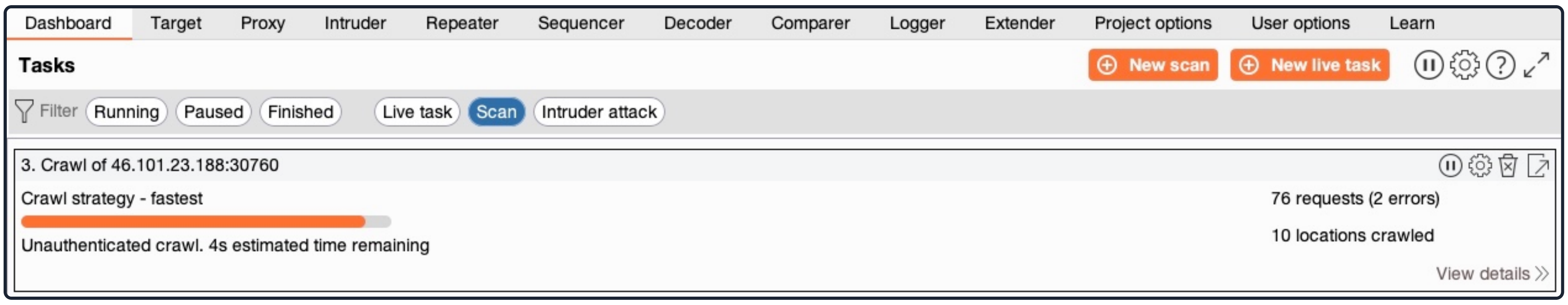
Let us select `Crawl` as a start and go to the `Scan configuration` tab to configure our scan. From here, we may choose to click on `New` to build a custom configuration, which would allow us to set the configurations like the crawling speed or limit, whether Burp will attempt to log in to any login forms, and a few other configurations. For the sake of simplicity, we will click on the `Select from library` button, which gives us a few preset configurations we can pick from (or custom configurations we previously defined):



Name	Function	Last used	Built-in
Crawl limit - 10 minutes	Crawling		✓
Crawl limit - 30 minutes	Crawling		✓
Crawl limit - 60 minutes	Crawling		✓
Crawl strategy - faster	Crawling		✓
Crawl strategy - fastest	Crawling		✓
Crawl strategy - more complete	Crawling		✓
Crawl strategy - most complete	Crawling		✓
Never stop crawl due to application errors	Crawling		✓

We will select the `Crawl strategy - fastest` option and continue to the `Application login` tab. In this tab, we can add a set of credentials for Burp to attempt in any Login forms/fields it can find. We may also record a set of steps by performing a manual login in the pre-configured browser, such that Burp knows what steps to follow to gain a login session. This can be essential if we were running our scan using an authenticated user, which would allow us to cover parts of the web application that Burp may otherwise not have access to. As we do not have any credentials, we'll leave it empty.

With that, we can click on the `Ok` button to start our Crawl scan. Once our scan starts, we can see its progress in the `Dashboard` tab under `Tasks`:



3. Crawl of 46.101.23.188:30760

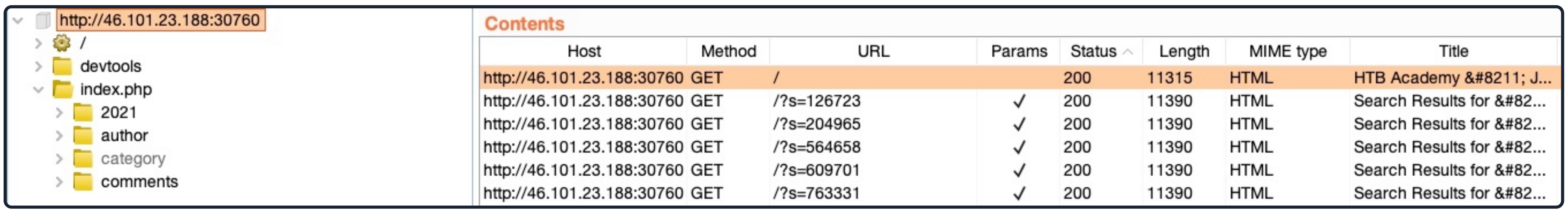
Crawl strategy - fastest

Unauthenticated crawl. 4s estimated time remaining

76 requests (2 errors)

