

Shunra's Deep Analysis analyzes the data from the Modeler's Packet Capture, and stores the analysis in the Reporter Database.

The Deep Analysis Export utility is a Windows™-based command-line tool that exports Deep Analysis results in Comma-Separated Value (CSV) format. This makes the data immediately accessible to spreadsheet applications (e.g., Microsoft Excel), scripts (e.g., PERL) and report generation (e.g., HTML, Microsoft Word).

The Deep Analysis Export utility:

- ▶ Converts the analysis into Java Script Object Notation (JSON) and
- ▶ Summarizes that information in a pair of CSV files (two files per test run)

Two files are created:

- ▶ **HTTP analysis:** results of the Deep Analysis; contain the detailed transaction data shown in the Deep Analysis Waterfall diagram data; the CSV columns include one row per resource request. File name ends in "http.csv"
- ▶ **Optimization:** contains the remediation analysis; includes a "scenario" column that indicates whether the recommendation applies to Desktop, Mobile or both. Name ends with "best practices.csv"

To install the Deep Analysis Export utility, unzip the contents of DeepAnalysisExport.zip file into a convenient folder.

Note: If you receive an error, it may be an indication that Visual C runtime DLL is not installed. For example, the system may display an error such as "The application has failed to start because its side-by-side configuration is incorrect" on a 64-bit operating system.

For more information on this issue, please see

<http://www.microsoft.com/downloads/details.aspx?FamilyID=9b2da534-3e03-4391-8a4d-074b9f2bc1bf&displaylang=en>

To verify that the Deep Analysis Export utility installed correctly, issue the following command from the directory where ExportDeepAnalysis.exe resides:

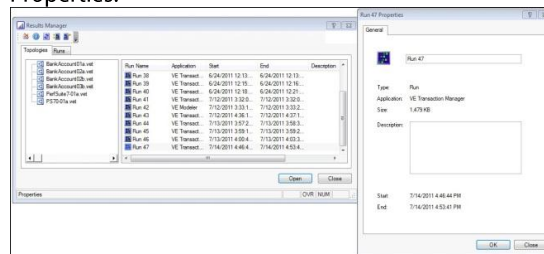
ExportDeepAnalysis.exe -h

Verify that the command-line Help instructions appear.

The utility will search for a test result whose name matches that in the command-line "-t" parameter. By default, Deep Analysis names each successive run by the name "Run ##" where ## is the next sequential number.

-t "Run 72"

Since the default test name contains a space, the command-line utility must reference it with quotes. Alternately, rename the test within the Results Manager to create a test name that has no spaces, thus obviating the quotes. Run names can be changed in Results Manager by right-clicking the Run Name and selecting Properties:



For PerformanceSuite users, it is mandatory to match the SQL database username and password (by default, "root" and "root").

To specify a different username, e.g., "root", with **-u** and a different password, e.g., "123" with **-p**, enter:

ExportDeepAnalysis.exe -t "Run 48" -u root -p 123

Tips for using the CSV file with Wireshark

1. The StartTime column can be used to filter packets in Wireshark, using a filter such as **frame.time_epoch > xxx && frame.time_epoch < yyy**

2. The TcpSession column can be used to filter packets in Wireshark, using a filter such as **tcp.stream eq xx**

Command Options

Using the **-h** parameter (help) will display the command line options that are available for **ExportDeepAnalysis**.

Usage:

ExportDeepAnalysis.exe [-h] [-v] [-s] [-t TESTNAME [-w WORKFOLDER] [-u USERID] [-p PASSWORD] [--version]

