

Software Version: 9.00 For Windows

User Guide

Document Release Date: February 2015 Software Release Date: February 2015

Legal Notices

Warranty

The only warranties for HP products and services are set forth in the express warranty statements accompanying such products and services. Nothing herein should be construed as constituting an additional warranty. HP shall not be liable for technical or editorial errors or omissions contained herein. The information contained herein is subject to change without notice.

Restricted Rights Legend

Confidential computer software. Valid license from HP required for possession, use or copying. Consistent with FAR 12.211 and 12.212, Commercial Computer Software, Computer Software Documentation, and Technical Data for Commercial Items are licensed to the U.S. Government under vendor's standard commercial license.

Copyright Notice

© Copyright 2015 Hewlett-Packard Development Company, L.P.

Trademark Notices

Adobe® is a trademark of Adobe Systems Incorporated. Microsoft® and Windows® are U.S. registered trademarks of Microsoft Corporation. Linux® is the registered trademark of Linus Torvalds in the U.S. and other countries. Java® is a registered trademark of Oracle and/or its affiliates. Red Hat® is a registered trademark of Red Hat, Inc. in the United States and other countries. VMware® is a registered trademark of VMware, Inc. in the United States and other countries.

Documentation Updates

The title page of this document contains the following identifying information:

- Software Version number, which indicates the software version.
- Document Release Date, which changes each time the document is updated.
- Software Release Date, which indicates the release date of this version of the software.

To check for recent updates or to verify that you are using the most recent edition of a document, go to: http://h20230.www2.hp.com/selfsolve/manuals

This site requires that you register for an HP Passport and sign in. To register for an HP Passport ID, go to: http://h20229.www2.hp.com/passport-registration.html

Or click the New users - please register link on the HP Passport login page.

You will also receive updated or new editions if you subscribe to the appropriate product support service. Contact your HP sales representative for details.

Support

Visit the HP Software Support Online web site at: http://www.hp.com/go/hpsoftwaresupport

This web site provides contact information and details about the products, services, and support that HP Software offers.

HP Software online support provides customer self-solve capabilities. It provides a fast and efficient way to access interactive technical support tools needed to manage your business. As a valued support customer, you can benefit by using the support web site to:

- · Search for knowledge documents of interest
- · Submit and track support cases and enhancement requests
- Download software patches
- Manage support contracts
- Look up HP support contacts
- Review information about available services
- Enter into discussions with other software customers
- Research and register for software training

Most of the support areas require that you register as an HP Passport user and sign in. Many also require a support contract. To register for an HP Passport ID, go to:

http://h20229.www2.hp.com/passport-registration.html

To find more information about access levels, go to:

http://h20230.www2.hp.com/new_access_levels.jsp

HP Software Solutions Now accesses the HPSW Solution and Integration Portal Web site. This site enables you to explore HP Product Solutions to meet your business needs, includes a full list of Integrations between HP Products, as well as a listing of ITIL Processes. The URL for this Web site is http://h20230.www2.hp.com/sc/solutions/index.jsp

About this PDF Version of Online Help

This document is a PDF version of the online help. This PDF file is provided so you can easily print multiple topics from the help information or read the online help in PDF format. Because this content was originally created to be viewed as online help in a web browser, some topics may not be formatted properly. Some interactive topics may not be present in this PDF version. Those topics can be successfully printed from within the online help.

Contents

Chapter 1: NV Analytics	7
System Requirements	
Installing NV Analytics	
Prerequisites	
Installing Software	
Silent Installation	
Silent Uninstallation	10
Network Virtualization for Mobile Licensing	11
Licensing Overview	11
Accessing the NV License Manager	12
Using Fixed Licenses	12
Using Floating Licenses	
Viewing Current License Usage	15
Log Files	
Chapter 2 ⁻ Analyzing Results	16
Settings	
Analyzing a Network Virtualization for Mobile Test Result File	
Exporting NV Analytics Results	
Viewing Reports	
Response Time	
Summaries	
Client Network Server Breakdown	23
Parameters	
General Analysis	24
Display Options	
Subtransaction Parameters	
Request/Response Details	
Throughput	
Endpoint Latencies	
TCP/UDP Errors & Sessions	
HTTP Analysis	
Subtransaction Details	
HTTP Parameters	

HTTP Optimization	34
HTTP Resources and Responses	35
Resources Breakdown	
HTTP Errors	
HLS Errors	
Response Summary	
Secure Communication	
Chapter 3: NV Analytics API	40
AnalysisEngines	40
Code sample	41
Extract Packet Lists	41
Code sample	42
AnalysisRequest	43
AnalysisSummary	
AnalysisArtifact	
Structure of an Analysis Report	49
Structure of the HTTP Waterfall Analysis Report	49
Structure of the Best Practices Analysis Report	
Chapter 4: NV Analytics Protocols	
Supported Protocols	54
Conversation Definition	54
Collecting Conversation Statistics	
Classification of TCP, UDP, IP	55
Sub-Transaction Grouping	
Understanding Protocol Association	55

User Guide

Chapter 1: NV Analytics

NV Analytics assists in pinpointing factors that negatively impact an application's performance across a network. NV Analytics performs an analysis based on packet list data, then displays the resulting data in informative reports that provide insight into an application's operation.

Analysis of HTTP, HTTPS and other protocols in waterfall diagrams provide a visual look into individual resource sizes and load times, enabling rapid analysis of transaction response times and the ability to quickly identify areas for optimization. In addition, the NV Analytics provides a method to automate the process of transaction analysis and grading.

This section describes:

- System Requirements: Reviews the minimum host requirements for NV Analytics. See "System Requirements" below for more information.
- Installing Software: Provides step-by-step instructions on how to install NV Analytics. See "Installing NV Analytics" on the next page for more information.
- Licensing NV Analytics: Provides instructions on how to license and activate NV Analytics. See "Network Virtualization for Mobile Licensing" on page 11 for more information.
- Log Files: Gives the location of the NV Analytics log files. See "Log Files" on page 15 for more information.

System Requirements

The minimum requirements for NV Analytics are as follows:

Processor	Dual core 2.5 GHz or stronger				
Memory	4 GB RAM				
Hard Disk	20 GB of free disk space				
Desktop Operating System (English Version only)	• Windows 7 SP1 (32/64 bit)				
	• Windows Server 2008 SP2 (32/ 64 bit)				
	• Windows Server 2008 R2 SP1 (64 bit)				
	• Windows 8 (32/64 bit)				
	• Windows 8.1 (32/64 bit)				

Browsers	Firefox
	Chrome
	Internet Explorer 9.0 and higher
	Safari 7 and higher
Microsoft Office (for exporting reports)	• Office 2007
	Office 2010
	Office 2013

Installing NV Analytics

Prerequisites

Wireshark version 1.10.8, 1.10.9, 1.10.10, or 1.10.11 must be installed prior to installation of NV Analytics.

The following will be installed during installation if not present:

- Java Runtime Environment 7.0 (32 bit) update 25 (will be installed if Java 6.0 update 24 or higher (32 bit) is not present)
- Microsoft .NET Framework 4.0 Full
- Microsoft Silverlight 5.0

Installing Software

To install NV Analytics, run the NVSetupWizard.exe setup file (as an administrator), select **NV Analytics**, click **Install**, and follow the on-line directions.

Upgrading NV Analytics: When you run the installation, the installer checks if there is a previous version of NV Analytics installed. Depending on the version you have installed, the installer does one of the following:

- Uninstalls the previous version and then continues the installation.
- Prompts you to first uninstall the previous version and then run the installation again.

After installing NV Analytics, you must reboot the computer.

After installation, access NV Analytics by clicking **Start > All Programs > HP Software > NV Analytics**, select **NV Analytics**).

Silent Installation

Note: During silent installation, if Wireshark is not downloaded and installed successfully, installation will abort.

To silently install NV Analytics:

- 1. Copy the file 'NVAnalyticsSetup.exe' setup file from the installation package to a convenient location.
- 2. Click Start > All Programs > Accessories, right-click Command Prompt, and then click Run as administrator (to run the CMD command with elevated privileges).
- 3. In the Command window, navigate to the location of the file copied in step 1, and enter the following command together with the required command line options:

NVAnalyticsSetup.exe /s /v"/qn command_line_options"

Command line options [* indicates a mandatory command line option]:

PORT=<port number>

The port used to connect to NV Analytics.

If another Network Virtualization component is already installed, NV Analytics will use the same port.

ENABLE_REMOTE=TRUE | FALSE

Opens the port in the firewall.

Default is TRUE.

DATA_FOLDER=\"<path to data folder>\"

The location where internal application data and user data is saved.

Default is <Common App Data folder>\HP\NV (C:\ProgramData\HP\NV in Win 7).

INSTALLDIR=\"<path to installation folder>\"

The location where the application files will be installed.

Default is C:\Program Files (x86)\HP\NV\

REBOOT_IF_NEED=TRUE | FALSE

If a reboot is needed, automatically reboots the computer after installation completes.

Default is TRUE.

For standalone authentication without the NV Server:

* ADMIN_NAME=administrator username

The username for NV Analytics admin user.

• * ADMIN_PASS=administrator password

The password for the admin user.

Silent Uninstallation

To silently uninstall NV Analytics:

- 1. Copy the file NVAnalyticsSetup.exe setup file from the installation package to a convenient location.
- 2. Click Start > All Programs > Accessories, right-click Command Prompt, and then click Run as administrator (to run the CMD command with elevated privileges).
- 3. In the Command Prompt window, navigate to the location of the file copied in step 1, and enter the following command together with the required command line options:

NVAnalyticsSetup.exe /s /removeonly /v"/qn command_line_options"

Command line options [All the command line options are optional]:

REBOOT_IF_NEED=TRUE | FALSE

If a reboot is needed, automatically reboots the computer after uninstall completes.

