Context App Tool

User Guide – Beta 4

Author: Michael Jordon

Email: cat@contextis.com

Web: http://cat.contextis.com

Contents

1 Introduction

Context App Tool (CAT) is an application to facilitate manual web application penetration testing. Conceptually it is similar to other proxies available both commercially and open source. CAT provides a richer feature set and greater performance, combined with a more intuitive user interface to aid a professional manual penetration tester.

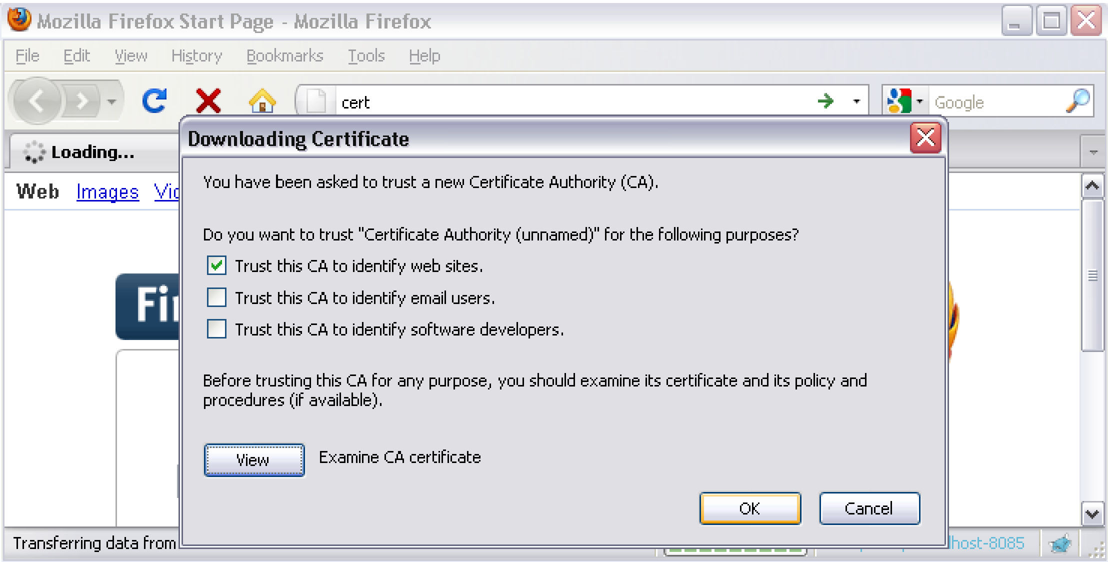
2 User Guide

2.1 Installation

Run the setup.exe from cat.contextis.com, follow the setup wizards. On installation CAT will be installed by default into:

"C:\Program Files\Context Information Security Ltd\CAT\

Also a shortcut will be added to the start menu under All Programs->CAT. On initial launching of CAT a CA certificate will be created for use with the proxy to man-in the-middle SSL. The CA is unique to each installation and the user will be prompted to install the certificate in the Windows certificate store. The purpose is so that CAT can create certificates on the fly for each HTTPS site. If the CA is trusted then Internet Explorer and other fat client application will fully trusted CAT. To install on Firefox and other browsers navigate to 'cert' in the browser and the user will be prompted to install the certificate.


Mozilla Firefox Start Page - Mozilla Firefox
File Edit View History Bookmarks Tools Help
cert
Google
Loading...
Web Images Vid
Transferring data from
host-8085

Downloading Certificate

You have been asked to trust a new Certificate Authority (CA).

Do you want to trust "Certificate Authority (unnamed)" for the following purposes?

☑ Trust this CA to identify web sites.

□ Trust this CA to identify email users.

□ Trust this CA to identify software developers.

Before trusting this CA for any purpose, you should examine its certificate and its policy and procedures (if available).

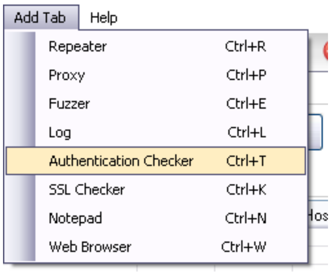
View Examine CA certificate

OK Cancel

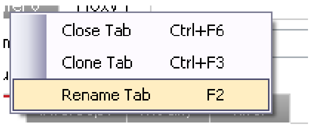
2.2 Menu

The outer menu structure allows for tabs of different tools to be loaded into the application. The tab types can be added by either the 'Add Tab' menu, buttons on the toolbar or short cut keys. The tab types are as follows:

1. Repeater (Ctrl+R) – Used for repeating a single request
2. Proxy (Ctrl+P) – Classic Inline proxy
3. Fuzzer (Ctrl+E) – Allows for batch of tests to be sent to a server for brute forcing, parameter fuzzing, forced browsing etc.
4. Log (Ctrl+L) – View a list of requests to sort, search repeat etc. Allows for a sequence of requests to be repeated and modified.
5. Authentication Checker (CTRL+T) – Two synchronised proxies which can be used to check authentication and authorisation controls.
6. SSL Checker (CTRL+K) – Request a specific page with various SSL ciphers and versions.
7. Notepad (CTRL+N) – A text/RTF editor which can be used as a scratch pad for conversions etc.
8. Web Browser (CTRL+W) – An integrated web browser with proxy pre-configured based on the Internet Explorer's rendering engine.



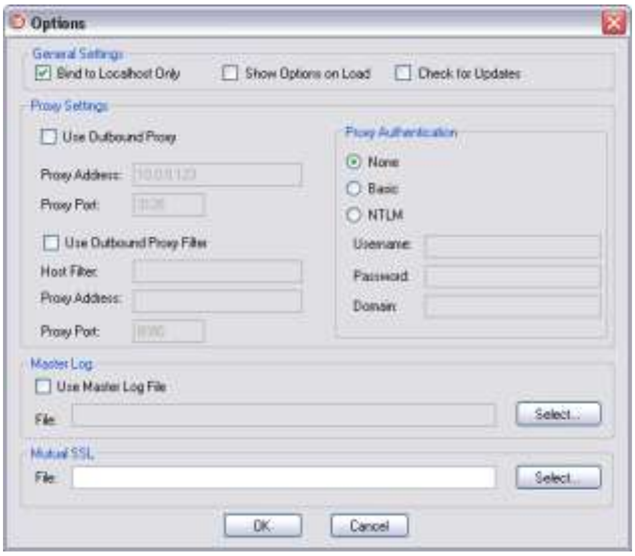
The open and save buttons can be used to save the project including all the tabs, and their data.



Right button on the name of a tab allows for the tab to be closed, cloned (all data copied into a new tab) and renamed. This allows for large projects to be managed more easily and will be saved with the project.

2.3 Options

From the file menu the options for the application can be selected. These options apply to all tabs (where appropriate).



General Settings
☑ Bind to Localhost Only □ Show Options on Load □ Check for Updates

Proxy Settings
□ Use Outbound Proxy
Proxy Address: 10.0.0.123
Proxy Port: 3128
□ Use Outbound Proxy Filter
Host Filter:
Proxy Address:
Proxy Port: 8080

Proxy Authentication
● None
○ Basic
○ NTLM
Username:
Password:
Domain:

Master Log
□ Use Master Log File
File:

Mutual SSL
File:

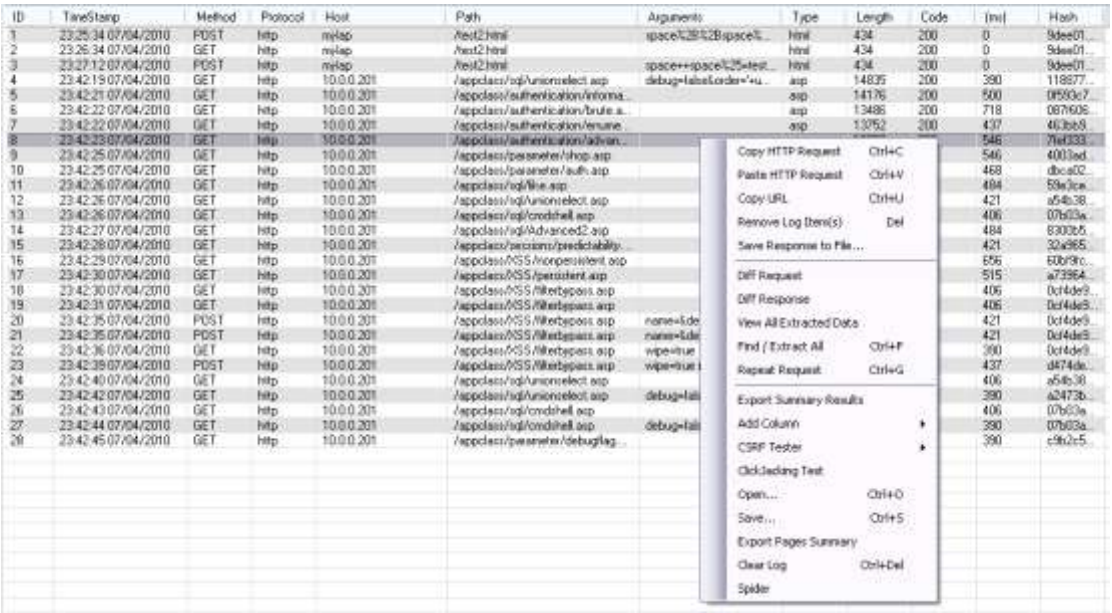
If an outbound proxy is set then all HTTP requests will be sent through that proxy. 'Use Outbound Proxy Filter' can be used to specify for certain hosts to be sent via a different proxy. Proxy authentication can be configured if an upstream proxy requires it.

A master log can be set which will record every request/response sent from the project, including repeater, proxy, fuzzer etc. to a single log file. This file can become very large if a large amount of fuzzing is performed. This is generally used to keep a record of all activity during an engagement. The file can be reloaded into a log panel at anytime.

If the application uses mutual SSL then the certificate can be configured in this window. This certificate will then be used for all HTTPS client connections.

2.4 Log View

Most of the different panels that make up the CAT tool use a log to record the results of the various activities. This log has various features that are common and allow the different components to interact.



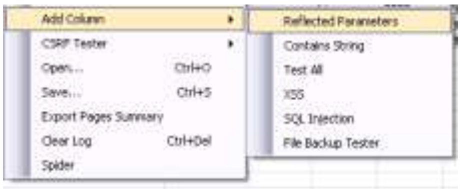
ID	TimeStamp	Method	Protocol	Host	Path	Arguments	Type	Length	Code	(ms)	Hash
1	23:25:34 07/04/2010	POST	http	mylap	/test2.html	space%28%2Bspace%...	html	434	200	0	9dee01...
2	23:26:34 07/04/2010	GET	http	mylap	/test2.html		html	434	200	0	9dee01...
3	23:27:12 07/04/2010	POST	http	mylap	/test2.html	space++space%25=test...	html	434	200	0	9dee01...
4	23:42:19 07/04/2010	GET	http	10.0.0.201	/appclass/sql/unionselect.asp	debug=false&order='+u...	asp	14835	200	390	118877...
5	23:42:21 07/04/2010	GET	http	10.0.0.201	/appclass/authentication/informa...		asp	14176	200	500	0f593c7...
6	23:42:22 07/04/2010	GET	http	10.0.0.201	/appclass/authentication/brute.a...		asp	13486	200	718	087f606...
7	23:42:22 07/04/2010	GET	http	10.0.0.201	/appclass/authentication/enume...		asp	13752	200	437	463bb9...
8	23:42:23 07/04/2010	GET	http	10.0.0.201	/appclass/authentication/advan...					546	7fef333...
9	23:42:25 07/04/2010	GET	http	10.0.0.201	/appclass/parameter/shop.asp					546	4003ad...
10	23:42:25 07/04/2010	GET	http	10.0.0.201	/appclass/parameter/auth.asp					468	dbca02...
11	23:42:26 07/04/2010	GET	http	10.0.0.201	/appclass/sql/like.asp					484	59a3ce...
12	23:42:26 07/04/2010	GET	http	10.0.0.201	/appclass/sql/unionselect.asp					421	a54b38...
13	23:42:26 07/04/2010	GET	http	10.0.0.201	/appclass/sql/cmdshell.asp					406	07b03a...
14	23:42:27 07/04/2010	GET	http	10.0.0.201	/appclass/sql/Advanced2.asp					484	8300b5...
15	23:42:28 07/04/2010	GET	http	10.0.0.201	/appclass/sessions/predictability...					421	32a965...
16	23:42:29 07/04/2010	GET	http	10.0.0.201	/appclass/XSS/nonpersistent.asp					656	60bf9fc...
17	23:42:30 07/04/2010	GET	http	10.0.0.201	/appclass/XSS/persistent.asp					515	a73964...
18	23:42:30 07/04/2010	GET	http	10.0.0.201	/appclass/XSS/filterbypass.asp					406	0cf4de9...
19	23:42:31 07/04/2010	GET	http	10.0.0.201	/appclass/XSS/filterbypass.asp					406	0cf4de9...
20	23:42:35 07/04/2010	POST	http	10.0.0.201	/appclass/XSS/filterbypass.asp	name=&de				421	0cf4de9...
21	23:42:35 07/04/2010	POST	http	10.0.0.201	/appclass/XSS/filterbypass.asp	name=&de				421	0cf4de9...
22	23:42:36 07/04/2010	GET	http	10.0.0.201	/appclass/XSS/filterbypass.asp	wipe=true				390	0cf4de9...
23	23:42:39 07/04/2010	POST	http	10.0.0.201	/appclass/XSS/filterbypass.asp	wipe=true				437	d474de...
24	23:42:40 07/04/2010	GET	http	10.0.0.201	/appclass/sql/unionselect.asp					406	a54b38...
25	23:42:42 07/04/2010	GET	http	10.0.0.201	/appclass/sql/unionselect.asp	debug=fal				390	a2473b...
26	23:42:43 07/04/2010	GET	http	10.0.0.201	/appclass/sql/cmdshell.asp					406	07b03a...
27	23:42:44 07/04/2010	GET	http	10.0.0.201	/appclass/sql/cmdshell.asp	debug=fal				390	07b03a...
28	23:42:45 07/04/2010	GET	http	10.0.0.201	/appclass/parameter/debugflag...					390	c9b2c5...

