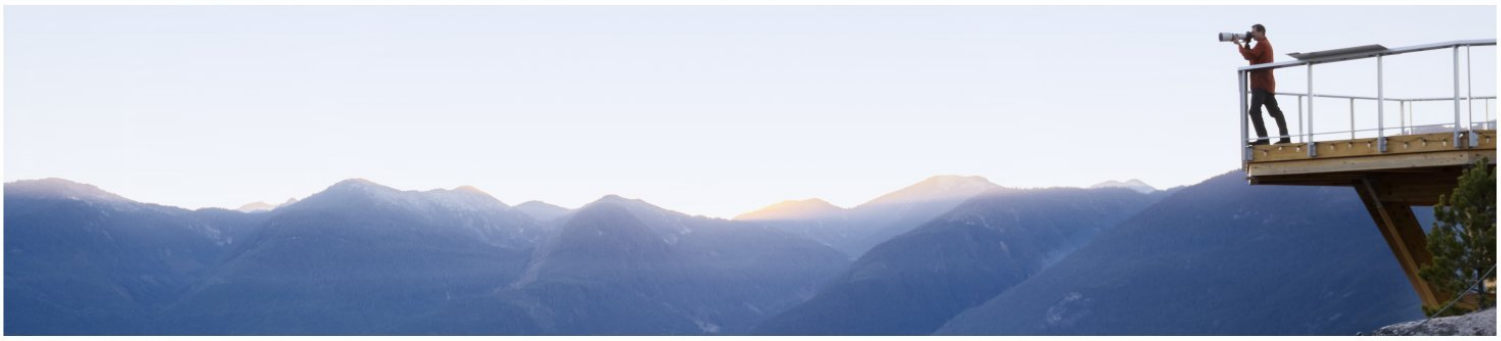
Real User Monitor

Version 9.51, Released November 2018

RUM Troubleshooting

Published November 2018



Legal Notices

Disclaimer

Certain versions of software and/or documents (“Material”) accessible here may contain branding from Hewlett-Packard Company (now HP Inc.) and Hewlett Packard Enterprise Company. As of September 1, 2017, the Material is now offered by Micro Focus, a separately owned and operated company. Any reference to the HP and Hewlett Packard Enterprise/HPE marks is historical in nature, and the HP and Hewlett Packard Enterprise/HPE marks are the property of their respective owners.

Warranty

The only warranties for products and services of Micro Focus and its affiliates and licensors (“Micro Focus”) are set forth in the express warranty statements accompanying such products and services. Nothing herein should be construed as constituting an additional warranty. Micro Focus 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

Contains Confidential Information. Except as specifically indicated otherwise, a valid license is 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

Trademark Notices

Adobe® and Acrobat® are trademarks of Adobe Systems Incorporated.

AMD and the AMD Arrow symbol are trademarks of Advanced Micro Devices, Inc.

Google™ and Google Maps™ are trademarks of Google Inc.

Intel®, Itanium®, Pentium®, and Intel® Xeon® are trademarks of Intel Corporation in the U.S. and other countries.

iPod is a trademark of Apple Computer, Inc.

Java is a registered trademark of Oracle and/or its affiliates.

Microsoft®, Windows®, Windows NT®, Windows® XP, and Windows Vista® are U.S. registered trademarks of Microsoft Corporation.

Oracle is a registered trademark of Oracle Corporation and/or its affiliates.

UNIX® is a registered trademark of The Open Group.

Contents

Chapter 1: Mobile Solution Troubleshooting 4
No Data in Reports 4
No Data in the Mobile Health Report but other Reports Display Data for this Application 6
Missing Crash Events 6
Chapter 2: Session Replay Troubleshooting 7
General Requirements 7
Logs from Session Replay Report 8
Internet Explorer Configuration Settings 9
Update for Configurations with Load Balancer 11
Problematic Session Replay UI in Internet Explorer 10 and 11 14
Poor Performance Environment 14
Chapter 3: Server Collector Troubleshooting 16
SSL Issues 16
How to Decrypt SSL Traffic in Wireshark 16
Switch Off SSL Communication Encryption 18
Connectivity Issues 19
Connect Wireshark to RUM Server Collector 19
Troubleshooting Low Connectivity between RUM Probe and RUM Server Collector 21
Setting Device Names on Windows 22
RUM Server Collector Workflow 23
Traffic Transferred to RUM Probe 24
UTC Time Difference 25
Port Number 25
Connect RUM Server Collector to Several Probes 25
Connect Several RUM Server Collectors to a Single RUM Probe 26
Starting the RUM Probe and RUM Server Collector 27
Reconnecting to a RUM Server Collector 28
TCP Offloading 28
Chapter 4: Performance Probe Packet Loss 30
Chapter 5: High CPU Utilization 31
Send Documentation Feedback 32

Chapter 1: Mobile Solution Troubleshooting

The following are possible issues that may arise when implementing the RUM mobile solution.

- "No Data in Reports" below
- "No Data in the Mobile Health Report but other Reports Display Data for this Application" on page 6
- "Missing Crash Events" on page 6

No Data in Reports

Possible Causes	Possible Solutions
There was no application traffic so there was no data to report.	Generate application traffic with at least 20 - 30 actions. Wait 10 - 15 minutes and then check the Session Analyzer Report.
When configuring the application, the wrong template was used.	Check that you used the **Mobile Application** template when you configured the application. 1. Select **APM > Admin > End User Management**. 2. Click the **Monitoring** tab. 3. Select the **End User Monitors** view. 4. Click the mobile application. 5. Click the **Real User Monitor** tab. 6. In the Template name field, verify that the name of the template is **Mobile Application**.
When instrumenting the application, the wrong application was used (using the wrong application key).	Re-instrument the application.
When instrumenting the application using a command line, the wrong key or RUM Client Monitor Probe URL was used.	Make sure you used the correct key and URL when you instrumented the application. See "Mobile Application Instrumentation" in the Real User Monitor Administration Guide. Test the probe URL defined during the instrumentation by opening a browser on the RUM Engine machine and accessing the Client Monitor Probe **(http://<RUM Client Monitor host>:<RUM Client Monitor port>**). The response should be "hello".

Possible Causes	Possible Solutions
When instrumenting the application, the wrong communications protocol was used (HTTP instead of HTTPs, or HTTPs instead of HTTP).	Re-instrument the application using the correct Client Monitor Probe URL. Test the probe URL defined during the instrumentation by opening a browser on the RUM Engine machine and accessing the Client Monitor Probe **(http://<RUM Client Monitor host>:<RUM Client Monitor port>**). The response should be "hello".
There is no connection between the device and the RUM Client Monitor Probe and therefore there is no data in the RUM Client Monitor Probe channel.	Open a browser on the RUM Engine machine and access the Client Monitor Probe **(http://<RUM Client Monitor host>:<RUM Client Monitor port>**). The response should be "hello". Make sure the probe is up and running. If the Client Monitor Probe is unreachable and an HTTP error is returned, contact network support. Make sure the firewall on the Client Monitor Probe machine does not block the Probe's services.
There is no connection between the Client Monitor Probe and the RUM Engine. In the System Status page in the RUM web console the **Probe** status is red (error) and the **Connect to Probe** status is red (error).	Open a browser on the RUM Engine machine and access the Client Monitor Probe (**http://<RUM Client Monitor host>:<RUM Client Monitor port>**). The response should be "hello". Make sure the firewall on the Client Monitor Probe machine does not block the Probe's services. Contact network support.
Client Monitor Probe is not in DMZ or firewall blocks the probe's services.	Work with your network administrator to make sure that the Client Monitor Probe is accessible from the outside.
Data is filtered out.	The instrumentation was performed on a different application configuration. Rerun the instrumentation on the correct application.
Client Monitor Probe is not up and running.	Open a remote desktop connection to the Client Monitor Probe and invoke the Client Monitor Probe's URL (**(http://<RUM Client Monitor host>:<RUM Client Monitor port>**). The response should be "hello"; if not, restart the Client Monitor Probe.
Although there is activity on the application, there are no network requests.	Work with the application developers or operators to validate if the application is working against a backend server.