Default is TRUE.

■ FORCE_REBOOT=TRUE | FALSE

Automatically reboots the computer after uninstall completes, whether or not a reboot is needed.

Default is FALSE.

DELETE_DATA=TRUE | FALSE

Deletes all stored NV Analytics data.

Default is FALSE.

Network Virtualization for Mobile Licensing

Network Virtualization for Mobile licenses are managed from the NV License Manager. The NV License Manager is installed automatically on each computer that has NV Server, NV Test Manager, NV Analytics, or NV Location Editor.

This topic includes:

Licensing Overview	11
Accessing the NV License Manager	
Using Fixed Licenses	
Using Floating Licenses	
Viewing Current License Usage	

Licensing Overview

The following licensing methods are available for the Network Virtualization for Mobile products:

- Fixed licenses
- Floating licenses
- Trial licenses

Fixed licenses

When using fixed licenses, HP sends you a license key for each computer that has a Network Virtualization for Mobile product installed (NV Test Manager, NV Agent, NV Analytics, or NV Global Library). The license is created for a specific Network Virtualization for Mobile product on a specific computer, and cannot be transferred to any other computer.

Fixed licenses are useful when you have not installed NV Server and you are therefore using a standalone installation of NV Test Manager or NV Analytics.

Floating licenses

You can use floating licenses only if NV Server is installed. When using floating licenses, the NV Server acts as a licensing server. Each NV Test Manager or NV Analytics that you install can check out licenses from the license server. When you are finished using the licenses, you return the licenses to the license server so that they can be used by other NV Test Manager or NV Analytics installations.

The advantage of floating licenses is that you can install Network Virtualization for Mobile on as many computers as you like, but you need licenses for only those computers that are actively using Network Virtualization for Mobile components.

Note: When using floating licenses, the license server still requires a fixed license.

Trial licenses

Each Network Virtualization for Mobile product is installed with a two-day trial license. The trial license gives access to all product functionality. The trial period begins the first time you log in to the product.

Caution: If you install any Network Virtualization for Mobile product on a virtual machine, do not clone the machine after the trial license has started.

Accessing the NV License Manager

You can access the NV License Manager in the following ways:

- From the Windows Start menu, select All Programs > HP Software > NV for Mobile > NV License Manager.
- From a web browser, navigate to the following URL:

http://<hostname>:<port>/shunra/license/

For example:

http://198.51.100.24:8182/shunra/license/

Note:

- If secured communication was selected when the NV Mobile components we installed, the URL begins with https://.
- If you used the default port during installation, the port number is 8182. To change the port, see "Changing the NV Test Manager Port" in the NV for Mobile User Guide.
- From within the NV Server or NV Test Manager, click the License link.

Using Fixed Licenses

When using fixed licenses, you must apply a license key to every computer that has one of the Network Virtualization for Mobile components installed on it.

Note: When using floating licenses, you must still apply a fixed license to the license server (see "Using Floating Licenses" below for more information).

To apply a fixed license key:

- 1. Open the NV License Manager on the desired computer.
- 2. Click the appropriate **Update license** button.
- 3. Click **Update via: > File**.
- 4. Click **Download the Product Key** and save the .c2v file in a convenient location.
- Click Contact HP to obtain a new license to connect to the HP Licensing site, and do one of the following:
 - If you have a valid license Entitlement Order Number (EON), enter your EON to activate your license.
 - To obtain a new license, click Contact HP Licensing to locate a Regional Licensing Support Center.

Your license activation request will be routed to the HP licensing team for processing. The licensing team will contact you to request the .c2v file, and send you a .v2c license key file.

Note: The .v2c license key you receive from HP can be used only on the computer on which you generated the .c2v file.

- 6. When you receive the .v2c license file, click the folder icon that appears to the right of the **License File** box, locate and upload the .v2c file.
- 7. Click **Update**. The updated license details are displayed in the NV License Manager main page.

Using Floating Licenses

When using floating licenses, licenses are held by the license server and are checked out as necessary by your NV Test Manager and NV Analytics machines. A license is checked out for a specified number of days, at the end of which it is automatically returned to the license server. A license can be returned early, if desired.

What do you want to do?

- "Set up your license server" on the next page
- "Check out a license" on the next page

- "Return a license" below
- "Change the maximum number of days a license can be checked out" on the next page

Set up your license server

The license server is installed automatically along with the NV Server. To begin using the license server, you must upload a bundle license key that includes your floating licenses. This bundle license key is a fixed license, and the instructions for uploading it are detailed under "Using Fixed Licenses" on page 12. Note that this bundle license includes a fixed license for the NV Server.

Note: You apply the bundle license using the NV License Manager of the machine where the NV Server is installed.

Check out a license

To check out a license:

Note: The machine that checks out a license must have access to the license server over TCP port 1947.

- 1. Open the NV License Manager on the machine that needs to check out a license. For details, see "Accessing the NV License Manager" on page 12.
- 2. Click the appropriate **Update License** button.
- 3. Choose Update via: License server.
- 4. In the **Local server address** field, select the machine where the license server is installed. If the license server does not appear in the list, enter its IP address.
- 5. In the **License duration (days)** field, choose for how long to check out the license. By default, the maximum number of days a license can be checked out is 14.
- 6. Click **Checkout license**. The license is checked out from the license server.

Return a license

To return a checked-out license:

- 1. Open the NV License Manager on the machine that needs to return a license.
- 2. Click the **Update License** button.
- 3. Choose Update via: Local licensing server.
- 4. Click **Return license**. The license is returned to the license server.

Change the maximum number of days a license can be checked out

You can change the maximum number of days that a license can be checked out. To do so:

- 1. Log into the computer that hosts the license server (NV Server).
- 2. Open the Sentinel Admin Control Center at the following URL:

http://localhost:1947

- 3. From the **Options** menu, click **Configuration**.
- 4. Click the **Detachable Licenses** tab.
- 5. In the **Max. Detach Duration** field, enter the maximum number of days that a license can be checked out.
- 6. Click the **Submit** button.

Viewing Current License Usage

If you are using floating licenses, the NV License Manager shows you the total number of available and in-use licenses, as well as a list of computers that have checked out licenses.

For a report of floating licenses for all product types, view this report on the NV Server.

Log Files

HP Network Virtualization for Mobile products' log files are located in the <installation directory>\logs, by default in \Program Files\HP\NV\logs or \Program Files (x86)\HP\NV\logs.

Chapter 2: Analyzing Results

This section describes how to view and analyze the results generated by NV Analytics.

NV Analytics reports consist of data captured using a network virtualization packet list which is subsequently processed and displayed in intuitive reports. These reports are then examined to decipher problems that may exist.

The Analytics reports provide detailed data about the breakdown of each transaction. Statistics of each resource being uploaded or downloaded in a transaction are displayed in both tabular and graphic format. Precise performance data includes load times, component download analysis, response time breakdown and details of errors received. Performance optimization recommendations how to improve and optimize mobile and non-mobile transaction performance are provided.

This section includes:

Settings	.16
Analyzing a Network Virtualization for Mobile Test Result File	. 18
Exporting NV Analytics Results	.19
Viewing Reports	. 20
Secure Communication	. 37

Settings

To restrict the analysis to a specific Port prior to analysis, click the Settings icon (wrench) on the Welcome page. The settings are also available from the main page.

HTTP Port Settings

To modify the default ports, select the HTTP ports tab and add one or more ports or range, each separated by a comma. HTTP/S analysis will run on the specified ports.

Analytics Settings						
HTTP ports	HTTP ports					
HTTPS keys	Enter ports or port ranges separated by commas. For example: 80,5000, 10345 – 10408					
General settings	80, 8080					
	Restore defaults					
1						
	Save Cancel					

HTTPS Keys

To enable analysis of secure data, enter the HTTPS key.

	Analytics Settings
HTTP ports HTTPS keys	HTTPS keys + ≫ ×
General settings	Server IP Port Key file path
	Save Cancel

To add a HTTPS Key:

- 1. Click the "+" sign, and in the New HTTPS Key window, enter the required information.
- 2. To edit the information, click the pencil icon; to delete the Key click the "x".

The IP, port and private key of the application server, or of the debugging proxy if installed, should

be provided.

	New HTTPS	6 key	
Server IP			
Port			
Key file path			8
	Sa	ave Cancel	

General Settings

Displays the NV Analytics server port.

Analytics Settings						
HTTP ports HTTPS keys General settings	General settings Analysis server port: 8182 Analysis server protocol: http					
	Save Cancel					

Analyzing a Network Virtualization for Mobile Test Result File

- 1. From the Start Menu, choose **Programs > HP > NV Analytics > NV Analytics**.
- 2. Click Open File (to specify the port and define other settings, see "Settings" on page 16). The following file types are supported: *.shunra, *.ved, *.cap, *.pcap and *.enc.

3. When the test result file contains more than one packet list, select some or all of the packet lists to be included in the analysis.

Packet Lists
The file contains multiple packet lists.
Analyze all packet Lists
✓ Packet List 4 (IP:192.168.1.85)
✓ Packet List 3 (IP:192.168.1.43)
✓ Packet List 2 (IP:192.168.1.130)
✓ Packet List 1 (IP:192.168.1.132)
Continue

4. After the NV Analytics window opens, in the toolbar click All Transactions to select a display of all the transactions, or select a specific transaction. The main page displays all the transactions, and each 'square' shows one transaction.