The log is driven by the user from the right click dropdown menu. An item or items are selected (multiple via shift or control key), then the right button provides various options.

- Copy HTTP Request – Used to move requests from one log to another log/repeater/fuzzer etc. (works on multiple items).
- Paste HTTP Request – Adds the current clipboard request or requests to the log
- Copy URL – Adds the URL of the current single request to the clipboard so it can be pasted into a web browser or notepad.
- Remove Log Item(s) – Remove the selected items from the log.
- Save Response to File… – Prompts the user for a filename and location to save the selected response. This can be used to save binary files that have been downloaded.
- Diff Request/Response – Does a visual diff between the two selected requests or responses.
- View All Extracted Data – Goes through all entries in the log and displays the HTML comments and the decoded viewstates.
- Find / Extract All – Displays the search box for searching the log for certain keywords or to regexing out certain values.
- Repeat Request – Send the request again and add the result to the end of the log.
- Export Summary Results – Creates a HTML file in the specified location which contains all the details in the log list (summary details no content).

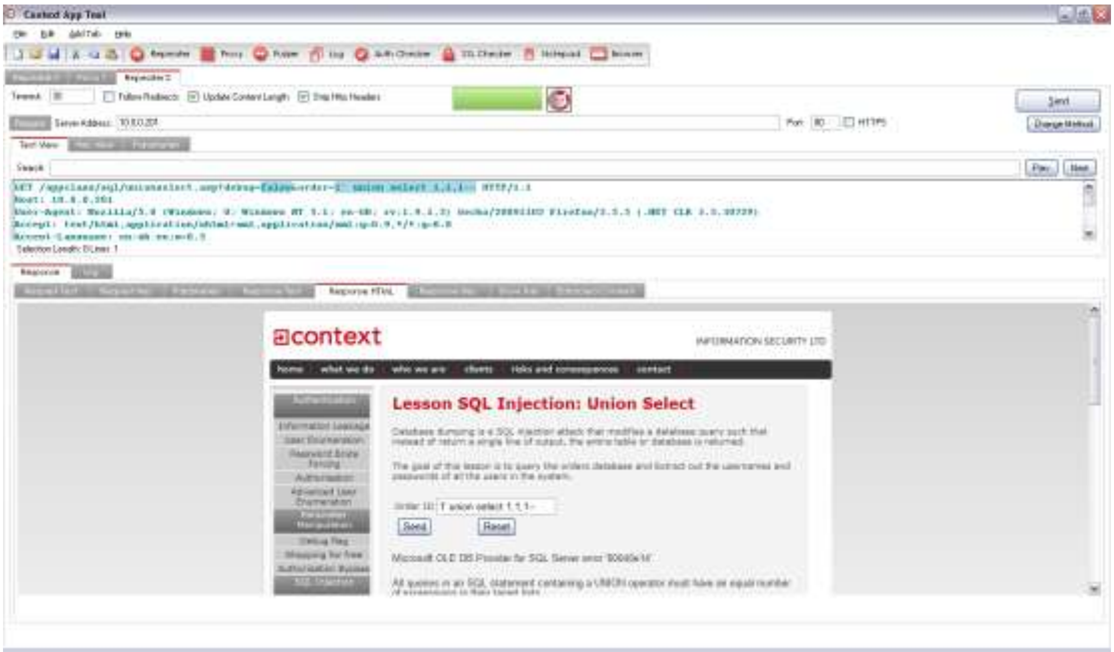
- Add Column – Adds extracted columns with certain extra information.
  - Reflected Parameters – Shows which of the supplied parameter's values appear in the response page. This is used to indentify potential non-persistent XSS. Be aware that a parameter with a very short value will often appear in the page quite frequently but not necessary be from the actual parameter.
  - Contains String – Shows a true/false value from a grep for the string specified.
  - Test All – Will perform the XSS, SQL Injection and backup tester on the request(s).
  - XSS – A basic cross site scripting checker, looking for reflected parameters and then determining if there is a XSS attack string that match is reflected. If an attack string is matched then it will be marked as (Red), if a parameter is just reflected then it will be listed separately.
  - SQL Injection – Enters a series of SQL injection strings into each parameter and then determiners if a SQL error message is returned (Red), the response took more than 20 seconds after three retries (Orange), some responses were different than the original (Yellow) or no change from the original (Green).
  - File Backup Tester – Tests the request(s) to see if there is another version with a backup type extension e.g. .bak, .old, .tmp.

- CSRF Tester – Tools to aid in testing for Cross Site Request Forgery. The create the URL version and the HTML for an auto-posting form.
- Clickjacking Test – Loads two copies of the selected URL, one in an IFRAME and another normally. It uses the cookies of the selected request. If the framed version still operates then the page is most likely vulnerable to Clickjacking.
- Open…, Save… - Allows for all log items to be saved to a XML or CAT file and then reloaded in any log tab at a later date.
- Export Page Summary – Creates a list of information about the pages in the log and any links that have not been followed in a HTML file in the location specified.
- Clear Log – Deletes all entries.
- Spider – Follows all links discovered on that host starting from the links on the request selected. A control box will appear at the bottom showing progress and a stop button. The result of the links and form submissions will be added into the same log view. Any pages with logout in the name will not be followed. Cookies from the original selected request will be used for the spider.



2.5 Repeater Panel

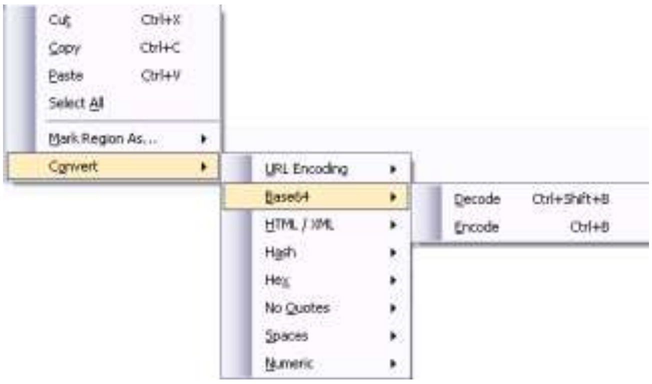
The repeater allows for a single HTTP request to be modified by hand and then repeated back to the server. All aspects of the request can be altered from the three views.



The three views consist of:

- Plain Text – Standard raw HTTP Request

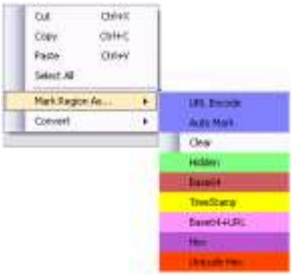
  The exact textural HTTP request that will be sent to the web server can be altered using a text editing box. The content length will be automatically updated when the request is sent. A variety of different encoding options are available under the drop down menu. To use the conversions highlight the text to alter and then either use the right button drop down menu or the short cut keys.



The encoding options are:

- URL – Encoding, Decoding, Unicode and Every character
- Base64 – Encode, Decode
- HTML / XML – Encode using the & => &amp;

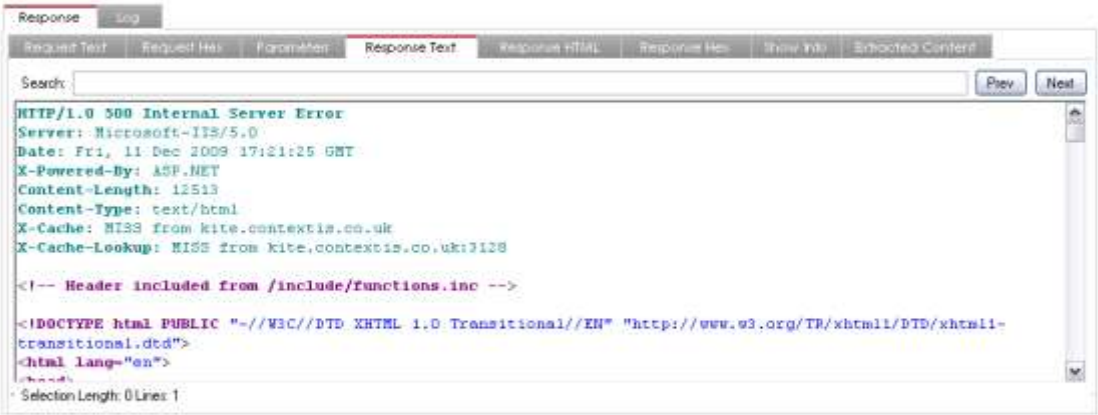
- Hash – MD5SUM or SHA1 the selection and replace it with the result
- Hex – ASCII to Hex e.g. A => 41
- No Quotes - Alters the text into a string of MySQL, SQL Server, Oracle or JavaScript without using quotes e.g. using a character concatenation representations e.g. XSS => String.fromCharCode(88,83,83) (JavaScript)
- Numeric – Hex to Decimal, Decimal to Hex
- The editor also allows for areas of the request to be highlighted in different colours. These colours will then be interpreted when the request is sent and these areas converted. The screenshot above shows areas highlighted in blue which will be URL encoded before being sent. The following screenshot shows the options supported.



- Hex View – For binary manipulations
- Parameter View – Only the GET, POST, MIME and Cookie values are show in a list so they can be altered individually. These values can be double clicked to show only single individual value. From this view the value will be URL encoded before sending and also the colour encoding options mentioned above are also available. This is useful when exploiting a single parameter e.g. SQL injection and the rest of the request is not important.

The response can be viewed in different forms (note that these apply across CAT when HTTP request/responses are shown):

- Request Text – The actual text sent including any conversations specified
- Request Hex – Hex view of the request
- Parameters – A list of the GET, POST, MIME and Cookies sent.
- Response Text – A syntax highlighted view of the actual text in the response

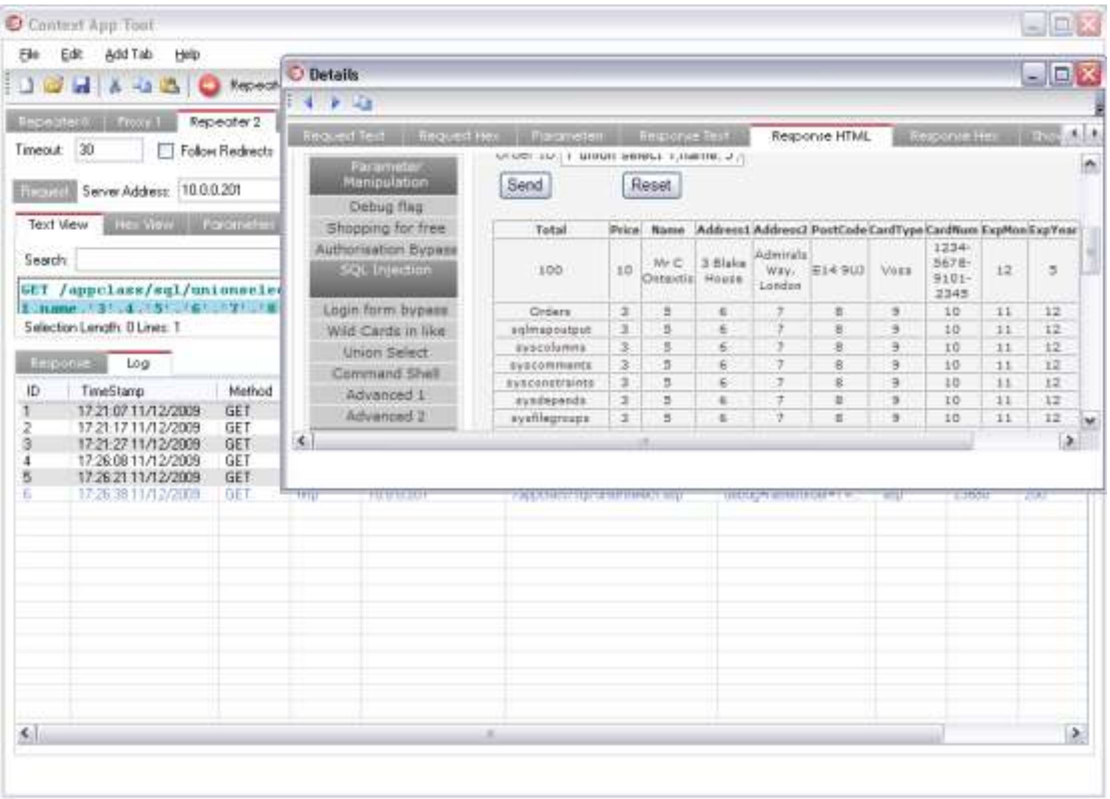


```
HTTP/1.0 500 Internal Server Error
Server: Microsoft-IIS/5.0
Date: Fri, 11 Dec 2009 17:21:25 GMT
X-Powered-By: ASP.NET
Content-Length: 12513
Content-Type: text/html
X-Cache: MISS from kite.contextis.co.uk
X-Cache-Lookup: MISS from kite.contextis.co.uk:3128

<!-- Header included from /include/functions.inc -->

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-
transitional.dtd">
<html lang="en">
<head>
```

Selection Length: 0 Lines: 1

- Response HTML – A rendered view of the HTML, this uses the Internet Explorer rendering engine and will download any resources needed (such as JavaScript, Images, CSS etc.). Furthermore this view can be interacted with so links can be followed (See limitations).
- Response Hex – For binary view of the actual response.
- Show Info – Various meta information about the request include the duration, sizes, server etc.
- Extracted Content – Shows a decoded ViewState and any HTML comments that are on the page. This information can also be extracted across multiple requests see log.