No Data in the Mobile Health Report but other Reports Display Data for this Application

Possible Causes	Possible Solutions
There is no data in the Requests and Domains panel.	Define requests manually (for example, from the session analyzer report) or wait until the classification starts defining requests.
It looks like only part of the requests appear in the requests table. This is because the URL parameters are blocked by default.	To unblock the URL parameters: 1. During instrumentation, enable the **Enable contact extraction from mobile application** option. 2. In APM, add the URL parameters to the **Sensitive Data** white list. See the Sensitive Data Area in the Real User Monitor section of the APM Application Administration Guide.

Missing Crash Events

Possible Causes	Possible Solutions
If during instrumentation it was discovered that ACRA crash reporting (which is a third-party tool) was already instrumented on your application, the RUM crash reporting was disabled.	Remove ACRA from the application and re-instrument the application.
The android crash engine detects Java related crashes only.	—
iOS crashes are reported in the following application launch if the application is re-launched within two hours of the crash.	—
Crash event is marked "Inactive".	Mark the Mobile Application Crash event "Active".
Android instrumentation failed.	Check the ApkInfuser log to determine why the instrumentation failed: In Engine: **<RUM>\tools\ApkInfuser\log** Or **<ApkInfuser>\log**

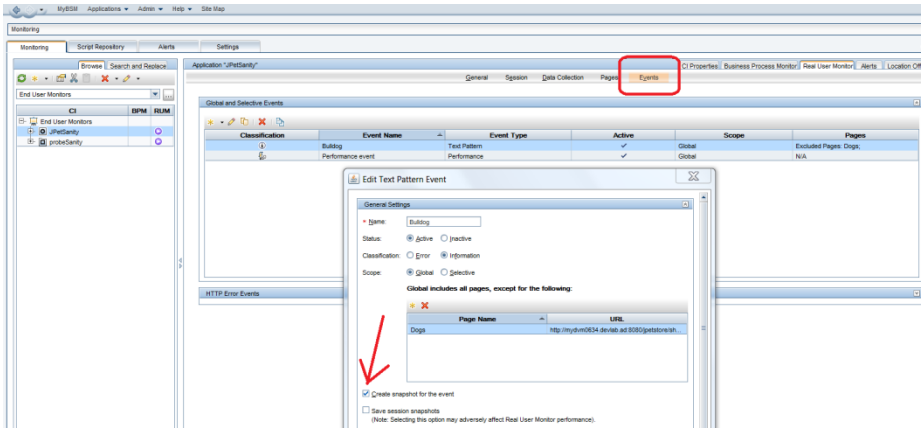
Chapter 2: Session Replay Troubleshooting

The section lists possible issues and their solutions that may arise when using RUM Session Replay.

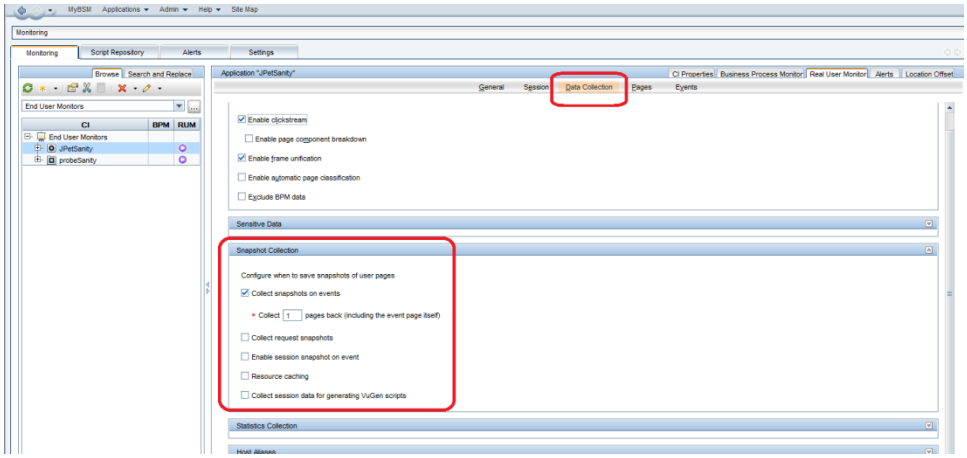
- "General Requirements" below
- "Logs from Session Replay Report" on the next page
- "Internet Explorer Configuration Settings" on page 9
- "Update for Configurations with Load Balancer" on page 11
- "Problematic Session Replay UI in Internet Explorer 10 and 11" on page 14
- "Poor Performance Environment" on page 14

General Requirements

- To view Session Replay, you must use Internet Explorer version 8 or later as your client browser when accessing APM.
- JRE must be installed on your client system.
- The first time you view Session Replay, ActiveX objects are installed on your client machine and are run each time you view Session Replay. Therefore, you must have the necessary permissions to install and run ActiveX objects on your client machine.
- ActiveX requires a root certificate in the client machine's local store of trusted root certification authorities. If such a certificate is not found, ActiveX prompts you to confirm the installation of a certificate called Rum-Probe CA for signing certificates. If you do not confirm installation of this certificate, you will be unable to view pages that were originally transferred over SSL.
- By default, RUM will take snapshots only for error events on configured applications. If you want to receive snapshots for other events, you have to configure them in the Events tab:



And make sure you collect the snapshot data:



Logs from Session Replay Report

To help troubleshoot problems, provide the following logs to R&D:

- **Applet logs**

  These logs are located on each machine that runs the report. These logs are located in the folder **C:\Users\<current user>\.hp_session_replay**.

  To retrieve troubleshooting information, in **log4j.properties**, set the log level to **DEBUG** and open the Session Replay report again.

  Each time you run a report a new folder is generated and contains logs for the report and cached page sources.

- **Logs on the RUM side**

  In **\RUM\conf\common\log4j.properties**, set the log level to **DEBUG**.

  Troubleshooting information will be saved in the following logs:

  - gatewayserver.log
  - rumproxy.log
  - repository.dataacesslayer.log
  - RUM_all.ejb.log

