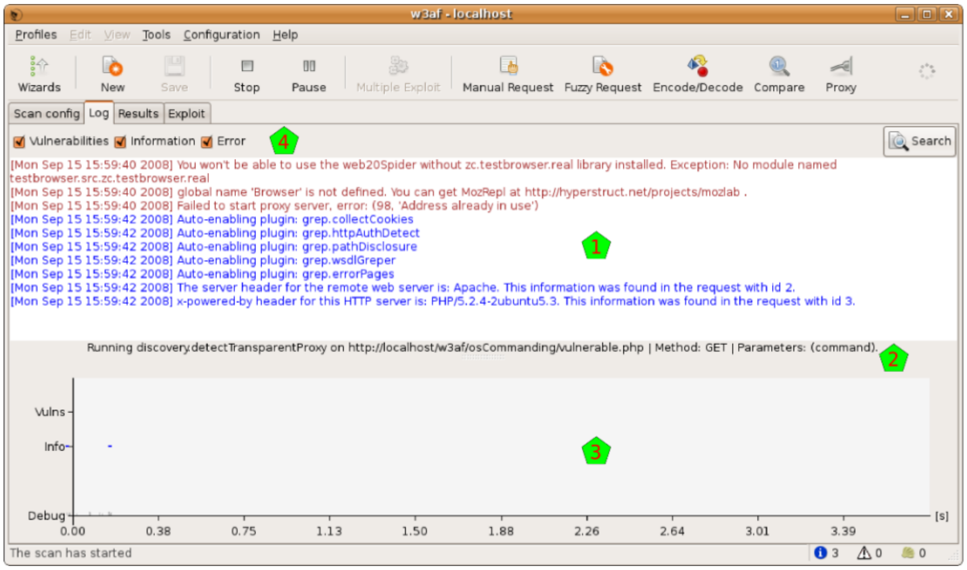
# W3af

- W3af (web application attack and audit framework) is an open-source web application security scanner.
- The project provides a vulnerability scanner and exploitation tool for Web applications.
- It provides information about security vulnerabilities and aids in penetration testing effort.
- Users have the choice between graphic user interface  and a command-line interface.
- W3af identifies most web application vulnerabilities using more than 130 plug-ins. After identification vulnerabilities like SQL injection, OS commanding, remote file inclusion, cross-site scripting, and unsafe file uploads, can be exploited in order to gain different types of access to the remote.

## Running the scan:



```
[Mon Sep 15 15:59:40 2008] You won't be able to use the web20Spider without zc.testbrowser.real library installed. Exception: No module named
testbrowser.src.zc.testbrowser.real
[Mon Sep 15 15:59:40 2008] global name 'Browser' is not defined. You can get MozRepl at http://hyperstruct.net/projects/mozlab .
[Mon Sep 15 15:59:40 2008] Failed to start proxy server, error: (98, 'Address already in use')
[Mon Sep 15 15:59:42 2008] Auto-enabling plugin: grep.collectCookies
[Mon Sep 15 15:59:42 2008] Auto-enabling plugin: grep.httpAuthDetect
[Mon Sep 15 15:59:42 2008] Auto-enabling plugin: grep.pathDisclosure
[Mon Sep 15 15:59:42 2008] Auto-enabling plugin: grep.wsdlGreper
[Mon Sep 15 15:59:42 2008] Auto-enabling plugin: grep.errorPages
[Mon Sep 15 15:59:42 2008] The server header for the remote web server is: Apache. This information was found in the request with id 2.
[Mon Sep 15 15:59:42 2008] x-powered-by header for this HTTP server is: PHP/5.2.4-2ubuntu5.3. This information was found in the request with id 3.
```

Running discovery.detectTransparentProxy on http://localhost/w3af/osCommanding/vulnerable.php | Method: GET | Parameters: (command).