The log tab keeps a record of each request that has been sent through this repeater. This is a standard HTTP log which is used throughout CAT. See Section 2.4 Log View for more details.



Timeout: 30 □ Follow Redirects

Server Address: 10.0.0.201

Search:

```
GET /appclass/sql/unionselec
1,name,'3',4,'5','6','7','8
```

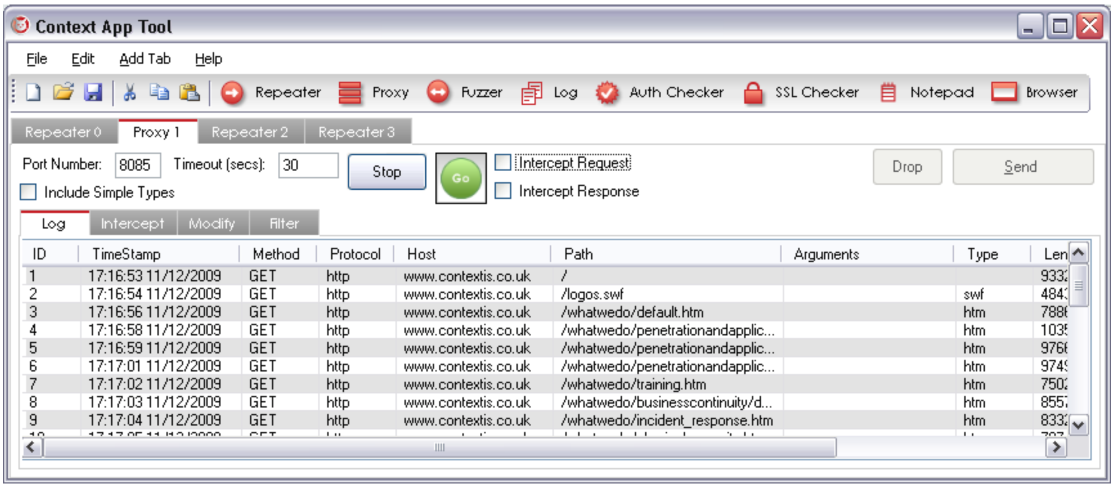
Selection Length: 0 Lines: 1

ID	TimeStamp	Method
1	17:21:07 11/12/2009	GET
2	17:21:17 11/12/2009	GET
3	17:21:27 11/12/2009	GET
4	17:26:08 11/12/2009	GET
5	17:26:21 11/12/2009	GET
6	17:26:38 11/12/2009	GET

Order ID: 1 union select 1,name, 3

Total	Price	Name	Address1	Address2	PostCode	CardType	CardNum	ExpMon	ExpYear
100	10	Mr C Ontextis	3 Blake House	Admirals Way, London	E14 9UJ	Visa	1234-5678-9101-2345	12	5
Orders	3	5	6	7	8	9	10	11	12
sqlmapoutput	3	5	6	7	8	9	10	11	12
syscolumns	3	5	6	7	8	9	10	11	12
syscomments	3	5	6	7	8	9	10	11	12
sysconstraints	3	5	6	7	8	9	10	11	12
sysdepends	3	5	6	7	8	9	10	11	12
sysfilegroups	3	5	6	7	8	9	10	11	12

2.6 Proxy Panel


Context App Tool
File Edit Add Tab Help
Repeater Proxy Fuzzer Log Auth Checker SSL Checker Notepad Browser
Repeater 0 Proxy 1 Repeater 2 Repeater 3
Stop
Go
Drop
Send
Log Intercept Modify Filter

Port Number: 8085 Timeout (secs): 30

□ Include Simple Types

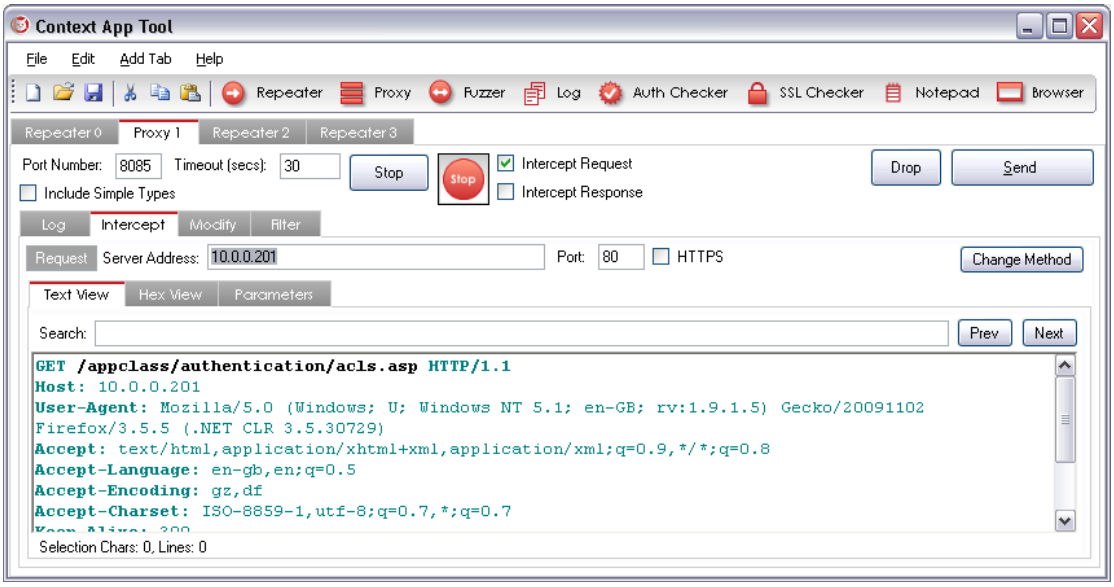
□ Intercept Request

□ Intercept Response

ID	TimeStamp	Method	Protocol	Host	Path	Arguments	Type	Len
1	17:16:53 11/12/2009	GET	http	www.contextis.co.uk	/			933
2	17:16:54 11/12/2009	GET	http	www.contextis.co.uk	/logos.swf		swf	484
3	17:16:56 11/12/2009	GET	http	www.contextis.co.uk	/whatwedo/default.htm		htm	788
4	17:16:58 11/12/2009	GET	http	www.contextis.co.uk	/whatwedo/penetrationandapplic...		htm	103
5	17:16:59 11/12/2009	GET	http	www.contextis.co.uk	/whatwedo/penetrationandapplic...		htm	976
6	17:17:01 11/12/2009	GET	http	www.contextis.co.uk	/whatwedo/penetrationandapplic...		htm	974
7	17:17:02 11/12/2009	GET	http	www.contextis.co.uk	/whatwedo/training.htm		htm	750
8	17:17:03 11/12/2009	GET	http	www.contextis.co.uk	/whatwedo/businesscontinuity/d...		htm	855
9	17:17:04 11/12/2009	GET	http	www.contextis.co.uk	/whatwedo/incident_response.htm		htm	833

The proxy is a standard inline proxy. The default port is 8085 and will increment from there for each proxy that is loaded. To use the proxy configure your browser to proxy through localhost on port 8085. If browsing requires a web proxy then this should be set in the file->options menu of CAT as the outbound proxy. Now every request will be sent through CAT. CAT by default will filter out 'simple types' of requests, namely images, style sheets etc. this can be changed using the 'Include Simple Types' checkbox. If only certain host should be captured or there are certain requests that are not wanted then the filter tab can be used to set these.

To intercept a request or response set the check box 'intercept Request/Response' on the next request the appropriate tab will be shown and CAT brought to the front. The same HTTP editing options will be shown as per the repeater.


Context App Tool
File Edit Add Tab Help
Repeater Proxy Fuzzer Log Auth Checker SSL Checker Notepad Browser
Repeater 0 Proxy 1 Repeater 2 Repeater 3
Stop
Stop
Drop
Send
Log Intercept Modify Filter
Request
Change Method
Text View Hex View Parameters
Prev
Next

Port Number: 8085 Timeout (secs): 30

□ Include Simple Types

☑ Intercept Request

□ Intercept Response

Server Address: 10.0.0.201 Port: 80 □ HTTPS

Search:

```
GET /appclass/authentication/acls.asp HTTP/1.1
Host: 10.0.0.201
User-Agent: Mozilla/5.0 (Windows; U; Windows NT 5.1; en-GB; rv:1.9.1.5) Gecko/20091102
Firefox/3.5.5 (.NET CLR 3.5.30729)
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: en-gb,en;q=0.5
Accept-Encoding: gz,df
Accept-Charset: ISO-8859-1,utf-8;q=0.7,*;q=0.7
Keep-Alive: 300
```

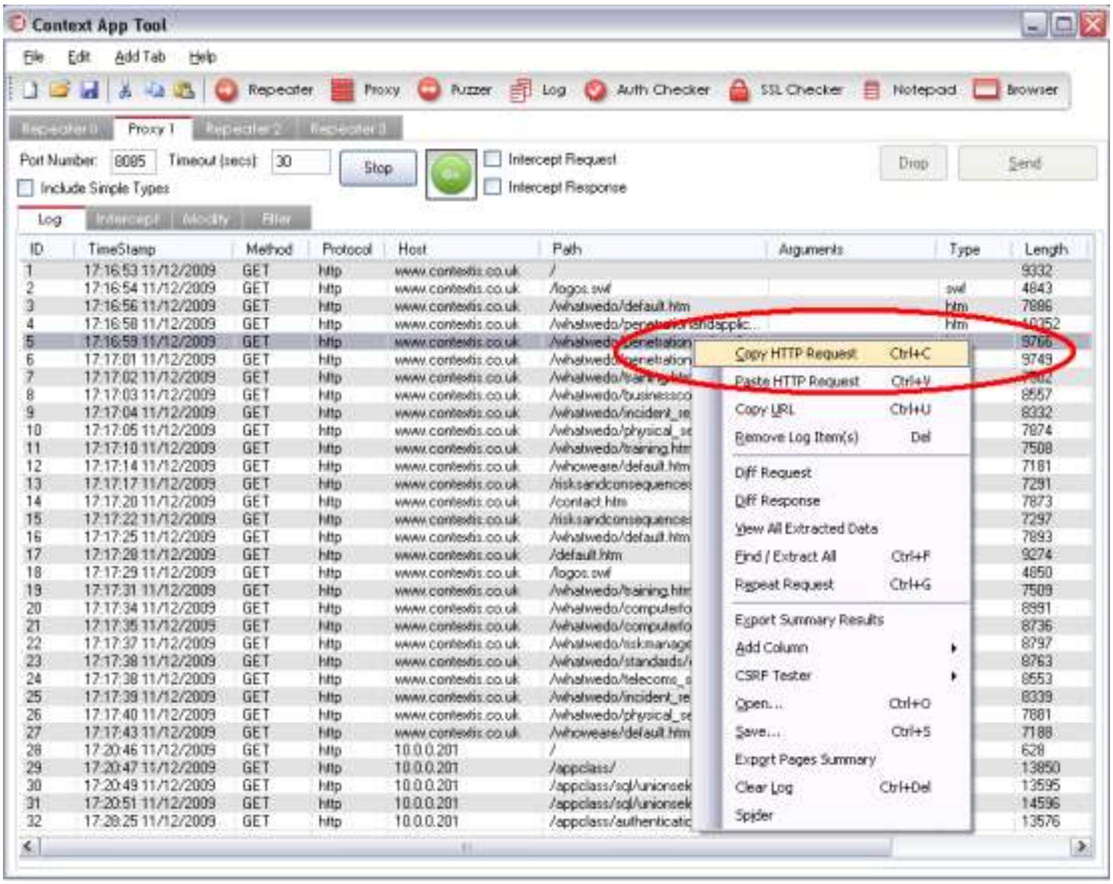
Selection Chars: 0, Lines: 0

Use the send or drop buttons to dispatch this request. If there are several requests stacked up then you can process these one by one or untick the 'intercept request/response' option and they will all be sent through.

The modify tab allows for changes to be made on the fly without the need to manually alter the request and responses. This includes regular expression alterations.

