Open Web Application Security Project (OWASP)

WebScarab

Training notes, 11th March 2011
Colin Watson colin.watson(at)owasp.org

WebScarab is a tool that allows you to record, inspect, modify and build requests and responses sent using the HTTP protocol. The tool supports HTTP over SSL and the framework of plug-ins is extensible through scripting.

> WebScarab's author:
> Rogan Dawes
>
> WebScarab project website:
> http://www.owasp.org/index.php/Category:OWASP_WebScarab_Project

These notes were created from using WebScarab, information on the project website and from Rogan Dawes' additional guidance documents (see Additional Resources).

Contents

Installing WebScarab

OWASP LiveCD

The OWASP Live CD Project by Matt Tesauro:

http://www.owasp.org/index.php/Category:OWASP_Live_CD_Project

has gathered together freely distributable application security tools into a single bootable CD. Apart from OWASP and non-OWASP tools, important OWASP documents such as the Testing Guide are also included. A full list of tools included is maintained at:

http://appseclive.org/node/46

The distribution includes the WebScarab tool (v20090122).

Requirements

You'll need:

- CD writer
- blank CD
- able to boot from a CD

Copies of the CD are also available at some OWASP conferences.

Method

Download the ISO image from:

http://appseclive.org/content/downloads

and burn it to a CD. Then use the CD created to boot your computer.

Download

WebScarab is available to download directly. Always start on the project's home page:

http://www.owasp.org/index.php/Category:OWASP_WebScarab_Project#Download

with links to:

- ZIP of the git tree
- Mac OS X package
- Java Web Start version

There is also some good guidance available in:

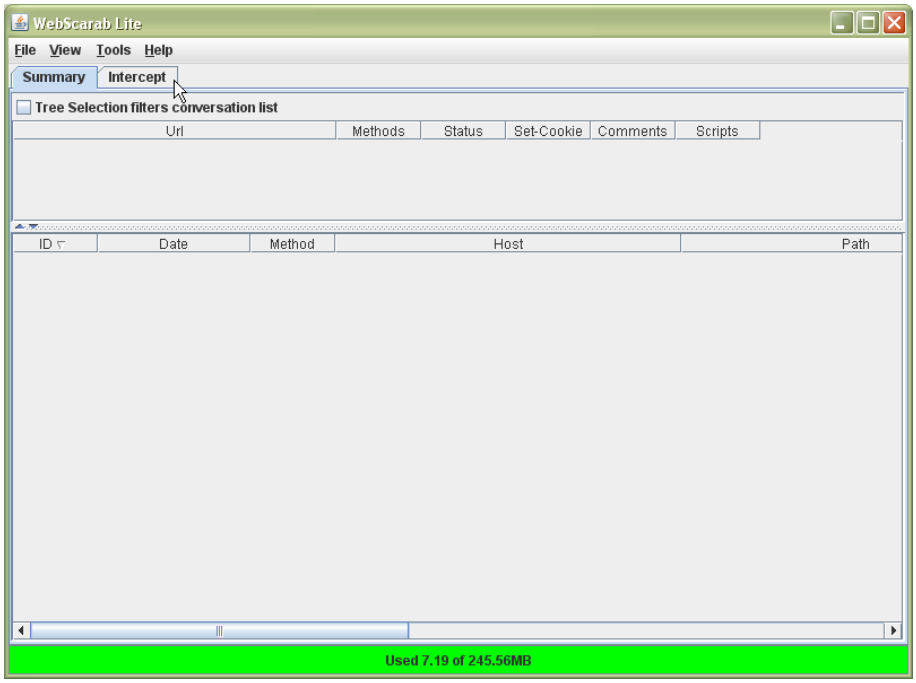
> The interactive HTTP proxy WebScarab – Installation and basic use
> Dr. Holger Peine, Fraunhofer IESE
> http://www.acsac.org/2007/downloads/t5-webscarab-instructions.pdf

You'll need Java (e.g. JRE) installed as well.

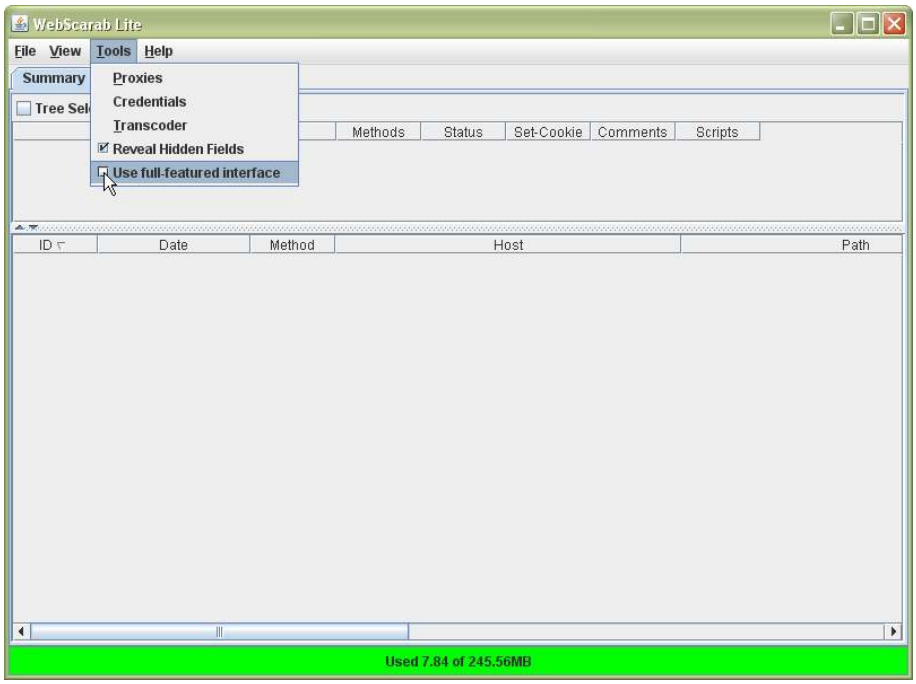
Setting up

Full interface

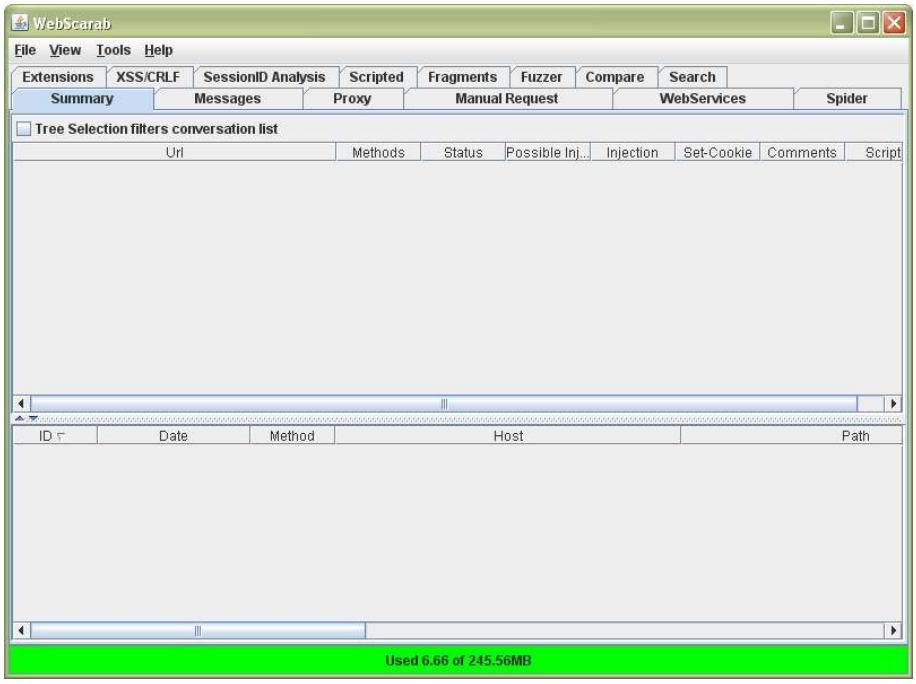
Check you have the full interface enabled. The Lite version has "WebScarab Lite" in the title bar and a reduced menu:



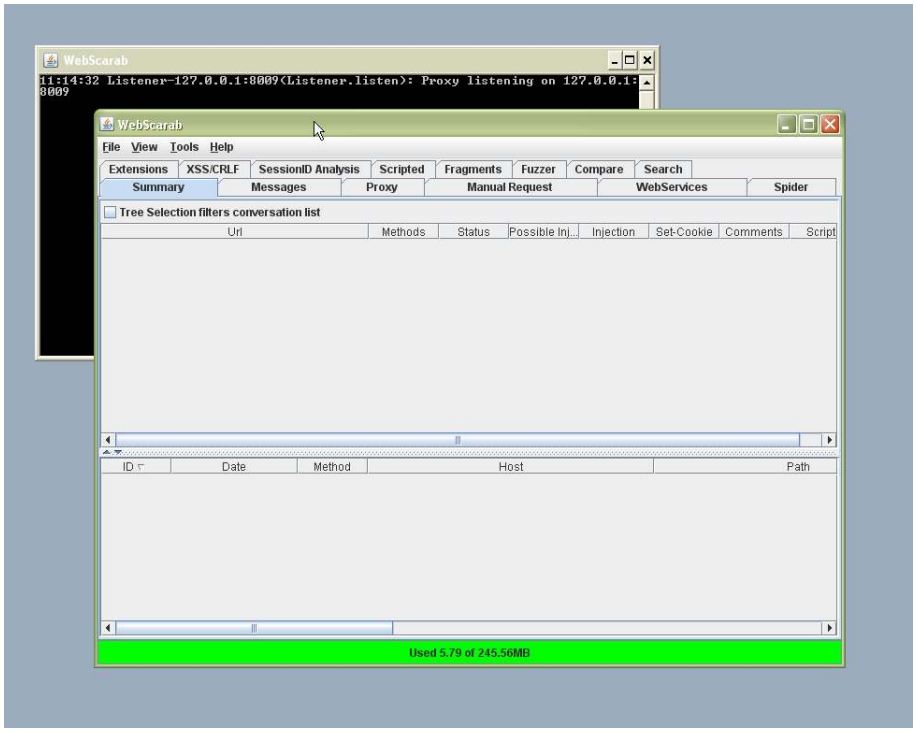
From the Tools menu, click on "Full-featured interface":



Then close, and run WebScarab again. The full interface looks like:



You will actually see two windows. Don't close the command prompt window:



Proxy setup

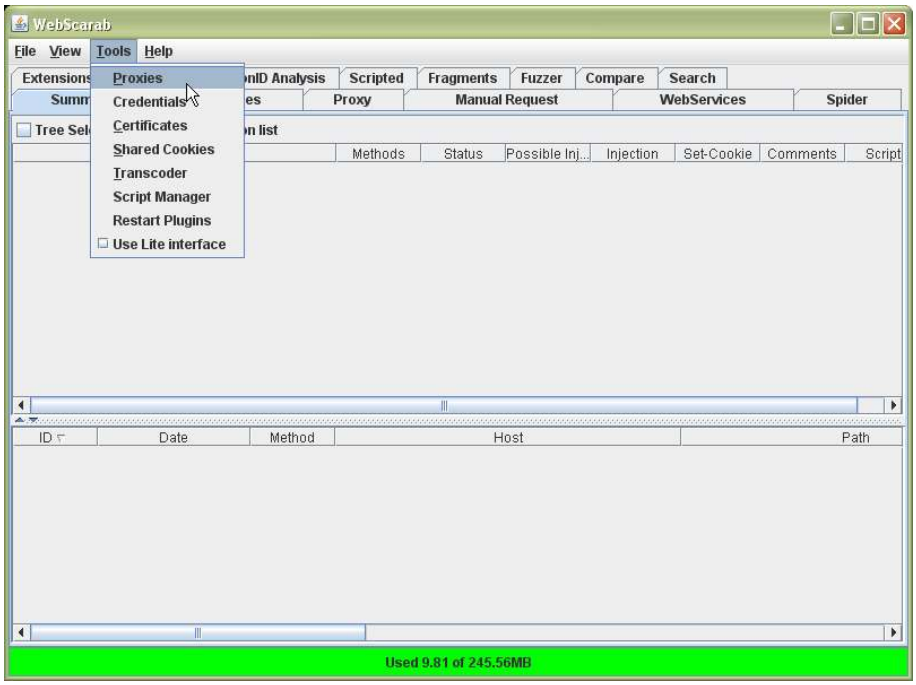
Network proxy

If you use a proxy to connect to the internet, you will need to configure WebScarab with the same settings. You can check whether your existing web browsers are using a network proxy as follows:

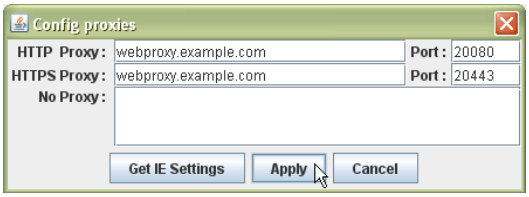
- Firefox
  Tools > Options > Advanced > Network > Settings
- Internet Explorer
  Tools > Internet Options > Connections > LAN settings > Advanced
- Opera
  Tools > Preferences > Advanced > Network > Proxy servers

If you are authorised to connect to someone else's network and cannot connect to the web application being investigated, ask whether you need to configure proxy settings to access.

To set the proxy, click on Proxies in the Tools menu:



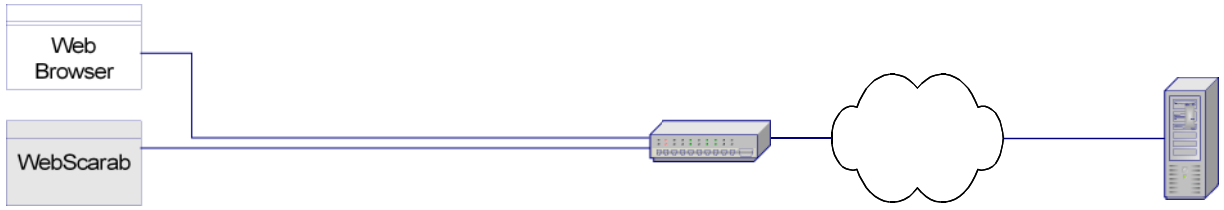
Type in (or clear) the network proxy settings:



The proxy name/address and port will vary.

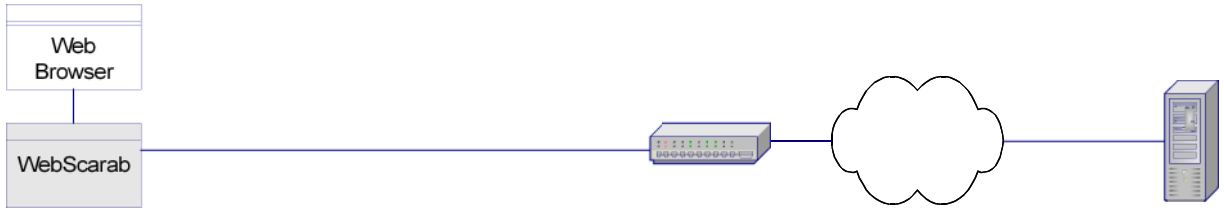
Local proxy

To send and receive HTTP requests, setting the network proxy is all that is required. WebScarab can act as a user agent just like your browser:



YOUR COMPUTER | NETWORK PROXY (OPTIONAL) | WEB SERVER

However, WebScarab is most useful when you configure it as a local proxy for your existing web browsers (and possibly other programs and tools):



YOUR COMPUTER | NETWORK PROXY (OPTIONAL) | WEB SERVER

To set up WebScarab as a local upstream proxy, you need to configure its listener(s) and then change your browser (and other tools/programs) to route their HTTP traffic via WebScarab.

Standard configuration

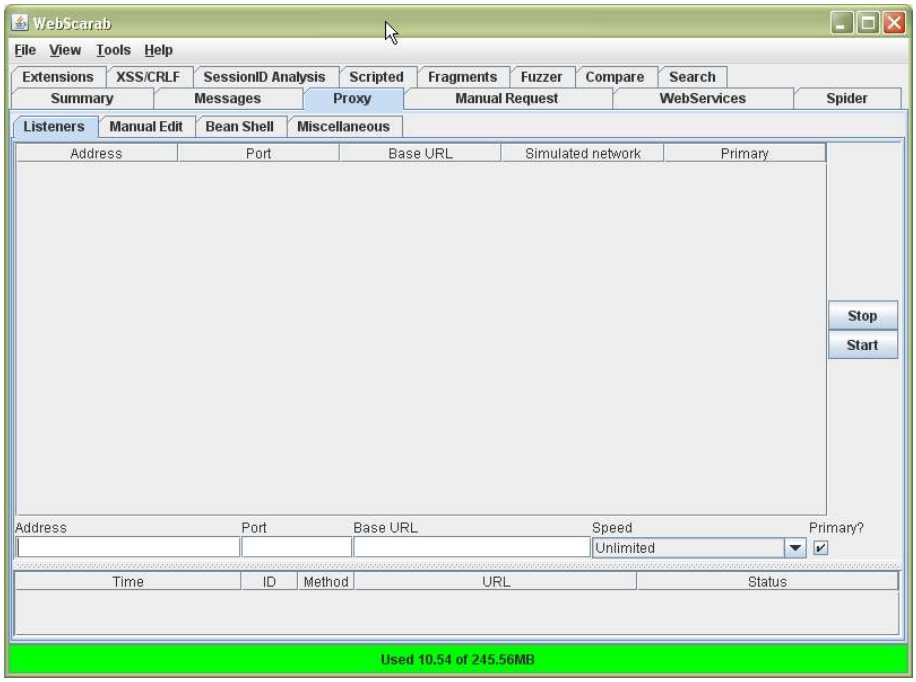
In the styandard configuration (as shown in the diagram immediately above), you typically only need one listener configured and this should be marked as "Primary". It is possible to add multiple listeners.

Tips:

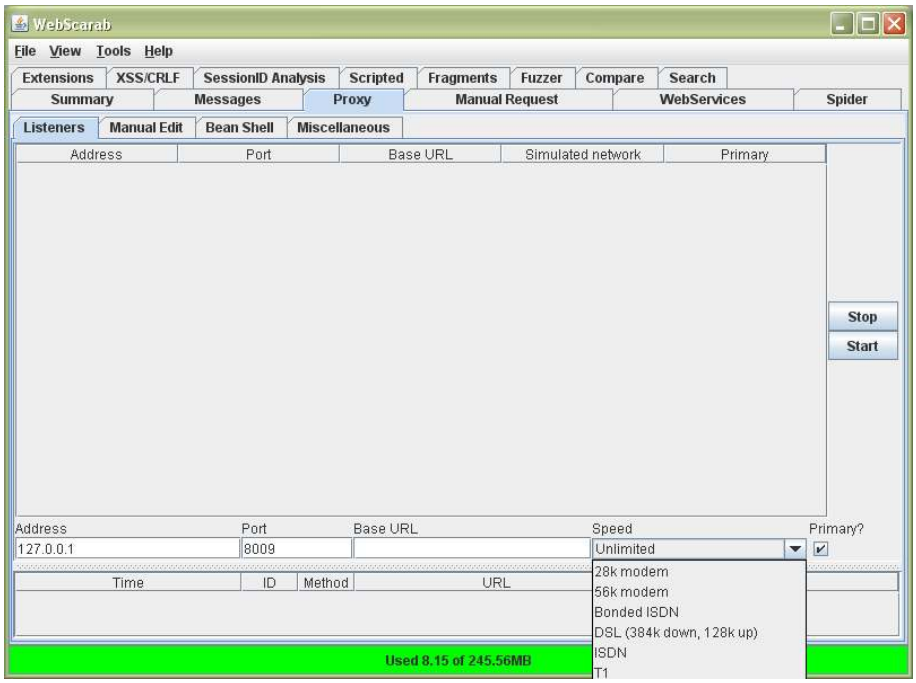
- Only have one primary listener (otherwise there will be a conflict)
- Do not set "Base URL" on the primary listener
- Setting the speed to "unlimited" is recommended unless network latency simulation is needed.

Listener configuration

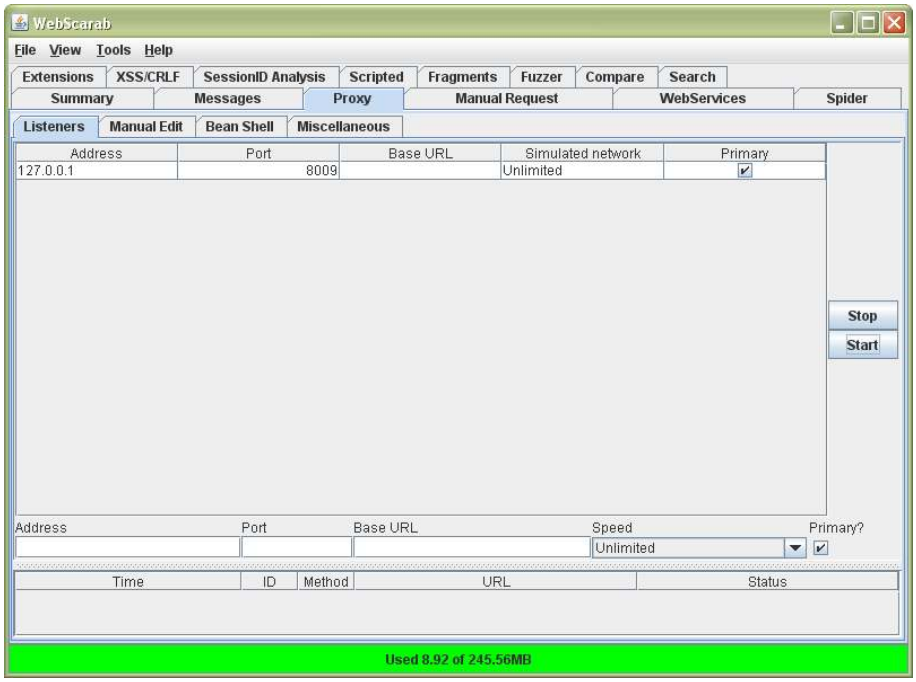
Click on the Proxy tab and ensure the Listeners sub-tab is highlighted:



Click in the "address" field towards the bottom of the screen and create a new listener. This will typically be a local host address e.g. 127.0.0.1 on an unused port number.