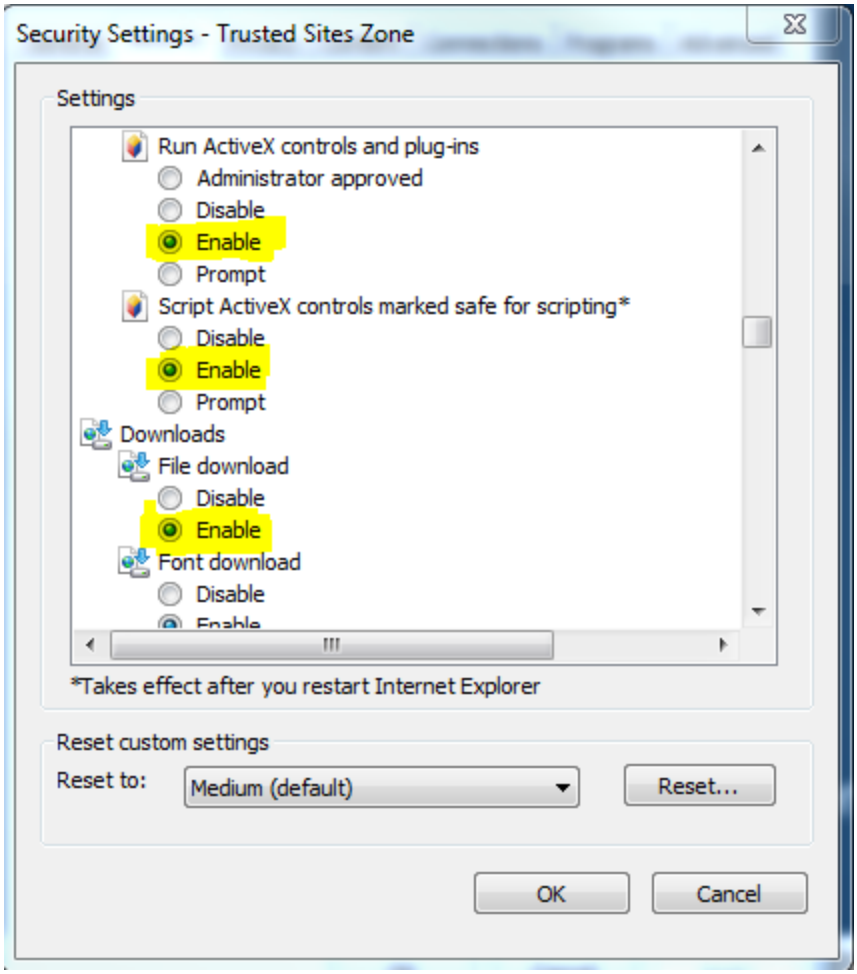
- **Logs on APM side**

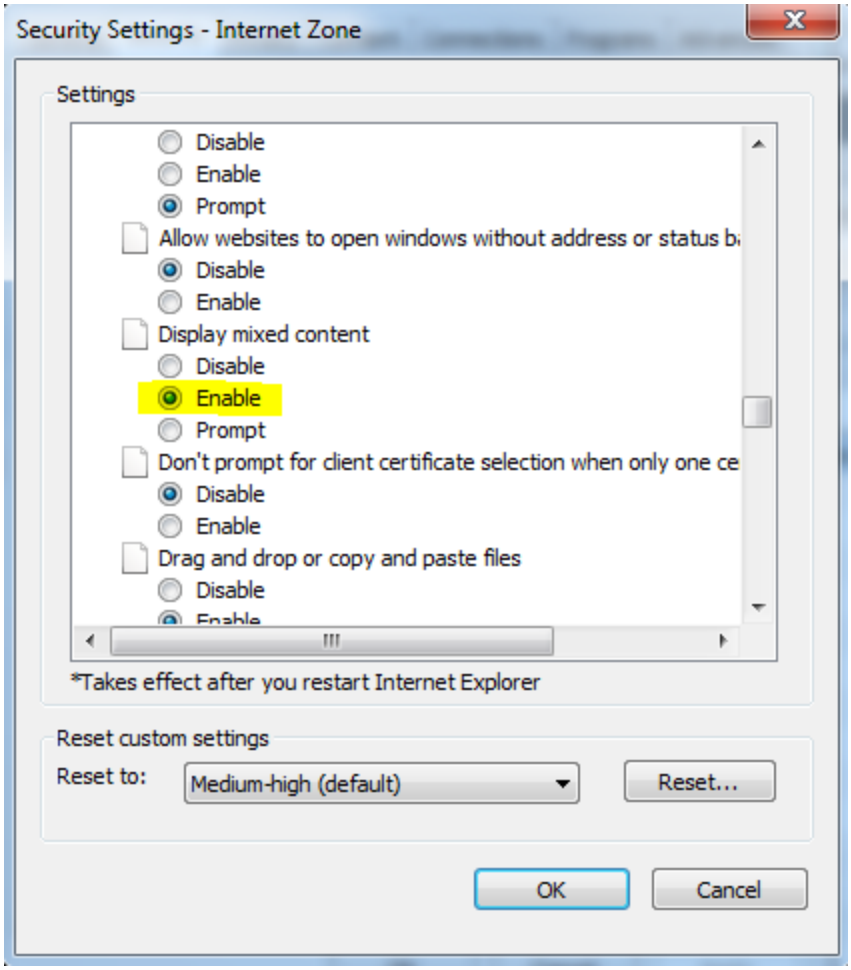
  The following logs are located in **\BSM\logs**

  - gdeGatewayClient.log
  - jboss-as-all.log

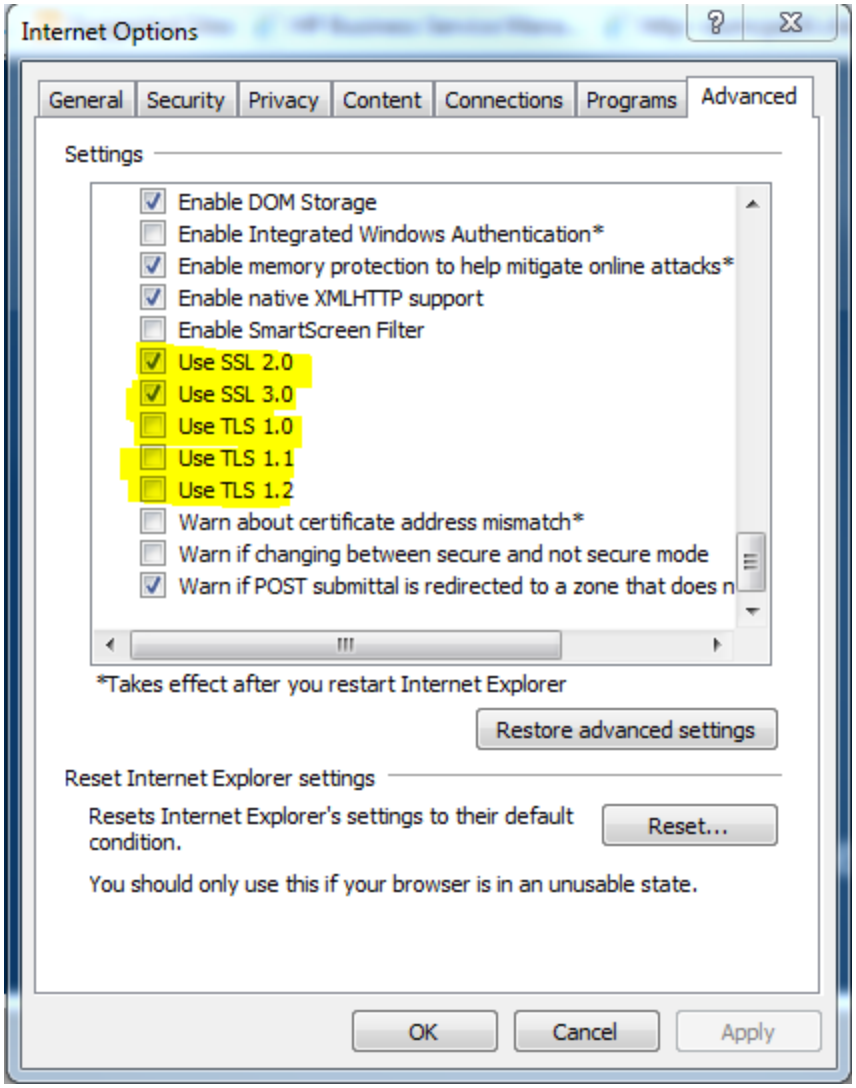
Internet Explorer Configuration Settings

1. In Internet Explorer, go to **Tools > Internet options**.
2. Click the **Security** tab.
3. Click **Custom Level**.
4. Enable the following Internet Explorer security settings:
   - Run ActiveX controls and plug-ins
   - Script ActiveX controls marked safe for scripting\
   - File download
   - Display mixed content

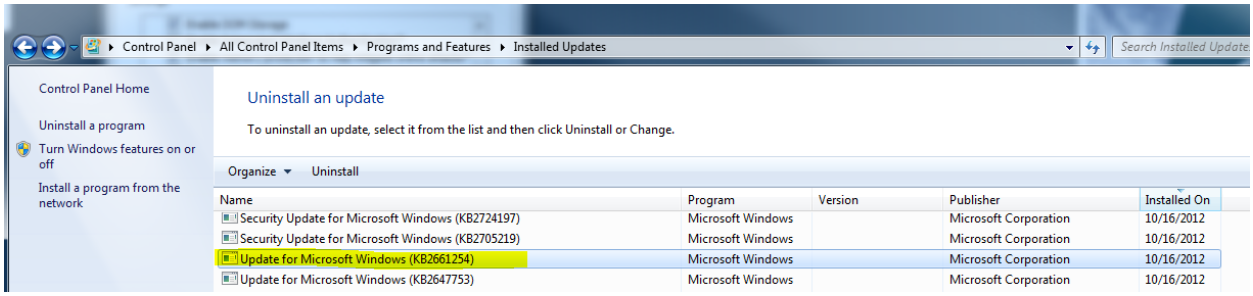




5. To retrieve snapshots for an https application, click the **Advanced** tab in the Internet Options dialog box and enable/disable the highlighted settings:



If you still get an error about a security certificate error for your ssl application, check if you installed an Update for Microsoft Windows KB2661254



If this update is installed, contact R&D to receive a hotfix for this problem.