The log tab shows the history of what requests have been seen through the proxy. From here they can be copied into the other tabs e.g. into a repeater for further investigation or the fuzzer for testing. Here are the steps to move a request from the proxy to the repeater:

1. Select the request in the log
2. Select "Copy HTTP Request"

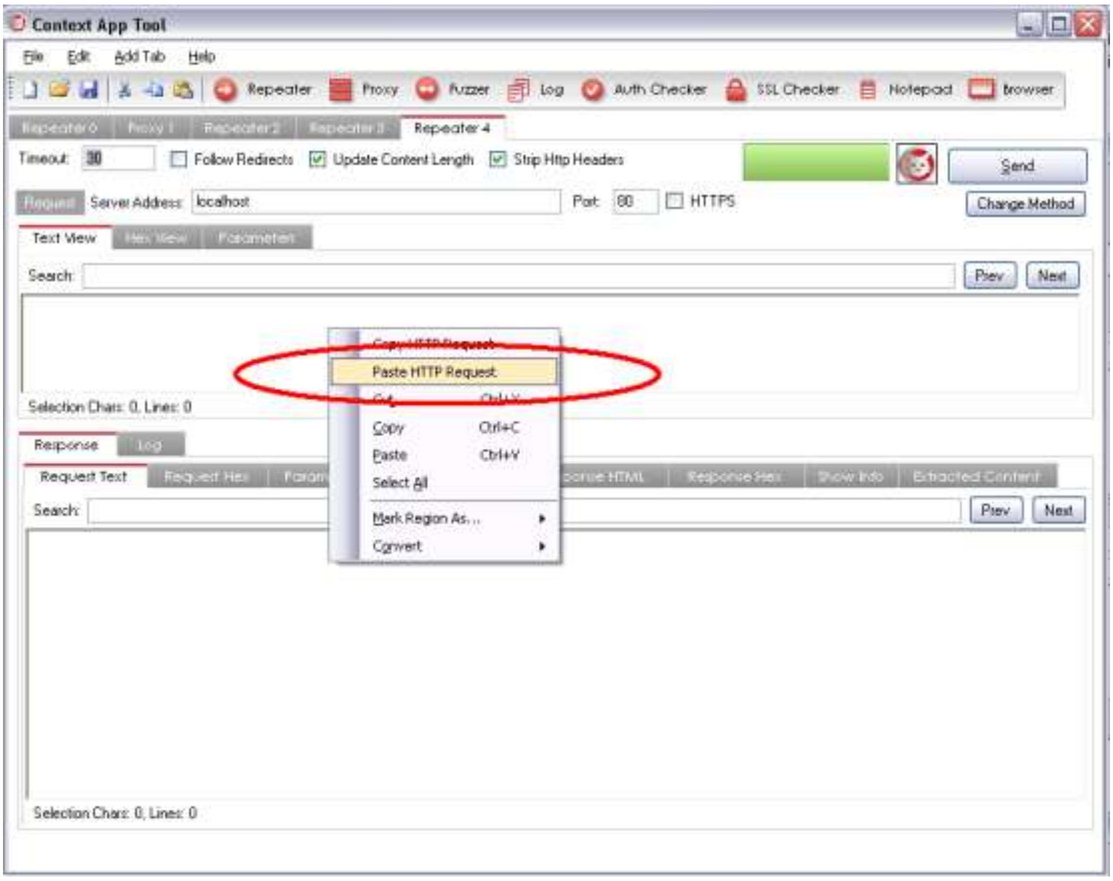


Port Number: 8085 Timeout (secs): 30

□ Include Simple Types □ Intercept Request □ Intercept Response

ID	TimeStamp	Method	Protocol	Host	Path	Arguments	Type	Length
1	17:16:53 11/12/2009	GET	http	www.contextis.co.uk	/			9332
2	17:16:54 11/12/2009	GET	http	www.contextis.co.uk	/logos.swf		swf	4843
3	17:16:56 11/12/2009	GET	http	www.contextis.co.uk	/whatwedo/default.htm		htm	7886
4	17:16:58 11/12/2009	GET	http	www.contextis.co.uk	/whatwedo/penetrationandapplic...		htm	10352
5	17:16:59 11/12/2009	GET	http	www.contextis.co.uk	/whatwedo/penetration			9766
6	17:17:01 11/12/2009	GET	http	www.contextis.co.uk	/whatwedo/penetration			9749
7	17:17:02 11/12/2009	GET	http	www.contextis.co.uk	/whatwedo/[illegible]			[illegible]
8	17:17:03 11/12/2009	GET	http	www.contextis.co.uk	/whatwedo/businessco			8557
9	17:17:04 11/12/2009	GET	http	www.contextis.co.uk	/whatwedo/incident_re			8332
10	17:17:05 11/12/2009	GET	http	www.contextis.co.uk	/whatwedo/physical_se			7874
11	17:17:10 11/12/2009	GET	http	www.contextis.co.uk	/whatwedo/training.htm			7508
12	17:17:14 11/12/2009	GET	http	www.contextis.co.uk	/whoweare/default.htm			7181
13	17:17:17 11/12/2009	GET	http	www.contextis.co.uk	/risksandconsequences			7291
14	17:17:20 11/12/2009	GET	http	www.contextis.co.uk	/contact.htm			7873
15	17:17:22 11/12/2009	GET	http	www.contextis.co.uk	/risksandconsequences			7297
16	17:17:25 11/12/2009	GET	http	www.contextis.co.uk	/whatwedo/default.htm			7893
17	17:17:28 11/12/2009	GET	http	www.contextis.co.uk	/default.htm			9274
18	17:17:29 11/12/2009	GET	http	www.contextis.co.uk	/logos.swf			4850
19	17:17:31 11/12/2009	GET	http	www.contextis.co.uk	/whatwedo/training.htm			7509
20	17:17:34 11/12/2009	GET	http	www.contextis.co.uk	/whatwedo/computerfo			8991
21	17:17:35 11/12/2009	GET	http	www.contextis.co.uk	/whatwedo/computerfo			8736
22	17:17:37 11/12/2009	GET	http	www.contextis.co.uk	/whatwedo/riskmanage			8797
23	17:17:38 11/12/2009	GET	http	www.contextis.co.uk	/whatwedo/standards/			8763
24	17:17:38 11/12/2009	GET	http	www.contextis.co.uk	/whatwedo/telecoms_s			8553
25	17:17:39 11/12/2009	GET	http	www.contextis.co.uk	/whatwedo/incident_re			8339
26	17:17:40 11/12/2009	GET	http	www.contextis.co.uk	/whatwedo/physical_se			7881
27	17:17:43 11/12/2009	GET	http	www.contextis.co.uk	/whoweare/default.htm			7188
28	17:20:46 11/12/2009	GET	http	10.0.0.201	/			628
29	17:20:47 11/12/2009	GET	http	10.0.0.201	/appclass/			13850
30	17:20:49 11/12/2009	GET	http	10.0.0.201	/appclass/sql/unionsek			13595
31	17:20:51 11/12/2009	GET	http	10.0.0.201	/appclass/sql/unionsek			14596
32	17:28:25 11/12/2009	GET	http	10.0.0.201	/appclass/authenticatic			13576

3. Click on Repeater on the tool bar or Add Tab->Repeater, to create a new repeater tab.
4. Then right button on the top box for HTTP request editing.
5. Select 'Paste HTTP Request'


Context App Tool
File Edit Add Tab Help
Repeater Proxy Fuzzer Log Auth Checker SSL Checker Notepad Browser
Repeater 0 Proxy 1 Repeater 2 Repeater 3 Repeater 4
Send
Change Method
Request
Text View Hex View Parameters
Search:
Prev Next
Copy HTTP Request
Paste HTTP Request
Cut Ctrl+X
Copy Ctrl+C
Paste Ctrl+V
Select All
Mark Region As...
Convert
Response Log
Request Text Request Hex Parameters Response HTML Response Hex Show Info Extracted Content

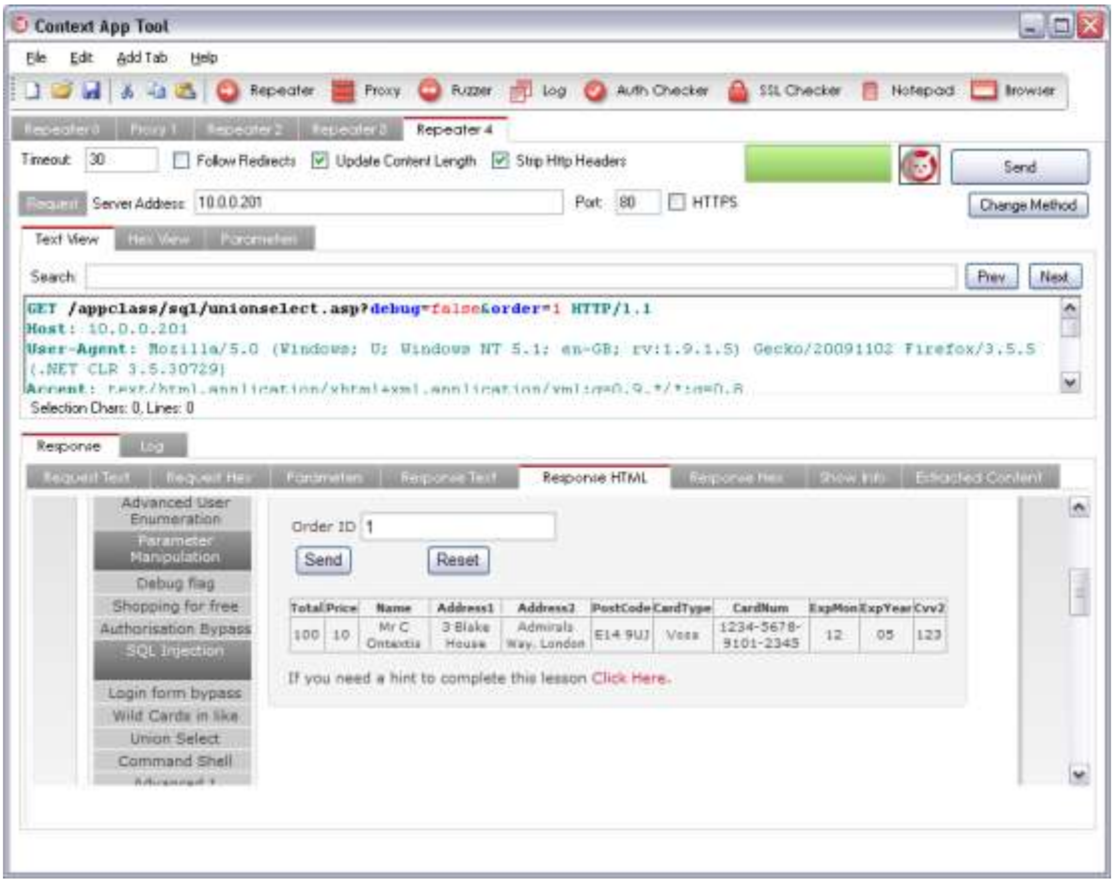
Timeout: 30 ☐ Follow Redirects ☑ Update Content Length ☑ Strip Http Headers

Server Address: localhost Port: 80 ☐ HTTPS

Selection Chars: 0, Lines: 0

Selection Chars: 0, Lines: 0

6. Press the send button to repeat the request.


Context App Tool
File Edit Add Tab Help
Repeater Proxy Fuzzer Log Auth Checker SSL Checker Notepad Browser
Repeater 0 Proxy 1 Repeater 2 Repeater 3 Repeater 4
Send
Change Method
Request
Text View Hex View Parameters
Search:
Prev Next
Response Log
Request Text Request Hex Parameters Response Text Response HTML Response Hex Show Info Extracted Content
Advanced User Enumeration
Parameter Manipulation
Debug flag
Shopping for free
Authorisation Bypass
SQL Injection
Login form bypass
Wild Cards in like
Union Select
Command Shell
Advanced 1
Send
Reset

Timeout: 30 ☐ Follow Redirects ☑ Update Content Length ☑ Strip Http Headers

Server Address: 10.0.0.201 Port: 80 ☐ HTTPS

```
GET /appclass/sql/unionselect.asp?debug=false&order=1 HTTP/1.1
Host: 10.0.0.201
User-Agent: Mozilla/5.0 (Windows; U; Windows NT 5.1; en-GB; rv:1.9.1.5) Gecko/20091102 Firefox/3.5.5
(.NET CLR 3.5.30729)
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
```

Selection Chars: 0, Lines: 0

Order ID 1

Total	Price	Name	Address1	Address2	PostCode	CardType	CardNum	ExpMon	ExpYear	Cvv2
100	10	Mr C Ontextis	3 Blake House	Admirals Way, London	E14 9UJ	Vosa	1234-5678-9101-2345	12	05	123

If you need a hint to complete this lesson Click Here.