Click on the "Start" button on the right hand side to add the listener.

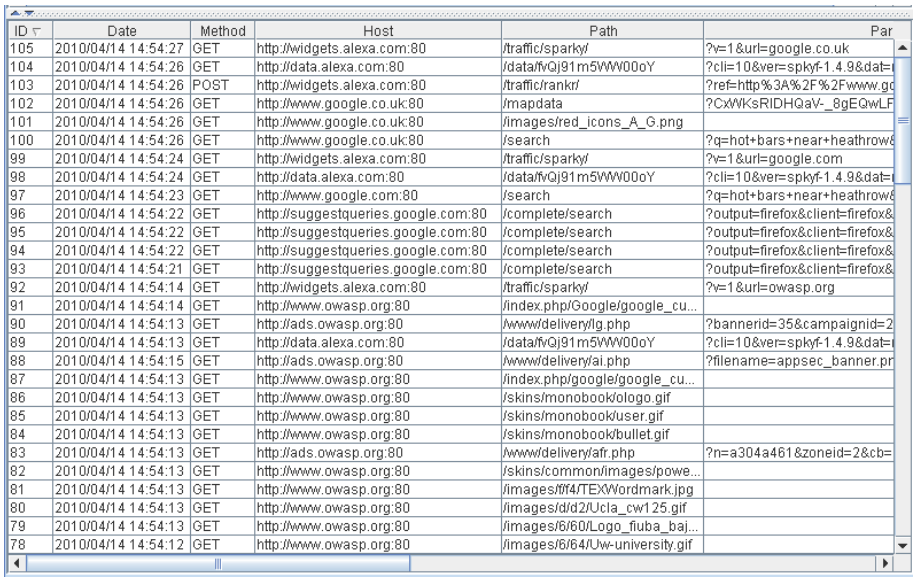


You can remove listeners by clicking on them and then click on "Stop".

Browser configuration

Configure each web browser and other tools/programs to route their HTTP traffic via the WebScarab primary listener.

Take care. All requests/responses from the reconfigured browser will pass through the forward proxy (WebScarab) and be recorded. You should remove, disable or avoid using tools/widgets/functions that your browser submits which are unrelated to the application investigation.



ID	Date	Method	Host	Path	Par
105	2010/04/14 14:54:27	GET	http://widgets.alexa.com:80	/traffic/sparky/	?v=1&url=google.co.uk
104	2010/04/14 14:54:26	GET	http://data.alexa.com:80	/data/fvQj91m5WW00oY	?cli=10&ver=spkyf-1.4.9&dat=
103	2010/04/14 14:54:26	POST	http://widgets.alexa.com:80	/traffic/rankr/	?ref=http%3A%2F%2Fwww.g
102	2010/04/14 14:54:26	GET	http://www.google.co.uk:80	/mapdata	?CxWKsRIDHQaV-_8gEQwLF
101	2010/04/14 14:54:26	GET	http://www.google.co.uk:80	/images/red_icons_A_G.png	
100	2010/04/14 14:54:26	GET	http://www.google.co.uk:80	/search	?q=hot+bars+near+heathrow&
99	2010/04/14 14:54:24	GET	http://widgets.alexa.com:80	/traffic/sparky/	?v=1&url=google.com
98	2010/04/14 14:54:24	GET	http://data.alexa.com:80	/data/fvQj91m5WW00oY	?cli=10&ver=spkyf-1.4.9&dat=
97	2010/04/14 14:54:23	GET	http://www.google.com:80	/search	?q=hot+bars+near+heathrow&
96	2010/04/14 14:54:22	GET	http://suggestqueries.google.com:80	/complete/search	?output=firefox&client=firefox&
95	2010/04/14 14:54:22	GET	http://suggestqueries.google.com:80	/complete/search	?output=firefox&client=firefox&
94	2010/04/14 14:54:22	GET	http://suggestqueries.google.com:80	/complete/search	?output=firefox&client=firefox&
93	2010/04/14 14:54:21	GET	http://suggestqueries.google.com:80	/complete/search	?output=firefox&client=firefox&
92	2010/04/14 14:54:14	GET	http://widgets.alexa.com:80	/traffic/sparky/	?v=1&url=owasp.org
91	2010/04/14 14:54:14	GET	http://www.owasp.org:80	/index.php/Google/google_cu...	
90	2010/04/14 14:54:13	GET	http://ads.owasp.org:80	/www/delivery/lg.php	?bannerid=35&campaignid=2
89	2010/04/14 14:54:13	GET	http://data.alexa.com:80	/data/fvQj91m5WW00oY	?cli=10&ver=spkyf-1.4.9&dat=
88	2010/04/14 14:54:15	GET	http://ads.owasp.org:80	/www/delivery/ai.php	?filename=appsec_banner.pr
87	2010/04/14 14:54:13	GET	http://www.owasp.org:80	/index.php/google/google_cu...	
86	2010/04/14 14:54:13	GET	http://www.owasp.org:80	/skins/monobook/ologo.gif	
85	2010/04/14 14:54:13	GET	http://www.owasp.org:80	/skins/monobook/user.gif	
84	2010/04/14 14:54:13	GET	http://www.owasp.org:80	/skins/monobook/bullet.gif	
83	2010/04/14 14:54:13	GET	http://ads.owasp.org:80	/www/delivery/afr.php	?n=a304a461&zoneid=2&cb=
82	2010/04/14 14:54:13	GET	http://www.owasp.org:80	/skins/common/images/powe...	
81	2010/04/14 14:54:13	GET	http://www.owasp.org:80	/images/f/f4/TEXWordmark.jpg	
80	2010/04/14 14:54:13	GET	http://www.owasp.org:80	/images/d/d2/Ucla_cw125.gif	
79	2010/04/14 14:54:13	GET	http://www.owasp.org:80	/images/6/60/Logo_fiuba_baj...	
78	2010/04/14 14:54:12	GET	http://www.owasp.org:80	/images/6/64/Uw-university.gif	

You'll quickly see these in WebScarab's logs, but common things are:

- search box suggestions

- use of other websites (e.g. search engines, news, web mail)
- website privacy and security blocking services
- traffic ranking tools (e.g. Alexa).

It is important to prevent these events occurring for three main reasons:

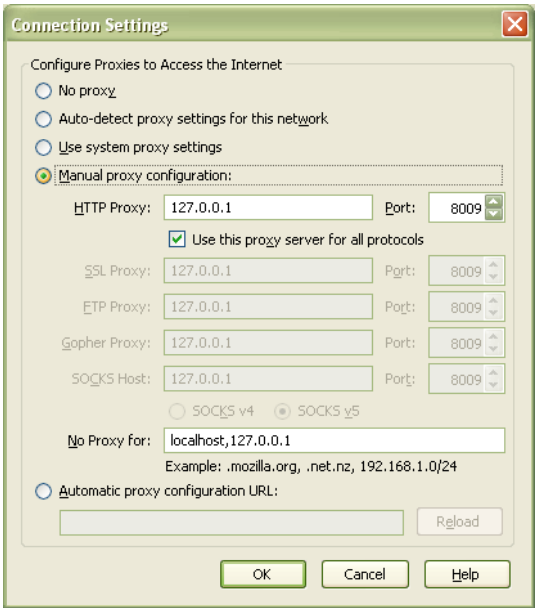
1. they add to WebScarab's logs which means you may miss important requests and responses with the application being investigated
2. you may keep the logs and someone else may look at them—you do not need to, or want to, let other people know what you are looking at or what is installed on your browser
3. third party sites may set or delete cookies.

If you are using Windows, WebScarab will try to configure Internet Explorer automatically when it starts up, and remove the proxy settings when it is closed again. This feature requires a JNI plugin/DLL, which has to be on the PATH. If the DLL was successfully located, you will see a button entitled "Get IE Settings"

To check IE, and to set up other web browsers, go to:

- Firefox
  Tools > Options > Advanced > Network > Settings
- Internet Explorer
  Tools > Internet Options > Connections > LAN settings > Advanced
- Opera
  Tools > Preferences > Advanced > Network > Proxy servers

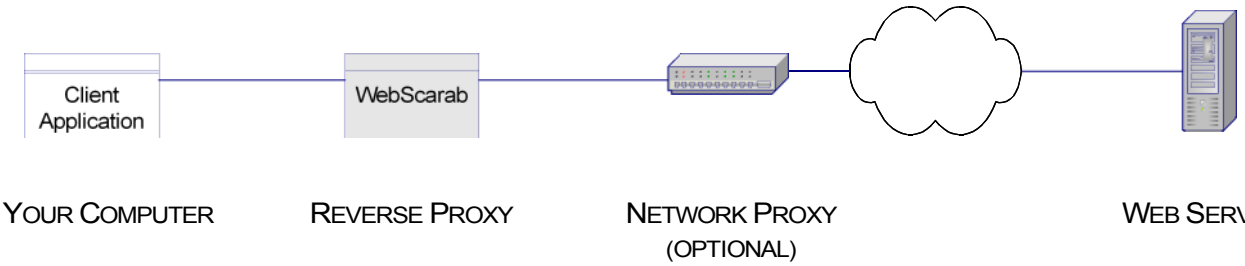
For example, in Firefox:



Reverse proxy configuration

Listener configuration

The Base URL setting mentioned previously is only used when you need WebScarab to act as a reverse proxy i.e. pretend to be the server for a client that has fixed or unconfigurable proxy settings. The Base URL will be the HTTP or HTTPS address of the web application, with matching port number—typically 80 or 443.



In this configuration, WebScarab should be set up on a different computer than the client, and it should listen on all network interfaces (leave "Address" blank or set as the network IP address of the machine WebScarab is installed on).

Browser configuration

The client application needs to be configured to use WebScarab's IP address (the reverse proxy) for the web application's address—by editing the hosts file on the computer running the client application (labelled "Your Computer" above). For an application using SSL, you will also usually have to import WebScarab's certificate to the client application's list of recognised certificates.