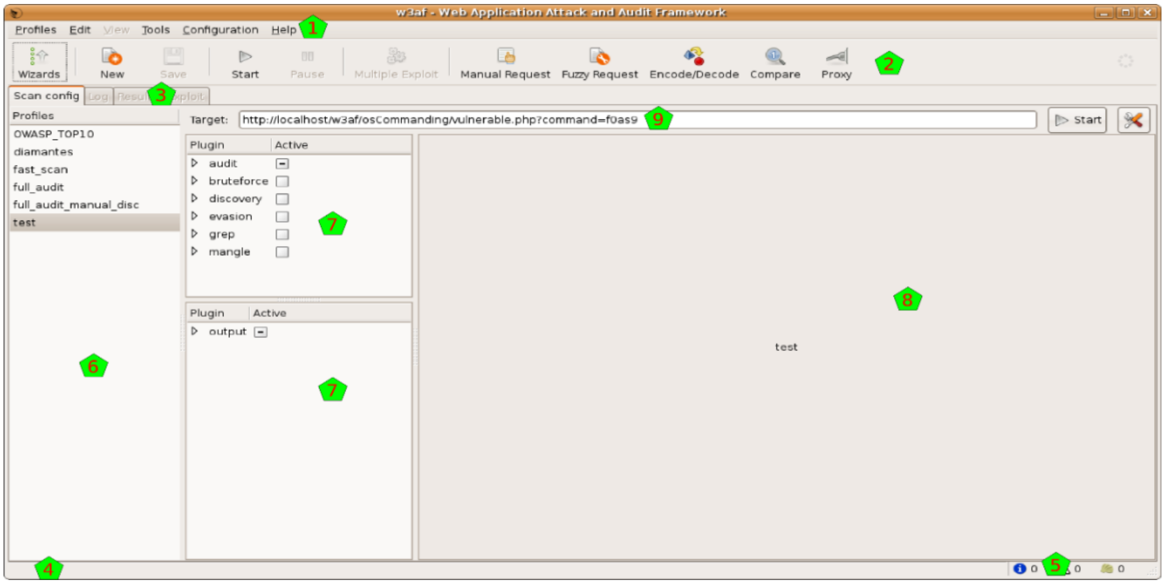
The scan has started

## Analyzing results:

- You can explore and analyze the scanning results after the scan process is completed (or before it's finished, because the system let's you work concurrently with that process).

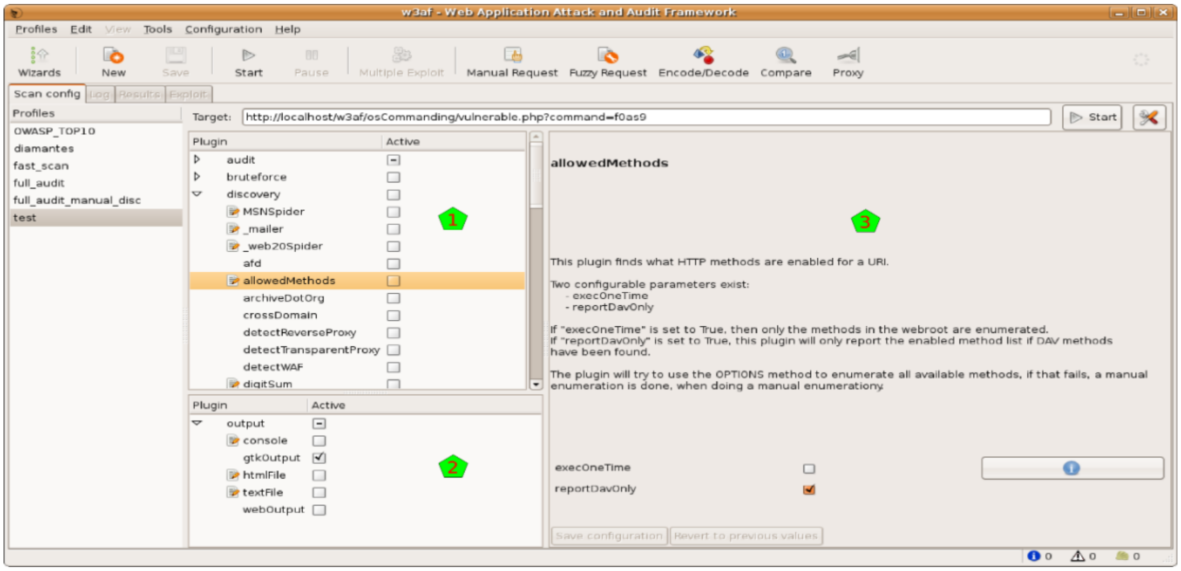
## General structure:

- Menu bar
- Toolbar
- Notebook Ta
- Toolbar
- Found element
- The Profile
- The Plugins Configuration Area
- Target URL.



Target: http://localhost/w3af/osCommanding/vulnerable.php?command=f0as9

- **Configuring the scan**
- To scan the web sites in different ways there are different plugins that can be configured in different ways.



Target: http://localhost/w3af/osCommanding/vulnerable.php?command=f0as9

gtkOutput ☑

**allowedMethods**

This plugin finds what HTTP methods are enabled for a URI.

Two configurable parameters exist:
 - execOneTime
 - reportDavOnly

If "execOneTime" is set to True, then only the methods in the webroot are enumerated.
If "reportDavOnly" is set to True, this plugin will only report the enabled method list if DAV methods have been found.

The plugin will try to use the OPTIONS method to enumerate all available methods, if that fails, a manual enumeration is done, when doing a manual enumerationy.