2.7 Fuzzer Panel

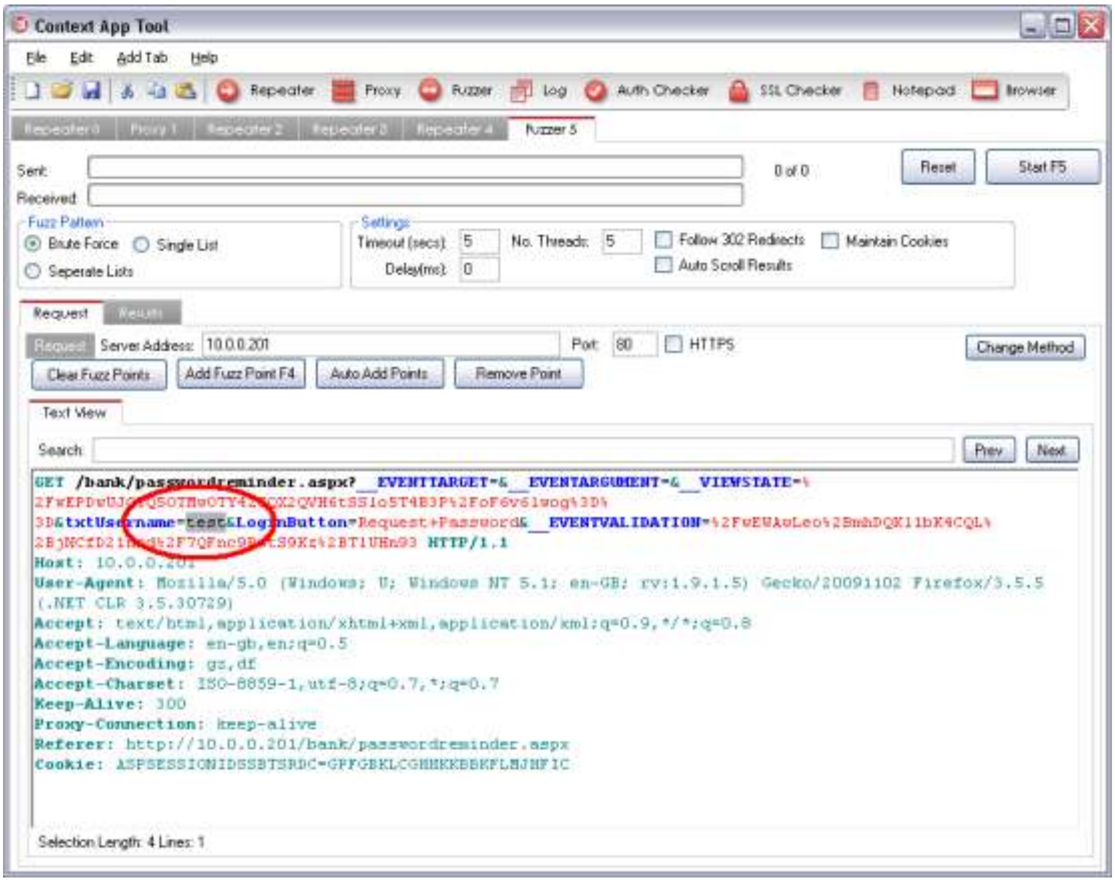
The fuzzer is used to create multiple requests based on a templated request. This is altered for each fuzz case, and can be used for example to:

- Directory Brute Forcing
- Username Enumeration
- Password Brute Forcing
- Parameter Fuzzing
- Parameter Brute Forcing
- SQL Injection exploit crafting
- Blind SQL/LDAP/XPATH data extraction
- Boundary Condition Checking

2.7.1 Username Enumeration Fuzz Example

The classic examples are username enumeration and password brute forcing. Where a template of a login request is captured and then repeated with the username being altered on each request. The results are then filtered for usernames that are valid and then a second test fuzz with the valid usernames that have been discovered and a list of common passwords. The results from this are then filtered again to find which requests were successful. The CAT fuzzer has a great range of flexibility in terms of the types of fuzzing that can be performed. For this document the above example will be shown.

1. Using either a web browser + proxy, the integrated web browser or via the repeater. Find a request which differentiates valid users. Copy this request into the fuzzer using the same copy and paste technique mentioned previously.
2. Highlight the characters that are to be replaced.



Fuzz Pattern: ● Brute Force ○ Single List ○ Seperate Lists

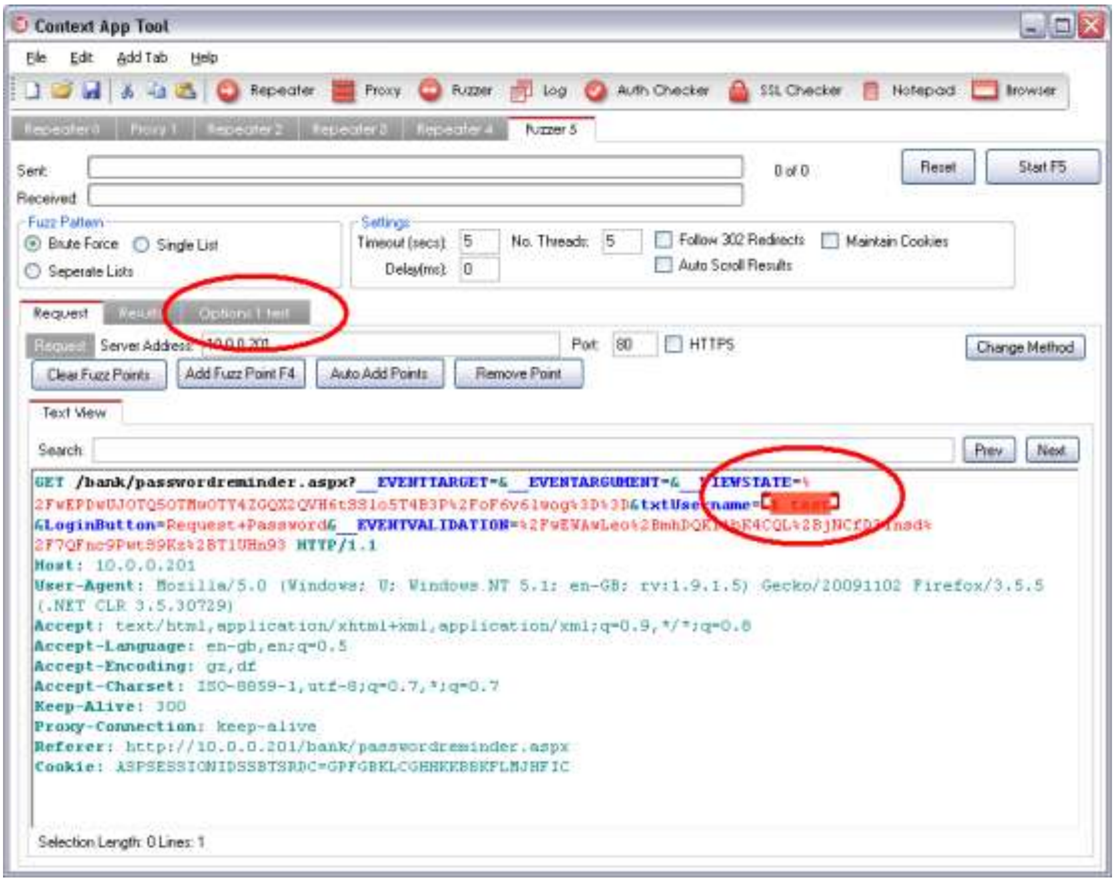
Settings: Timeout (secs): 5 No. Threads: 5 □ Follow 302 Redirects □ Maintain Cookies
Delay(ms): 0 □ Auto Scroll Results

Server Address: 10.0.0.201 Port: 80 □ HTTPS

```
GET /bank/passwordreminder.aspx?__EVENTTARGET=&__EVENTARGUMENT=&__VIEWSTATE=%
2FwEPDwUJ[illegible]Q5OTMwOTY4[illegible]CX2QVH6tSS1o5T4B3P%2FoF6v61wog%3D%
3D&txtUsername=test&LoginButton=Request+Password&__EVENTVALIDATION=%2FwEWAwLeo%2BmhDQK11bK4CQL%
2BjNCfD21[illegible]d%2F7QFnc9[illegible]tS9Kz%2BT1UHn93 HTTP/1.1
Host: 10.0.0.201
User-Agent: Mozilla/5.0 (Windows; U; Windows NT 5.1; en-GB; rv:1.9.1.5) Gecko/20091102 Firefox/3.5.5
(.NET CLR 3.5.30729)
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: en-gb,en;q=0.5
Accept-Encoding: gz,df
Accept-Charset: ISO-8859-1,utf-8;q=0.7,*;q=0.7
Keep-Alive: 300
Proxy-Connection: keep-alive
Referer: http://10.0.0.201/bank/passwordreminder.aspx
Cookie: ASPSESSIONIDSSBTSRDC=GPFGBKLCGHMKKBBKFLMJNFIC
```

Selection Length: 4 Lines: 1

3. Select 'Add Fuzz Point' or F4. This results in the existing word being highlighted and a new tab being added for the setting for that fuzz point.


Context App Tool
File Edit Add Tab Help
Repeater Proxy Fuzzer Log Auth Checker SSL Checker Notepad Browser
Repeater 0 Proxy 1 Repeater 2 Repeater 3 Repeater 4 Fuzzer 5
Reset Start F5
Request Result Options 1 test
Change Method
Clear Fuzz Points Add Fuzz Point F4 Auto Add Points Remove Point
Text View
Prev Next

Sent:

Received:

0 of 0

Fuzz Pattern: ● Brute Force ○ Single List ○ Seperate Lists

Settings: Timeout (secs): 5 No. Threads: 5 □ Follow 302 Redirects □ Maintain Cookies Delay(ms): 0 □ Auto Scroll Results

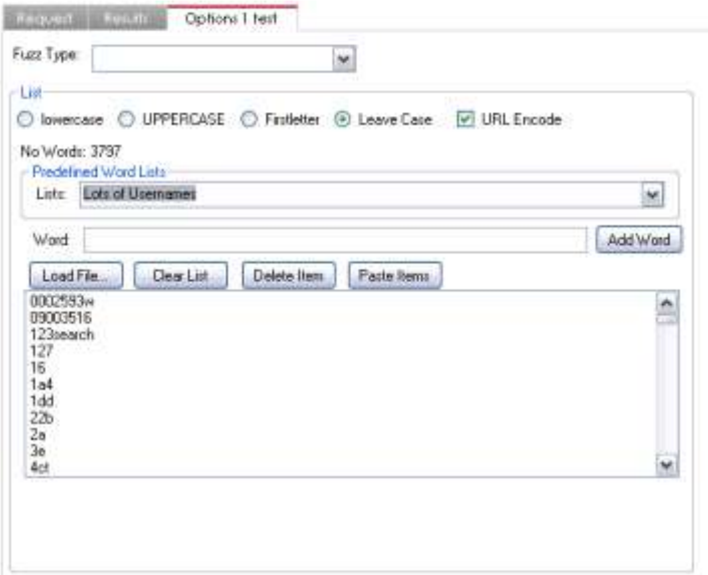
Server Address: 10.0.0.201 Port: 80 □ HTTPS

Search:

```
GET /bank/passwordreminder.aspx?__EVENTTARGET=&__EVENTARGUMENT=&__VIEWSTATE=%
2FwEPDwUJOTQ5OTMwOTY4ZGQX2QVH6t881o5T4B3P%2FoF6v61wog%3D%3D&txtUsername=test
&LoginButton=Request+Password&__EVENTVALIDATION=%2FwEWAwLeo%2BmhDQKMbF4CQL%2BjNCfDJnsd%
2F7QFnc9Pwt89Kz%2BT1UHn93 HTTP/1.1
Host: 10.0.0.201
User-Agent: Mozilla/5.0 (Windows; U; Windows NT 5.1; en-GB; rv:1.9.1.5) Gecko/20091102 Firefox/3.5.5
(.NET CLR 3.5.30729)
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: en-gb,en;q=0.5
Accept-Encoding: gz,df
Accept-Charset: ISO-8859-1,utf-8;q=0.7,*;q=0.7
Keep-Alive: 300
Proxy-Connection: keep-alive
Referer: http://10.0.0.201/bank/passwordreminder.aspx
Cookie: ASPSESSIONIDSSBTSRDC=GPFGBKLCGHHKKBBKFLMJHFIC
```

Selection Length: 0 Lines: 1

4. On the new tab ‘Options 1 test’ select ‘Word List’ as the fuzz type and then select a predefine list of words ‘Lots of Usernames’. (These predefined lists are by default in C:\Program Files\Context Information Security Ltd\CAT\WordLists, which can be manually modified).


Request Result Options 1 test
Add Word
Load File... Clear List Delete Item Paste Items

Fuzz Type:

List: ○ lowercase ○ UPPERCASE ○ Firstletter ● Leave Case ☑ URL Encode

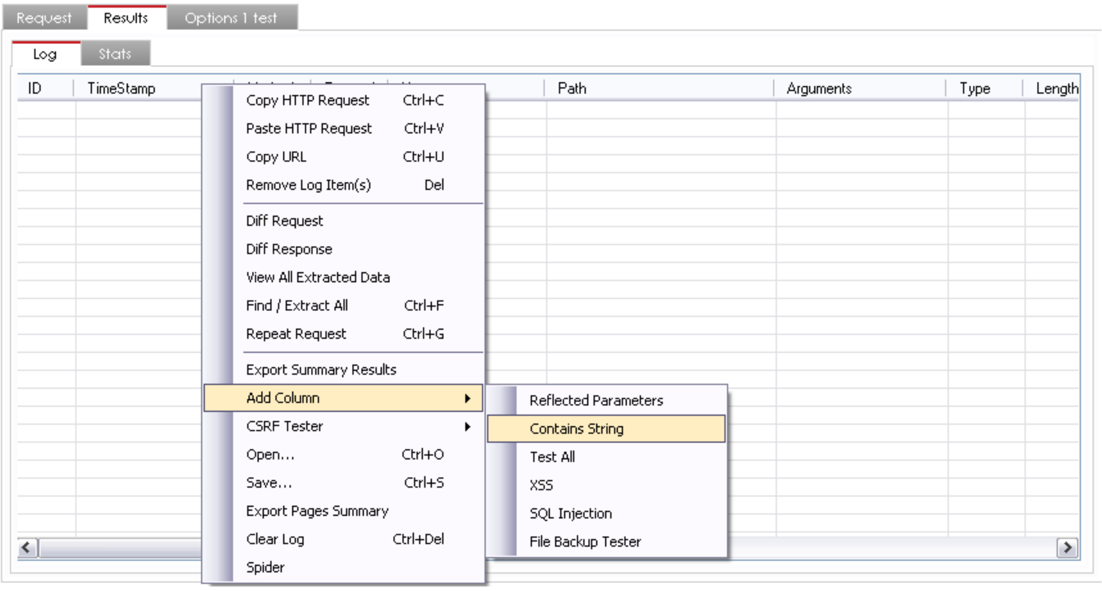
No Words: 3797

Predefined Word Lists: Lists: Lots of Usernames

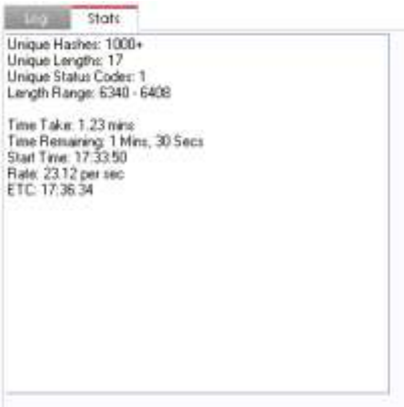
Word:

```
0002593w
09003516
123search
127
16
1a4
1dd
22b
2a
3e
4ct
```

5. On the Results tab press the right button to get the pop-up menu and Add Column->Contains String. In the popup box add the string which will indentify a valid username in the response text. This adds a column to the results which can be sorted showing which response contained the string that indentifies them as being valid. This feature can be used pre, post and during a fuzz test.