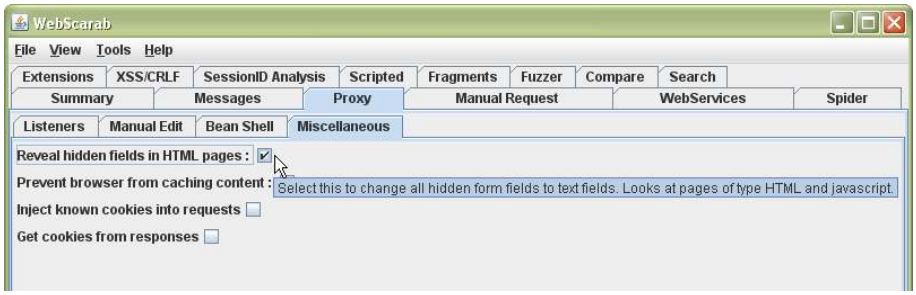
Tip: For Java applications, you may need to define another keystore. See: http://forums.sun.com/thread.jspa?threadID=220329&tstart=165

Other proxy settings

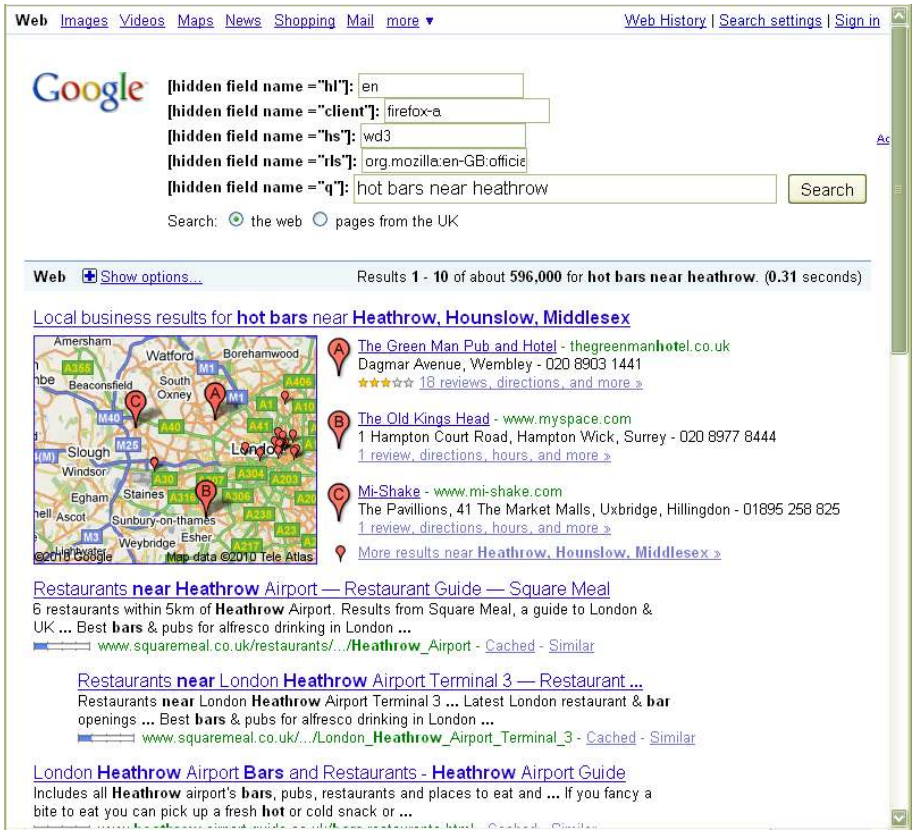
Some other settings of interest in the Proxy tab, under Miscellaneous.

Reveal hidden fields

Begin with this disabled, but it can be useful to help identify (and alter) hidden fields in web forms.



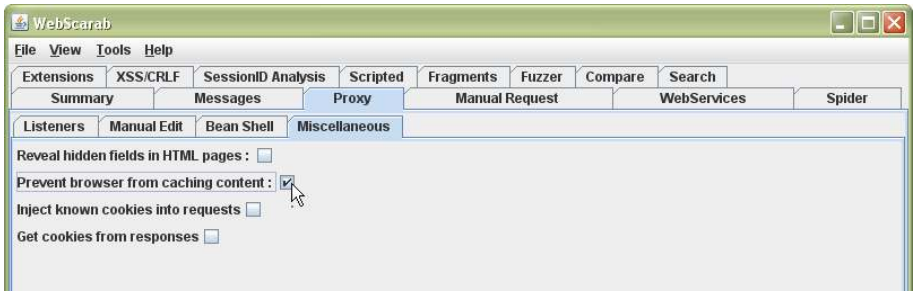
It does mess up the layout of pages, and you may find yourself toggling this on and off.



You can also amend the values in captured requests (see below).

Prevent browser from caching content

This option removes any "if-modified-since" from requests to ensure that WebScarab always has a copy of the response body, rather than allowing the browser to use a locally cached copy.



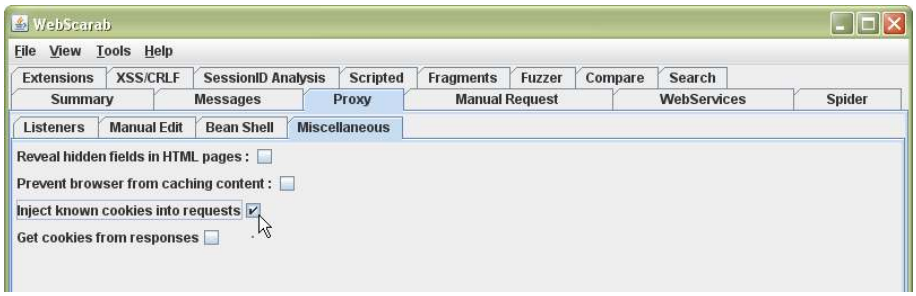
An example header that would be removed from a request is:

If-Modified-Since: Tue, 05 Jan 2011 09:59:47 GMT

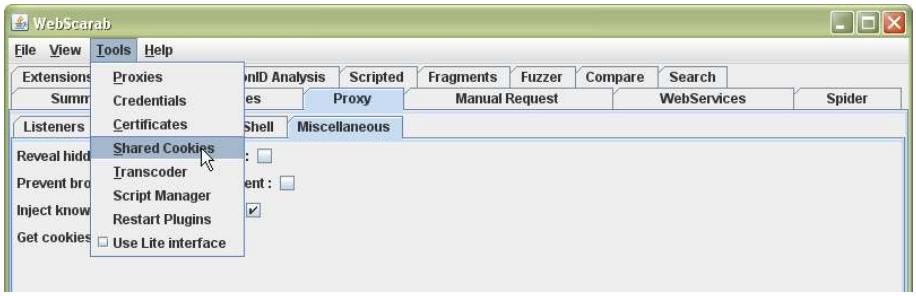
It is recommended that this option is usually set. You may want to uncheck this if you are investigating the application's caching.

Inject known cookies into requests

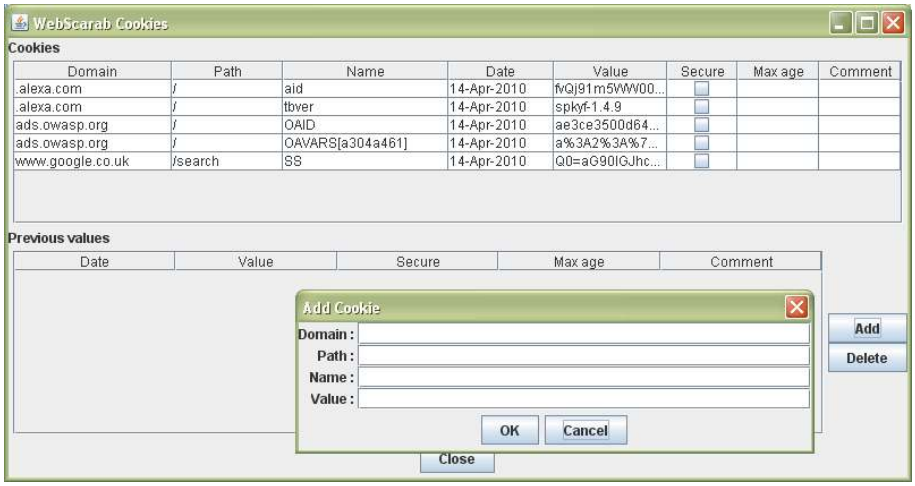
WebScarab can override any cookies in the browser and use its own "known cookies".



The cookies injected are defined in the Shared Cookies list (Tools > Shared Cookies):

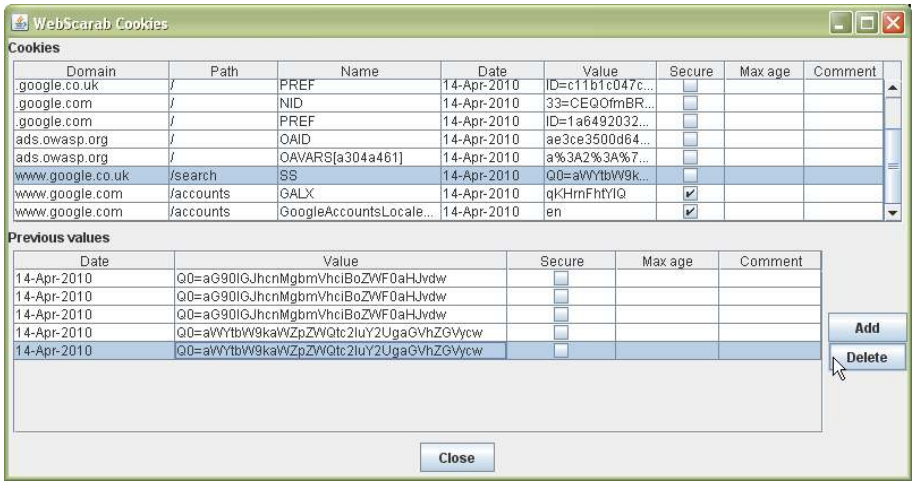


Cookies are not automatically collected here (see below), but you can also manually add or delete cookies:



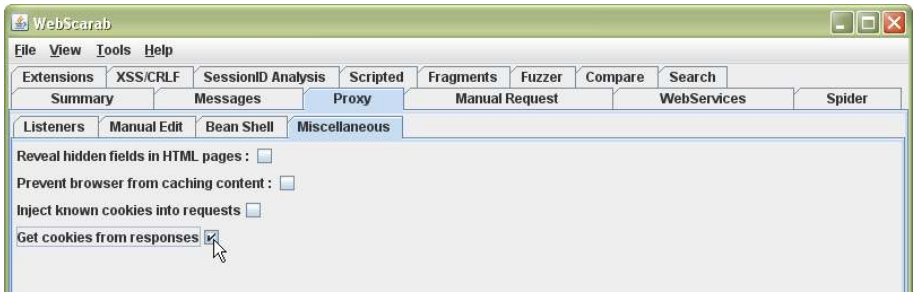
Normally don't use "Inject known cookies into requests" until you need to need to sue it to to help fix cookie values or to replay a previous session for example.

Tip: To delete cookies in the list, click on their name in the top section and then click on a value below. Then click on the Delete button. You will need to do this for each value.



Get cookies from responses

This will force WebScarab to extract Set Cookie headers from responses and adds them to the Shared Cookies list.