Update for Configurations with Load Balancer

In some cases if you have a load balancer in https, an exception error may appear in sessionReplay.log:

```
2013-09-13 10:27:52,297 [AWT-EventQueue-2] (BaseAppletViewer.java:221) DEBUG - onAllTasksDone
2013-09-13 10:27:52,297 [AWT-EventQueue-2] (TasksExecutorImpl.java:154) ERROR - Error getting session
java.util.concurrent.ExecutionException: com.mercury.rum.engine.snapshotreplay.gateway.GatewayExecption:
org.apache.wink.client.ClientRuntimeException: java.lang.RuntimeException: javax.net.ssl.SSLHandshakeException:
sun.security.validator.ValidatorException: PKIX path building failed:
sun.security.provider.certpath.SunCertPathBuilderException: unable to find valid certification path to requested target
    at java.util.concurrent.FutureTask$Sync.innerGet(Unknown Source)
    at java.util.concurrent.FutureTask.get(Unknown Source)
    at com.mercury.rum.engine.snapshotreplay.utils.swingworker.SwingWorker.get(SwingWorker.java:558)
```

When the JRE is used to connect to an SSL Web server, or whenever it accepts a certificate, it must be able to validate and trust the certificate to establish the SSL session.

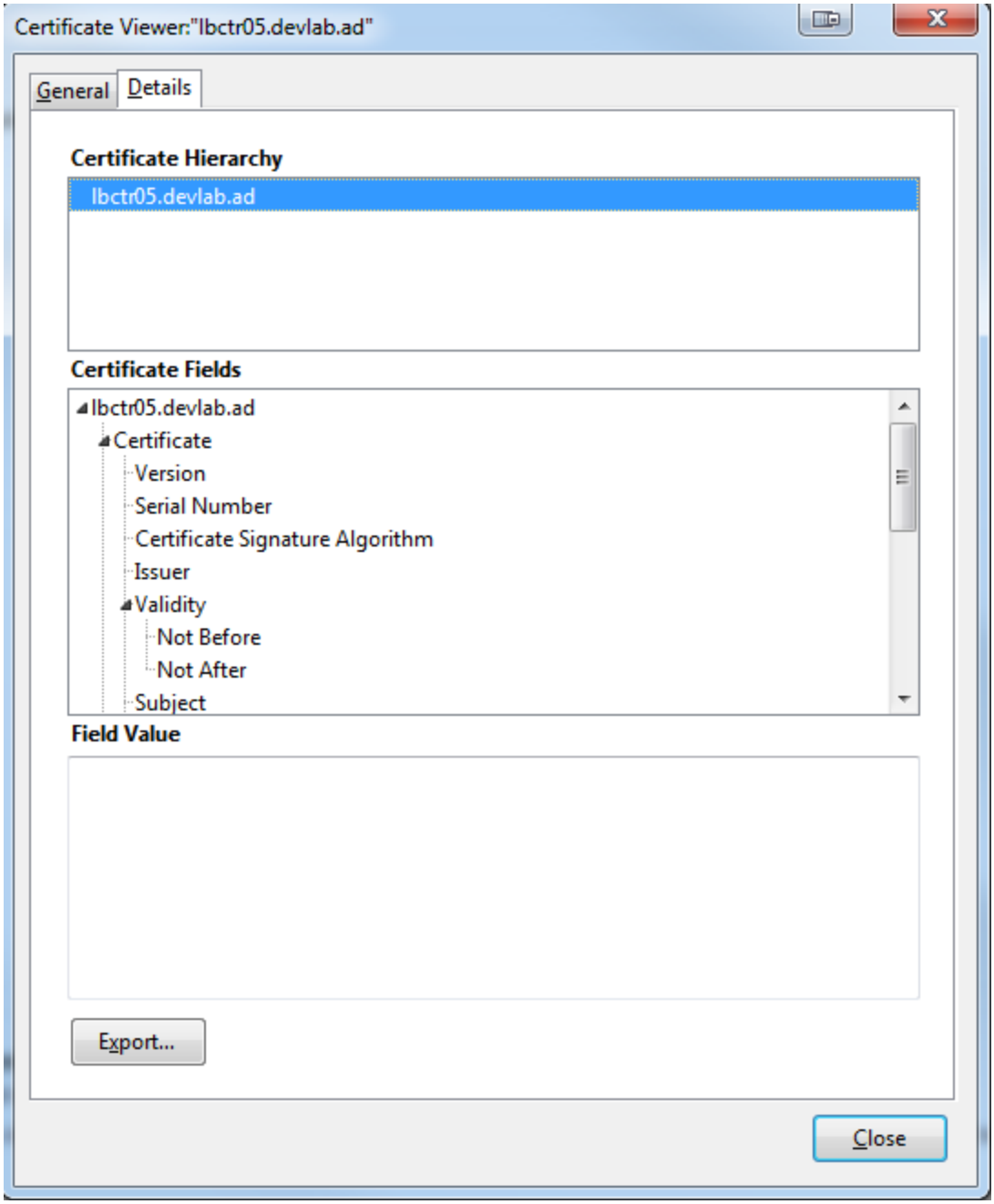
To trust and validate a certificate, JRE uses a trusted certificates store called a truststore. If the JRE can find a certificate in its truststore that is identical to the certificate requiring validation, validation is completed and the establishment of the session continues. Otherwise, the JRE will try to validate the digital signature of the certificate signed by the certificate issuer, using the issuing chain.

In order to validate a certificate signed by an issuer, or chain, the issuer's certificate must be included in the truststore used by the JRE. A certificate issuer is a Certification Authority (CA) that signs certificates. If you import the certificate of the CA into the JRE truststore, each certificate issued by this CA can be validated by the JRE.

When a session is started between the JRE and the Gateway Server, the Gateway Server's Web server sends the browser a server-side certificate that was issued by a Certification Authority (CA). If the certificate used by the Web server is issued by a known CA, the certificate can generally be validated by the JRE and no configuration is required. However, if the CA is not trusted by the JRE, the JRE must be configured to validate the server-side certificate that is sent.

Generally, it can be solved by installing the certificate on the Gateway machines.

1. In your browser, go to the https of the load balancer. Click the https certificate chain (a lock icon appears in Internet Explorer, or the domain name appears to the left of the URL in Firefox) and navigate the certificate hierarchy. At the top of the hierarchy there should be a Primary Root CA. This could be missing from your Java cacerts file.



2. Export the Root CA. Choose the **X.509 Certificate (DER)** type, so the exported file has a **der** extension.
3. On each gateway machine run the following two commands. (For this example, assume that the certificate is named ca.der.)

   <APM installation directory>\JRE64\bin\keytool -import -alias ca_root -file c:\ca.der -keystore <APM installation directory>\JRE64\lib\security\cacerts

   <APM installation directory>\JRE\bin\keytool -import -alias ca_root -file c:\ca.der -keystore <APM installation directory>\JRE\lib\security\cacerts

   You will be asked for a password. The default password of the truststore is **changeit**.

   Once you restart the gateway machines and the command has been run, the JRE is able to validate the certificate sent by the SSL Web server.

Problematic Session Replay UI in Internet Explorer 10 and 11

If when running Session Replay in Internet Explorer 10 or 11, the Session Replay UI looks as if the left frame is expanded over the entire page so you cannot see the snapshot content, you need to add the APM address to the Compatibility View list.

In Internet Explorer, open **Tools > Compatibility View Settings** and add the APM domain name to the list and run the browser in Compatibility mode.

Or

Add the following line of code in
C:\BSM\AppServer\webapps\site.war\jsps\eum\snapshot\SessionReplayFrames.jsp

```
<%@ taglib prefix="swing" tagdir="/WEB-INF/tags/swing" %>
<%
    String getRequestURL = request.getRequestURL().toString().replaceAll("Session
    getRequestURL = getRequestURL.substring(getRequestURL.indexOf("sessionApplet.
    String getQueryString = request.getQueryString();
    String targetURL = getRequestURL + "?" + getQueryString;
%>

<html>
<head>
    <meta http-equiv="X-UA-Compatible" content="IE=EmulateIE9">
    <script>
        function createTargetUrl(fromJsp, toJsp) {
            var location = window.location;

            var protocol = location.protocol;   // includes ':'
            var host = location.host;           // host name and port
```

You do not need to restart APM after these changes.