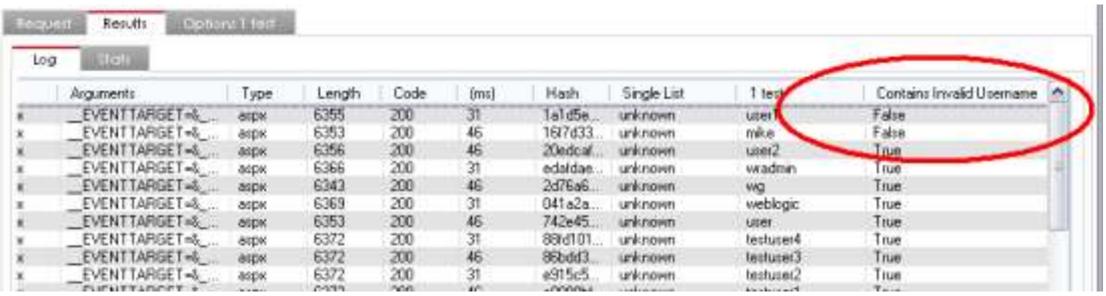


6. Press start on the fuzzer and the brute force will begin.
7. In the stats page it shows various information including the number of unique response size, hashes code, and ETA of the fuzz test.



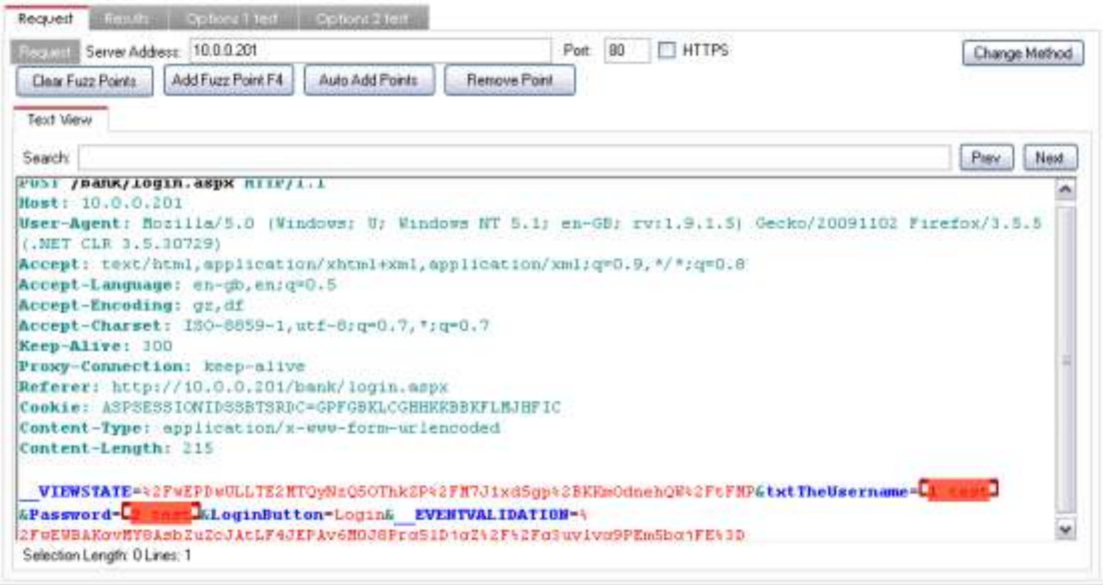
```
Unique Hashes: 1000+
Unique Lengths: 17
Unique Status Codes: 1
Length Range: 6340 - 6408

Time Take: 1.23 mins
Time Remaining: 1 Mins, 30 Secs
Start Time: 17:33:50
Rate: 23.12 per sec
ETC: 17:36:34
```

8. By clicking on the header in the log tab the list is sorted by that column. By sorting by the ‘Contains String’ column the results with valid usernames will be shown together at the top.



	Arguments	Type	Length	Code	(ms)	Hash	Single List	1 tes	Contains Invalid Username
x	__EVENTTARGET=&_...	aspx	6355	200	31	1a1d5e...	unknown	user1	False
x	__EVENTTARGET=&_...	aspx	6353	200	46	1617d33...	unknown	mike	False
x	__EVENTTARGET=&_...	aspx	6356	200	46	20edcaf...	unknown	user2	True
x	__EVENTTARGET=&_...	aspx	6366	200	31	edafdae...	unknown	wadmin	True
x	__EVENTTARGET=&_...	aspx	6343	200	46	2d76a6...	unknown	wg	True
x	__EVENTTARGET=&_...	aspx	6369	200	31	041a2a...	unknown	weblogic	True
x	__EVENTTARGET=&_...	aspx	6353	200	46	742e45...	unknown	user	True
x	__EVENTTARGET=&_...	aspx	6372	200	31	88d101...	unknown	testuser4	True
x	__EVENTTARGET=&_...	aspx	6372	200	46	86bdd3...	unknown	testuser3	True
x	__EVENTTARGET=&_...	aspx	6372	200	31	e915c5...	unknown	testuser2	True

9. To brute force the passwords, a tag is added as per step 3 for the username and the password on the login page.



Server Address: 10.0.0.201 Port: 80 □ HTTPS

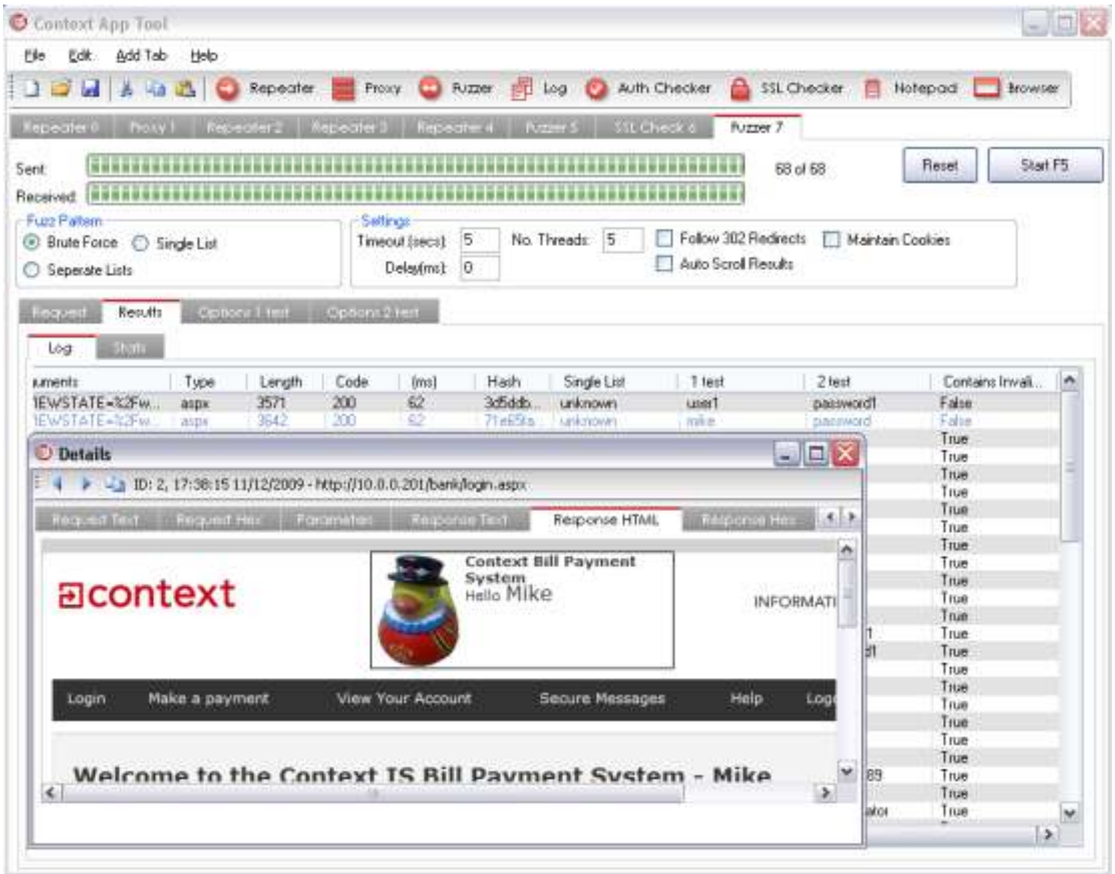
```
POST /bank/login.aspx HTTP/1.1
Host: 10.0.0.201
User-Agent: Mozilla/5.0 (Windows; U; Windows NT 5.1; en-GB; rv:1.9.1.5) Gecko/20091102 Firefox/3.5.5
(.NET CLR 3.5.30729)
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: en-gb,en;q=0.5
Accept-Encoding: gz,df
Accept-Charset: ISO-8859-1,utf-8;q=0.7,*;q=0.7
Keep-Alive: 300
Proxy-Connection: keep-alive
Referer: http://10.0.0.201/bank/login.aspx
Cookie: ASPSESSIONIDSSBTSRDC=GPFGBKLCGHHKKBBKFLMJBFIC
Content-Type: application/x-www-form-urlencoded
Content-Length: 215

__VIEWSTATE=%2FwEPDwULLTE2MTQyNzQ5OThkZP%2FM7J1xd5gp%2BKKmOdnehQW%2FtFMP&txtTheUsername=1 test
&Password=2 test&LoginButton=Login&__EVENTVALIDATION=%
2FwEWBAKovMY8AsbZuZcJAtLF4JEPAv6MOJ8Fra51D1aZ%2F%2Fa3uvlva9PEm5ba1FE%3D
```

Selection Length: 0 Lines: 1

10. The first tab is then populated with the valid usernames and the second has the predefined word list of ‘common passwords’ selected.

11. A column is added to the results to detect either successful login or unsuccessful. This is then set as the column to order the list by.

12. Start the fuzzer and successful authentications will be at the top of the list.



Sent: 68 of 68

Received:

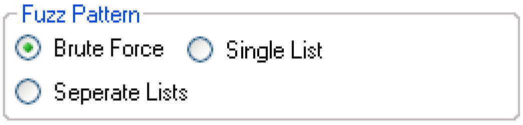
Fuzz Pattern: ● Brute Force ○ Single List ○ Seperate Lists

Settings: Timeout (secs): 5 No. Threads: 5 □ Follow 302 Redirects □ Maintain Cookies Delay(ms): 0 □ Auto Scroll Results

uments	Type	Length	Code	(ms)	Hash	Single List	1 test	2 test	Contains Invali...
IEWSTATE=%2Fw...	aspx	3571	200	62	3d5ddb...	unknown	user1	password1	False
IEWSTATE=%2Fw...	aspx	3642	200	62	71e65fa...	unknown	mike	password	False
									True
									True
									True
									True
									True
									True
									True
									True
									True
									True
									True
								1	True
								d1	True
									True
									True
									True
									True
									True
									True
								89	True
									True
								ator	True

ID: 2, 17:38:15 11/12/2009 - http://10.0.0.201/bank/login.aspx

2.7.2 Fuzz Patterns



Fuzz Pattern

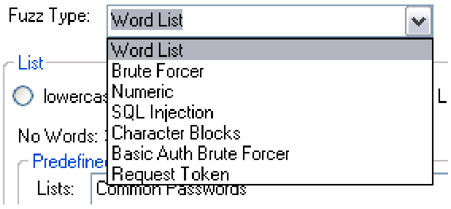
● Brute Force ○ Single List

○ Seperate Lists

The fuzz pattern defines how the fuzz points and fuzz lists will be combined into the actual test cases. If there is a single fuzz point then all patterns result in the same requests. The difference is how many lists and how many resulting test cases are generated.