Probably have this checked, so you can review the cookies being set at a glance.

Logging

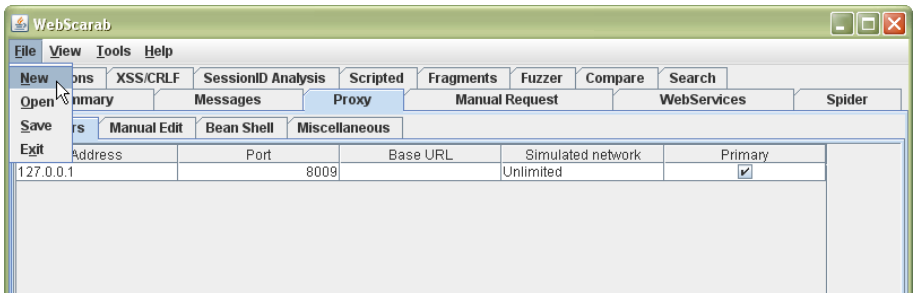
WebScarab sessions

WebScarab maintains a record of requests, responses and associated data from when it is started to exiting, when the data are lost. This log is called the "session" and should not be confused with session management of the application being investigated.

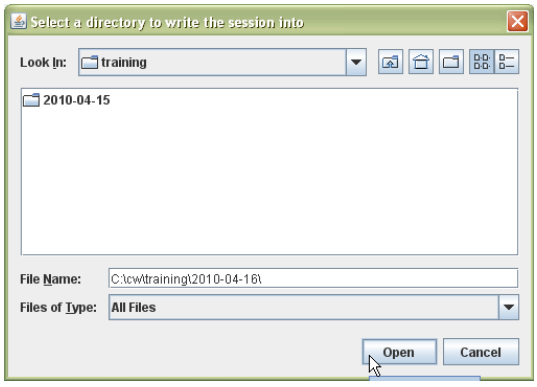
Each time you run WebScarab, the first thing to do after checking configuration, is to create a new session or open an existing session.

Creating a new session

From the File menu, select "new":

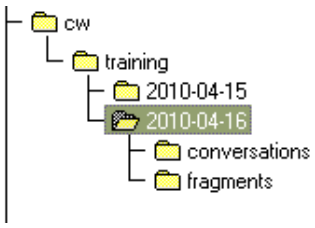


In the file dialogue box, navigate to the directory you want to save the session data in, and provide a name for the session. You may want to use dates and initials to identify sessions.



WebScarab will create two sub-directories within your selected destination called "conversations" for request and response data and "fragments" for scripts and comments extracted from HTML content. Both are initially empty.

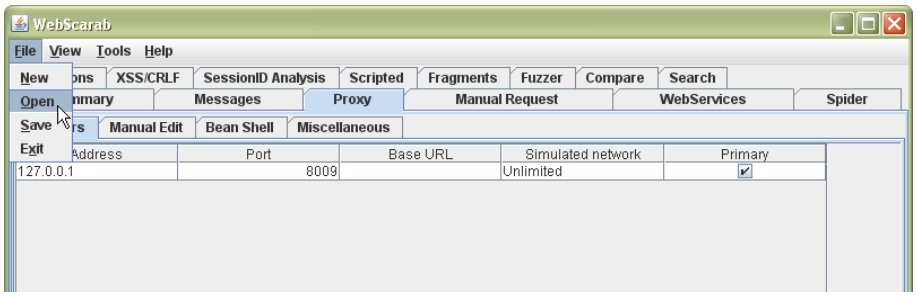
The example directory structure matching the screen captures above is shown below:



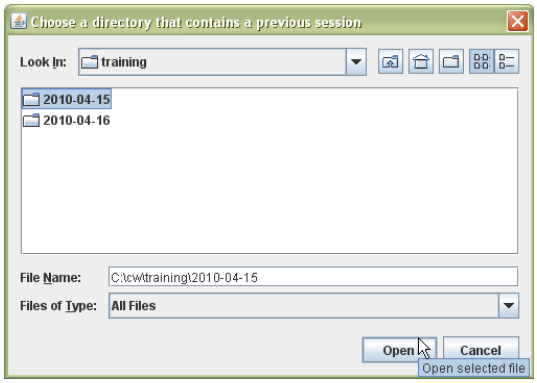
Tip: Check WebScarab is writing data to the correct directory as soon as you begin using using the tool.

Opening an existing session

To work with previously saved session, save any existing session and then select "Open" from the File menu.



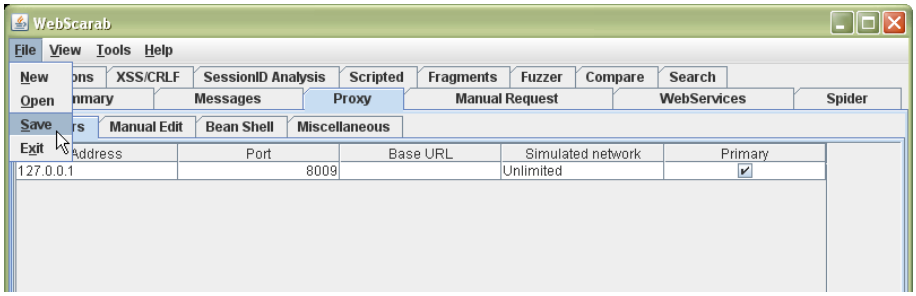
Select the session directory and click Open button:



Further activity will append data into this session.

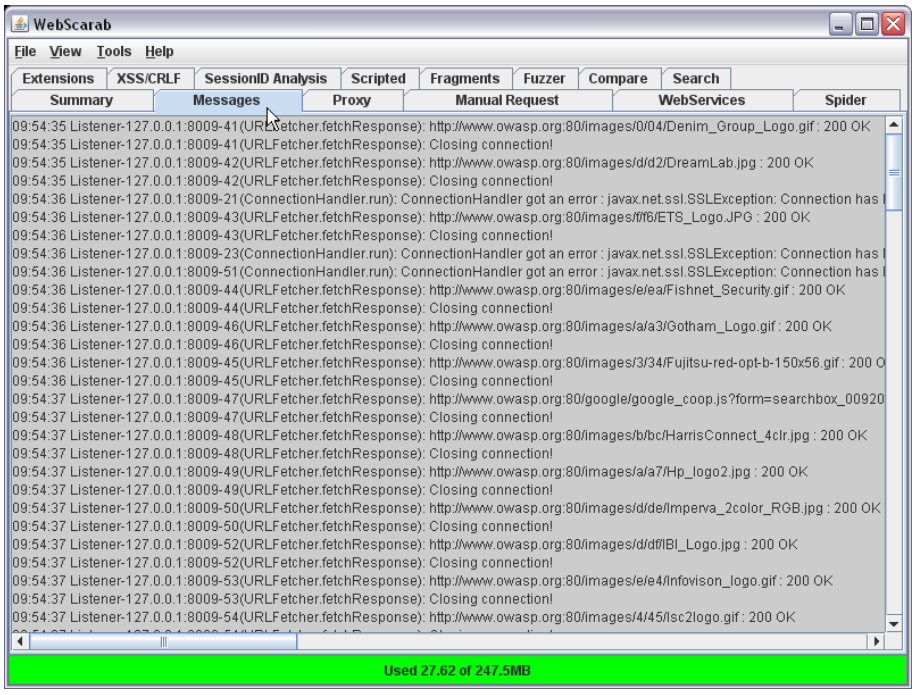
Saving sessions

Periodically, select "save" in the File menu to force all temporary data to be written to the session directory. Save before exiting WebScarab.



Messages

The Messages tab logs actions and displays system messages. Use Help > Log Level to control the level of detail. "Info" is usually sufficient.



Session log protection and preservation

The session data is likely to be confidential and may also contain sensitive data. Ensure you store and protect the data in a manner instructed by the client (e.g. contract, non-disclosure agreement), your own policies, the law (e.g. Data Protection Act) and other mandates.

Be especially careful with tracking where data are copied and the secure deletion of files.

Requests and responses

HTTP - Hypertext Transfer Protocol

We need to understand the HTTP protocol. Full details:

> HTTP - Hypertext Transfer Protocol, W3C
> http://www.w3.org/Protocols/

We don't need to craft our own requests, we can see what real web browsers and sending and receiving back from web applications using WebScarab.

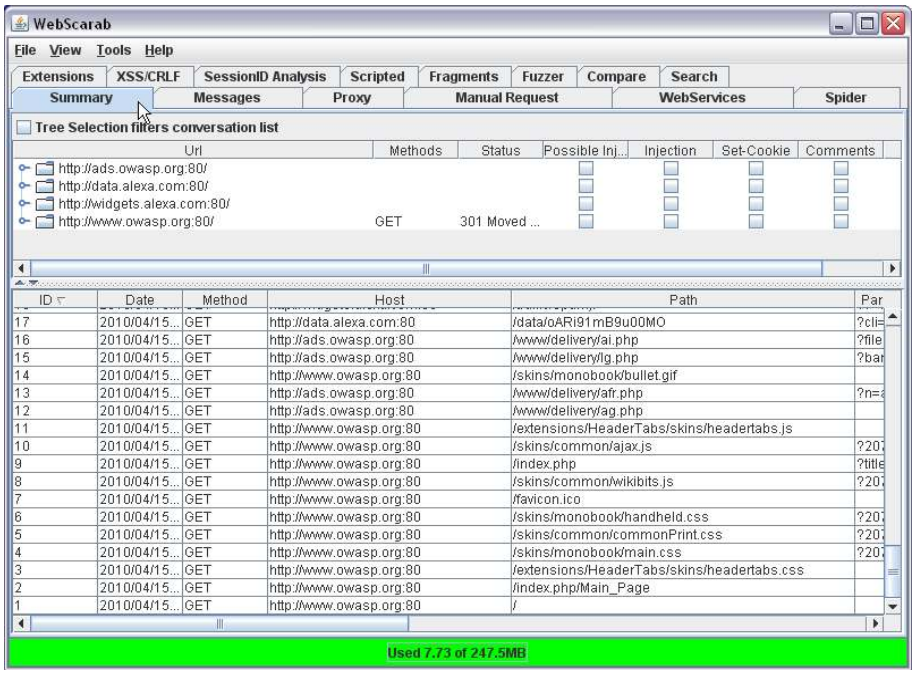
But it is useful to keep to and a list of HTTP response codes:

> HTTP/1.1 Status Code Definitions
> http://www.w3.org/Protocols/rfc2616/rfc2616-sec10.html

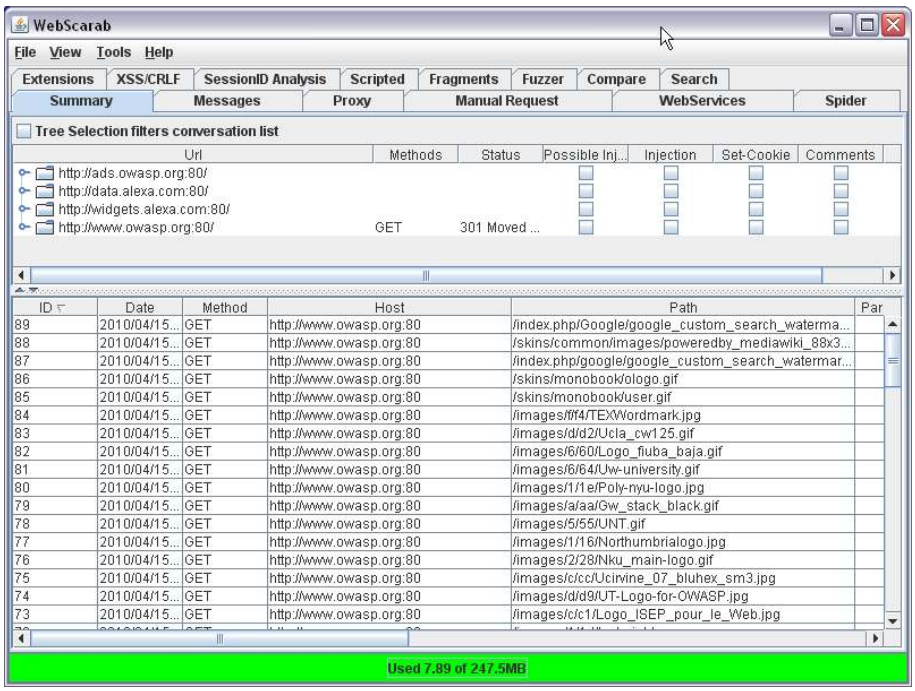
These will help us to identify redirects, not founds and server error messages.