Poor Performance Environment

If the APM environment is running slowly, (for example, if you connect to APM via Remote Desktop Protocol (RDP) and the UI response is slow) it may take time to open a pop-up window after you press **Session Replay**. In this case, you may get an error message about an ActiveX problem.

To resolve this problem, increase the timeout and maxFailedAttempts values in

C:\BSM\AppServer\webapps\site.war\jsps\eum\snapshot\starterApplet.jsp

```
var failedAttempts = 0;
var maxFailedAttempts = 3;

if (setProxy != null) {
    try {
        failedAttempts = 0; // in case SR button is clicked more than once, we need to start over.
        setTimeout(callCopyCookie, 5000);
    }
    catch (e){
        <%-- Session Replay ActiveX did not start properly, will try to open Session Replay anyway. --%>
        alert('<bean:message key="session.replay.proxy.start.error" bundle="eum-reports" />');
        callCopyCookie();
    }
}

function callCopyCookie() {
    try {
        retVal = setProxy.CopyCookie(wantedTitle, targetUrl);

        if (isDebugEnabled()) {
            sendDebugLog("DEBUG message from starterApplet.jsp: callCopyCookie() - " + " Current retVal is: " + retVal);
        }
    }
    catch(e) {
        failedAttempts = failedAttempts + 1;
        if (failedAttempts < maxFailedAttempts) {
            setTimeout(callCopyCookie, 5000);
        }
        else {

            sendErrorLog("Exception in starterApplet.jsp: callCopyCookie() - " + e.message + " Current retVal is: " + retVal);
            <%-- Session Replay ActiveX did not work properly, will try to open Session Replay anyway. --%>
            alert('<bean:message key="session.replay.proxy.work.error" bundle="eum-reports" />');
        }
    }
}
```

You do not need to restart APM after these changes.

Chapter 3: Server Collector Troubleshooting

The section describes possible issues and solutions may arise when using the RUM Server Collector and includes:

- "SSL Issues" below
- "Connectivity Issues" on page 19
- "RUM Server Collector Workflow" on page 23
- "Traffic Transferred to RUM Probe" on page 24
- "UTC Time Difference" on page 25
- "Port Number" on page 25
- "Connect RUM Server Collector to Several Probes" on page 25
- "Connect Several RUM Server Collectors to a Single RUM Probe" on page 26
- "Starting the RUM Probe and RUM Server Collector" on page 27
- "Reconnecting to a RUM Server Collector" on page 28
- "TCP Offloading" on page 28

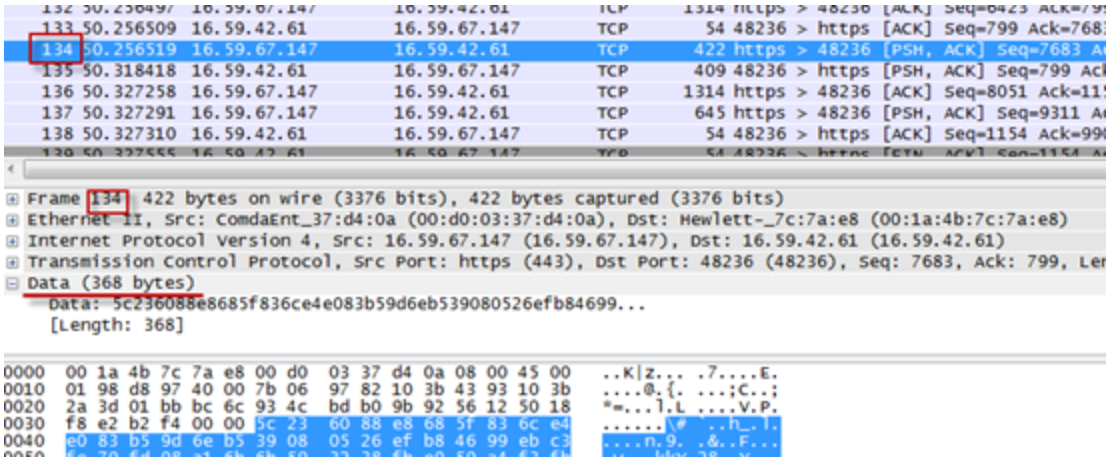
SSL Issues

This section includes:

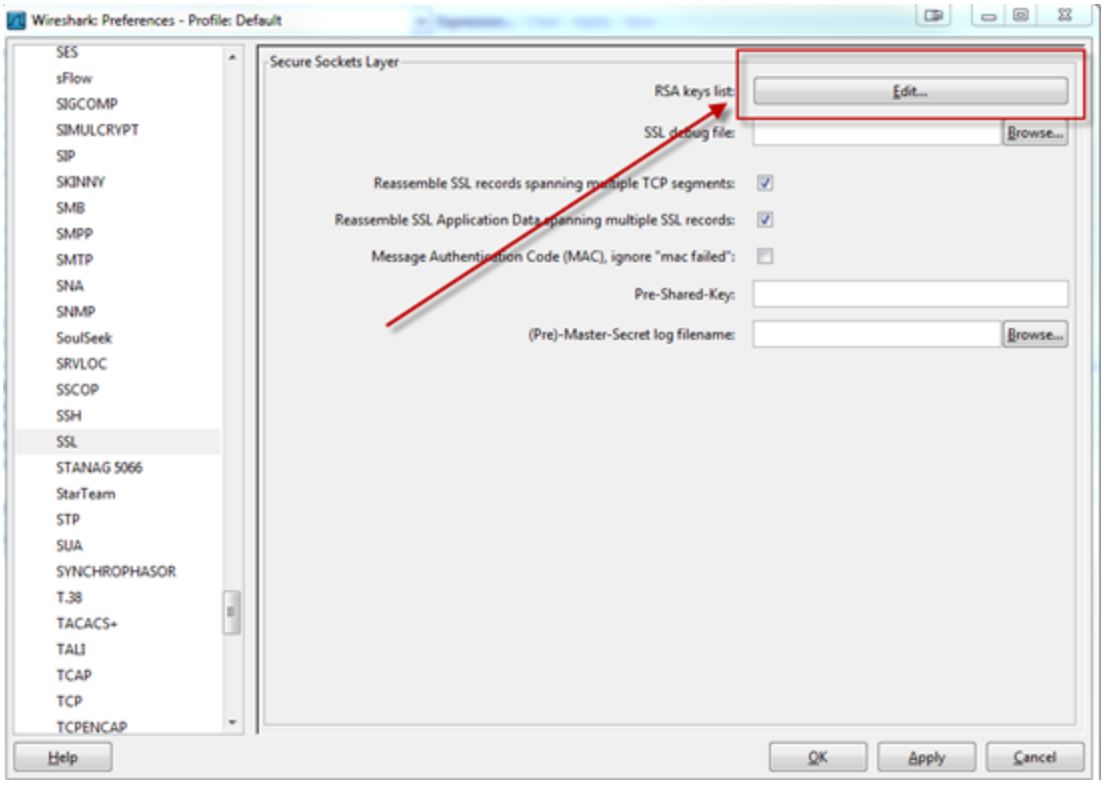
- "How to Decrypt SSL Traffic in Wireshark" below
- "Switch Off SSL Communication Encryption" on page 18

How to Decrypt SSL Traffic in Wireshark

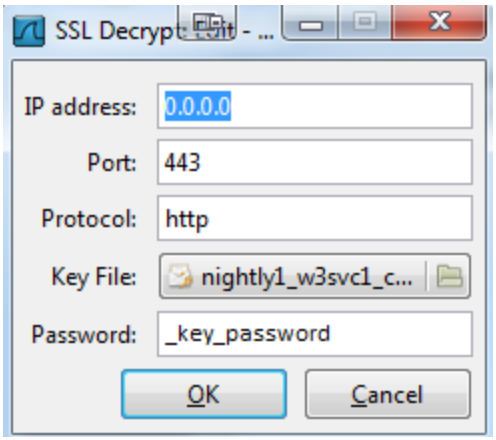
1. Open the pcap file (or access SSL traffic in another way, for example, by using a remote agent or local device. Wireshark displays encrypted traffic. It does not have a server key to decrypt SSL.