- A 'brute force' will do every item on the first list with every item on the second list etc. This is the classic username and password brute force e.g. Total Cases = L0 x L1 X L2 (L0 is the length of the first list etc.)
- 'Single List' is where for each fuzz point a global list that will be applied for each parameter in turn. This is normally used for fuzzing each parameter one by one by applying a pre-define list of fuzz characters. Total cases = L0 x No. Fuzz Points
- 'Separate Lists' will have a tab for each fuzz point. For each fuzz case an item from each of the lists will be taken. Total Cases = Min(L0,L1,L2)

2.7.3 Fuzz Types

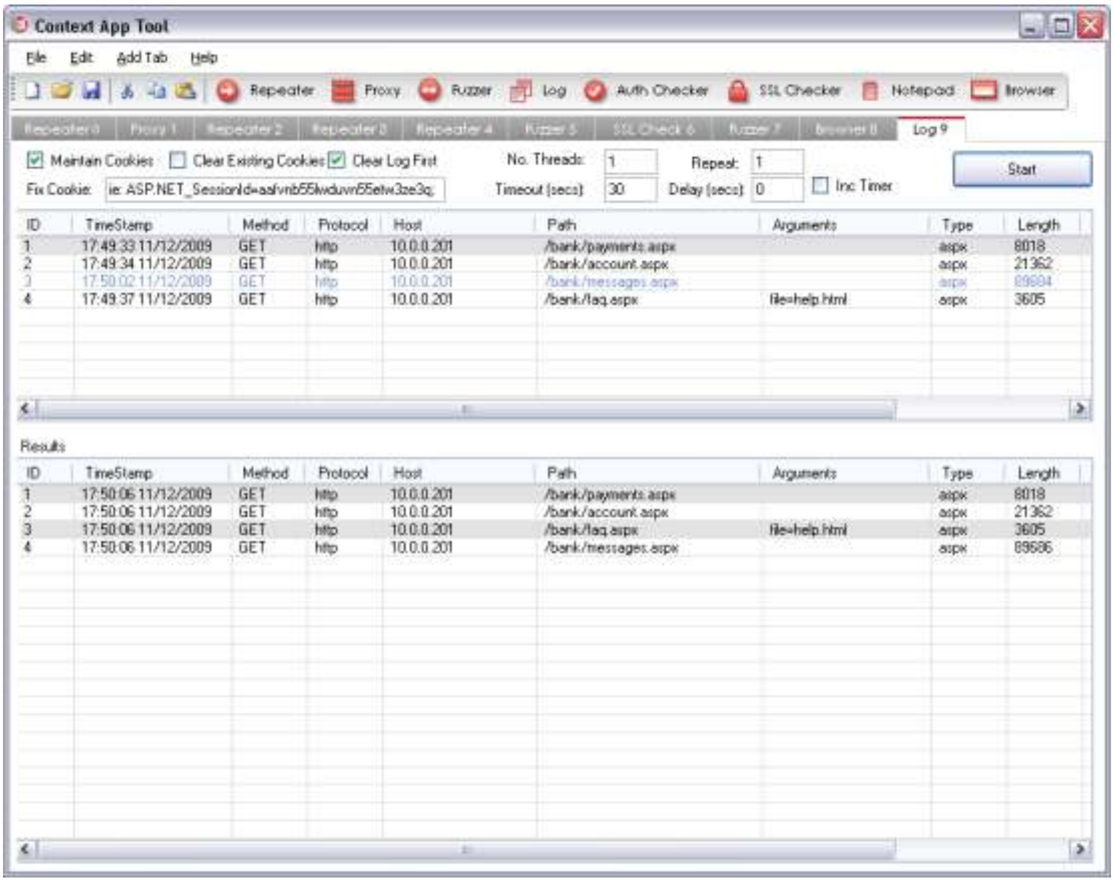


Fuzz Type: Word List

- Word List – A list of words either entered manually, loaded from a file or pre-defined. The capitalisation can be altered.
- Brute Forcer – A range of characters where every permutation will be tested.
- Numeric – A range of numbers either hex or decimal.
- SQL Injection – Two tools for brute forcing the UNION SELECT length and types.
- Character Blocks – A string of increasing lengths from a base character(s). e.g 10xA.
- Basic Authentication Brute Forcer – Performs a brute force against a HTTP basic Authentication web site.

2.8 Log Panel

The log panel contains a HTTP log which is used in various places throughout the application for storing a history of activity or results. The log panel provides a log for processing the data and repeating a sequence of requests.



☑ Maintain Cookies ☐ Clear Existing Cookies ☑ Clear Log First

No. Threads: 1 Repeat: 1

Fix Cookie: ie: ASP.NET_SessionId=aafvnb55lwduvn55etw3ze3q;

Timeout (secs): 30 Delay (secs): 0 ☐ Inc Timer

ID	TimeStamp	Method	Protocol	Host	Path	Arguments	Type	Length
1	17:49:33 11/12/2009	GET	http	10.0.0.201	/bank/payments.aspx		aspx	8018
2	17:49:34 11/12/2009	GET	http	10.0.0.201	/bank/account.aspx		aspx	21362
3	17:50:02 11/12/2009	GET	http	10.0.0.201	/bank/messages.aspx		aspx	89684
4	17:49:37 11/12/2009	GET	http	10.0.0.201	/bank/faq.aspx	file=help.html	aspx	3605

Results

ID	TimeStamp	Method	Protocol	Host	Path	Arguments	Type	Length
1	17:50:06 11/12/2009	GET	http	10.0.0.201	/bank/payments.aspx		aspx	8018
2	17:50:06 11/12/2009	GET	http	10.0.0.201	/bank/account.aspx		aspx	21362
3	17:50:06 11/12/2009	GET	http	10.0.0.201	/bank/faq.aspx	file=help.html	aspx	3605
4	17:50:06 11/12/2009	GET	http	10.0.0.201	/bank/messages.aspx		aspx	89686

By double clicking on any log item the item will be loaded into a new window with the usual view of the request, response etc. From the log panel this window can be used to edit the request for the purpose of replaying with different values. This is useful in complex SSO login processes or where a set work flow is used. If a parameter at the beginning of the sequence is not used until the end then this is the interface that can be used to test this case.

The ‘Fix Cookie’ allows for the requests to be repeated with a different cookie for authentication/authorisation checking.

The ‘Maintain Cookie’ will pick up new cookies if they are set during the sequence.

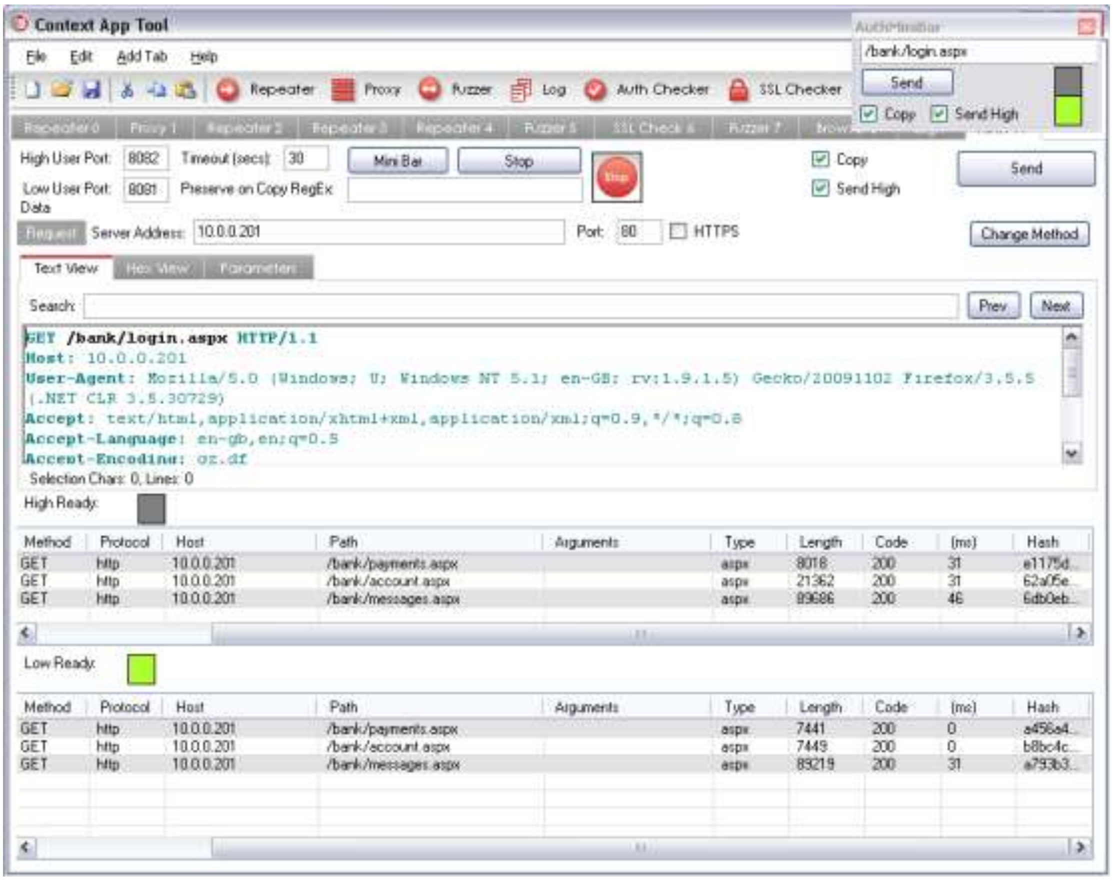
The ‘no. threads’ setting is used to control how many concurrent requests will occur at any one time. This is set to 1 by default so the requests will be performed in order, if increased to 5, 10 then the requests will be repeated simultaneously and quicker but not necessarily in order.

‘Repeat’ will request the log entries multiple times. This can be used to grab cookies or tokens that are different per-request and then use the find/extract option to remove them for further analysis.

‘Delay’ and ‘inc timer’ are used to put pauses in the sequence either fixed or by that value incrementing. E.g. if delay =60 and inc timer is on then the requests would have a delay between them of 1min then 2mins, 3mins, 4mins. This can be useful for determining the session timeout.

2.9 Auth Checker Panel

The auth checker panel is used in determining the authorisation of particular requests as different users. The auth checker has two proxies running on different port numbers. To use the tool two separate browsers (e.g. Firefox and IE) are configured to proxy through each port. The 'low user' then logs into the application with a low level of privileges than the 'high user', and the high user logs in with a high level of access. When both browsers are correctly configured and authenticated the 'copy' tick box is selected from either the main panel or the 'minibar' window. From this point the proxies are synchronised so that any action that the high user performs will be performed by the low user but with the low user's cookie. For example if an admin user clicks on 'manage users' the high user will pause and wait for the low user (the lights will change to green when a user is paused and waiting for the other). Then a low user would click on any link within the application. This will result in the low user's link replaced with the high users 'mange user' link and the low user would attempt to force browse to this area. This can be used with POST as well as GET requests to ensure that the ACLs are correctly implemented.



/bank/login.aspx

☑ Copy ☑ Send High

High User Port: 8082 Timeout (secs): 30

Low User Port: 8081 Preserve on Copy RegEx Data

☑ Copy

☑ Send High

Server Address: 10.0.0.201 Port: 80 ☐ HTTPS

Search:

```
GET /bank/login.aspx HTTP/1.1
Host: 10.0.0.201
User-Agent: Mozilla/5.0 (Windows; U; Windows NT 5.1; en-GB; rv:1.9.1.5) Gecko/20091102 Firefox/3.5.5
(.NET CLR 3.5.30729)
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: en-gb,en;q=0.5
Accept-Encoding: gz,df
```

Selection Chars: 0, Lines: 0

High Ready:

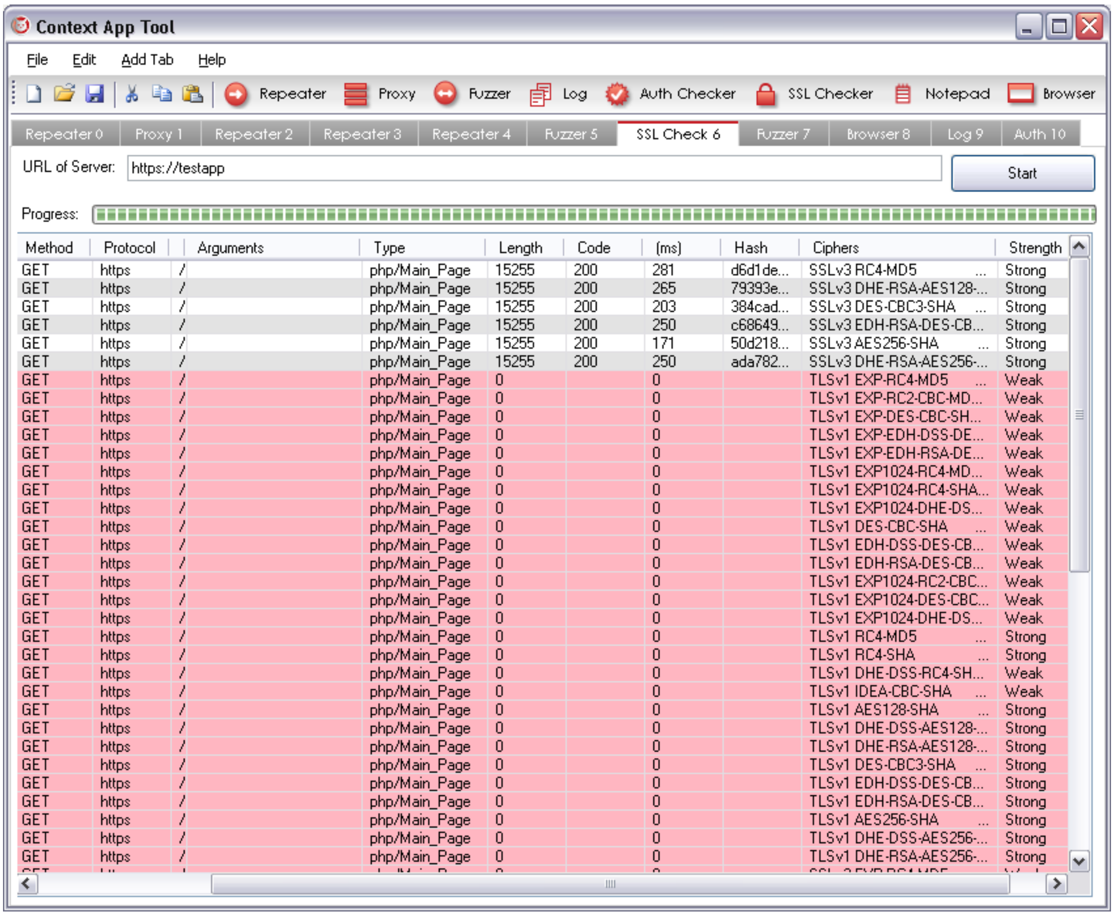
Method	Protocol	Host	Path	Arguments	Type	Length	Code	(ms)	Hash
GET	http	10.0.0.201	/bank/payments.aspx		aspx	8018	200	31	e1175d...
GET	http	10.0.0.201	/bank/account.aspx		aspx	21362	200	31	62a05e...
GET	http	10.0.0.201	/bank/messages.aspx		aspx	89686	200	46	6db0eb...