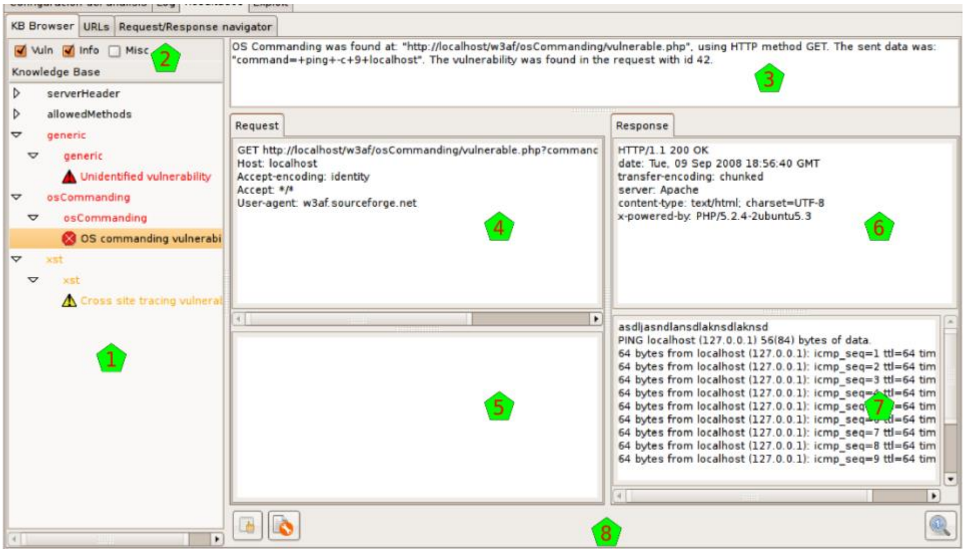
execOneTime ☐
reportDavOnly ☑

- The first section has all the scan plugins, in the upper part of the column. There you have the different plugins grouped regarding the scan type. They are separated in:
- `audit`
- `bruteforce`
- `crawl`

- `infrastructure`
- `evasion`
- `grep`
- `mangle`
- `output`

- **Using the profiles**

In the profiles you can save different configurations.We can think a Profile as the collection of configured plugins and target URL. In the column of the left we can see which plugins do we have:



OS Commanding was found at: "http://localhost/w3af/osCommanding/vulnerable.php", using HTTP method GET. The sent data was: "command=+ping+-c+9+localhost". The vulnerability was found in the request with id 42.

```
GET http://localhost/w3af/osCommanding/vulnerable.php?commanc
Host: localhost
Accept-encoding: identity
Accept: */*
User-agent: w3af.sourceforge.net
```

```
HTTP/1.1 200 OK
date: Tue, 09 Sep 2008 18:56:40 GMT
transfer-encoding: chunked
server: Apache
content-type: text/html; charset=UTF-8
x-powered-by: PHP/5.2.4-2ubuntu5.3
```

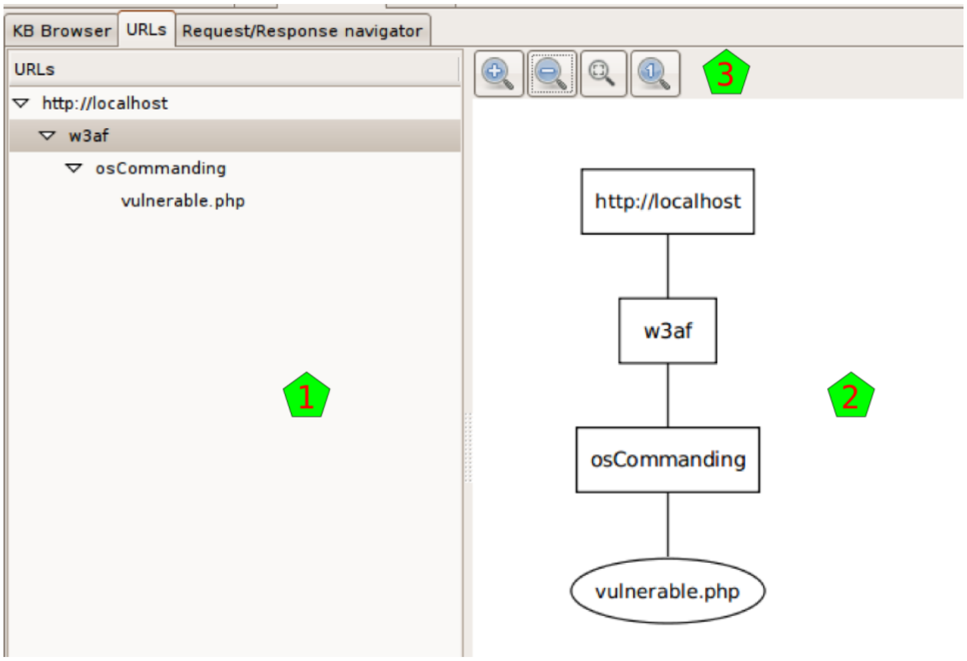
```
asdljasndlansdlaknsdlaknsd
PING localhost (127.0.0.1) 56(84) bytes of data.
64 bytes from localhost (127.0.0.1): icmp_seq=1 ttl=64 tim
64 bytes from localhost (127.0.0.1): icmp_seq=2 ttl=64 tim
64 bytes from localhost (127.0.0.1): icmp_seq=3 ttl=64 tim
64 bytes from localhost (127.0.0.1): icmp_seq=4 ttl=64 tim
64 bytes from localhost (127.0.0.1): icmp_seq[illegible]=64 tim
64 bytes from localhost (127.0.0.1): icmp_seq=[illegible]tl=64 tim
64 bytes from localhost (127.0.0.1): icmp_seq=7 ttl=64 tim
64 bytes from localhost (127.0.0.1): icmp_seq=8 ttl=64 tim
64 bytes from localhost (127.0.0.1): icmp_seq=9 ttl=64 tim
```