10 locations crawled

We may also click on the `View details` button on the tasks to see more details about the running scan or click on the gear icon to customize our scan configurations further. Finally, once our scan is complete, we'll see `Crawl Finished` in the task info, and then we can go back to (`Target>Site map`) to view the updated site map:



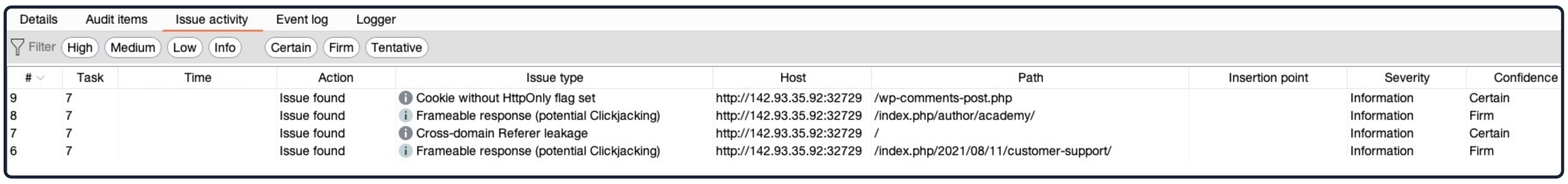
Host	Method	URL	Params	Status	Length	MIME type	Title
http://46.101.23.188:30760	GET	/		200	11315	HTML	HTB Academy &#8211; J...
http://46.101.23.188:30760	GET	/?s=126723	✓	200	11390	HTML	Search Results for &#82...
http://46.101.23.188:30760	GET	/?s=204965	✓	200	11390	HTML	Search Results for &#82...
http://46.101.23.188:30760	GET	/?s=564658	✓	200	11390	HTML	Search Results for &#82...
http://46.101.23.188:30760	GET	/?s=609701	✓	200	11390	HTML	Search Results for &#82...
http://46.101.23.188:30760	GET	/?s=763331	✓	200	11390	HTML	Search Results for &#82...

Passive Scanner

Now that the site map is fully built, we may select to scan this target for potential vulnerabilities. When we choose the `Crawl and Audit` option in the `New Scan` dialog, Burp will perform two types of scans: A `Passive Vulnerability Scan` and an `Active Vulnerability Scan`.

Unlike an Active Scan, a Passive Scan does not send any new requests but analyzes the source of pages already visited in the target/scope and then tries to identify `potential` vulnerabilities. This is very useful for a quick analysis of a specific target, like missing HTML tags or potential DOM-based XSS vulnerabilities. However, without sending any requests to test and verify these vulnerabilities, a Passive Scan can only suggest a list of potential vulnerabilities. Still, Burp Passive Scanner does provide a level of `Confidence` for each identified vulnerability, which is also helpful for prioritizing potential vulnerabilities.

Let's start by trying to perform a Passive Scan only. To do so, we can once again select the target in (`Target>Site map`) or a request in Burp Proxy History, then right-click on it and select `Do passive scan` or `Passively scan this target`. The Passive Scan will start running, and its task can be seen in the `Dashboard` tab as well. Once the scan finishes, we can click on `View Details` to review identified vulnerabilities and then select the `Issue activity` tab:



#	Task	Time	Action	Issue type	Host	Path	Insertion point	Severity	Confidence
9	7		Issue found	Cookie without HttpOnly flag set	http://142.93.35.92:32729	/wp-comments-post.php		Information	Certain
8	7		Issue found	Frameable response (potential Clickjacking)	http://142.93.35.92:32729	/index.php/author/academy/		Information	Firm
7	7		Issue found	Cross-domain Referer leakage	http://142.93.35.92:32729	/		Information	Certain
6	7		Issue found	Frameable response (potential Clickjacking)	http://142.93.35.92:32729	/index.php/2021/08/11/customer-support/		Information	Firm

Alternately, we can view all identified issues in the `Issue activity` pane on the `Dashboard` tab. As we can see, it shows the list of potential vulnerabilities, their severity, and their confidence. Usually, we want to look for vulnerabilities with `High` severity and `Certain` confidence. However, we should include all levels of severity and confidence for very sensitive web applications, with a special focus on `High` severity and `Confident/Firm` confidence.