Browser requests

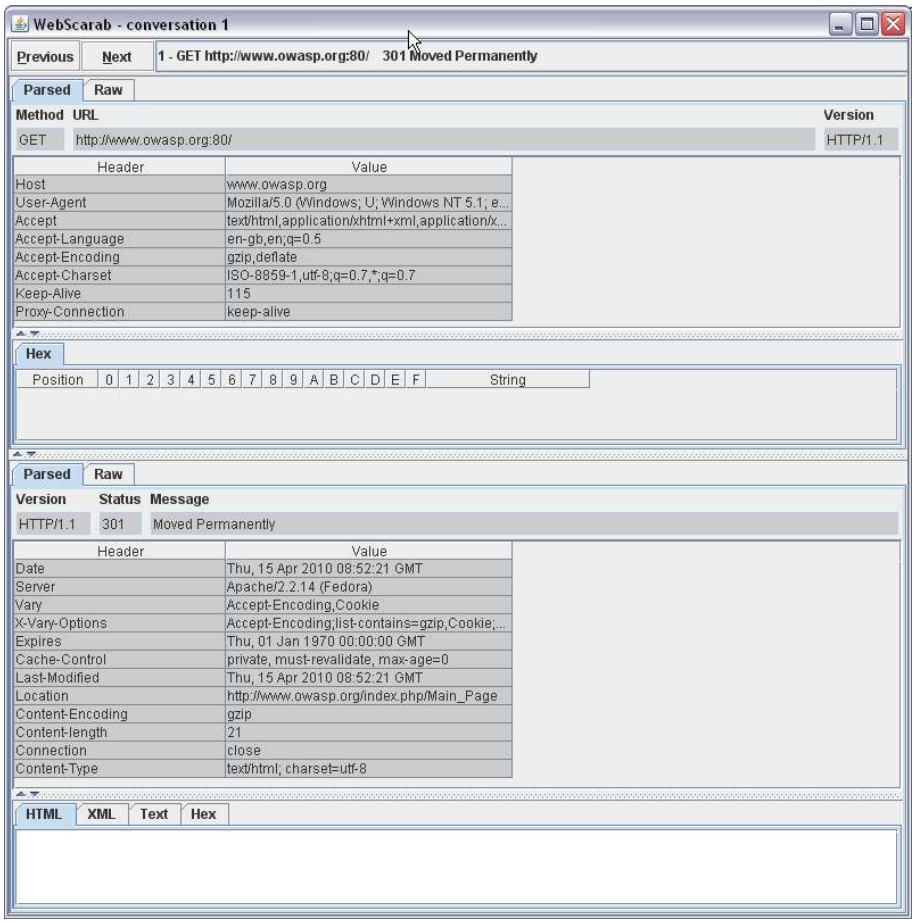
First specify where the session will be stored. Then request a page in your browser. The Summary tab of this Main window fills up with details of the requests being sent by the browser:



The IDs show the order in which the request are made. In this case ID 1 is the page we requested, and the following ones are other page components (images, JavaScript, style sheets, APIs, etc) the page references.



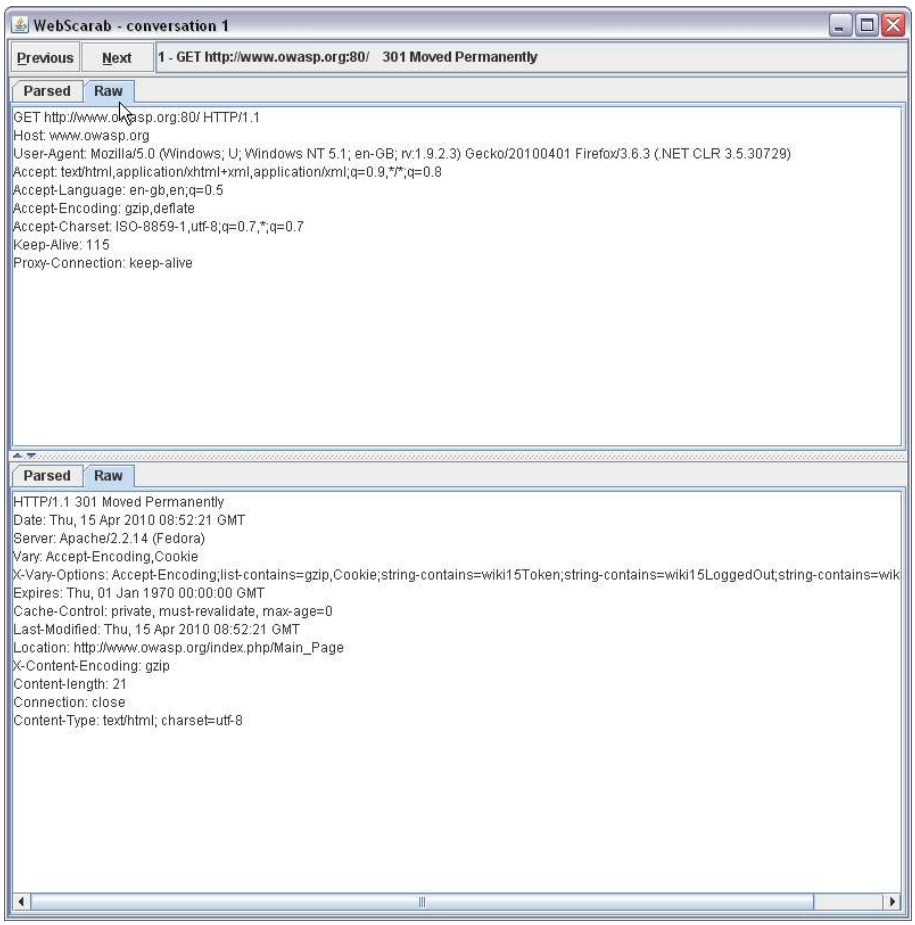
Right click and select "Show Conversation" or double-clicking on an entry (e.g. ID 1) to display the full request and response in a new Conversation window:



Tip: Use the HTML, XML, Text and Hex tabs to display the response in different

formats.

Change to the raw view tabs to see the conversation in a simpler format:

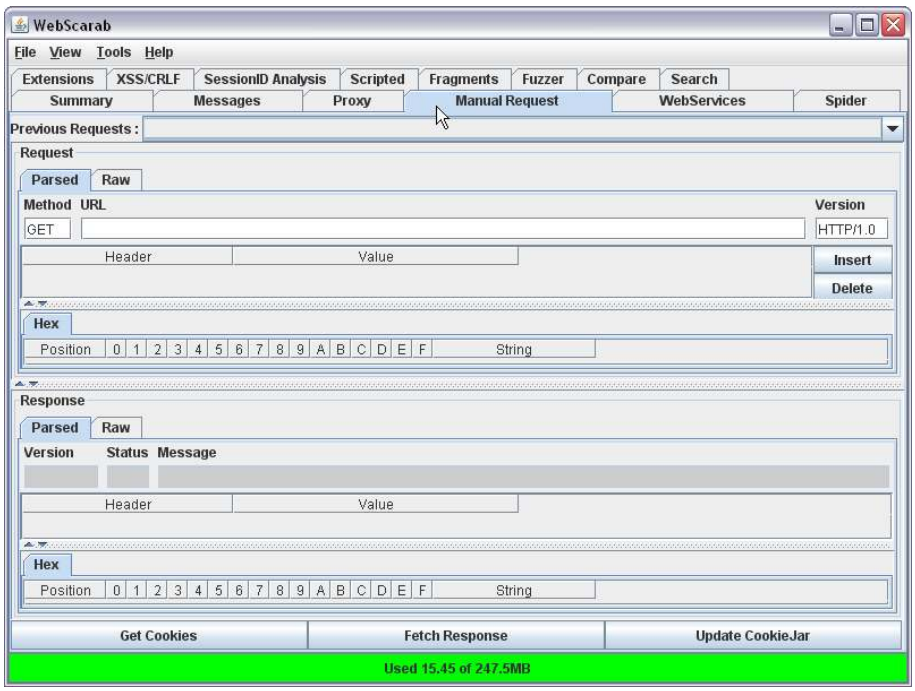


Here we can see the response was a redirect (301 = Moved Permanently) to /index.php/Main_Page

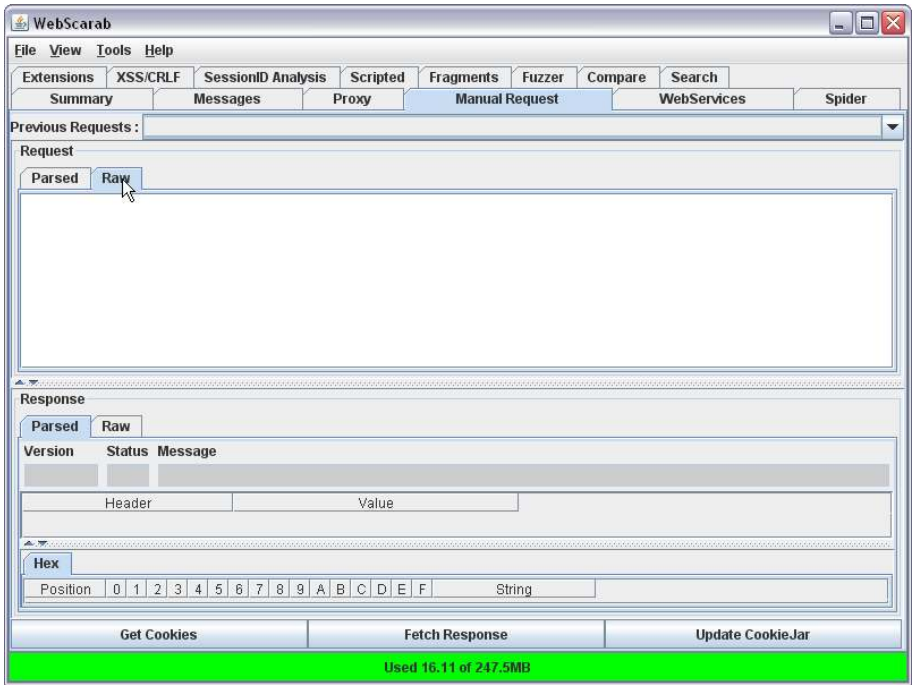
Manual requests

We can create and submit an HTTP request manually in WebScarab. The easiest way is to use an existing one and then edit it. We can copy text from the raw request. Click and drag to highlight the first two lines and then Ctrl-C to copy to the clipboard:

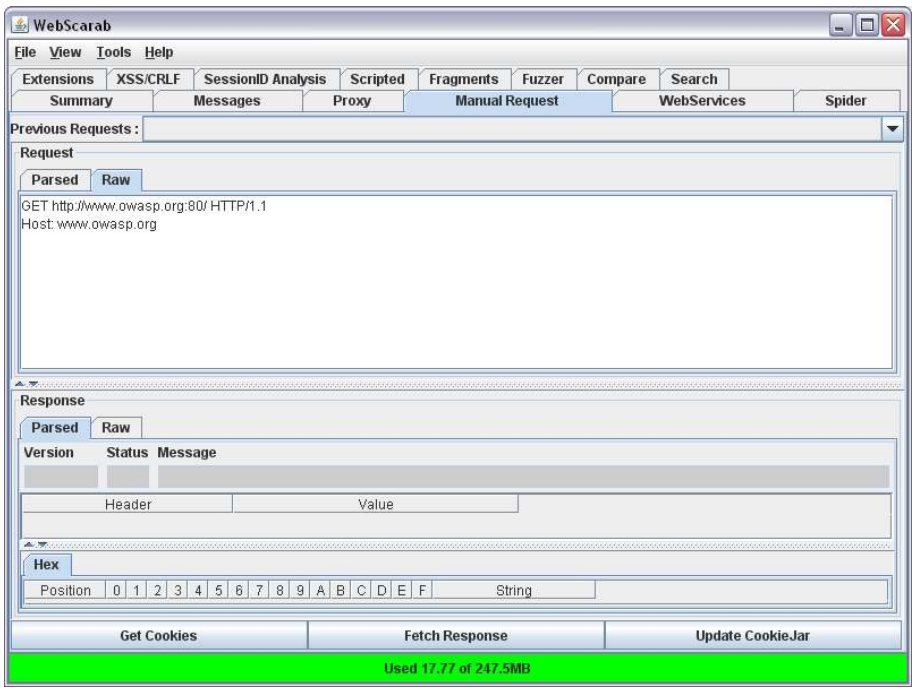
Then go to the Manual Request tab in the Main window (this leaves the Conversation Window open):



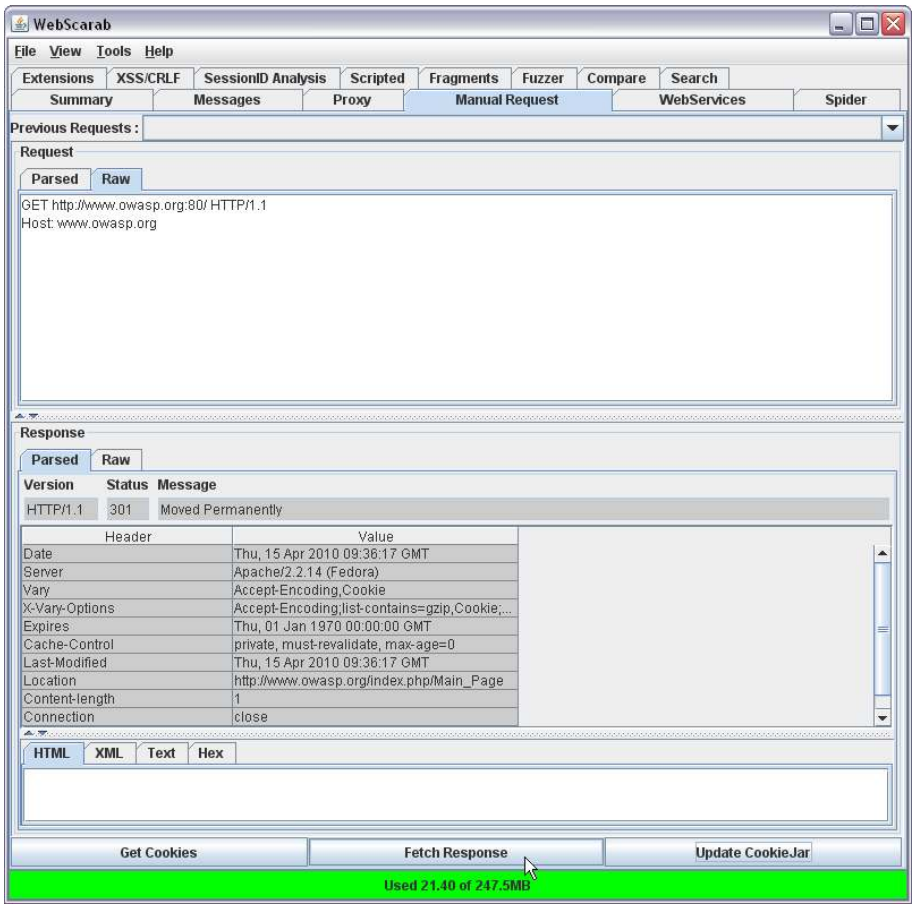
Then click on the Raw tab in the Request section:



and paste in (Ctrl-V) the two copied lines:

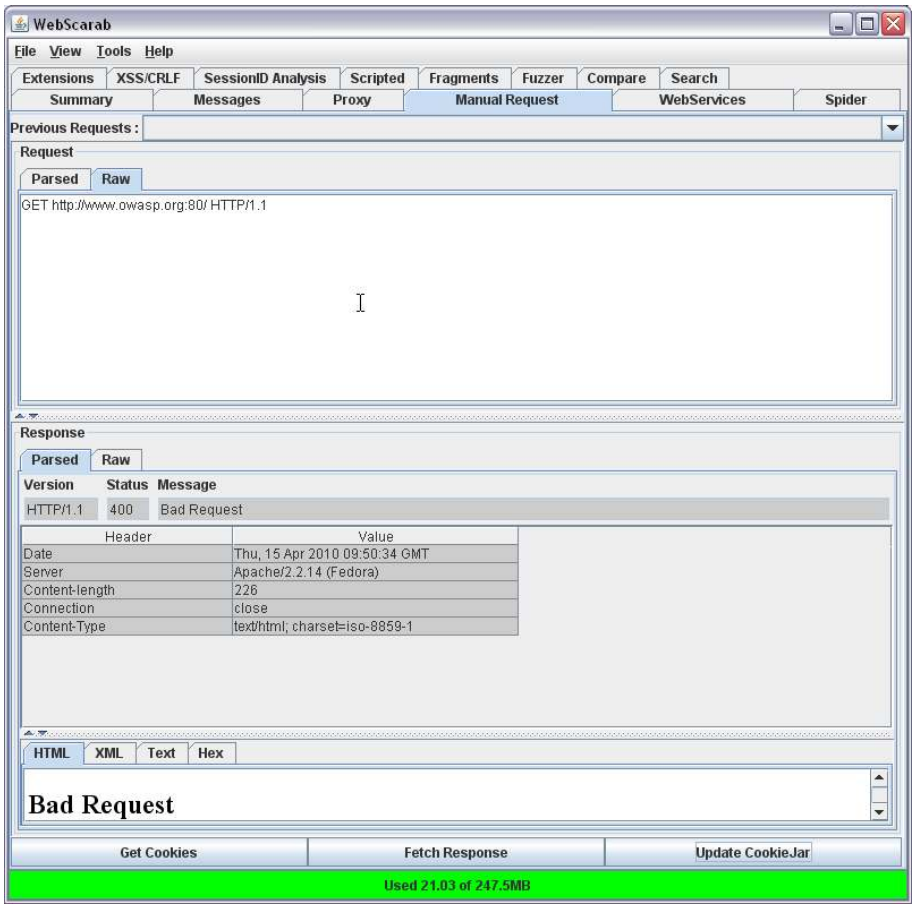


Then click on the "Fetch Response" button and the response from the server appears in the Response section:



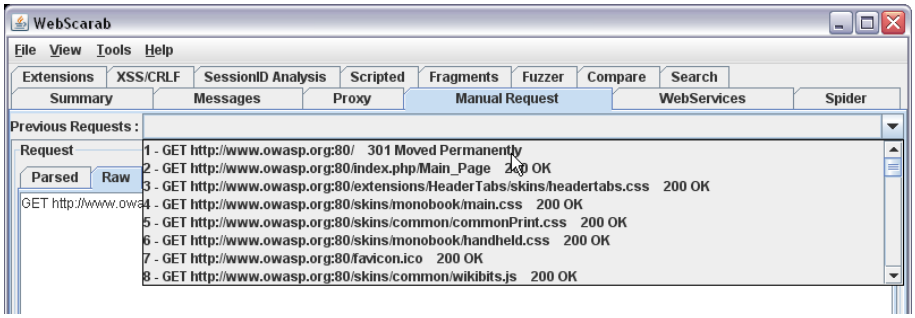
Tip: Click and drag on the textured horizontal bars to change the height of the sections.