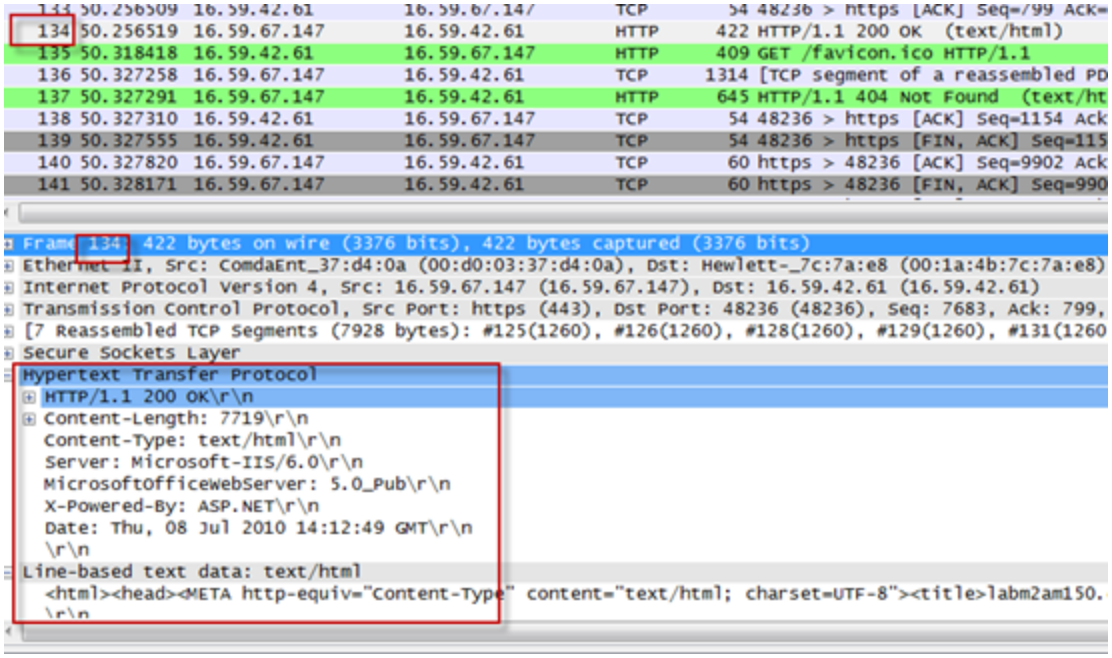
2. Click **Edit > Preferences** and select **SSL protocol > RSA keys list > Edit**.



3. Add a key file with password.



4. View the results in Wireshark.



Switch Off SSL Communication Encryption

The following provides instructions for switching off the SSL communication encryption between the RUM Server Collector and the RUM Probe

1. Stop the RUM Probe.
2. Stop the RUM Server Collector services.
3. On the Probe side, in **<HPRUMProbe>\etc\rum_probe\rpsecurity.conf**, set **[servercollector]/ collector_enable_ssl** to **false**.

```
[servercollector]
#whether to use ssl encryption
collector_enable_ssl              false

# rproxy server authentication
collector_ca_cert                 /etc/rum_probe/rum-collector-ca.crt
```

4. On the RUM Server Collector side, in **<RUMSC>\etc\rum_collector\collector.conf**, set **[security] use_ssl=false**.

```
[security]
# Use SSL manage/data connections. If set - this will violate backward
# compatibility with wireshark. Only HP RUM Probe will be able to connect
use_ssl=false

# optional client authentication
ssl_ca_file=/etc/rum_collector/rum-collector-ca.crt

# mandatory server key/certificate.
# the certificate must be signed by a CA to pass optional server authentication on client
ssl_key=/etc/rum_collector/rum-collector-server.key
ssl_cert=/etc/rum_collector/rum-collector-server.crt
```

5. Start the RUM Server Collector service.
6. Start the RUM Probe.

Connectivity Issues

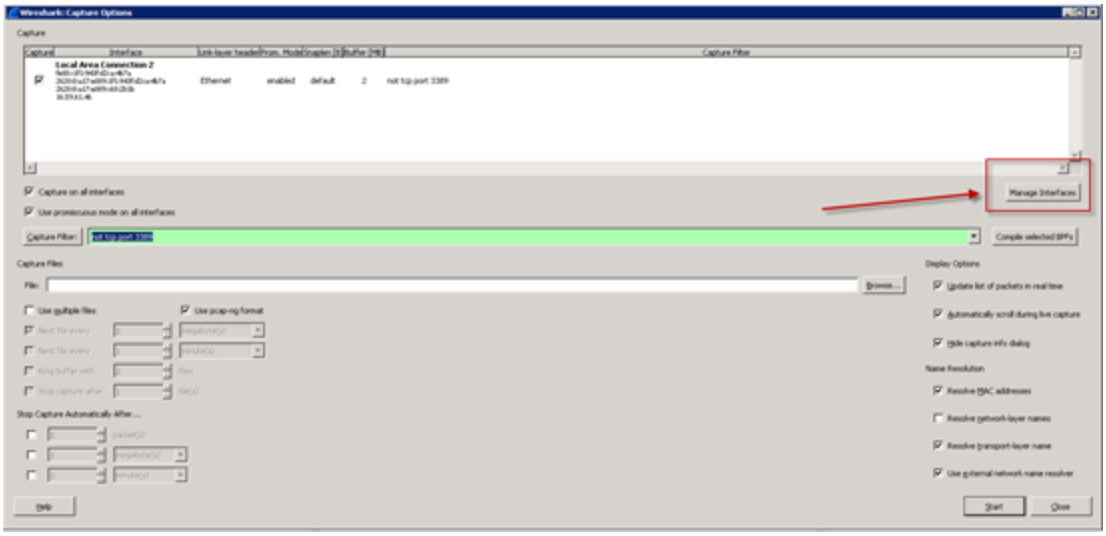
This section includes:

- "Connect Wireshark to RUM Server Collector" below
- "Troubleshooting Low Connectivity between RUM Probe and RUM Server Collector" on page 21

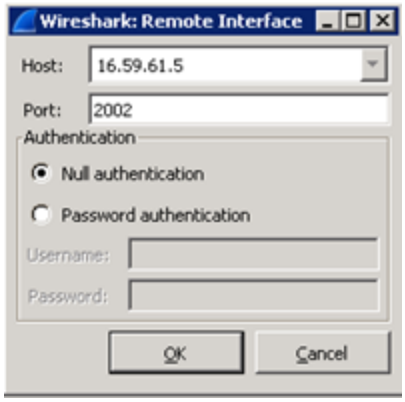
Note: Disconnect the RUM Probe from the RUM Server Collector prior to connecting Wireshark to the RUM Server Collector instance to reduce the load on the RUM Server Collector and implicitly on the web/application server.

Connect Wireshark to RUM Server Collector

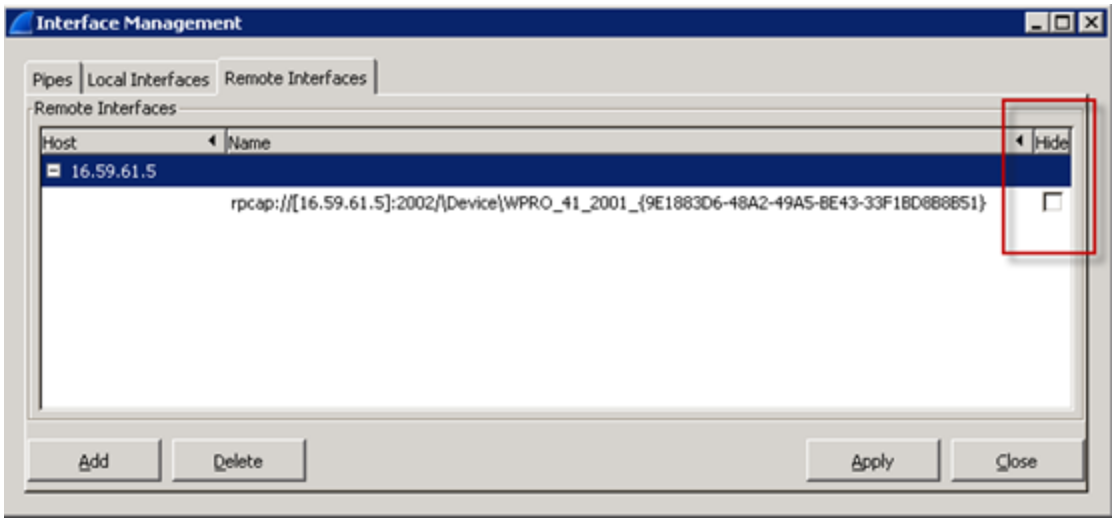
1. Switch off SSL encryption on the RUM Server Collector (see How to Switch Off SSL Encryption of Communication between RUM Server Collector and RUM Probe on page 6. Do not switch off the RUM Probe.)
2. On a different server, typically the server where the RUM probe is installed, open Wireshark.
3. Click the **Capture Options** button (or **Capture > Interfaces > Options**).
4. Click the **Manage Interfaces** button.