Exporting NV Analytics Results

Data of the selected view and recommendations can be exported to a .csv file and in MS Word format.

Exporting in .csv format

The export to .csv includes the throughput, the network conditions displayed in the waterfall graph, and recommendations such as the rules, grade achieved and the violations for each rule.

To export results in .csv format:

1. Click the ¹ icon in the toolbar.

Note: When using a zoom view, the export will use the selected portion of the transaction, and not export the entire transaction.

2. In the Export settings dialog, select all or some of the reports, and any or all of the options, then click Export.

Exporting in MS Word format

When exporting in MS Word format, the HTTP Report and Recommendations can be exported.

Note: When using a zoom view, the export will use the selected portion of the transaction, and not export the entire transaction. The NV Analytics template file must be installed in its original location, usually in the Documents folder.

To export results in MS Word format:

1. Click the will icon in the toolbar.

Note: When using a zoom view, the export will use the selected portion of the transaction, and not export the entire transaction. Results of searches that are highlighted are also highlighted in the exported reports.

Viewing Reports

The Overview page shows you transactions contained in the test result file you analyzed. Click on any transaction to view details of that transaction. Each Transaction section provides details of the throughput and errors, and shows a performance score with a letter and a percentage. The display is interactive, so that clicking any metric, such as "Total" displays the detailed report for that metric.



The NV Analytics Overview page displays detailed breakdown of each transaction in the following categories:

- "Response Time" below
- "Summaries" on the next page
- "General Analysis" on page 24
- "Endpoint Latencies" on page 27
- "TCP/UDP Errors & Sessions" on page 28
- "HTTP Analysis" on page 29
- "HTTP Optimization" on page 34
- "HTTP Resources and Responses" on page 35

Below the list of reports (on the left side), the total TRT for the transaction is listed. If a description was added when creating the transaction, it is also displayed.

In each of the report views, to return to the main page click the Home icon, or use the Back icon to return to the previous view, or the Forward icon to advance to the next view. A report that does not contain data displays an icon.

Response Time

In the All Transactions tab, click the *life* icon at the top left to display the Transaction Response Times of all transactions.



Click any bar to display the HTTP Analysis of that transaction.

Summaries

Displays the Client Network Server Breakdown of the transaction according to protocol, showing results in a pie chart and also additional details for the following protocols:

- TCP
- UCP
- HTTP
- HTTPS (secure communication)
- HLS (HTTP Live Streaming)



Client Network Server Breakdown

The legend for the chart is below the table. The values for the fields shown in each pie are:

- **Connect duration:** portion of time in which the client connected to the server, such as the request in TCP or the triple handshake in SSL (the establishment of a secure channel)
- Client duration: portion of time that the client processes (does not include time waiting for a server response)
- Response wait duration: portion of time spent waiting for the server's response
- Transmission duration: portion of time data was downloaded or uploaded

Parameters

- Total packets: total number of packets associated with the protocol
- Total KB: total throughput associated with the protocol; includes headers, such as TCP, IP, etc.

- Application turns: the number of times a communication flow change occurs from the request to the response, per protocol
- KB per application turn: the average of the throughput per Application Turn, per protocol
- Max concurrent connections: the highest number of concurrent connections, per protocol
- **Protocol overhead %:** the percentage of the total throughput per protocol used by non-data elements such as headers
- Errors: the number of errors that occurred in the transaction

General Analysis

The General Analysis displays details of all subtransactions for all protocols.

						cars.pcap	0				→ I	N →≣≣	30
Overview	0	Gener	al Ana	alysis								-	
Summaries		Low	High	Source IP	Source Port	Destination IP	Destina	Session	Time	Size	Informati	Timeline	^
General Analysis	Þ	UDP	DNS	172.30.2.117	56749	172.30.1.1	53	0	1 ms	86 bytes	DNS query		
Codeciet Lateraire	Þ	TCP	HTTP	172.30.2.117	50126	213.218.135.59	80	1	314 ms	2093 bytes	HTTP GET I		
Endpoint Latencies	Þ	TCP	нттр	172.30.2.117	50127	213.218.135.59	80	2	417 ms	10459 bytes	HTTP GET F		
TCP\UDP	Þ	ТСР	HTTP	172.30.2.117	50128	213.218.135.59	80	3	319 ms	3365 bytes	HTTP GET F		
TCP Errors & Sessions	Þ	тср	HTTP	172.30.2.117	50130	213.218.135.59	80	5	245 ms	842 bytes	HTTP GET F		
UDP Errors & Sessions	Þ	ТСР	HTTP	172.30.2.117	50131	213.218.135.59	80	6	239 ms	612 bytes	HTTP GET I		
UTTO	Þ	ТСР	HTTP	172.30.2.117	50129	213.218.135.59	80	4	244 ms	1018 bytes	HTTP GET h		
	Þ	ТСР	HTTP	172.30.2.117	50126	213.218.135.59	80	1	114 ms	621 bytes	HTTP GET F		
HTTP Analysis	Þ	ТСР	нттр	172.30.2.117	50126	213.218.135.59	80	1	202 ms	7709 bytes	HTTP GET F		
Optimization	Þ	TCP	нттр	172.30.2.117	50131	213.218.135.59	80	6	561 ms	29809 bytes	HTTP GET F		
Resources & Responses	E	Breakdown	1						Т	hroughput		0 351777035	5#05531240
		ТСР								Total			
TRT		UDP											
Sec 211.1	L	DNS res	solution						, a r (
No Description		HTTP	239ms										
]	Top sessions To	op services		

Display Options

The following options are available from the toolbar of the General Analysis and HTTP Analysis reports. The first two options are also available in the right-click menu in the table:

- Highlight options: click to select from the various options that highlight the resources according to the same Source IP, etc.
- Display in graph options: click 🚾 💌 to select to display a session, service, etc. in the graph
- Filter options: click is to select parameters that limit the display of the subtransactions to the selected criteria

Subtransaction Parameters

To adjust the display of the subtranscations:

- Sort the rows according to ascending or descending order in each column (available in all tables)
- Expand or decrease the area by dragging the borders of the area (available in the General and HTTP Analysis).

The following data is provided in table format. Each row represents a subtransaction:

- Low-level Protocol: includes TCP and UDP
- High-level Protocol: includes HTTP and HTTPS, DNS, etc.
- Source IP
- Source Port
- Destination IP
- Destination Port
- Session: number of the TCP or UDP session in which the resource was uploaded or downloaded; color coding scheme indicates the tasks occurring in the download, such as DNS resolution, TCP setup, etc.
- Time: response time of the request in milliseconds
- Size: the number of bytes used in the request/response
- Information: protocol specific information (when available)
- Timeline: the position of the Resource within the Transaction sequence

Request/Response Details

To display the Request/Response details, expand or collapse a row with the ricon in the left column. The Client and Server portion of each Request/Response are displayed in bytes.

Breakdown

The breakdown of the share of the traffic for each protocol is displayed in the graph at the lower left of the window, including TCP, UDP, DNS Resolution and HTTP.

Throughput Graph

Note: The breakdown and throughput area can be collapsed and expanded by clicking the double arrow below the main table.

Throughput

The Throughput edit options are used to modify the display in the graph.

Use the zoom bar to focus on a specific period within the transaction.

- Top Sessions: displays up to five Sessions with the most traffic
- Top Services: displays up to five Services with the most traffic

In each tab of the Throughput area, click a bar to include that item, such as a specific session in the graph. The Raw data displays the throughput in KB/sec.

To add/delete components in the Throughput graph display:

1. Click one of the tabs, for example "TCP Session 18"; the graph displays the throughput for TCP Session 18 in maroon.



2. To select other sessions, services, etc., click the pencil icon for each session or hosts that is to be displayed.

Breakdow	n		Throughput
Select ad	ditional throughput graph sions UDP Sessions Source IP Destination IP Services	×	Total
	Name	Raw Data(KB)	()
-	TCP Session6	873	TCP Session6
-	TCP Session2	402	TCP Session5
-	TCP Session5	398	TCP Session11
-	TCP Session11	385	
·	TCP Session19	247	
-	TCP Session18	217	
·	TCP Session24	96	
			Top sessions Top services

- 3. The following tabs are available and each displays the traffic (in KB/Sec) for each of the following values:
 - TCP Sessions
 - UDP Session
 - Source IP
 - Destination IP
 - Services (IP Address and Port)
- 4. Select the required item, for example "TCP session 16" or a Service Name. The metric is displayed in the graph with the selected color.

Endpoint Latencies

The Endpoint Latencies report displays details of the latency observed at the client and server endpoints.