Active Scanner

We finally reach the most powerful part of Burp Scanner, which is its Active Vulnerability Scanner. An active scan runs a more comprehensive scan than a Passive Scan, as follows:

1. It starts by running a Crawl and a web fuzzer (like dirbuster/ffuf) to identify all possible pages
2. It runs a Passive Scan on all identified pages
3. It checks each of the identified vulnerabilities from the Passive Scan and sends requests to verify them
4. It performs a JavaScript analysis to identify further potential vulnerabilities
5. It fuzzes various identified insertion points and parameters to look for common vulnerabilities like XSS, Command Injection, SQL Injection, and other common web vulnerabilities

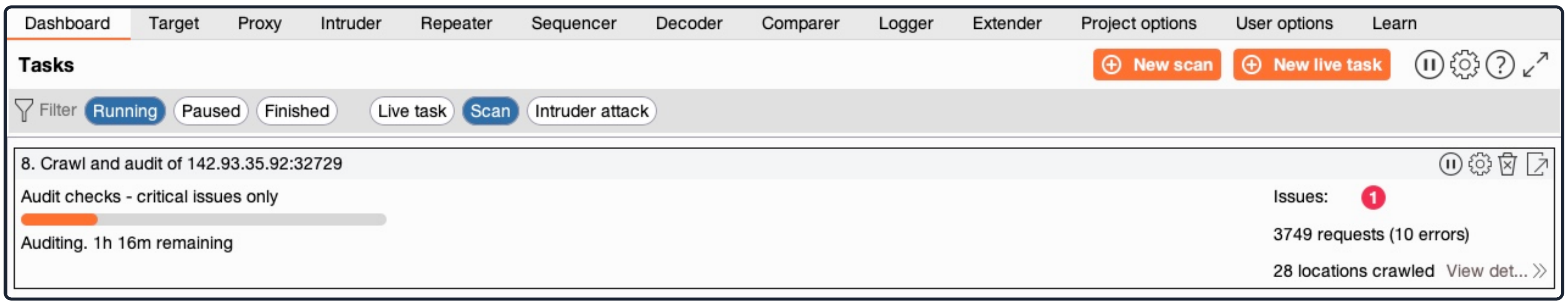
The Burp Active scanner is considered one of the best tools in that field and is frequently updated to scan for newly identified web vulnerabilities by the Burp research team.

We can start an Active Scan similarly to how we began a Passive Scan by selecting the `Do active scan` from the right-click menu on a request in Burp Proxy History. Alternatively, we can run a scan on our scope with the `New Scan` button in the `Dashboard` tab, which would allow us to configure our active scan. This time, we will select the `Crawl and Audit` option, which would perform all of the above points and everything we have discussed so far.

We may also set the Crawl configurations (as we discussed earlier) and the Audit configurations. The Audit configurations enable us to select what type of vulnerabilities we want to scan (defaults to all), where the scanner would attempt inserting its payloads, in addition to many other useful configurations. Once again, we can select a configuration preset with the `Select from library` button. For our test, as we are interested in `High` vulnerabilities that may allow us to gain control over the backend server, we will select the `Audit checks -`

critical issues only option. Finally, we may add login details, as we previously saw with the Crawl configurations.

Once we select our configurations, we can click on the Ok button to start the scan, and the active scan task should be added in the Tasks pane in the Dashboard tab:



8. Crawl and audit of 142.93.35.92:32729

Audit checks - critical issues only

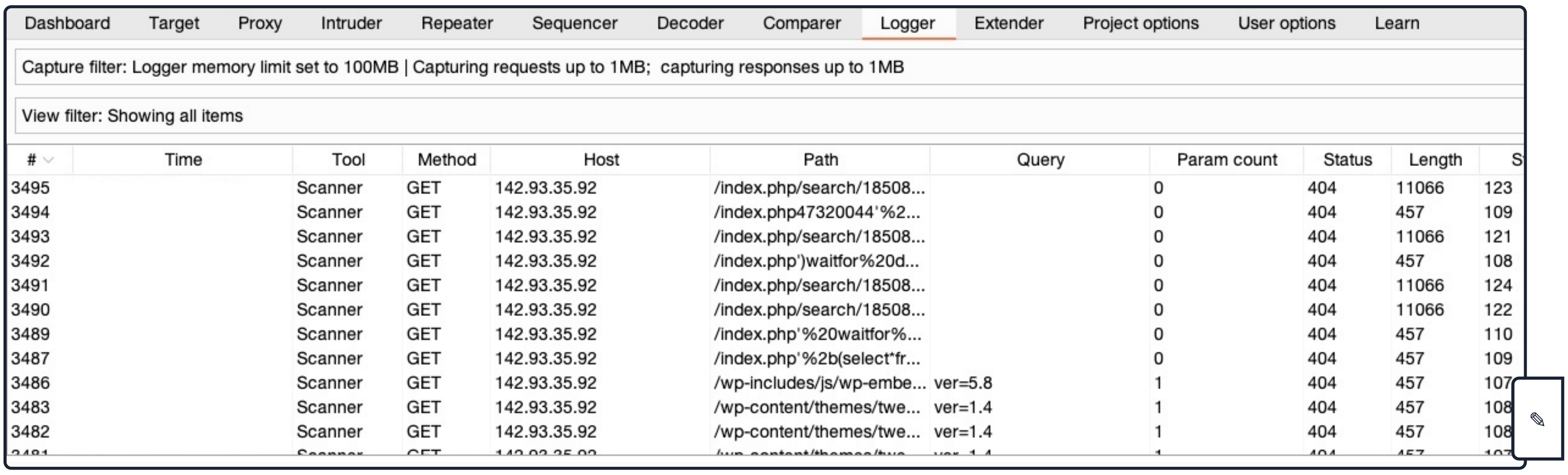
Auditing. 1h 16m remaining

Issues: 1

3749 requests (10 errors)

28 locations crawled View det...

The scan will run all of the steps mentioned above, which is why it will take significantly longer to finish than our earlier scans depending on the configurations we selected. As the scan is running, we can view the various requests it is making by clicking on the View details button and selecting the Logger tab, or by going to the Logger tab in Burp, which shows all requests that went through or were made by Burp:

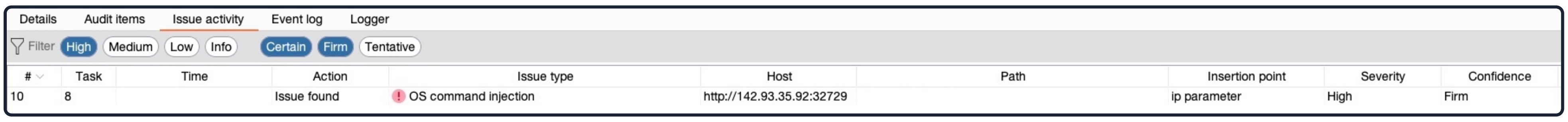


Capture filter: Logger memory limit set to 100MB | Capturing requests up to 1MB; capturing responses up to 1MB