## Browsing the Knowledge Base

The Knowledge Base is a collection of discovered items, that can be classified in Vulnerabilities, Informations, and other stuff. The KB Browser tab lets you dive into this information.

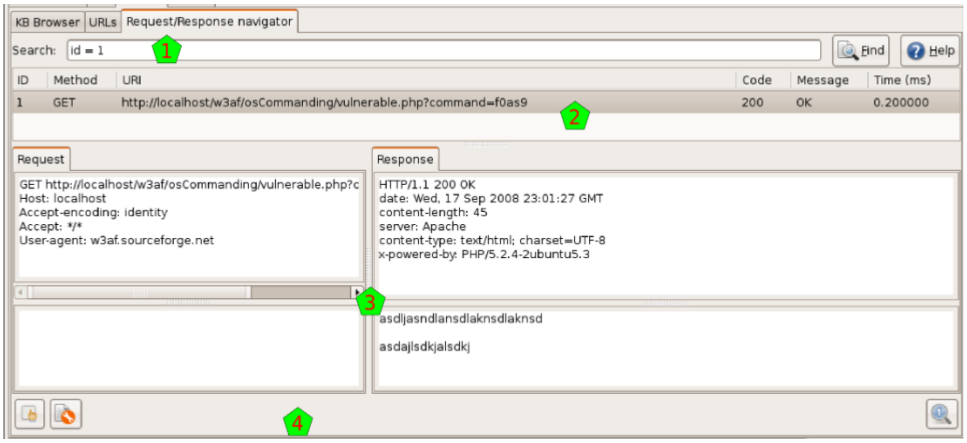
## Site structure

The URLs tab shows the structure of the site that the system worked on. It’s separated In two parts, but both parts show actually the same information, although they show it in different ways.



## Requests and responses

In this window you will be able to search for any request (and the associated response) that the system had generated during the scanning.



Search: id = 1

| ID | Method | URI | Code | Message | Time (ms) |
|---|---|---|---|---|---|
| 1 | GET | http://localhost/w3af/osCommanding/vulnerable.php?command=f0as9 | 200 | OK | 0.200000 |

```
GET http://localhost/w3af/osCommanding/vulnerable.php?c
Host: localhost
Accept-encoding: identity
Accept: */*
User-agent: w3af.sourceforge.net
```

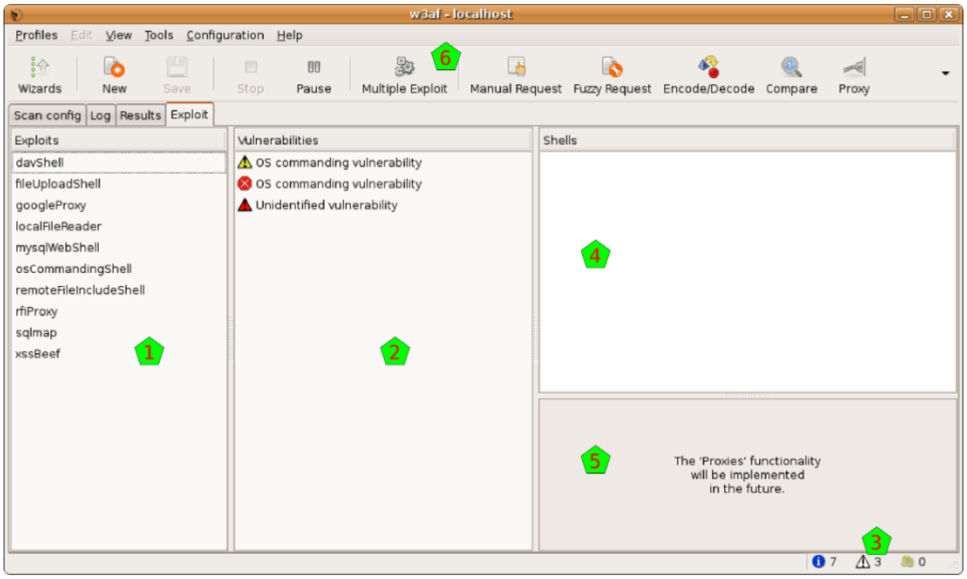
```
HTTP/1.1 200 OK
date: Wed, 17 Sep 2008 23:01:27 GMT
content-length: 45
server: Apache
content-type: text/html; charset=UTF-8
x-powered-by: PHP/5.2.4-2ubuntu5.3
```

```
asdljasndlansdlaknsdlaknsd

asdajlsdkjalsdkj
```