- Source IP
- Destination IP
- Names: The name of the server

- Best estimate (ms): Best estimate of the latency between the client and server, as deduced from the TCP connections between the two, taking into consideration any additional latency found in the packet capture due to bandwidth constraints (thus, it's possible for this value to be lower than the Min value)
- Min (ms): the minimum value observed in the packet capture
- 95th percentile (ms): the maximum value observed in the packet capture, excluding outermost conditions
- Max (ms): the maximum value observed in the packet capture
- Samples: the number of packets used in the calculation of the latency

	Endpoin	t Latend	cies						
verview									
ummaries	Source IP	Destination IP	Name\s	Best	Min(ms)	5th	95th	Max	Sam
eneral Analysis	172.30.2.145	217.163.21.34	ad.yieldmanager.com	43.081	25	25	67	225	108
ndpoint Latencies	172.30.2.145	65.55.33.49	view.atdmt.com	70.065	25	25	222	224	55
	172.30.2.145	209.85.148.94	www.google.co.il, clients1.c	96.447	96	96	117	117	16
P/UDP	172.30.2.145	88.221.158.10	secure-media.ventsiaents:re	97	97	97	97	97	4
P Errors & Sessions	172.30.2.145	64.208.186.42	media.eldreldmanaret.com	99.333	95	95	116	116	6
P Errors & Sessions	172.30.2.145	23.14.86.102	libs.coremetrics.com	99.731	99	99	115	115	8
ГР	172.30.2.145	217.163.21.41	ad.yieldmanager.com	101.197	93	94	112	144	44
TP Analysis	172.30.2.145	209.85.148.13	clients5.google.com	101.233	100	100	111	111	4
imization	172.30.2.145	63.215.202.6	media.fastclick.net	105.333	104	104	107	107	3
ources & Responses	172.30.2.145	82.112.106.35	crl.microsoft.com	108	107	107	126	126	3
	172.30.2.145	64.208.186.25	www.viswit@sswit@t.com	109	98	98	120	120	3
	172.30.2.145	2.16.18.102	libs.coremetrics.com	113.5	113	113	114	114	3
	172.30.2.145	65.55.33.52	switch.atdmt.com	114.833	112	112	117	117	6
	172.30.2.145	46.137.18.97	segment-pixel.invitemedia.	152	150	150	164	164	30
RT	172.30.2.145	64.236.116.18	leadback.advertising.com	152.768	150	150	162	192	11
	172.30.2.145	94.245.121.17	view.atdmt.com	153.967	151	151	157	178	67
292.9	172.30.2.145	184.50.246.89	secure.veldmaldmanat.com	168.266	163	164	175	208	12
c	172.30.2.145	205.177.70.16	media.vientsenentset.com	171	170	170	173	173	3
	172.30.2.145	46.137.28.208	segment-pixel.invitemedia.«	208.416	206	207	210	235	86
Description	170 00 0 145	170 1 045 107		00F 7F	004	004	220	220	10

TCP/UDP Errors & Sessions

Select **TCP errors & sessions** to display the details of transaction that used the TCP protocol. Select **UDP errors & sessions** to display the details of transaction that used the UDP protocol.

TCP Errors & Sessions

Errors

No protocol errors found

Sessions

	Session	Source Address	Source Port	Destination Address	Destination Port	Total Responses	Average	Total Errors	Error Types
Þ	0	10.0.0.88	49208	10.0.0.56	80	1	1	0	No Errors
Þ	1	10.0.0.88	49209	10.0.0.56	80	1	0	0	No Errors
Þ	2	10.0.0.88	49210	10.0.0.56 🗟	80	1	1	0	No Errors
Þ	3	10.0.0.88	49211	10.0.0.56	80	1	1	0	No Errors
Þ	4	10.0.0.88	49212	10.0.0.56	80	1	0	0	No Errors
₽	5	10.0.0.88	49213	10.0.56	80	1	0	0	No Errors

Session details for TCP and UDP are displayed separately, and each includes:

- Session
- Source Address
- Source Port
- Destination Address
- Destination Port
- Total Responses
- Average Response Time
- Total Errors
- Error Types

HTTP Analysis

The details of the transaction are displayed in both table and graph format.

						cars.pcap)	•				→W →≣≣	30
Overview	E	Т	ΓΡ Α	nal	ysis				-	• • •	(
Summaries			Status	Туре	Resource	Host	Session	Instances	Time	Size	Timeline		
General Analysis	Þ	+			/	www.cattronic.	1	1	340 ms	2093 bytes	U		
Endpoint Latencies	Þ	+	⊘ 200	2	/styleTransverse.css	www.cattronic.	2	1/11	419 ms	10459 bytes	l		
	Þ	۰	⊘ 200	2	/styleTransverse.css	www.cattronic.	3	1/11	320 ms	3365 bytes	1		
TCP\UDP	Þ	R	⊘ 200	2	/styleLogin.css	www.cattronic.	5	1	247 ms	842 bytes	1		
TCP Errors & Sessions	Þ	÷	⊘ 200	2	/modalbox.css	www.cattronic.	6	1	241 ms	612 bytes	l		
UDP Errors & Sessions	Þ	+	<mark>⊘</mark> 200	2	/styleLogin.css	www.cattronic.	4	1	246 ms	1018 bytes	l		
нттр	Þ	Ŧ	⊘ 200	2	/modalbox.css	www.cattronic.	1	1	116 ms	621 bytes	1		
HTTP Analysis	Þ	+	⊘ 200	JS	/include.js	www.cattronic.	1	1	203 ms	7709 bytes	1		
	Þ	+	⊘ 200	JS	/prototype.js	www.cattronic.	6	1	562 ms	29809 bytes	1		
Optimization	Þ	+	⊘ 200	z	/scriptaculous.js	www.cattronic.	5	1/11	130 ms	2136 bytes	1		
Resources & Responses	Þ	+	⊘ 200	JS	/effects.js	www.cattronic.	4	1	291 ms	9951 bytes	1		
	Þ	+	⊘ 200	JS	/modalbox.js	www.cattronic.	3	1	179 ms	7674 bytes	1		
	Þ	+		JS	/iopopup.js	www.cattronic.	5	1	2524 ms	1831 bytes	1		
	•		A 200		llogo logio dif	unum otteopia	1	•	01 mc	EELE buton		70454 405500	1 40 700 1 7
TRT		HTT	P request	/respo	nse			Thro	Total		0 331//	70354 105552	14070917
		DNS	;	24 r	ns (wait 23 ms)				TCP				
211.1		Coni	nect time		97 ms (wait 1 ms)				HTTP •	•			
No Description		Req	uest		1 ms								
		Wait	t		110 ms							<u>iliiin</u>	
		Resp	oonse	_	108 ms						<u> </u>	. A	: @

The following options are available in the toolbar of the General Analysis and HTTP Analysis reports. The first two options are also available in the right-click menu in the table:

• Highlight options: click to select from the various options that highlight the resources according to the same Source IP, etc. This example shows all the Session's participants.

	-17		P A	nal	ysis						
			Status	Туре	Resource	Host	Session	Instances	Time	Size	Timeline
1		Ł	<mark>⊘</mark> 200		1	www.cattronic.	1	1	340 ms	2093 bytes	0
1		ŀ	⊘ 200	2	/styleTransverse.css	www.cattronic.	2	1/11	419 ms	10459 bytes	U
1		F	⊘ 200	2	/styleTransverse.css	www.cattronic.	3	1/11	320 ms	3365 bytes	l
1		ŀ	⊘ 200	2	/styleLogin.css	www.cattronic.	5	1	247 ms	842 bytes	I
3		Ł	⊘ 200	2	/modalbox.css	www.cattronic.	6	1	241 ms	612 bytes	I
1		ŀ	⊘ 200	2	/styleLogin.css	www.cattronic.	4	1	246 ms	1018 bytes	I
1		ŀ	⊘ 200	2	/modalbox.css	www.cattronic.	1	1	116 ms	621 bytes	l
1		ŀ	⊘ 200	JS	/include.js	www.cattronic.	1	1	203 ms	7709 bytes	l
ŀ		ŀ	⊘ 200	JS	/prototype.js	www.cattronic.	6	1	562 ms	29809 bytes	l
1		ŀ	⊘ 200	JS	/scriptaculous.js	www.cattronic.	5	1/11	130 ms	2136 bytes	l

• Display in graph options: click 🜌 🔽 to select to display a session, service, etc. in the graph

To search for similar resources according to URL:

1. In the toolbar, enter the required string in the Search area, in this example "in".

F		rp a	nal	ysis		-	•	•		(in			×
		Status	Туре	Resource	Host	Session	Instances	Time	Size Sea	archResults 🕊	《 1/:	13 👌	>>
Þ	Ŧ	o 200		/redirect.html	10.0.0.56	0	1	13 ms	1746 bytes	l			
Þ	+	i 304	2	/frames.htm	10.0.0.56	1	1	7 ms	797 bytes	1			
Þ	+	e 304	2	/main.htm	10.0.56	2	1	10 ms	799 bytes	l l			
Þ	+	🔕 404		/UntitledFrame-1.htm	10.0.0.56	3	1	10 ms	2301 bytes				
Þ	۰	() 304	2	/menu.htm	10.0.0.56	4	1	5 ms	799 bytes				
Þ	+	😢 404		/loop-relax.mp3	10.0.0.56	5	1/6	5 ms	2193 bytes				
Þ	+	i 304	?	/osho_menu.jpg	10.0.0.56	6	1	6 ms	696 bytes	1			
Þ	+	i 304	2	/star.gif	10.0.0.56	7	1	6 ms	691 bytes				
Þ	۰	() 304	2	/oshobw.jpg	10.0.0.56	8	1	5 ms	693 bytes	l			
Þ	+	i 304	?	/mcloud2.jpg	10.0.0.56	9	1	5 ms	694 bytes	1			
Þ	+	😢 404		/loop-relax.mp3	10.0.0.56	10	2/6	22 ms	2193 bytes	1			
Þ	Ŧ	e 304	?	/info.htm	10.0.0.56	11	1	5 ms	802 bytes	1			
Þ	+	e 304	?	/info.jpg	10.0.0.56	12	1	6 ms	691 bytes	1			

All matching resources are highlighted, in this case strings that contain "in". Use the arrows below the Search area to navigate between the results.

