

ZebraTester

Release Notes

Version 5.4-A



Table of Contents

1	V5.4.....	3
1.1	New Functions and Features.....	3
1.1.1	Major Release V5.4-A (Installation Kit dated July 17, 2015)	3
1.2	Discontinued Functions	8
1.3	Version 5.4 Compatibility.....	9

1 V5.4

The brand of the product has been renamed from ProxySniffer to ZebraTester.

1.1 New Functions and Features

1.1.1 Major Release V5.4-A (Installation Kit dated July 17, 2015)

ZebraTester V5.4-A contains the following new features:

- **Revised Content Test Menu**

The "Content Test Menu" is now an overlay instead of a pop-up window and has been revised and improved. It's now also supported to verify text fragments of a received HTTP response header.

- **Other Revised Menus**

The following menus are now overlays instead of pop-up windows:

- Cumulated Response Time per Media Type
- Recorded URL Performance Data
- Delete URLs
- Delete Pages

The filter for the hosts in the main menu is now also an overlay which contains checkboxes per host name, rather than a simple text input field as before. This new overlay can be activated only if the recorded session contains more than one host name.

- **Enhanced Var Extractor Wizard**

A new variable extractor type has been implemented to extract values from unstructured HTTP responses. This new variable extractor type is no longer text-line based and uses as algorithm a search text and a relative token number, counted from the search text (in comparison to the already existing variable extractor whose token number always starts at the beginning of a line). Both types of variable extractors are now supported by the Var Extractor Wizard.

- **Reduced Size of Log Output for Load Test Jobs**

Load test jobs are writing now only a small set of overall values about the measured data to the log, but no longer detailed error data in case when an URL call fails. This reduces the size of the job *.out file normally by more than 95%. All detailed error data are still available in the "Error Snapshots" as before. To re-enable the old functionality for the log output you can select the new debug option "debug failed loops" when starting a load test. Furthermore, it's now also possible to suppress completely to write any data to the job *.out file by using the new "Additional Option" -nostdoutlog.

- **Real-Time Control of Load Test Jobs**

The "Abort Job" button shown during load test execution has been replaced by an overlay which allows you:

- To abort the load test job (as before).
- To suspend and to resume the load test job.
- To increase or decrease the planned test duration.
- To increase and to decrease the number of simulated users.
- To modify the values of "User Input Fields" in real-time.

This is supported for normal jobs as well as for cluster jobs. Any manual actions made in this overlay are journaled in the test result (*.prxres file) and can be retraced in the "Load Test Result Detail Menu" → "Test Scenario".

Notes: a) If a load test job is suspended it will nevertheless be terminated when the planned test duration is exceeded.

b) If you decrease the number of users to zero then the load test job will stop as soon as zero users are reached.

c) Only already running users can be decreased.

d) If you trigger an increase or a decrease during an already increasing/decreasing process then a second increasing/decreasing process will run in parallel.

e) User Input Fields can only be modified in real-time if they have been created/declared by enabling the (new) checkbox "Enable Value Changes in Real-Time - During Test Execution".

- **Reduced Overhead Between Executed Loops**

The overhead of a couple of 10 Milliseconds between the executed loops of a simulated user has been reduced to almost zero.

- **Reduced File Size / Memory Size for Cluster Job Results**

When acquiring the result of a cluster job you can now select if all of the "raw" Exec Agent member data should be part of the test result (as before). If you de-select this (new) option then the size of the result file – respectively the size of the required Java memory to display the result – will be up to 90% smaller – depending on the number of cluster members. You will not lose any data – but the cluster job result cannot be expanded later into single Exec Agent job results.

- **Optimized Resources for “One Loop Only per User” Load Tests**

The Java memory and the references to terminated threads has been substantially reduced for “One Loop Only per User” load tests

- **Measuring of the SSL Handshake Time**

The time elapsed to perform SSL Handshakes is now measured and displayed in real-time during the test execution.

- **Removed Load Test Options**

The load test options "-log", "-nolog" and "-sslreset" have been removed. Passing such options take now no effect. The SSL session cache is now cleared at the start of each executed loop (as by setting "-sslreset" before).

- **New PrxJob Commands**

The following PrxJob Commands have been added:

- changeJobNumSimulatedUser <exec agent name> <job id> <delta number> [<startup delay ms>]
- setJobSuspend <exec agent name> <job id> <"false"|"true"> [<startup delay ms>]
- isJobSuspend <exec agent name> <job id>
- changeJobTestDuration <exec agent name> <job id> <delta seconds>
- getJobRealTimeUserInputFields <exec agent name> <job id>
- getClusterJobRealTimeData <cluster name> <cluster job id> [-detailed]
- changeClusterJobNumSimulatedUser <cluster name> <cluster job id> <delta number> [<startup delay ms>]
- setClusterJobSuspend <cluster name> <cluster job id> <"false"|"true"> [<startup delay ms>]
- isClusterJobSuspend <cluster name> <cluster job id>
- changeClusterJobTestDuration <cluster name> <cluster job id> <delta seconds>
- getClusterJobRealTimeUserInputFields <cluster name> <cluster job id>

• **Revised PrxJob checkUrl Command**

The PrxJob checkUrl Command supports now SNI (Server Name Indication). Furthermore, the following new options have been added:

- [-ssl <version>] (Optional. Set the SSL or TLS version. Possible values are "all", "v3", "tls", "tls11" or "tls12")
- [-dnsstatistic] (Optional. Measure also the DNS resolve time.)
- [-dnssrv <IP-name-server-1>[,<IP-name-server-N>]] (Optional. Set an alternative list of DNS servers.)