## Exploitation

When the scan is running or after the scan finished running, as you can check the results, you also can start with the exploitation. For this, go to the fourth tab in the system, called Exploit:



Exploits

- davShell
- fileUploadShell
- googleProxy
- localFileReader
- mysqlWebShell
- osCommandingShell
- remoteFileIncludeShell
- rfiProxy
- sqlmap
- xssBeef

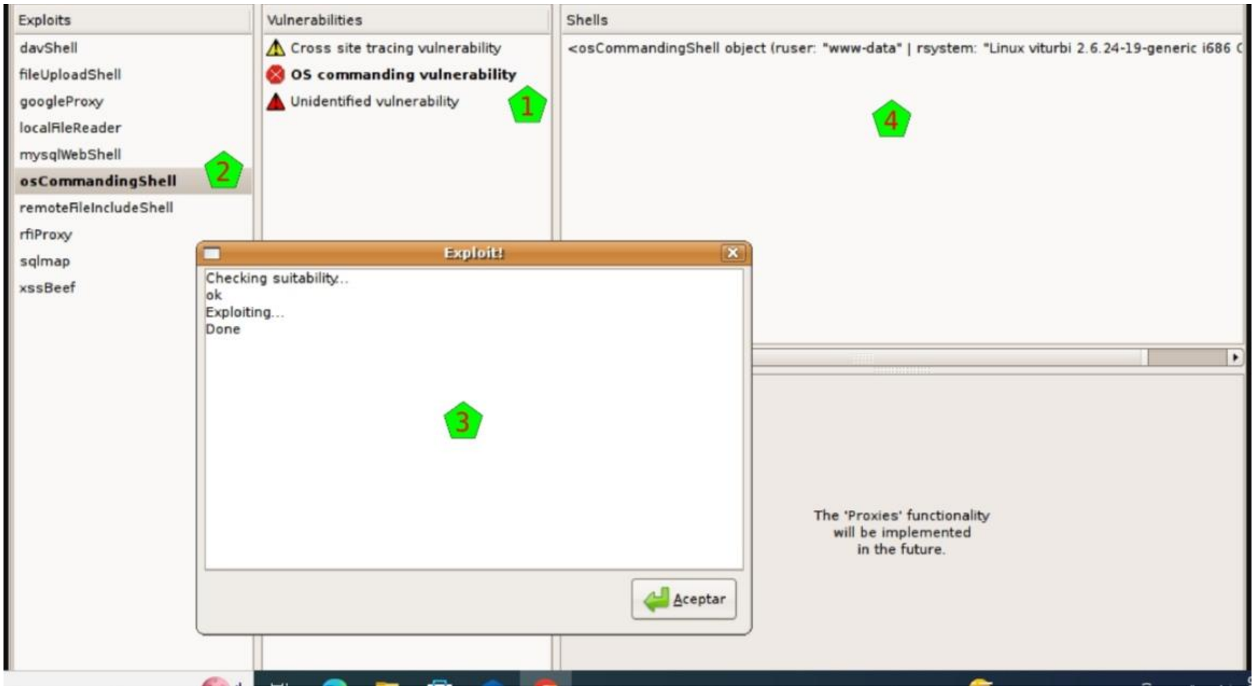
Vulnerabilities

- ⚠ OS commanding vulnerability
- ⊗ OS commanding vulnerability
- ▲ Unidentified vulnerability

The 'Proxies' functionality
will be implemented
in the future.