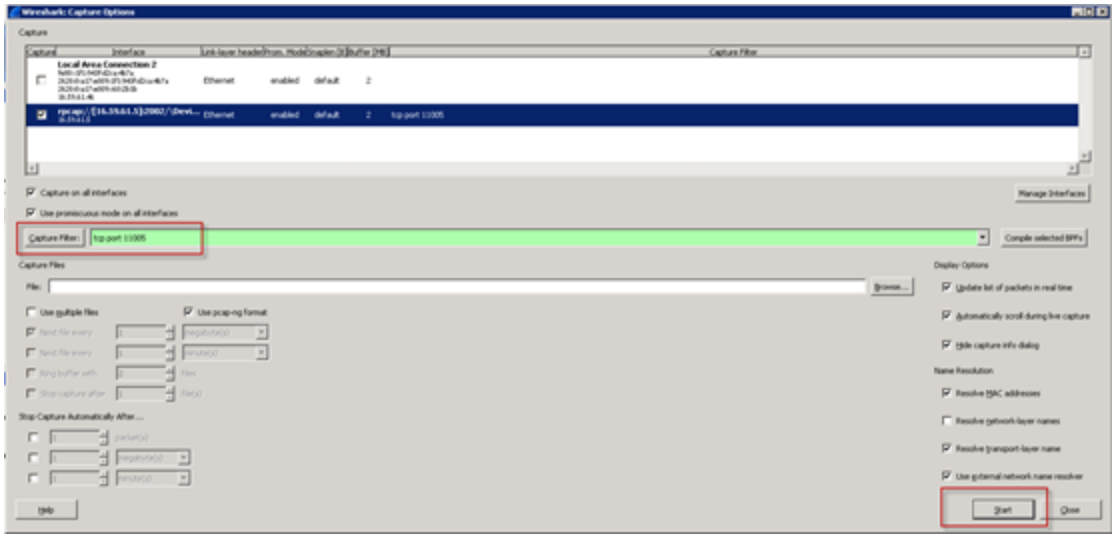
5. Select **Interface Management > Remote Interfaces > Add**.
6. Enter the **Host** and **Port** number.



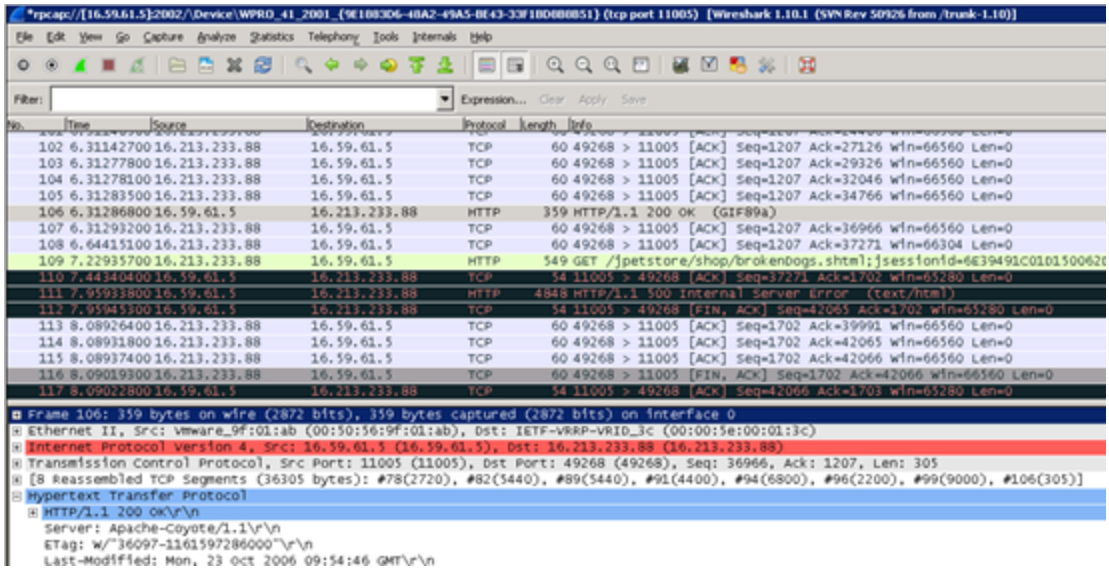
7. Click **OK** to add the remote interface. The new remote interface appears in the Remote Interfaces tab of the Interface Management window.



8. To hide an interface, click the **Hide** checkbox.
9. To display only concrete traffic (for example, web traffic), set filters for the remote interface. In the following example, we are interested in traffic from the Web Server (http://<server name>:11005/jpetstore/), therefore, in the Capture Filter field, type **tcp port 11005**.



10. Select the remote device and click **Start** to capture the traffic.



The traffic the RUM Server Collector transfers to the RUM Probe is displayed.

IN the RUM Server Collector logs, the following information about the connection to Wireshark appears.

```
2013-11-26 15:43:14 [5392] INFO  main <..\..\rproxy\rproxy.cpp:338> - Waiting for
incoming connections 0.0.0.0:2002

2013-11-26 15:43:16 [5288] INFO  daemon <..\..\rproxy\daemon.cpp:1300> - Started
client data connection 16.59.61.46:60518<-->16.59.61.5:2002 Device: \Device\WPRO_41_
2001_{9E1883D6-48A2-49A5-BE43-33F1BD8B8B51}

2013-11-26 15:44:06 [5584] INFO  daemon <..\..\rproxy\daemon.cpp:438> - Finished
client control connection: 16.59.61.46:60517<-->16.59.61.5:2002

2013-11-26 15:44:06 [5012] INFO  daemon <..\..\rproxy\daemon.cpp:1275> - Finished
packet sending thread

2013-11-26 15:44:06 [5288] INFO  daemon <..\..\rproxy\daemon.cpp:1324> - Finished
client data connection 16.59.61.46:60518<-->16.59.61.5:2002 Device: \Device\WPRO_41_
2001_{9E1883D6-48A2-49A5-BE43-33F1BD8B8B51}
```

Troubleshooting Low Connectivity between RUM Probe and RUM Server Collector

When the RUM Server Collector is working, it captures traffic from configured NIC(s), encrypts it (or not depending on the configuration), then wraps it with RPCAP protocol and sends it to the RUM Probe. This means that the total amount of traffic generated on the RUM Server Collector machine by the App/Web server is duplicated and some overhead is added (in general).

Therefore, if have 100 Mb/s of traffic on the NIC, using the RUM Server Collector (assuming that it will transfer all traffic from NIC, for example when in Discovery mode) the following is true:

100 Mb/s of native traffic + 100 Mb/s RUM SC transferred to RUM Probe + 10% overhead (RPCAP and SSL wrapping) ≈ 210 Mb/s

This is a not a problem when you have a 1G or 10G network card, but in general you need to verify that your bandwidth is enough to send additional traffic to the RUM Probe. If necessary, you can use an additional network card to deliver captured traffic to the RUM Probe.

If the RUM Server Collector is not ready to send packets to the network, it drops them and publishes the statistics in a log. There is no mechanism to guarantee delivery of the packets.

```
[delivery]

# Threshold fo packet queue. Packets are dropped if queue size reaches the threshold,
default is 10000

packet_queue_max_size=10000

# Thresholds for logging of dropped packets

# drop_packets_threshold specifies amount of dropped packets on which warning message
is logged, default is 10000

# drop_packets_timeout sets time duration in seconds, on which to log warning message
about dropped packets if any, default is an hour = 3600

drop_packets_threshold=10000

drop_packets_timeout=3600
```

where:

packet_queue_max_size – Packets are dropped if the queue size reaches the threshold (default is 10000).

drop_packets_threshold and drop_packets_timeout – By default, the RUM Server Collector collects statistics about dropped packets during a one hour period. If the amount of dropped packets exceeds **drop_packets_threshold**, the information is published in the log.