2. Clear the search results by closing the Search area (clicking the "X").

Subtransaction Details

To adjust the display of the request/responses:

- Sort the rows according to ascending or descending order in each column.
- Expand or decrease the area by dragging the borders of the area.

The following columns are displayed in the report:

- Expand/Collapse icon **•**: Show or hide details of each Request.
- **Up/down arrows:** The up arrow indicates a POST or PUT; down arrow indicates a GET; a head indicates an icon header; N/A indicates HTTPS; for all other types a star is shown.
- Status: The HTTP status, such as 404 (page not found) or 200 (OK).
- **Type:** The icon indicates the type of file requested, for example a graphic file.
- Resource: Displays the path that was accessed by the subtransaction.
- Host: The host (domain, server, etc.) from which the resource is uploaded or downloaded.
- Session: Number of the TCP session in which the resource was uploaded or downloaded.
- Instances: Number of times a resource with the same name appears in the transaction.
- Time: Response time of the request in milliseconds.
- Size: The number of bytes used in the request/response.
- Timeline: The position of the Resource within the Transaction sequence.

The following data is displayed per Transaction (at the left):

- **TRT**: The time it takes to transaction to complete (between the first packet of the request and the last packet of the transaction).
- **Description:** Displays the transaction's description.

To customize the display of the graph:

- The graph at the bottom-right displays the throughput. The 'handles' can be used to focus on a specific period within the Request/Response.
- Select HTTP Throughput to display the HTTP data only, or Total Throughput to display all the data in the Request/Response.

HTTP Parameters

When a resource is highlighted, the area in the below the table displays the following breakdown:

- **DNS Resolution:** Includes the wait time from the resolve time to the start of the connection (the time between the DNS query and the first SYN to the server whose name was queried).
- **Connect Time:** The TCP Setup time and any wait time between the initialization of the connection and sending of the first request data packet.
- TLS Time: SSL/TLS secure channel establishment.
- **Request:** Time required for the client to send the request to the server.
- Wait: The time (in ms) between the last packet of the request and the first packet of the response.
- Response: The time (in ms) between the first packet and the last packet of the response.
- Encrypted Data Transmission: The duration of an encrypted HTTPS session.

To display the details of a Request/Response:

- 1. Double-click the Request/Response name, or click the Expand/Collapse icon; the following columns are displayed:
 - **Request Headers:** The syntax of the request header.
 - **Response Headers:** The syntax of the response header.
 - Request Content/Response Content: Displays the image, HTML data, etc. (non-printable characters may be replaced by a ". " Content in certain formats such as PDF, Audio, Video, Flash, and Fonts are not displayed.
 - Details: Throughput of the request/response; for a HTTP POST the full URL is also displayed.

۲	Ŧ	⊘ 200	JS	/prototype.js	www.cattronic.	6	1	562 ms	29809 bytes	I
Ŧ	+	⊘ 200	z	/scriptaculous.js	www.cattronic.	5	1/11	130 ms	2136 bytes	
	Res Res Det	quest Hei sponse H sponse C cails	aders eaders ontent	<pre>// script.aculo.us scriptacu // Copyright (c) 2005-200 // // Permission is hereby gra // a copy of this software a // "Software"), to deal in t // without limitation the rig // distribute, sublicense, a // permit persons to whom // the following conditions</pre>	Ilous.js v1.8.0, 7 7 Thomas Fuchs anted, free of ch and associated c the Software witl ghts to use, cop ind/or sell copies in the Software is :	Tue Nov 0 (http://s arge, to a locumenta hout restr y, modify, s of the So furnished	6 15:01:40 cript.aculo.u ny person c ation files (ti iction, inclus merge, put ftware, and d to do so, s	+0300 200 us, http://r ubtaining he ding olish, I to ubject to	07 nir.aculo.us)	

Note: When the Resource is selected (highlighted), details of the Resource are displayed below.

HTTP Optimization

NV Analytics provides a number of Best Practice recommendations, based on data obtained from external sources, in addition to the knowledge obtained within HP from current application testing methodology.

Each transaction is given a score comparing the transaction to the best practice. The Total Score (out of 100) is the summary of the individual scores, based on the level of compliance to website programming rules. Prioritization emphasizes which transactions most affect the results. Each recommendation is weighted, based on the potential performance improvement to be obtained if the recommendation is performed, and is displayed in a negative numeric point value, such as -4 Points. In addition, a value from A - F is provided; F indicates that a significant improvement would be obtained in reducing transaction response time by implementing the recommendation, whereas A indicates that minimal benefit will be obtained.

HTTP Optimization Recommend for: Desktop I Mobile Sort by: Priority & Name	Total score: 56/100
Make fewer HTTP requests (desktop)	- 8 Point
Don't download the same data twice	- 8 Point
Avoid image scaling in HTML	- 4 Point
Minify your textual components	- 4 Point
Try to reduce the size of the cookies	- 4 Point
Avoid loading javascripts in the head section	- 4 Point
F Reduce the size of your images (desktop)	- 4 Point
E Avaid 404 array code (Not Found) arrays	2 Doint

To display the list of transactions with the relevant recommendations:

- 1. Select **Optimization** and select the rules to be included in the report; by default all are selected. The report can be displayed more than once with different rules.
- 2. Select **Desktop** or **Mobile**; you can also sort the results according to the **Priority** (according to the number of points given to each result) or by **Name** (alphabetical order).
- 3. To display additional details, click a recommendation. For example, for the recommendation "Don't download the same data twice", the details show the number of times the file appeared. Each can be clicked and viewed in the HTTP Analysis.

Note: The score and recommendations may be different for desktop and mobile results, due to platform specific optimizations.

To filter the list of rules for both mobile and desktop:

- 1. Click the Edit (pencil) icon dat the top right.
- 2. Select or deselect a rule, then click Save. When only some of the rules are selected, an

"information icon" appears beside the Edit icon

HTTP Resources and Responses

The Resources Breakdown displays the Instances and Total Throughput for each type of resource that was present in the transaction.

Resources Breakdown

The Instances pie chart shows the number of times that each type of resource appears in the transaction. The chart is divided according to type of resource, so that .jpeg images are one category, and css files are another category.

The Total Throughput chart shows the breakdown according to the same categories as the Instances chart, but the size is calculated according to the Total Throughput in KB.

HTTP Resources & Responses



HTTP Errors

The table displays each error according to the subtransaction for which it occurred. The Errors by Class graph displays the totals according to the class, such as 4xx for HTTP client errors, and 5xx for server errors.

Ŧ	Errors	
		·

Subtransaction	Total	Error Name	Erro	rs By Class
GET http://10.0.0.56/UntitledFrame-1.htm	1	HTTP 404	4	
GET http://10.0.0.56/main/loop-relax.mp3	3	HTTP 404	3	
		·	2	
			1	
			0	4vv

HLS Errors

The table displays a summary of errors that occurred during video streaming.

HLS Errors

	Subtransaction	Total	Error Name							
•	http://mlblive-l3c.mlb.com/ls02/mlbam/2011/05/27/	4	Video Buffering							
	Video Buffering - Video Buffering (4 time)									

Response Summary

Displays the number of occurrences of each HTTP response code.

Response Summary

	Response code	Occurrence
E	2xx	
Þ	200 OK	6
E	4xx	
Þ	404 Not found	4

Secure Communication

Secure HTTPS communication can be viewed in the NV Analytics, including the Host Name, TCP Session Establishment and SSL Session Establishment data.

Note: When the private key is not known or unavailable, a debugging proxy such as Fiddler or mitmproxy can be utilized. For details, see "Setting up the NV Environment" in the *HP Network Virtualization for Mobile User Guide*.

The private key must be a text file in the Privacy Enhanced Mail (PEM) format, as in this example:

```
-----BEGIN RSA PRIVATE KEY-----

MIICXAIBAAKBgQC20yQBpiE1atCfIuiOqa9fnPyHrCsYCoCJ/hKlS8z6IOcmBsNq

kZckzlMqcQP7i9o3SGVBXv1rAsGN3SnkqNUD4PFukUHQrjPpc0KPX9KdDCcLFu5f

Bxq+7/7hDzQA+rKWqixO3WmBVcKQm+3WvpF5M+jJuIsRY806r0xb+FrpvwIDAQAB

AoGBAKWpszHPUL4vmPTqU+KZ/bDc9rMVnkL9mTXxRQIFKqroT6vUaxTQ8i1GzHNj

zyELu+NmNWJD6cwixjJ/Wmu5VLLT9syvBnAuLNDwn02/TrOurxdk/FeEuPEF6exo

GKKDAAzDCYvEBDOEbYBx/3s+v0n3XLeW6dKX76JZkv97LokZAkEA2kO1u+WDAGcB

s5pq3I2R10ELeao4d/1dY7x4Efap3HJNWMF0byZEwPyC5yKKEZQDKt3n549nTPxM

wD09geHjqwJBANZu41xS/+4WkqwN5yzh8VCKmMUc0PQvXw+niXrjSucc5k5VdHw4

qOobXalp1PyrafHj6YV8PfX9XHpPRAzs/j0CQBq8iJqHttBpzc+0buCthtPR7XHt

5BTuuG3rPFoW+R/D8K2apQSoj2uEgxSFLcvpcaninPHEo0b08SfqLqCmZxkCQADb

dKA13LwQ7wktDQ2K4bIWu8Gd+d/gCJtBajVJj1UZMnqBsPOGLnaxIVC6EZXpAYVs

CdT0yDKhjqsWggkjMWkCQHZvP1E28M51k1pLsQx43nq7zbueKZWkDg/biA3yOaLb

FJ9TSJeufAXAmG/US+zCfgLuzrSuJwHiCMnhRrBOm+Y=

-----END RSA PRIVATE KEY-----
```

If it is not in this format, use Open SSL to change the format of the private key. Determine the operating system of the Server. On Microsoft® Windows, keys are often stored in PKCS7/DER format (locally) or in .NET format (from any directory server). To convert use these commands:

for PKCS7/DER keys (as held on disk)
openssl pkcs8 -nocrypt -in derfile.key -inform DER -out key.pem -outform PEM
for NET keys (from the directory server)
openssl pkcs8 -nocrypt -in file.ick -inform NET -out key.pem -outform PEM

On Mac OSX, Solaris, and other systems the file format used is often PKCS#12. To convert use this command:

openssl pkcs12 -nodes -in file.p12 -out key.pem -nocerts -nodes

On Linux use these commands:

openssl x509 -nocrypt -in foo.der -informat DER -out key.pem -outformat PEM openssl x509 -nocrypt -in foo.net -informat NET -out key.pem -outformat PEM

Note: To analyze secure communication when the server's private key is known, refer to the SSL options in "HTTPS Keys" on page 17.