## Executing an exploit:

Exploits act on vulnerabilities. But not all exploits act on every vulnerabilities. It is well known if any exploit could act on some vulnerability, though, but to be sure and actually exploit it some verification needs to be done. Fortunately, the system easies very much this process to you.system easies very much this process to you.



```
<osCommandingShell object (ruser: "www-data" | rsystem: "Linux viturbi 2.6.24-19-generic i686 (
```

```
Checking suitability...
ok
Exploiting...
Done
```

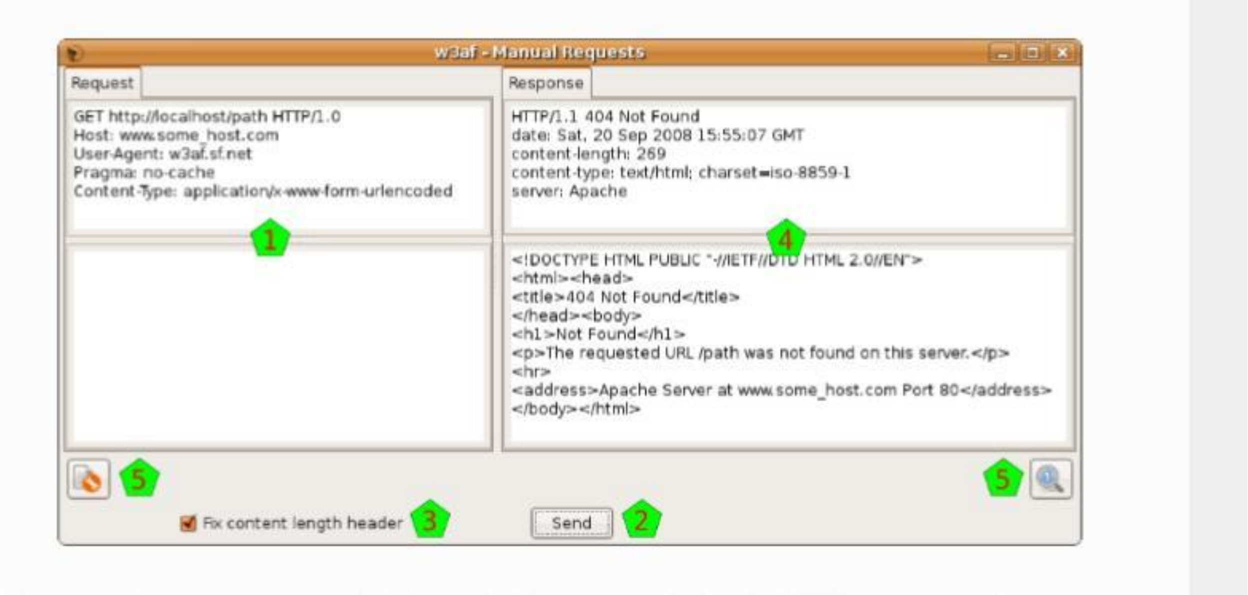
The 'Proxies' functionality
will be implemented
in the future.

## Tools

Apart from the w3af core functionality, that is to scan for vulnerabilities and exploit them, there are other tools that help you in the day by day work.

## Manual Requests

This tool lets you write and send HTTP requests.



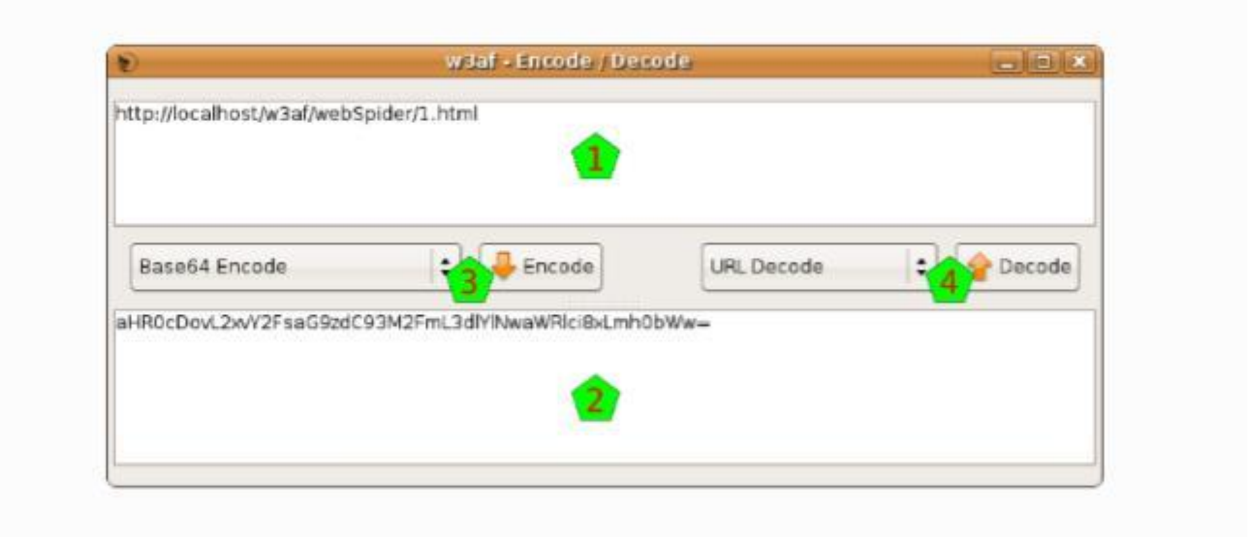
```
GET http://localhost/path HTTP/1.0
Host: www.some_host.com
User-Agent: w3af.sf.net
Pragma: no-cache
Content-Type: application/x-www-form-urlencoded
```

```
HTTP/1.1 404 Not Found
date: Sat, 20 Sep 2008 15:55:07 GMT
content-length: 269
content-type: text/html; charset=iso-8859-1
server: Apache
```

```
<!DOCTYPE HTML PUBLIC "-//IETF//DTD HTML 2.0//EN">
<html><head>
<title>404 Not Found</title>
</head><body>
<h1>Not Found</h1>
<p>The requested URL /path was not found on this server.</p>
<hr>
<address>Apache Server at www.some_host.com Port 80</address>
</body></html>
```