Run	Start	End	Type	TestName	Status	Method	BaseURL	Query	RequestSize	ResponseSize	ReqContentSize	ReqContent
1	1:12:12-09	Run 48:0025	Connect	HTTP	2024	200 GET	192.16.1.4/3	/index.htm	360	321	0	
2	1:12:12-09	Run 48:0025	Connect	HTTP	2024	200 GET	192.16.1.4/3	/index.htm	370	320	0	
3	1:12:12-09	Run 48:0025	Connect	HTTP	884	200 GET	192.16.1.4/3	/images/mainmenu.gif	390	320	0	
4	1:12:12-09	Run 48:0025	Connect	HTTP	1708	200 GET	192.16.1.4/3	/images/mainmenu_image.jpg	390	321	0	
5	1:12:12-09	Run 48:0025	Log on	HTTP	2091	200 GET	192.16.1.4/3	/bank_login.htm	572	321	0	
6	1:12:12-09	Run 48:0025	Log on	HTTP	2092	200 GET	192.16.1.4/3	/images/bank_image.gif	366	320	0	
7	1:12:12-09	Run 48:0025	Log on	HTTP	1242	200 GET	192.16.1.4/3	/images/bank_image.gif	366	321	0	
8	1:12:12-09	Run 48:0025	Log on	HTTP	1248	200 GET	192.16.1.4/3	/images/bank_image.gif	366	321	0	
9	1:12:12-09	Run 48:0025	Log on	HTTP	2885	200 GET	192.16.1.4/3	/images/bank_image.gif	400	320	0	
10	1:12:12-09	Run 48:0025	Log on	HTTP	1118	200 GET	192.16.1.4/3	/images/bank_image.jpg	391	321	0	
11	1:12:12-09	Run 48:0025	Log on	HTTP	2041	200 GET	192.16.1.4/3	/index.htm	365	320	0	
12	1:12:12-09	Run 48:0025	Log on	HTTP	2936	200 GET	192.16.1.4/3	/index.htm	389	321	0	
13	1:12:12-09	Run 48:0025	Log on	HTTP	863	200 GET	192.16.1.4/3	/images/bank_image.gif	391	321	0	
14	1:12:12-09	Run 48:0025	Log on	HTTP	812	200 GET	192.16.1.4/3	/images/bank_image.gif	401	321	0	
15	1:12:12-09	Run 48:0025	Log on	HTTP	1919	200 GET	192.16.1.4/3	/images/bank_image.jpg	366	321	0	
16	1:12:12-09	Run 48:0025	Log on	HTTP	4279	0 **UNDEF**	192.16.1.4/3	/bank_login.htm	528	321	0	
17	1:12:12-09	Run 48:0025	Log on	HTTP	2168	200 GET	192.16.1.4/3	/index.htm	379	320	0	
18	1:12:12-09	Run 48:0025	Log on	HTTP	2297	200 GET	192.16.1.4/3	/index.htm	379	324	0	
19	1:12:12-09	Run 48:0025	Log on	HTTP	1148	200 GET	192.16.1.4/3	/images/bank_image.gif	391	321	0	
20	1:12:12-09	Run 48:0025	Log on	HTTP	1068	200 GET	192.16.1.4/3	/images/bank_image.gif	385	322	0	
21	1:12:12-09	Run 48:0025	Log on	HTTP	0 **UNDEF**	0 **UNDEF**	192.16.1.4/3	/images/bank_image.gif	385	322	0	
22	1:12:12-09	Run 48:0025	Log on	HTTP	1248	0 **UNDEF**	192.16.1.4/3	/images/bank_image.gif	385	322	0	
23	1:12:12-09	Run 48:0025	Log on	HTTP	0 **UNDEF**	0 **UNDEF**	192.16.1.4/3	/images/bank_image.gif	385	322	0	

M	I	O	A	Start	P	Q	R	S	T	U	V	W	X	Y
ReqContent	ReqContentSize	ReqContentType	ReqContentEncoding	TcpSession	TotalTime	Dir	TcpStart	TLSHandshake	Request	Wait	Response			
0	2648	text/html	0	2625	0	1346	0	1	1247	23				
1	309	text/css	1	2055	0	1079	0	1	984	1				
2	867	image/gif	0	866	0	0	0	1	868	1				
3	4099	image/jpeg	1	1709	0	0	0	1	1824	784				
4	6708	text/html	0	2296	0	532	0	1	723	1038				
5	809	text/css	1	2055	0	872	0	1	1130	1				
6	1027	image/gif	0	1246	0	0	0	1	1242	1				
7	180	image/gif	1	1240	0	0	0	1	1244	1				
8	217	image/gif	2	2387	0	1242	0	1	1143	1				
9	5305	image/jpeg	0	1229	0	0	0	1	1011	117				
10	2648	text/html	3	3212	0	1047	0	1	1086	1078				
11	390	text/css	4	2049	0	1037	0	1	1010	1				
12	3429	application/javascript	5	2927	0	1038	0	1	1026	872				
13	1027	image/gif	3	865	0	0	0	1	863	1				
14	217	image/gif	3	854	0	0	0	1	812	1				
15	5305	image/jpeg	4	1338	0	0	0	1	715	622				
16	0		4	1	0	0	0	1	0	0				
17	8050	text/html	6	4280	0	910	0	1	1070	2299				
18	309	text/css	7	2158	0	1060	0	1	1106	1				
19	441	application/javascript	8	2299	0	1183	0	1	1114	1				
20	1027	image/gif	8	1150	0	0	0	1	1148	1				
21	5154	image/gif	6	1249	0	0	0	1	1156	112				
22	0		6	1	0	0	0	1	0	0				
23	0		8	1270	0	1269	0	1	0	0				

The following folder structure is created, with additional subfolders such as CSV, JSON, etc. In this example, the Run Name is "Run 48".

C:\DeepAnalysis\workFolder\Run 48 – contains the data for a specific test

For more information about Shunra Analytics and the Deep Analysis Report, refer to the Analytics product manuals, and www.shunra.com