0 t						Untitled	Trans	sa (→	W →II
Overview Summaries	Н	Т	гр а	nal	lysis					-		C				٩
General Analysis			Status	Туре	e Resource	Host	Session	Instances	Time	Size	Timeline		•	-		
	P	+	↔ 302		/	shunra.jira.con	2	1	2466 ms	990 bytes	1					
TCP\UDP	•	+	200		/Dashboard.jspa	shunra.jira.con	3	1	2685 ms	44333 bytes		1				
TCP Errors & Sessions	•	+	Ø 200	CSS	/com.atlassian.gadgets.dashbo	shunra.jira.con	4	1	1254 ms	25362 bytes					-	
UDP Errors & Sessions		Re	auest He	aders	HTTP/1.1 200 OK											
нттр		Po		loador	Date: Sun, 10 Jul 2011 (Server: Apache/2.2.3 (C	7:48:05 GMT entOS)										
HTTP Analysis		De.	sponse i	leavers	X-AREQUESTID: 648x26	754x1										
Optimization		Re	sponse C	ontent.	Expires: Sun, 10 Jul 201	1 16:33:41 GMT										
Resources & Responses		De	tails		Cache-Control: max-age Cache-Control: public	=31536000										
					Last-Modified: Wed, 06 ETag: "1309993840000" Vary: Accept-Encoding	ul 2011 23:10:40	GMT									
		нття	PS decry	pted se	ession			Thr	oughput		4440 5544	6647	7750	8854	9958	MS ¥
TRT	L	D	NS resol	ution				-	 Total 						,	(B/Sec 0.4
11.1		Coni	nect time	e 47 r	ms (wait 46 ms)				■ TCP HTTP							0.2
sec		TLS	time	32	ms			ø 💻	•							
No Description		Req	uest	11	ms				•							
		Wait			1163 ms			*	•							
		Resp	oonse		11 ms			To	p sessions	Top hosts		1			€ ,	Q. Q.

This is an example of the detailed view when HTTPS traffic is decrypted.

When the traffic is not decrypted, the transaction details are not visible, as in this example.

0 t				Untitle	d Transa 💿		→W →	51
Overview Summaries	HTTP Analysis					• •	(م	
General Analysis	Status Type Resource	Host Session www.cattronic. 6	Instances Time 1 91 ms	Size 1269 bytes	Timeline			
TCP\UDP	🕨 🖡 🥑 200 📠 /fleche_panier.png	www.cattronic. 4	1 86 ms	1024 bytes	1			
TCP Errors & Sessions	🕨 🐺 👩 200 📠 /filet_total.png	www.cattronic. 2	1 87 ms	973 bytes	I. I.			
UDP Errors & Sessions	▶ ♣ Ø 200 mm /pct_fleche_right.png	www.cattronic. 5	1 87 ms	1393 bytes	l.			1
	▶ ⊚ ⊗ N/A 🔤	74.125.39.19 9	1 136881 ms	148 bytes				
нттр	▶ © © N/A 🔤	74.125.39.17 10	1 282 ms	375 bytes				
HTTP Analysis	🖊 🖡 🥑 200 📠 /croix_fermeture.png	www.cattronic. 6	1 84 ms	2659 bytes	I. I.			
Optimization	🕨 🔻 👩 200 📠 /spinner.gif	www.cattronic. 5	1 172 ms	8288 bytes	1			
Resources & Responses	🕨 🔻 🕑 200 📠 /cg	www.cattronic. 2	1 93 ms	1214 bytes	I. I.			
	🕨 🔻 🕑 200 📠 /cg.gif	www.cattronic. 6	1 646 ms	44013 bytes	I			
	🕨 🖡 🥑 200 📠 /startSearchIdent.do	www.cattronic. 6	1/2 280 ms	14202 bytes				
	+ + 📀 200 🝙 /styleTransverse.css	www.cattronic. 2	3/11 85 ms	3384 bytes	l I			
	▶ ♦ Ø 200 💼 /styleTransverse.css	www.cattronic. 5	3/11 257 ms	10478 bytes				
	🕨 🖡 🥑 200 💼 /applicabilite.css	www.cattronic. 11	2/10 183 ms	634 bytes				
	🕨 🕸 🥑 200 💼 /applicabilite.css	www.cattronic. 12	2/10 196 ms	662 bytes	1			
	🕨 🔻 😝 304 💼 /scriptaculous.js	www.cattronic. 13	3/11 179 ms	691 bytes	l			
	🕨 🛡 🥑 200 💼 /styleQuantite.css	www.cattronic. 2	2/10 84 ms	948 bytes	1			
	🕨 🛡 🥑 200 💼 /styleDevis.css	www.cattronic. 6	3/20 82 ms	623 bytes	1			
		www.cattronic 2	2/20 92 mc	632 hutor	n ' 35177 ' 70354 ' 105	532 140709	175886 MS	
	HTTPS encrypted session		- Total				KB/Sec	
	DNS resolution		- TCP				80	
			- HTTP				70	
TRT			TCP Sess	ion6	1		60	
	Connect time		💉 📼 TCP Sess	ion2			50	
0 211.1			TCP Sess	ion5			40	
sec			💉 💳 TCP Sess	ion19			30	
	TLS time						20	
No Description							10	
	Https 136881	ms	Top sessions	Top hosts	A AM			
			100 0000000					

Chapter 3: NV Analytics API

NV Analytics API uses Representation State Transfer (REST) web services architecture. The analysis API requests all have a same URL structure, the prefix is:

[base address]/shunra/api/analysis

Note: A code sample "analyzer.py" in Python is available in the installation folder. Updates can be found in https://gist.github.com/2773832

It can be used to access the API. Segments of the code are also provided in this document.

The following methods are exposed:

AnalysisEngines	40
Extract Packet Lists	41
AnalysisRequest	43
AnalysisSummary	47
AnalysisArtifact	48

AnalysisEngines

Provides a JSON contents list of the installed analysis engines.

GET

[base address]/shunra/api/analysis/engines

Example: http://localhost:8182/shunra/api/analysis/engines

Response

The response includes the id and names of all the analysis engines.

```
{
  "supportedAnalysisEngines":[{"name":"harExport","id":"harExport"},
  {"name":"networkmeasurements","id":"networkmeasurements"},
  {"name":"generalWaterfall","id":"generalWaterfall"},{"name":"http","id":"http"},
  {"name":"iostats","id":"iostats"},{"name":"metrics","id":"metrics"},
  {"name":"best practices","id":"best practices"}]}
}
```

User Guide Chapter 3: NV Analytics API

Returns

- 200 "OK"
- 404 "Not Found"
- 500 "Internal Server Error"

Code sample

```
def get_engine_id(engine_name):
"""
    Returns the analysis engine id, given its name.
    This can also be used as a sort of a sanity test for the analysis api.
    >>> get_engine_id('best practices')
    u'best practices'
    """
    resp = get('/shunra/api/analysis/engines')
    engines = dict([(entry['name'], entry['id']) for entry in resp
['supportedAnalysisEngines']])
    return engines[engine_name]
```

Extract Packet Lists

Provides a JSON contents of packet list names, IDs, endpoints and .pcap and .ved file unique IDs.

PUT

[base address]/shunra/api/analysis/packetlistmetadata

Example: http://localhost:8182/shunra/api/analysis/packetlistmetadata

Body

{

The JSON defines the analyzed emulation result (.ved or .pcap file) ID. It is a file system path for NV Analytics:

```
{
"id":"C:\\tmp\\Sample.ved"
}
```

The response includes the ID of the analyzed run result and the packet list metadata (names, IDs, endpoints):

```
"packetLists": [{
"endpoints": [{
```

```
"name":"Tokyo Office",
                "id":"6d0652db88c349de9382a54dc350349f"
        }],
        "name":"Packet List 3",
        "id":"c6064d9bf25d405382e374795fef35fe"
},
{
        "endpoints": [{
                "name":"London Office",
                "id":"de358779547c4eea8caeef62bfbbb493"
        }],
        "name":"Packet List 2",
        "id":"59220e1cb4d248eba3b89a695918be91"
},
{
        "endpoints": [{
                "name":"NY Office",
                "id":"8c95498f7bb04c7598dde1d5e609082a"
        }],
        "name": "Packet List 1",
        "id":"620984c9a31b4ef694a1ac47d61b6a7e"
}],
"runResultId": "b80de7f5ffa97428b2324c8b3a9d469b"
```

}

Returns

- 200 "OK"
- 404 "Not Found"
- 500 "Internal Server Error"

Code sample

```
def get_packetlists(inputfilepath):
"""
```

```
Returns a dictionary of the available packet lists in the given file.
The dictionary keys are the packet lists names, and the dictionary values are the
packet lists ids.
```

```
>>> packetlists = get_packetlists(os.path.join(SAMPLE_FOLDER, 'Sample.ved'))
>>> len(packetlists)
3
>>> 'Packet List 1' in packetlists
True
"""
resp = put('/shunra/api/analysis/packetlistmetadata', {'id':inputfilepath})
```

```
return dict([(entry['name'], entry['id']) for entry in resp['packetLists']])
def get_run_result_id(inputfilepath):
resp = put('/shunra/api/analysis/packetlistmetadata', {'id':inputfilepath})
return resp['runResultId']
```

AnalysisRequest

Represented by JSON; provides contents of current status of analysis process per packet list, per transaction and per analysis engine.