## Encode and Decode

This tool allows you to apply a lot of encoding and decoding functions in the text that you want.



```
http://localhost/w3af/webSpider/1.html
```

```
aHR0cDovL2xvY2FsaG9zdC93M2FmL3dlYlNwaWRlci8xLmh0bWw=
```

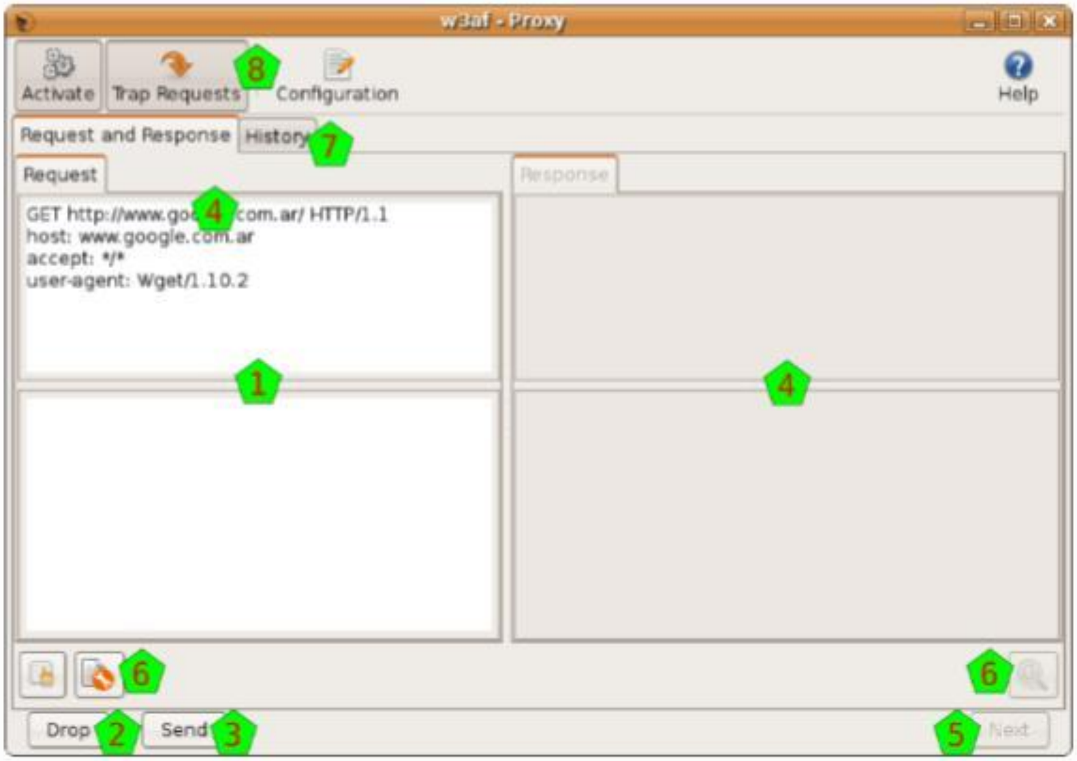
## Comparing HTTP traffic

With this tool you will be able to compare different requests and responses.

The Comparator window is separated mainly in two panes: both request and responses that you’re comparing. In this tool all the information is concatenated in the same text, to ease the comparison, but you have four buttons to control which part of the information appear in the text: request headers, request body, response headers, and response body.

## Using the Proxy

This tool is a proxy that listen to a port in the machine you’re running the w3af program. You can configure any program that issues HTTP request (like your internet browser, for example) to use this proxy.