Low Ready: 🟩

Method	Protocol	Host	Path	Arguments	Type	Length	Code	(ms)	Hash
GET	http	10.0.0.201	/bank/payments.aspx		aspx	7441	200	0	a456a4...
GET	http	10.0.0.201	/bank/account.aspx		aspx	7449	200	0	b8bc4c...
GET	http	10.0.0.201	/bank/messages.aspx		aspx	89219	200	31	a793b3...

2.10 SSL Checker Panel

The SSL Checker takes a HTTPS URL and will request the URL with different SSL versions and ciphers. CAT uses OpenSSL for the implementation of SSL. If the request is successful then the resulting page is returned into the log. The page can then be opened to ensure that the page is the actual page and not a warning page that the version of SSL in use is not supported.

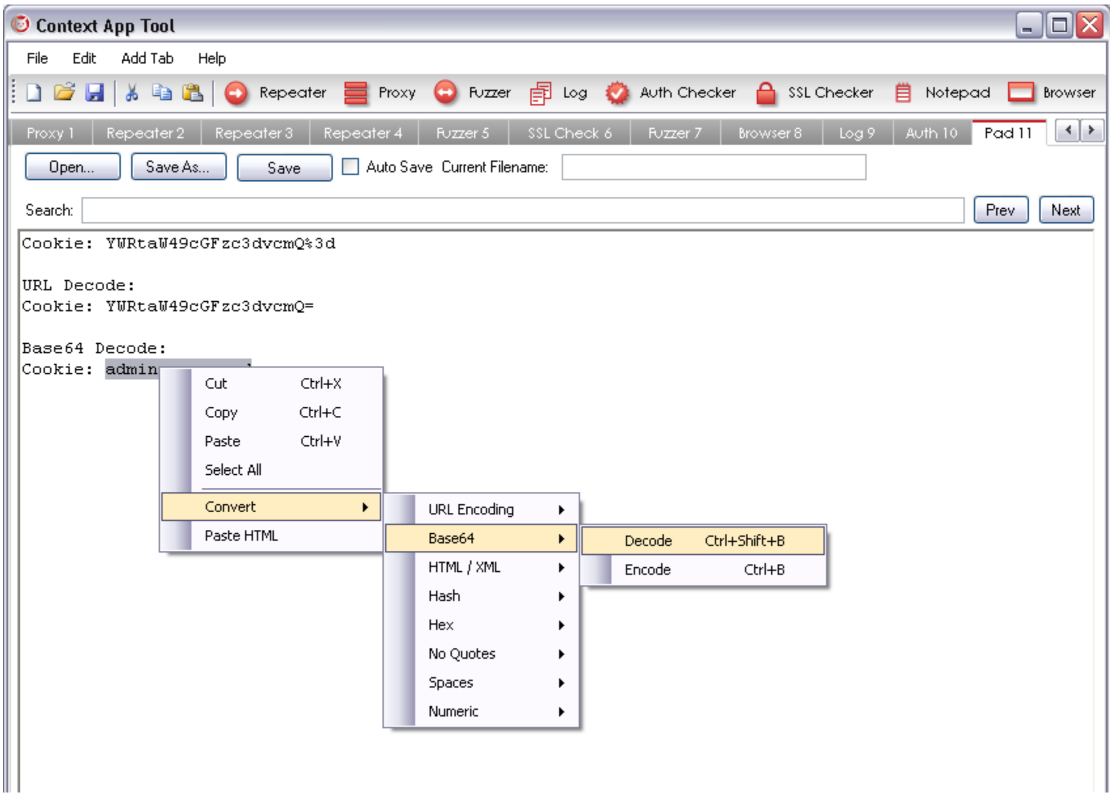


URL of Server: https://testapp

Method	Protocol		Arguments	Type	Length	Code	(ms)	Hash	Ciphers	Strength
GET	https	/		php/Main_Page	15255	200	281	d6d1de...	SSLv3 RC4-MD5 ...	Strong
GET	https	/		php/Main_Page	15255	200	265	79393e...	SSLv3 DHE-RSA-AES128-...	Strong
GET	https	/		php/Main_Page	15255	200	203	384cad...	SSLv3 DES-CBC3-SHA ...	Strong
GET	https	/		php/Main_Page	15255	200	250	c68649...	SSLv3 EDH-RSA-DES-CB...	Strong
GET	https	/		php/Main_Page	15255	200	171	50d218...	SSLv3 AES256-SHA ...	Strong
GET	https	/		php/Main_Page	15255	200	250	ada782...	SSLv3 DHE-RSA-AES256-...	Strong
GET	https	/		php/Main_Page	0		0		TLSv1 EXP-RC4-MD5 ...	Weak
GET	https	/		php/Main_Page	0		0		TLSv1 EXP-RC2-CBC-MD...	Weak
GET	https	/		php/Main_Page	0		0		TLSv1 EXP-DES-CBC-SH...	Weak
GET	https	/		php/Main_Page	0		0		TLSv1 EXP-EDH-DSS-DE...	Weak
GET	https	/		php/Main_Page	0		0		TLSv1 EXP-EDH-RSA-DE...	Weak
GET	https	/		php/Main_Page	0		0		TLSv1 EXP1024-RC4-MD...	Weak
GET	https	/		php/Main_Page	0		0		TLSv1 EXP1024-RC4-SHA...	Weak
GET	https	/		php/Main_Page	0		0		TLSv1 EXP1024-DHE-DS...	Weak
GET	https	/		php/Main_Page	0		0		TLSv1 DES-CBC-SHA ...	Weak
GET	https	/		php/Main_Page	0		0		TLSv1 EDH-DSS-DES-CB...	Weak
GET	https	/		php/Main_Page	0		0		TLSv1 EDH-RSA-DES-CB...	Weak
GET	https	/		php/Main_Page	0		0		TLSv1 EXP1024-RC2-CBC...	Weak
GET	https	/		php/Main_Page	0		0		TLSv1 EXP1024-DES-CBC...	Weak
GET	https	/		php/Main_Page	0		0		TLSv1 EXP1024-DHE-DS...	Weak
GET	https	/		php/Main_Page	0		0		TLSv1 RC4-MD5 ...	Strong
GET	https	/		php/Main_Page	0		0		TLSv1 RC4-SHA ...	Strong
GET	https	/		php/Main_Page	0		0		TLSv1 DHE-DSS-RC4-SH...	Weak
GET	https	/		php/Main_Page	0		0		TLSv1 IDEA-CBC-SHA ...	Weak
GET	https	/		php/Main_Page	0		0		TLSv1 AES128-SHA ...	Strong
GET	https	/		php/Main_Page	0		0		TLSv1 DHE-DSS-AES128-...	Strong
GET	https	/		php/Main_Page	0		0		TLSv1 DHE-RSA-AES128-...	Strong
GET	https	/		php/Main_Page	0		0		TLSv1 DES-CBC3-SHA ...	Strong
GET	https	/		php/Main_Page	0		0		TLSv1 EDH-DSS-DES-CB...	Strong
GET	https	/		php/Main_Page	0		0		TLSv1 EDH-RSA-DES-CB...	Strong
GET	https	/		php/Main_Page	0		0		TLSv1 AES256-SHA ...	Strong
GET	https	/		php/Main_Page	0		0		TLSv1 DHE-DSS-AES256-...	Strong
GET	https	/		php/Main_Page	0		0		TLSv1 DHE-RSA-AES256-...	Strong

2.11 Notepad

The notepad is a large version of the editor used to modify the HTTP requests in the repeater and other tabs. This allows for the built-in conversions to be used on a free set of text, e.g. for cookie analysis from cookies gathered.

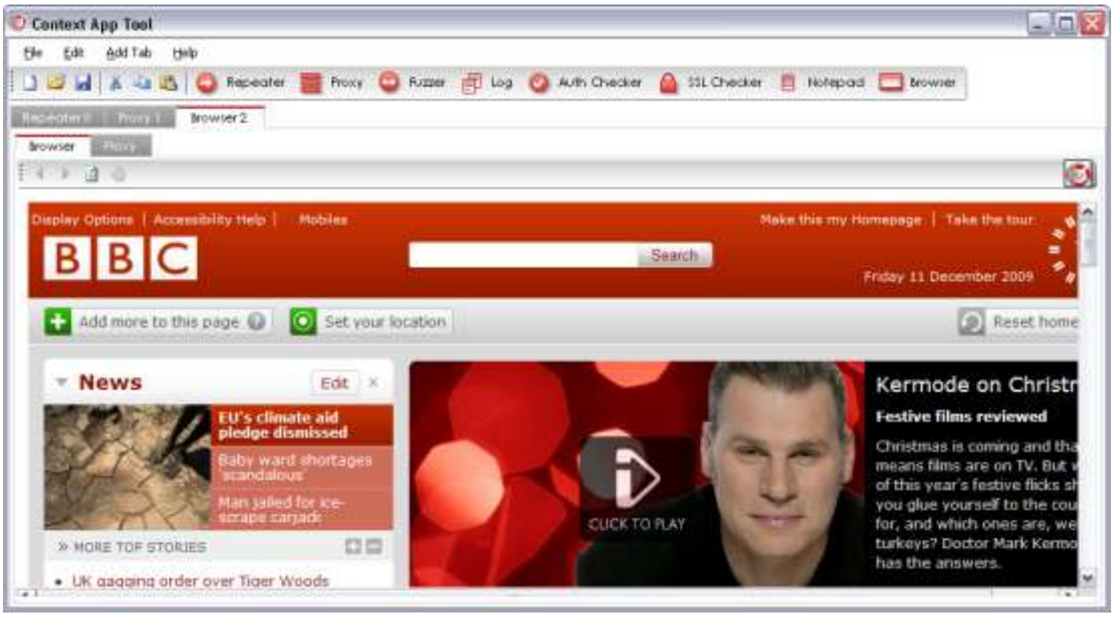


```
Cookie: YWRtaW49cGFzc3dvcmQ%3d

URL Decode:
Cookie: YWRtaW49cGFzc3dvcmQ=

Base64 Decode:
Cookie: admin
```

The conversions are accessed by selecting the text to process and pressing right button. Under the menu there is a ‘convert’ item which then lists the various ways supported to manipulate the text in ways often used with web applications.

2.12 Integrated Web Browser Panel

The integrated Web Browser uses an Internet Explorer rendering engine with the CAT proxy pre-configured for ease of accessing and testing applications without the need to setup a web browser. Furthermore, each Web Browser tab uses separate cookies so different user accounts can be logged in separately into each tab.



Christmas is coming and tha means films are on TV. But v of this year's festive flicks sl you glue yourself to the cou for, and which ones are, we turkeys? Doctor Mark Kermo has the answers.

The log tab shows a record of all the requests that the browser has sent through such that they can be intercepted or copied into other tabs e.g. repeater and fuzzer.

3 Additional Information

3.1 Limitations

CAT is still in beta and therefore is yet to be totally bug free. Here is a list of some of the currently know bugs:

1. Brute Forcer Fuzz Type does not support over a length of one.
2. The integrated web browser has certain limitations where new windows are opened. These will break out of CAT and be real IE windows. JavaScript errors can also appear.

3.2 Acknowledgements

CAT makes use of OpenSSL for certificate manipulation and SSL version checking, DirBuster's directory listings are part of the fuzzers word lists and Ionic zip utilities for CATX projects. To those authors I thank you.

3.3 Bug Reporting

The current version of CAT is Beta 3, therefore requires more testing and additional work to remedy bugs within the code. Please report any bugs or feature requests to: CAT@contextis.co.uk. Please include the version of CAT, Windows version, and any information provided by error handling like Exceptions and stack traces. Information about how to recreate the issue, including the types of web servers and any Screenshots would also be useful (where possible). Context will ensure that anyone who participates in the beta testing will be updated with new versions CAT as they become available.

3.4 Upgrades

CAT will check on start up cat.contextis.com for a newer version of the software, if one is found the user will be informed but it will not automatically upgrade. You can also keep up to date with cat by visiting the website. If a new version of the software is available then download the new installer and your current instance will be upgraded.

3.5 Change Log

The changes between Beta 3 and Beta 4 are as follows:

- Added Support for 64 bit windows.
- Change certificate storage location to allow CAT to run as a non-admin user
- Add Column for test for auto completion on forms
- Add Column for cache controls
- Add Click jacking test support
- SSL connection keep-alive to increase performance.
- CAT was fixed to use SSLV3 from the proxy this is now relaxed to include SSLV2 and TLS1.
- Can show options on first load.
- Can disable check for update call back
- SSL non-standard support
- Fixed bugs related to non-standard HTTP response headers
- Fixed a bug in fuzzer relating to concurrency
- Can save CSRF post forms to files
- Save text in notepad
- Counter on the length of text
- Update UI to better route usage.
- Response bodies can be saved to a file