The response is a dictionary with the following entries:

- transactionAnalysisStatus a list of transactions as described below
- reportId an identifier for the analysis process
- name name of the analyzed packet list
- id id of the analyzed packet list

The list of transactions contains an entry for each transaction assoicated with the packet list.

Each entry is a dictionary, containing the transaction id (id), name (name) and analysis status (analysisStatusPerEngine).

Analysis status is a dictionary whose keys are the analysis engine, and the values are their status as specified in the API documentation.

PUT

```
[base address]/shunra/api/analysis/request/{plid}
```

```
Example:
http://localhost:8182/shunra/api/analysis/request/620984c9a31b4ef694a1ac47d61b6a
7e
```

Where "plid" is a packet list unique ID that has been returned by the Extract Packet List request.

Body

Contains the analysis parameters such as ports, SSL Encryption Key, and the analyzed emulation result (.ved or .pcap file) ID. and the file system path, because the file is not persisted (retained?) by the system. The body is in JSON format:

```
{
  "ports":"80, 8080",
  "sslEncryptionKey": "172.30.2.31,443,http,C:\\keys\\secret.key",
  "runResultHandle":"C:\\tmp\\Sample.ved"
```

}

The response includes a current analysis status per transaction, per installed analysis engines and the generated analysis report id identifying the analysis parameters:

```
{
```

```
"transactionAnalysisStatus": [{
        "analysisStatusPerEngine": {
                "networkmeasurements":"Started",
                "harExport":"Started",
                "generalWaterfall":"Started",
                "http":"Started",
                "iostats":"Started",
                "metrics":"Started",
                "best practices":"Started"
        },
        "name":"Undefined",
        "id":"ccb8713e522241c9a691c4ed1ce72d27"
}],
"reportId":"-561678026",
"name":"Packet List 1",
"id":"620984c9a31b4ef694a1ac47d61b6a7e"
```

}

Where possible analysis statuses are:

public enum WorkStatus {

// a job still has not been started, not proceeded, analyzed, etc

Idle(0),

// a job (for example emulation or analysis) started

Started(1),

// a job (for example emulation or analysis) finished

```
Finished(2),
```

// a job (for example analysis) failed

Failed(3);

```
}
```

Note: The heuristic for analysis process completeness is that all the items have either a Finished or Failed status; otherwise some items in the analysis jobs pool have not completed yet.

The client side should continue to process analysis requests until the analysis process has completed.

Returns

- 200 "OK"
- 404 "Not Found"
- 500 "Internal Server Error"

Code Sample

def analyze(inputfilepath, packetlist_id, settings={}):

calls analysis on a given file (use settings to pass special analysis parameters

such as port numbers and ssl keys)

packetlist_id should be the id return by get_packetlists for a specific packet list.

The response is a dictionary with the following entries:

- * transactionAnalysisStatus a list of transactions as described below
- * reportId an identifier for the analysis process
- * name name of the analyzed packet list
- * id id of the analyzed packet list

The list of transactions contains an entry for each transaction assoicated with the packet list.

```
Each entry is a dictionary, containing the transaction id (id), name (name) and analysis status (analysisStatusPerEngine).
```

Analysis status is a dictionary whose keys are the analysis engine, and the values are their status as specified in the API documentation.

```
>>> inputfilepath = os.path.join(SAMPLE_FOLDER, 'Sample.ved')
       >>> packetlists = get packetlists(inputfilepath)
       >>> packetlist id = packetlists['Packet List 1']
       >>> resp = start_analysis(inputfilepath, packetlist_id)
['transactionAnalysisStatus']
       >>> len(resp) # only one tranasction is associated with this packet list
       1
       >> resp[0]['name']
       u'Undefined'
       >>> resp[0]['analysisStatusPerEngine']['http'] in ['Idle', 'Started', 'Finished',
'Failed']
       True
       .....
       params = dict(settings)
       params['runResultHandle'] = inputfilepath
       resp = put('/shunra/api/analysis/request/'+packetlist_id, params)
       return resp
       def get_report_id(inputfilepath, packetlist_id, settings={}):
       return analyze(inputfilepath, packetlist_id, settings)['reportId']
```

```
def get_transactions(inputfilepath, packetlist_id, settings={}):
.....
       Gets all the transactions associated with a given packetlist.
       The result is a list of pairs, the first element of each pair is the transaction
id, and the second is the transaction's name
       >>> inputfilepath = os.path.join(SAMPLE_FOLDER, 'Sample.ved')
       >>> packetlists = get_packetlists(inputfilepath)
       >>> packetlist_id = packetlists['Packet List 1']
       >>> result = get_transactions(inputfilepath, packetlist_id)
       >>> len(result) # only one transaction is associated with this packet list
       1
       >>> result[0][1]
       u'Undefined'
       .....
       return [(transaction['id'], transaction['name']) for transaction in analyze
(inputfilepath, packetlist_id, settings)['transactionAnalysisStatus']]
       def start_analysis(inputfilepath, packetlist_id, settings={}):
.....
Starts analysis on a given file.
       The response is a list, with an entry for each transaction assoicated with the
packet list.
       Each entry is a dictionary, containing the transaction id (id), name (name) and
analysis status (analysisStatusPerEngine).
       Analysis status is a dictionary whose keys are the analysis engine, and the values
are their status as specified in the API documentation.
       .....
       return analyze(inputfilepath, packetlist_id, settings)
def is_analysis_done(inputfilepath, packetlist_id, settings={}):
       .....
       Returns True if all the transactions associate with the given packet list were
analyzed and their reports are ready to be fetched.
       resp = analyze(inputfilepath, packetlist_id, settings)
['transactionAnalysisStatus']
       for transaction in resp:
       for engine_status in transaction['analysisStatusPerEngine'].values():
if engine_status in ['Idle','Started']:
return False
return True
```

AnalysisSummary

Represented by JSON; provides the contents of analysis summary per packet list, per transaction, and per analysis engine.

GET

[base address]/shunra/api/analysis/summary/{runresulthandle}/{plid}/{reportId}/
{engineId}

Example:

```
http://localhost:8182/shunra/api/analysis/summary/b80de7f5ffa97428b2324c8b3a9d46
9b/620984c9a31b4ef694a1ac47d61b6a7e/-561678026/best%20practice
```

GET

[base address]/shunra/api/analysis/summary/{runresulthandle}/{plid}/{reportId}/
{trId}/{engineId}

Example:

http://localhost:8182/shunra/api/analysis/summary/b80de7f5ffa97428b2324c8b3a9d46
9b/620984c9a31b4ef694a1ac47d61b6a7e/-561678026/best%20practices

The first call returns the requested analysis report for all transactions in packet list. The second returns it for the specified transaction only.

The following report types are currently supported:

- http: HTTP Analysis
- best practices: Optimization report
- iostats: Throughput report
- general/waterfall: General Analysis
- metrics: the protocols' summary and metrics report
- networkmeasurements: Endpoints Latencies report
- harExport: a report containing the HTTP subtransaction in HAR format (experimental)

Response

GET

'/shunra/api/analysis/summary/%s/%s/%s/%s'%(run_result_handle, packetlist_id, report_id, transaction_id, engine_id)

Returns

resp['successfullTransactionAnalysis'][0]['result'] ???

- 200 "OK"
- 404 "Not Found"
- 500 "Internal Server Error"

Code Sample

```
def get_analysis_report(run_result_handle, packetlist_id, report_id, transaction_
id, engine_id):
    """
    Get the result of running one of the analysis engines on a given packet list
    """
    resp = get('/shunra/api/analysis/summary/%s/%s/%s/%s/%s'%(run_result_handle,
packetlist_id, report_id, transaction_id, engine_id))
    return resp['successfullTransactionAnalysis'][0]['result']
```

AnalysisArtifact

A file found within a transaction, such as a picture, movie, doc, text, etc.

This section contains:

- "Structure of an Analysis Report" on the next page
- "Structure of the HTTP Waterfall Analysis Report" on the next page
- "Structure of the Best Practices Analysis Report" on page 52

GET

[base address]/shunra/api/analysis/artifact/{filehandle}

Where the artifact handle is taken from the analysis report (See "Structure of an Analysis Report" on the next page).