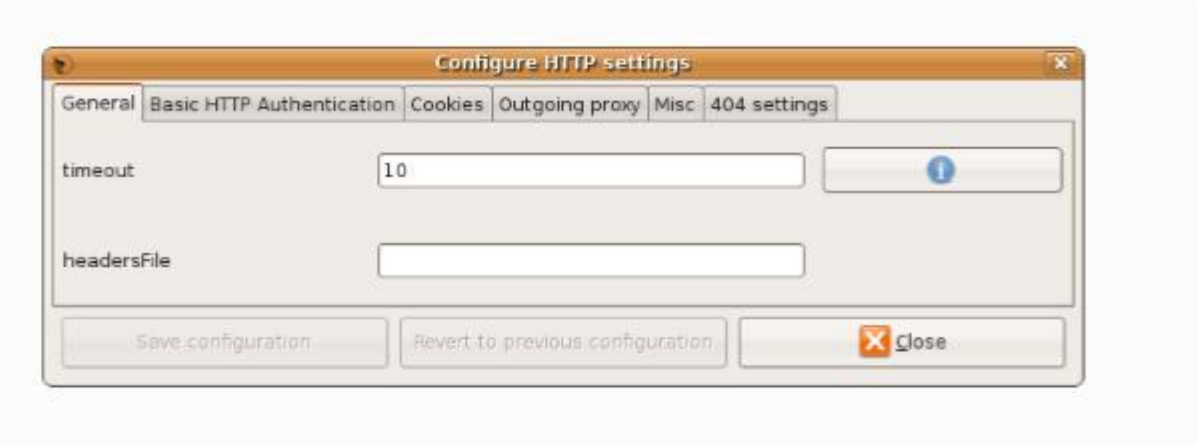
```
GET http://www.go    com.ar/ HTTP/1.1
host: www.google.com.ar
accept: */*
user-agent: Wget/1.10.2
```

## Configurations

There're different configuration panels all across the w3af system. Here all of them are explained.

## HTTP configuration

This section is used to configure URL settings that affect the core and all plugins.

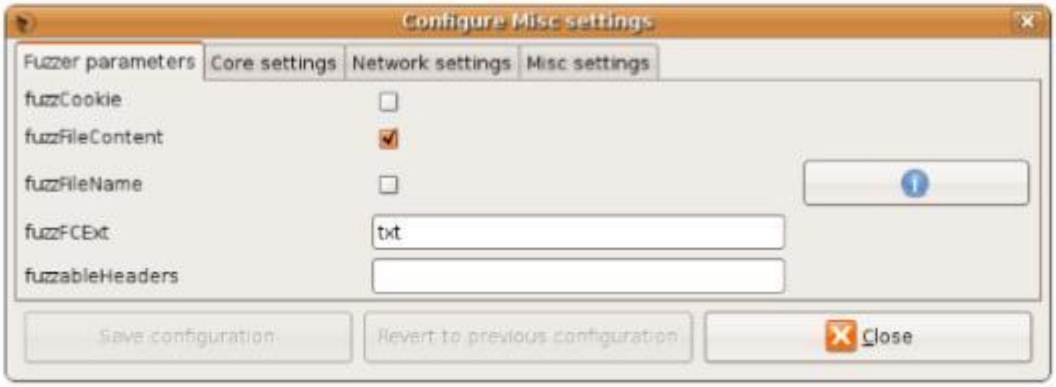


timeout 10

headersFile

## Miscellaneous configuration

This section is used to configure misc settings that affect the core and all plugins.



fuzzCookie ☐

fuzzFileContent ☑

fuzzFileName ☐

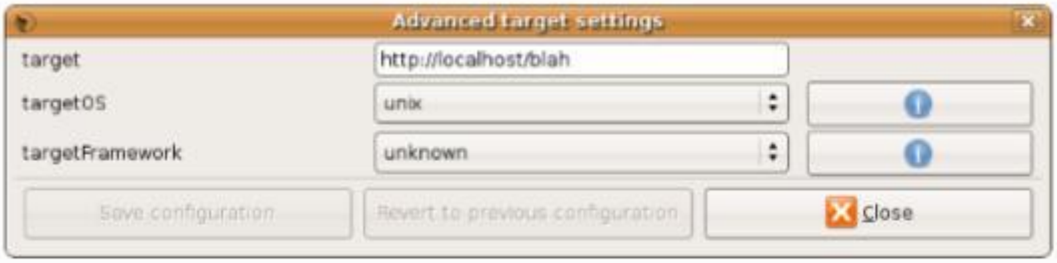
fuzzFCExt txt

fuzzableHeaders

## Advanced target configuration

This section is used to provide detailed information about the target system.

This section is used to provide detailed information about the target system.



target http://localhost/blah

targetOS unix

targetFramework unknown