You may want to try adding or removing other headers from the original request. Here's a request without the HOST header:



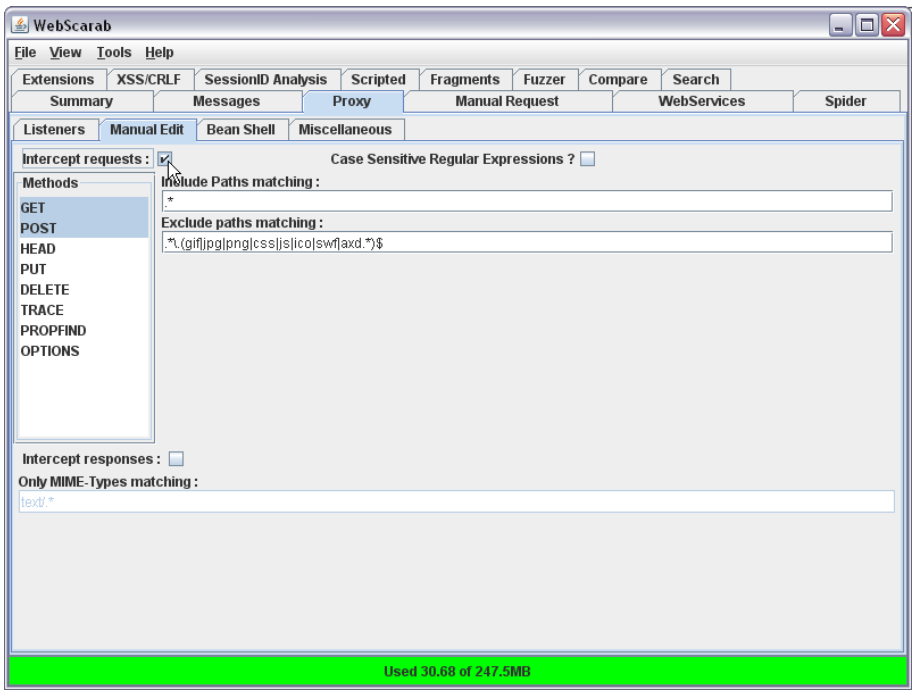
This begins to provide information on the configuration (in this case of the IOWASP web site). Other sites may require some Accept headers too.

You can also select previous requests from the drop-down list on the Muanil Request tab:



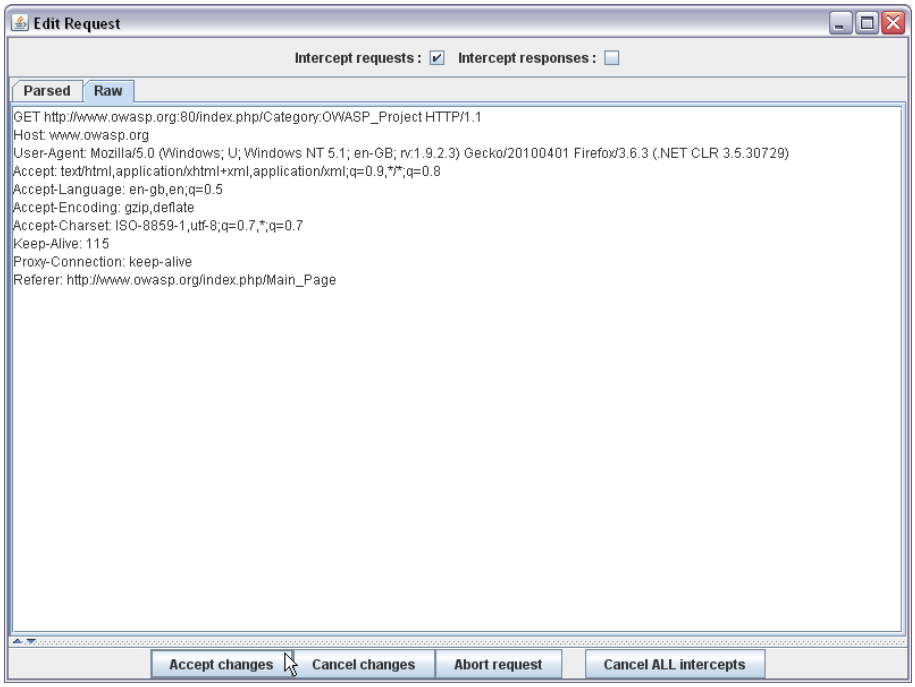
Intercepting and modifying conversations

Under the Proxy tab, select the Manual Edit tab. Click on "Intercept Requests". Leave "Intercept Responses" unchecked for now:



It is possible to use regular expressions here to limit which paths are included and excluded. Typically, you won't want to intercept every request since each page may contain scores of requests, most of which relate to static content, and you don't want to have to intervene with every one.

Now request a URL using your web browser. The browser will wait, and WebScarab will open an Edit Request window:



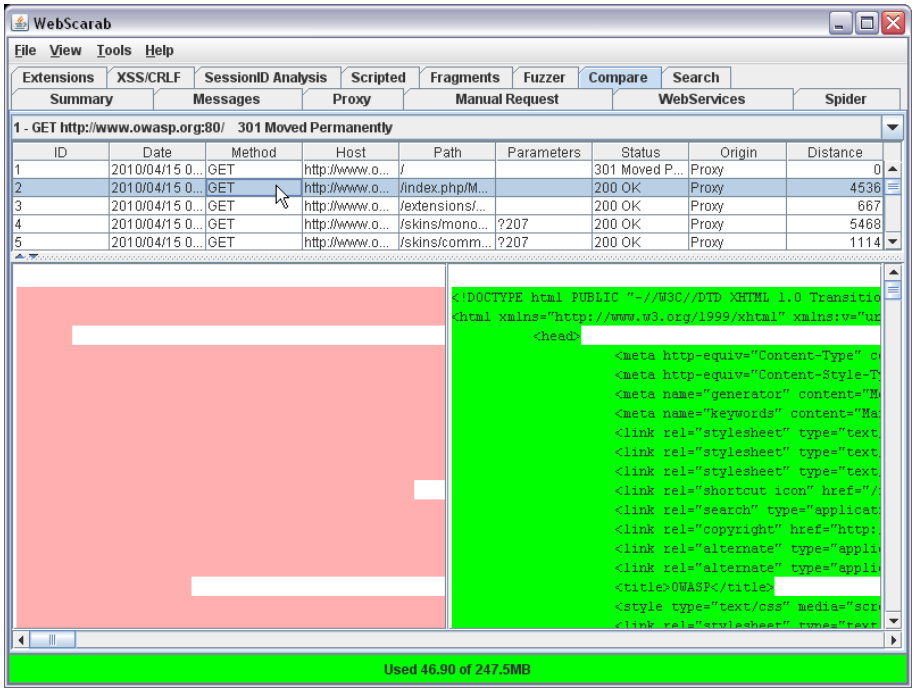
Here you can edit the raw (or parsed) request and then accept/cancel the changes to send the request to the web server. Abort request will prevent anything being sent to the server (and an error message will be displayed in the browser).

The headers and/or payloads can be modified.

Similarly responses can be captured and modified. This may be useful if the web page's behaviour changes in response to returned data.

Response comparison

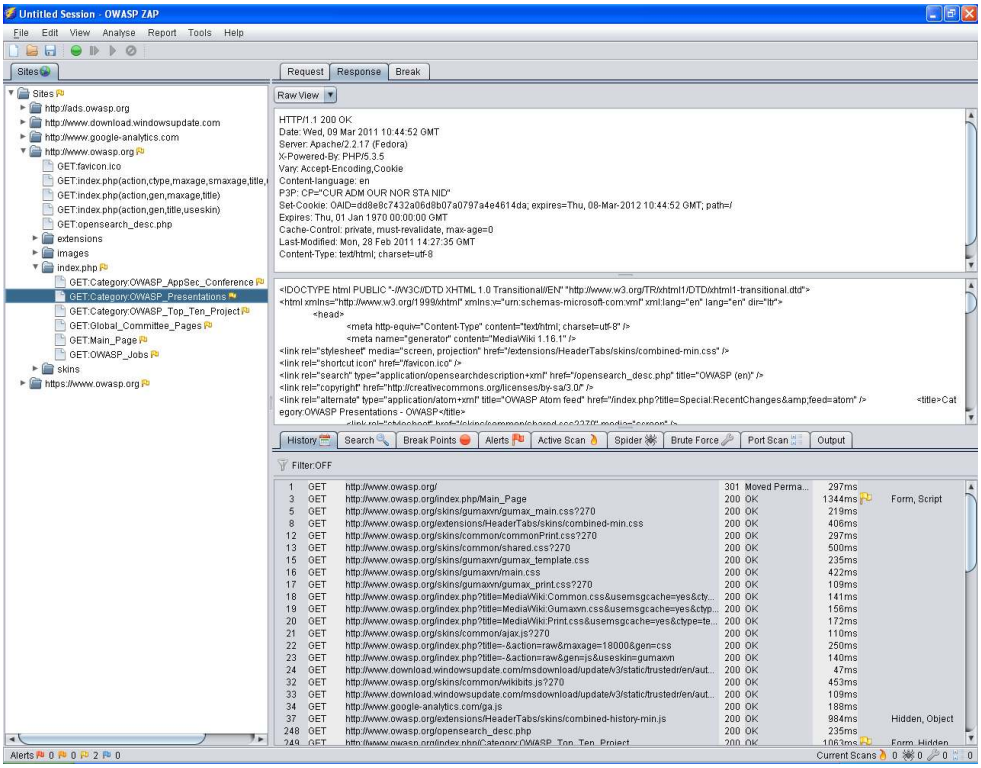
On the Compare tab, select a conversation ID from the drop-down list (the base response) and then another (the target response) from the grid below. The two responses are shown side-by-side:



The Distance column in the grid is a measure of the difference between the two responses - actually the Levenshtein Edit Distance, showing the number of edits (inserts, changes, and deletes) that are required to transform the base response into the current target.

See also.... Zed Attack Proxy

ZAP is another, more recent, OWASP project - an intercepting proxy somewhat like WebScarab.



The current advantages of each are:

For WebScarab:

- more flexible
- session ID analysis
- fragment collector
- beanshell extensions
- parameter fuzzer
- XSS/CRLF - passive analysis
- new NG project lead (late 2010)

For Zed Attack Proxy:

- great usability
- port scanner
- encode/decode/hash tool
- comprehensive help pages
- report generation
- multi-lingual support
- under active development

ZAP may be more suited to developers, testers, auditors, etc who are not focused on security testing all the time. But it is under development with more features being added.

Additional resources

- Project wiki, OWASP
  http://www.owasp.org/index.php/Category:OWASP_WebScarab_Project
- Quick start guide, Rogan Dawes
  http://dawes.za.net/rogan/webscarab/quickstart.php
- Getting started tutorial, project wiki, OWASP
  http://www.owasp.org/index.php/WebScarab_Tutorial
- Manual, Rogan Dawes
  http://dawes.za.net/rogan/webscarab/docs/
- Presentations
  - Uncovering WebScarab's Hidden Treasures, Rogan Dawes
    https://www.owasp.org/images/8/88/OWASP_EU_Summit_2008_WebScarab_treasures.ppt
  - WebScarab-NG, Dave Wichers
    http://www.owasp.org/index.php/File:OWASPAppSec2007Milan_WebScarabNG.ppt
- Project mailing list
  https://lists.owasp.org/mailman/listinfo/owasp-webscarab
- OWASP WebGoat and WebScarab, Lulu books
  http://www.lulu.com/product/paperback/owasp-webgoat-and-webscarab/1889624
- WebScarab NG
  http://www.owasp.org/index.php/OWASP_WebScarab_NG_Project
- Zed Attack Proxy
  - Project
    http://www.owasp.org/index.php/OWASP_Zed_Attack_Proxy_Project
  - Download
    http://code.google.com/p/zaproxy/downloads/list
  - Survey
    http://www.kwiksurveys.com/online-survey.php?surveyID=IIEOLN_d6d2ce53