Example:

```
http://localhost:8182/shunra/api/analysis/artifact/620984c9a31b4ef694a1ac47d61b6
a7e%2F-
561678026%2Fccb8713e522241c9a691c4ed1ce72d27%2F94660f9c01724f63bedfefb370dc4575%
2Fabf8bde63762421dbe29cab1cecae661
```

Returns

User Guide Chapter 3: NV Analytics API

- 200 "OK"
- 404 "Not Found"
- 500 "Internal Server Error"

Structure of an Analysis Report

As a result of executing step (4) "Get Analysis Result" for a specific transaction, the API returns a JSON document of the following format:

```
{
       "name": "Packet List 1",
       "successfullTransactionAnalysis": [{
               "status": "Finished",
               "result": {
                       "type": "Best Practices Report",
                       "subtype": "Web Applications Best Practices Report",
                       "version": "0.5",
                       -- Other, analysis engine dependent fields --
               },
               "name": "Undefined",
               "id": "fe15bdf3eafe4ec8bb1b055c49ca622b"
       }],
       "reportType": "best practices",
       "reportId": "129778102",
       "id": "1ae2d2ee02144e69801e0f3d1cb39d89",
       "failedTransactionAnalysis": []
}
```

Note: The text in blue will list the actual report. Notice that since the request is for a specific engine for a specific transaction, the results for "successfullTransactionAnalysis" contains only one entry, which lists the report for that transaction, wrapped with the transaction's name and ID. The reports all share a common structure, with "type", "subtype" and "version" fields.

Structure of the HTTP Waterfall Analysis Report

A typical HTTP Waterfall report contains many entries in the "subTransactions" list, each is one of two possible types:

- HTTP request/response: contains a singles HTTP request coming either from a HTTP session or a decrypted HTTPS session
- HTTPS session: contains details about an non-decrypted HTTPS session (highlighted in red)

Not all fields are mandatory. Below, only the fields marked in red are guaranteed to be available in each entry. For example, if the response to a given request was not captured in the packet list, all the fields associated with a response do not appear in the entry. Thus, only the component that details the request's timestamps is guaranteed to be available. (Currently, the report contains only HTTP request/response pairs where the request was captured).

Timestamps are marked in blue and represent the number of seconds since January 1st, 1970. The handle to the response data is marked in orange. The response data itself can be retrieved by using the "Get Analysis Artifact" (5) API call.

```
{
       "type": "Waterfall report",
       "subtype": "Http Waterfall report",
       "version": "0.80",
       "subTransactions": [{
               "type": "HTTP request/response",
               "start": 1333054863953,
               "end": 1333054864640,
               "recomendations": "",
               "attributes": {
                       "RequestContentSize": 0,
                       "ResponseContentType": "application/json; charset\u003dUTF-8",
                       "StatusCode": 401,
                       "TcpReset": false,
                       "Method": "POST",
                       "Scheme": "https",
                       "ResponseContentSize": 104,
                       "RequestHeaders": "POST /setup/ws/1/validate HTTP/1.1\r\nHost:
setup.example.com\r\ nAccept-Encoding: gzip, deflate\r\nConnection: keep-
alive\r\nProxy-Connection: keep-alive\r\n",
                       "TcpSession": 4,
                       "RequestData": "",
                       "RequestContentType": "text/plain",
                       "URI": "/setup/ws/1/validate",
                       "ResponseData":
                       "\\74b85bbff75340a9b744bf8b4d1f5f6b\\-
1019702096\\5d35c39d1db84f3fa16786dc78eff622\\0703708ccbda491ba6d59944c1ef1114/7882
0e83d8634265900999172f134389",
                       "ResponseHeaders": "HTTP/1.1 401 Unauthorized\r\nDate: Thu, 29 Mar 2012 21:01
GMT \r\nConnection: Keep-alive\r\n",
                       "host": "setup.example.com",
                       "Referer": "https://www.example.com/"
               },
               "components": [{
                       "type": "DNSResolution",
                       "start": 1333054863853,
                       "end": 1333054863900
               },
               {
                       "type": "TCPSetup",
```

```
"start": 1333054863900,
                "end": 1333054863953
        },
        {
                "type": "ClientWaitAfterTCPSetup",
                "start": 1333054863953,
                "end": 1333054864418
        },
        {
                "type": "TLSHandshake",
                "start": 1333054864418,
                "end": 1333054864632
        },
        {
                "type": "request",
                "start": 1333054864632,
                "end": 1333054864632
        },
        {
                "type": "wait",
                "start": 1333054864632,
                "end": 1333054864640
        },
        {
                "type": "response",
                "start": 1333054864640,
                "end": 1333054864640
        }
]},
        "type": "HTTPS session",
        "start": 1333054861902,
        "end": 1333054863281,
        "recomendations": "",
        "attributes": {
                "SentBytes": 384,
                "ReceivedBytes": 5792,
                "host": "www.example.com",
                "TcpReset": false,
                "TcpSession": 0
        },
        "components": [{
                "type": "TCPSetup",
                "start": 1333054861902,
                "end": 1333054861902
        },
        {
                "type": "ClientWaitAfterTCPSetup",
```

{

```
"start": 1333054861902,
            "end": 1333054862731
        },
        {
            "type": "TLSHandshake",
            "start": 1333054862731,
            "end": 1333054863230
        },
        {
            "type": "EncryptedDataTransmission",
            "start": 1333054863230,
            "end": 1333054863281
        }
}
```

Structure of the Best Practices Analysis Report

The best practices report is quite simple. report is a list of entries, each representing a best practice; In the example below two best practices are highlighted in blue. Each best practice contains the following fields:

• Name

]}

- Description
- List of applicable scenarios (currently DesktopWeb or MobileSafari)
- Score: which measures the how much the transaction follows the given best practice (a number in [0,1]),
- Weight: measures the impact of the best practice on the transaction (a number in [0,1]).
- A dictionary of violations: each entry in this dictionary is a specific type of violation on the best practice and a list of resources (or TCP sessions) that are committing that violation. Notice that a transaction may not have any violation for a given best practice, as is the case with "Compress Components" below.

```
{
    "type": "Best Practices Report",
    "subtype": "Web Applications Best Practices Report",
    "version": "0.5",
    "report": [{
        "violations": {},
        "name": "Compress Components",
```

```
"scenarios": [
                       "DesktopWeb",
                       "MobileSafari"
               ],
               "description": "Checks that textual elements are transferred in a compressed
format. Compression usually reduces the response size by about 70%. Approximately
90% of current Internet traffic travels through browsers that claim to support
gzip. ",
               "score": 100.0,
               "weight": 1.0
       },
       {
               "violations": {
                       "An expiration header was not found": [
                                "http://platform.example.com/widgets.js"
                       ],
                       "Expiration date is within the next two days": [
                       "HTTP://media.example.com/media-proxy/picture1.jpg",
                       "http://media.example.com/media-proxy/picture2.jpg",
                       "http://media.example.com/media-proxy/picture3.jpg"
               ]},
               "name": "Add long term headers expiration dates",
               "scenarios": [
                       "DesktopWeb",
                       "MobileSafari"
               ],
               "description": "Near future headers expiration dates prevent effective caching.
This results in a repeat visit to your site to be slower than necessary.",
               "score": 65.0,
               "weight": 0.8
       }]
}
```

Chapter 4: NV Analytics Protocols

This section provides details regarding how NV Analytics identifies and works with various protocols in the legacy reports, including:

Supported Protocols	54
Conversation Definition	. 54
Collecting Conversation Statistics	. 55
Classification of TCP, UDP, IP	55
Sub-Transaction Grouping	. 55
Understanding Protocol Association	. 55

Supported Protocols

The following protocols are supported and analyzed by NV Analytics.

Layer 2 - 3	Web
IP	HTTP
TCP	HTTPS
UDP	

Conversation Definition

The definition and identification of a conversation depends on the type of analysis being performed.

The definitions (identifications) are based on:

- IP IP address pair (e.g. 10.0.0.1 10.0.0.2)
- UDP IP address & port number pair (e.g. 10.0.0.1:6789 10.0.0.2:3456)
- TCP IP address & port number pair (e.g. 10.0.0.1:6789 10.0.0.2:3456)
- HTTP URL (e.g. www.google.com/images)

Collecting Conversation Statistics

NV Analytics collects statistics per conversation instance (e.g. a single Get of www.google.com URL). Metrics shown will be for the following groupings:

- All applications
- Per application
- Per application conversation (including sub-conversation)

Classification of TCP, UDP, IP

A conversation is classified as TCP only if no higher level protocol is present. In this case it is identified as TCP Other.

A conversation is classified as UDP only if no higher level protocol is present. In this case it is identified as UDP Other.

A conversation is classified as IP only if there is no higher level protocol e.g. not TCP or UDP). In this case it is identified as IP Other.

Sub-Transaction Grouping

The NV Analytics groups sub-conversations into a single flow so that you can get data of the whole conversation or on each of the sub-conversations that it contains. How this grouping takes places is determined by how you configure NV Analytics for grouping.

Understanding Protocol Association

Conversations are associated based on the relevant application protocol. This means that if we're looking at HTTP then the underlying TCP / IP communication and communication metrics will be associated with the HTTP conversation they are part of. If we found TCP Retransmissions during a HTTP Get Request-Response conversation, these TCP Retransmissions will be associated with the HTTP conversation.

When a sequence number is received that is lower than expected (i.e. either a retransmission, a fast retransmission, or an out of order segment), NV Analytics assumes that it is a fast retransmission if:

- It has seen >= 2 duplicate ACKs for this segment (i.e. >=3 ACKs).
- If this segment is the next un-ACKed segment.
- If this segment came within 20ms of the last duplicate ACK (20ms is arbitrary; it should be small enough to not be confused with a retransmission timeout).