View filter: Showing all items

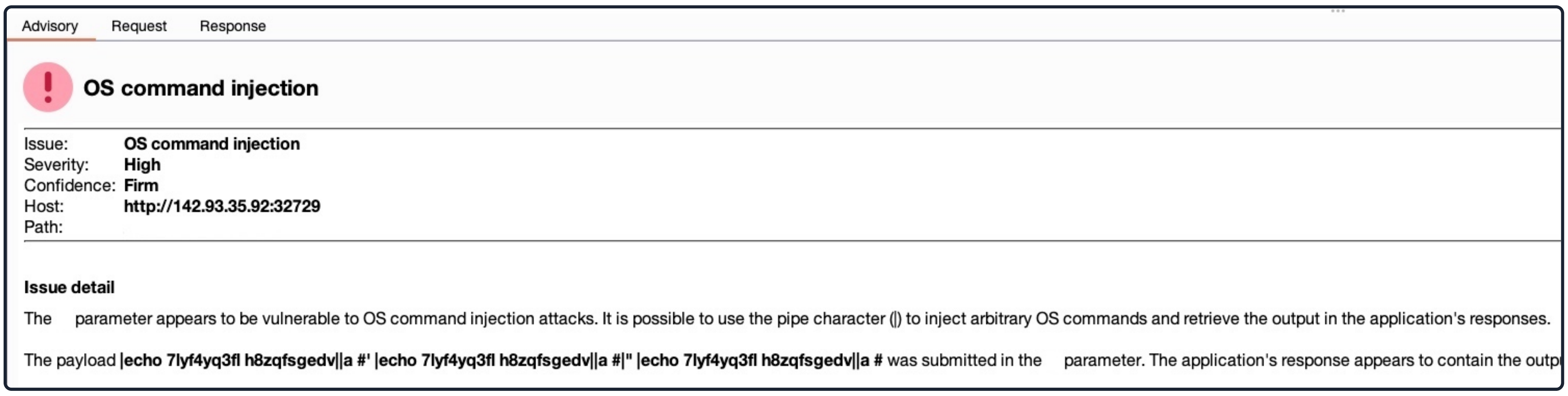
#	Time	Tool	Method	Host	Path	Query	Param count	Status	Length	S
3495		Scanner	GET	142.93.35.92	/index.php/search/18508...		0	404	11066	123
3494		Scanner	GET	142.93.35.92	/index.php47320044'%2...		0	404	457	109
3493		Scanner	GET	142.93.35.92	/index.php/search/18508...		0	404	11066	121
3492		Scanner	GET	142.93.35.92	/index.php')waitfor%20d...		0	404	457	108
3491		Scanner	GET	142.93.35.92	/index.php/search/18508...		0	404	11066	124
3490		Scanner	GET	142.93.35.92	/index.php/search/18508...		0	404	11066	122
3489		Scanner	GET	142.93.35.92	/index.php'%20waitfor%...		0	404	457	110
3487		Scanner	GET	142.93.35.92	/index.php'%2b(select*fr...		0	404	457	109
3486		Scanner	GET	142.93.35.92	/wp-includes/js/wp-embe...	ver=5.8	1	404	457	107
3483		Scanner	GET	142.93.35.92	/wp-content/themes/twe...	ver=1.4	1	404	457	108
3482		Scanner	GET	142.93.35.92	/wp-content/themes/twe...	ver=1.4	1	404	457	108

Once the scan is done, we can look at the Issue activity pane in the Dashboard tab to view and filter all of the issues identified so far. From the filter above the results, let's select High and Certain and see our filtered results:



#	Task	Time	Action	Issue type	Host	Path	Insertion point	Severity	Confidence
10	8		Issue found	! OS command injection	http://142.93.35.92:32729		ip parameter	High	Firm

We see that Burp identified an OS command injection vulnerability, which is ranked with a High severity and Firm confidence. As Burp is firmly confident that this severe vulnerability exists, we can read about it by clicking on it and reading the advisory shown and view the sent request and received response, to be able to know whether the vulnerability can be exploited or how it poses a threat on the webserver:



OS command injection

Issue: OS command injection
Severity: High
Confidence: Firm
Host: http://142.93.35.92:32729
Path:

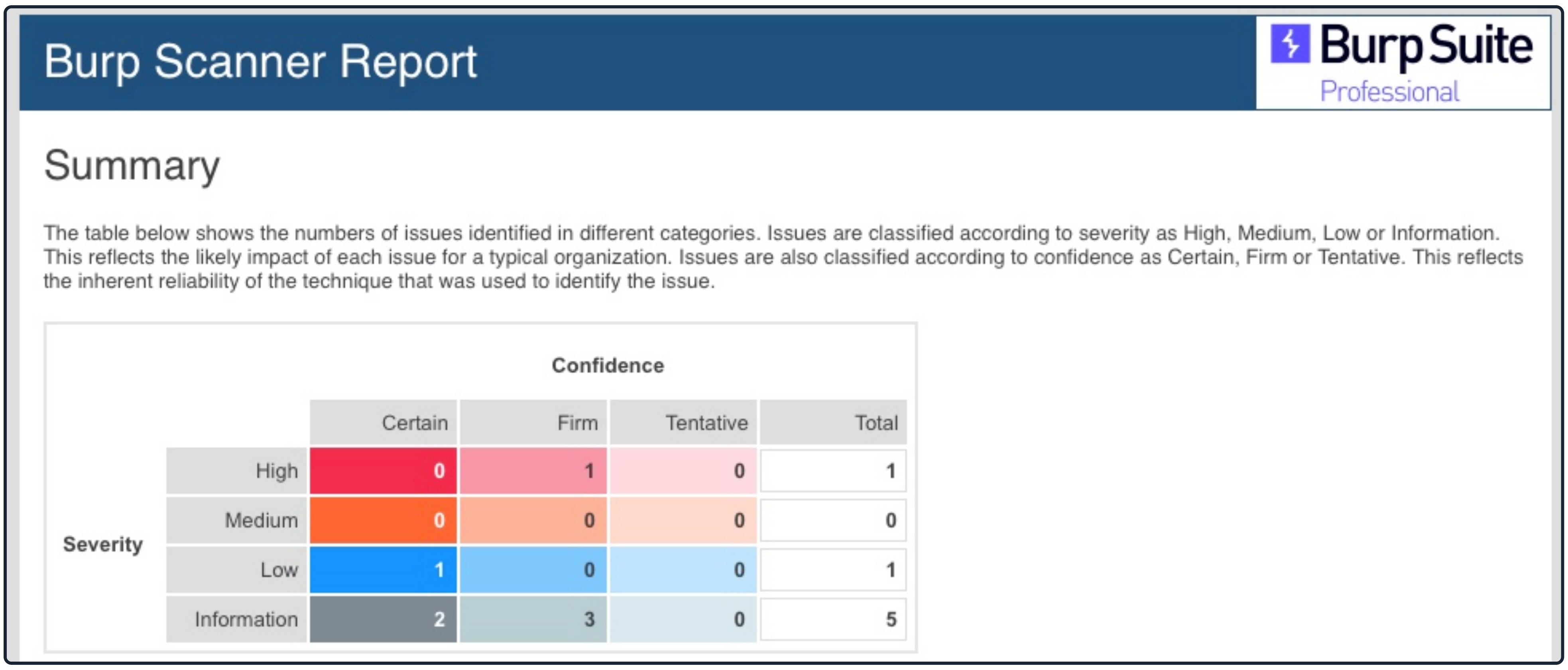
Issue detail

The parameter appears to be vulnerable to OS command injection attacks. It is possible to use the pipe character (|) to inject arbitrary OS commands and retrieve the output in the application's responses.

The payload **|echo 7lyf4yq3fl h8zqfsgedv||a #' |echo 7lyf4yq3fl h8zqfsgedv||a #|" |echo 7lyf4yq3fl h8zqfsgedv||a #** was submitted in the parameter. The application's response appears to contain the outp

Reporting

Finally, once all of our scans are completed, and all potential issues have been identified, we can go to (Target>Site map), right-click on our target, and select (Issue>Report issues for this host). We will get prompted to select the export type for the report and what information we would like to include in the report. Once we export the report, we can open it in any web browser to view its details:



Burp Scanner Report

Summary

The table below shows the numbers of issues identified in different categories. Issues are classified according to severity as High, Medium, Low or Information. This reflects the likely impact of each issue for a typical organization. Issues are also classified according to confidence as Certain, Firm or Tentative. This reflects the inherent reliability of the technique that was used to identify the issue.

		Confidence			
		Certain	Firm	Tentative	Total
Severity	High	0	1	0	1
	Medium	0	0	0	0
	Low	1	0	0	1
	Information	2	3	0	5

As we can see, Burp's report is very organized and can be customized to only include select issues by severity/confidence. It also shows proof-of-concept details of how to exploit the vulnerability and information on how to remediate it. These reports may be used as supplementary data for the detailed reports that we prepare for our clients or the web application developers when performing a web penetration test or can be stored for our future reference. We should never merely export a report from any penetration tool and submit it to a client as the final deliverable. Instead, the reports and data generated by tools can be helpful as appendix data for clients who may need the raw scan data for remediation efforts or to import into a tracking dashboard.

← Previous Next →

+10 Streak pts ✓ Mark Complete & Next

Cheat Sheet

Table of Contents

Getting Started

Web Proxy

Web Fuzzer

Web Scanner

Burp Scanner

ZAP Scanner

Extensions

Skills Assessment

Skills Assessment - Using Web Proxies

My Workstation

OFFLINE

▶ Start Instance

∞ / 1 spawns left

✎