• **Two New Internal Processing States for Measured URL Calls**

The internal processing states "DNS Resolve" and "SSL/TLS Handshake" have been implemented for measured URL calls. In case if an URL call fails (for example when a timeout occurs), you will see now if such an error did potentially happen during query of a DNS server or during a SSL/TLS handshake.

Complete list of all internal processing states:

Internal Processing State of URL Call	Value	Meaning
No Step / Not Initialized	-1	The URL call had not yet started.
DNS Resolve (new)	10	The URL call failed during resolving the DNS name of the web server.
Open Network Connection to Proxy	0	The URL call failed during the opening of a network connection to an outbound proxy server. This processing state is executed only if the load test is performed via a proxy server (unusual, not recommended test scenario).
Open Network Connection	1	The URL call failed during the opening of a TCP network connection to the web server.
SSL/TLS Handshake (new)	11	The URL call failed during executing the SSL/TLS handshake with the web server.
Transmit HTTP Request	2	The URL call failed during the transmission of the HTTP request data.
Wait for Server Response	3	The URL call failed while waiting for the first byte of the HTTP response data from the web server.

Receive HTTP Header	4	The URL call failed while receiving the HTTP response header from the web server.
Receive Content	5	The URL call failed while receiving the HTTP response content from the web server (HTML data, images, ...).
Close Network Connection	6	The URL call failed while closing the network connection to the web server.
All Done	7	The URL call itself completed successfully (all data transmitted and received), but the received HTTP status code was incorrect, or the received MIME type (text/html, image/gif, ...) was incorrect, or an error was detected inside the received content data.

- **DNS Resolve Time and SSL Handshake Time Shown in URL Performance Data**

The DNS Resolve Time and the SSL Handshake Time are now shown in all URL performance data.

- **Support to Terminate a Simulated User due to a Failed URL Call**

The “Failure Action” in the “HTTP Response Verification” menu contains a new option which allows to terminate a simulated user when a “red error” occurred for the (particular) configured URL. All simulated users for which this error occurs are removed from further load test execution. The users are also terminated if a variable extraction for that URL fails or when a load test plug-in which is bound to that URL releases a self-generated error. The corresponding captured error snapshots are marked with a yellow colored warning sign in order that you can retrace why a user has been terminated.

- **Accessible X509 Server Certificate Chain**

The X509 certificate chain received from the web server during the SSL/TLS handshake is now accessible from load test plug-ins. This allows also to extract the public key of a Web server certificate during the load test execution.

- **New Functions to the Proxy Recorder REST API**

The following functions have been added to the Proxy Recorder REST API:

- getRecorderURLBlacklist
- setRecorderURLBlacklist
- getRecorderURLWhitelist
- setRecorderURLWhitelist
- getRecorderOutboundProxyTimeout (Apica BNet only)
- setRecorderOutboundProxyTimeout (Apica BNet only)
- getRecorderRecordFailedTransmitRequests (Apica BNet only)
- setRecorderRecordFailedTransmitRequests (Apica BNet only)
- getRecorderSSLVersion
- setRecorderSSLVersion

The updated Application Reference Manual contains further information about these new REST API functions.

- **Additional Java Methods for Recorded Sessions and Recorder Plug-Ins**

- Class `dfischer.proxysniffer.ProxyDataRecord`
 - New method `getRecorderSequenceID()`
 - New method `getExecutionStep()`
 - New method `getFailureExceptionText()`

- **Additional Java Methods for Recorder Plug-Ins**

- Class `dfischer.proxysniffer.ProxyRecorderContext`
 - `getRecorderSequenceID()`
- Class `dfischer.proxysniffer.ProxyRecorderPluginURLContext`
 - `getRecorderSequenceID()`
 - `getExecutionStep()`
 - `getFailureException()`

- **Revised Installation Kits**

The installation kits for Windows, Mac OS X and Linux have been revised. The Mac OS X installation kit contains now an integrated Java compiler and interpreter.

1.2 Discontinued Functions

- The support for "SSL V2" has been removed
- Kerberos authentication has been removed

1.3 Version 5.4 Compatibility

Previous Version Data Files / Version 4.6-Z – 5.2-*	Compatibility with Version 5.4
Recorded Web Surfing Sessions (*.prxdat files)	Compatible
Created Load Test Programs (*.java and *.class files)	Compatible
Load Test Results (*.prxres files)	Compatible

Previous Version Data Files / Version 4.0 – Version 4.6-Y	Compatibility with Version 5.4	
Recorded Web Surfing Sessions (*.prxdat files)	Compatible	¹ Workaround: you can load any old web surfing session (*.prxdat file) into V5.4 by using the Project Navigator. After that you have to generate and compile the load test program again before you can run it on V5.4.
Created Load Test Programs (*.java and *.class files)	NOT Compatible ¹	
Load Test Results (*.prxres files)	Compatible	