In order to troubleshoot connectivity problems, you can adjust the **drop_packets_threshold** parameter so that the information that some packets are dropped is documented in the log.

Setting Device Names on Windows

The best practice for using RUM Server Collector is to configure listening concrete devices on the remote server.

1. Set the RUM Probe to listen to all devices:

   ```
   device rpcap://[16.59.61.5]:2002/

   device rpcap://server_host:2002/
   ```

   When the RUM Probe and RUM Server Collector are connected, all available devices are listed in the RUM Server Collector log:

   ```
   2013-02-20 11:09:12 [4568] INFO  daemon <..\..\rproxy\daemon.cpp:1252> - Started
   client data connection 16.59.57.35:49618<-->16.59.61.5:2002 Device: \Device\WPRO_
   41_2001_{32DE4B0B-6C58-4C16-B822-FC353B50675A}
   ```

```
2013-02-20 11:09:12 [2884] INFO  daemon <..\..\rproxy\daemon.cpp:1252> - Started
client data connection 16.59.57.35:49617<-->16.59.61.5:2002 Device: \Device\WPRO_
41_2001_{FE45A5F9-A4FC-4F7A-8C18-21097D13F3DB}

2013-02-20 11:09:12 [4428] INFO  daemon <..\..\rproxy\daemon.cpp:1252> - Started
client data connection 16.59.57.35:49621<-->16.59.61.5:2002 Device: \Device\WPRO_
41_2001_{43079F95-A18E-42DB-8F43-131B146596CF}

2013-02-20 11:09:13 [2176] INFO  daemon <..\..\rproxy\daemon.cpp:1252> - Started
client data connection 16.59.57.35:49623<-->16.59.61.5:2002 Device: \Device\WPRO_
41_2001_{BAA5AADA-3787-4DF7-A64F-DE89C8307A9C}

2013-02-20 11:09:13 [4512] INFO  daemon <..\..\rproxy\daemon.cpp:1252> - Started
client data connection 16.59.57.35:49624<-->16.59.61.5:2002 Device: \Device\WPRO_
41_2001_{D4A27079-3385-47FE-8EF6-40365891306E}

2013-02-20 11:09:13 [4476] INFO  daemon <..\..\rproxy\daemon.cpp:1252> - Started
client data connection 16.59.57.35:49653<-->16.59.61.5:2002 Device: \Device\WPRO_
41_2001_{799A034E-3E22-4E9E-AEEE-3149DE866B34}

2013-02-20 11:09:13 [5128] INFO  daemon <..\..\rproxy\daemon.cpp:1252> - Started
client data connection 16.59.57.35:49656<-->16.59.61.5:2002 Device: \Device\WPRO_
41_2001_{8D714585-93CD-497A-9E74-9C925B8BAE1C}
```

2. Copy the required device from the log and configure it on the RUM Probe side.

   ```
   device rpcap://[16.59.61.5]:2002/\Device\WPRO_41_2001_{32DE4B0B-6C58-4C16-B822-
   FC353B50675A}
   ```

The device name is created by the driver. The same network device can be named differently. Following is an example of the same network device name differently:

- `\Device\WPRO_41_2001_{32DE4B0B-6C58-4C16-B822-FC353B50675A}`
- `\Device\WPRO_41_2001_{32DE4B0B-6C58-4C16-B822-FC353B50675A}`

The naming format from collector.log must be used when specifying the remote network interface manually on the RUM Probe side.

To detect which network interface corresponds to which device name, type the following:

- (Windows) **c:\Program Files\Wireshark\tshark.exe –D**
- (Linux) **ifconfig**

RUM Server Collector Workflow

1. When the RUM Server Collector starts, it publishes the following in the logs:

   ```
   Starting HPRUMServerCollector

   Waiting for incoming connections [::]:2002

   Waiting for incoming connections 0.0.0.0:2002
   ```

This means that the RUM Server Collector started and waits for an IPv4/IPv6 connection on any existing network interface on the server. You can indicate which address to use in the collector.conf file:

```
[general]

# Address the daemon has to bind to

# Default: it binds to all local IPv4 and IPv6 addresses.

address=16.59.61.5
```

The RUM Server Collector will accept connections on the 16.59.61.5 IP address only:

```
Waiting for incoming connections 16.59.61.5:2002
```

2. The RUM Server Collector receives a connection request. In order for the request to be granted, the request must be within the terms of the RUM Server Collector control connection:

```
Started client control connection: 16.59.61.46:60342<-->16.59.61.5:2002

Finished client control connection: 16.59.61.46:60342<-->16.59.61.5:2002
```

Therefore, the following occurs:

a. The system checks the client to ensure that it is an acceptable client according to the configuration.
b. The system performs a full server and client authentication using the SSL handshake.
c. The system tests the local devices to ensure that they are sniffable, and available devices are listed.

The client connection is completed

3. If the previous step succeeded, new connections are created and control and data is transferred.

```
Waiting for incoming connections 16.59.61.5:2002

Started client control connection: 16.59.61.46:60343<-->16.59.61.5:2002

Waiting for incoming connections 16.59.61.5:2002

Started client data connection 16.59.61.46:60344<-->16.59.61.5:2002 Device:
\Device\WPRO_41_2001_{9E1883D6-48A2-49A5-BE43-33F1BD8B8B51
```

Traffic Transferred to RUM Probe

By default, the RUM Server Collector sets a filter on transferred traffic according to the port number of the monitored application. For example, if you are monitoring an HTTP application, by default the RUM Server Collector filters traffic from the connected network device by port 80 (or 8080). This behavior is due to the following reasons:

- Reduce resources consumption on unnecessary traffic (network, CPU, memory) both at the RUM Probe and RUM Server Collector side.
- Reduce performance impact.

When you enable the Discovery mechanism (Tier or Traffic Discovery), the RUM Server Collector disables all filters and transmits all traffic from the configured devices.

Therefore, enabling\disabling Discovery can increase\reduce traffic from the RUM Server Collector to the RUM Probe.

UTC Time Difference

The RUM Probe relies on the fact that the UTC time on the RUM Probe and RUM Server Collector machines are almost the same, differing no more than several minutes. If the difference was greater than several minutes, all RUM Probe channels on the engine side could be rejected. The RUM Probe has a detection mechanism for this issue. If an error message appears in the RUM Probe's log about a UTC time difference, you should verify the system time on the RUM Probe and RUM Server Collector machine.

```
ERROR sniffer.device - Too big time shift between Probe and SC machine detected:
02:00:09!
For proper functioning of RUM you need to synchronize UTC times on Probe and SC
machines.
```

Port Number

The RUM Server Collector opens a port number on the server (port 2002 by default). Sometimes it can be blocked by an antivirus or firewall. If it is blocked, the RUM Probe cannot establish a connection to the RUM Server Collector and publishes the following message in the log file:

```
Could not obtain NIC list from remote host rpcap://[16.59.61.5]:6666/:
Is the server properly installed on 16.59.61.5?
connect() failed: An attempt was made to access a socket in a way forbidden by its
access permissions.  (code 10013)
```

At the same time, the RUM Server Collector starts and waits for a connection:

```
Starting HPRUMServerCollector

Waiting for incoming connections 16.59.61.5:6666
```

Connect RUM Server Collector to Several Probes

We do not recommend connecting one RUM Server Collector to several RUM Probes due to the impact on performance.

It is theoretically possible to connect one RUM Server Collector to several RUM Probes. However, since resource consumption by the RUM Server Collector is multiplied, the impact on App/Web server performance increases.

Connect Several RUM Server Collectors to a Single RUM Probe

You can connect a single RUM Probe to several RUM Server Collectors and configure the RUM Probe to simultaneously use local devices and several RUM Server Collectors. For example:

```
[collector]

device all

device rpcap://LinuxHost:2002/eth0

device rpcap://[16.59.61.5]:2002/ \Device\WPRO_41_2001_{9E1883D6-48A2-49A5-BE43-33F1BD8B8B51}
```

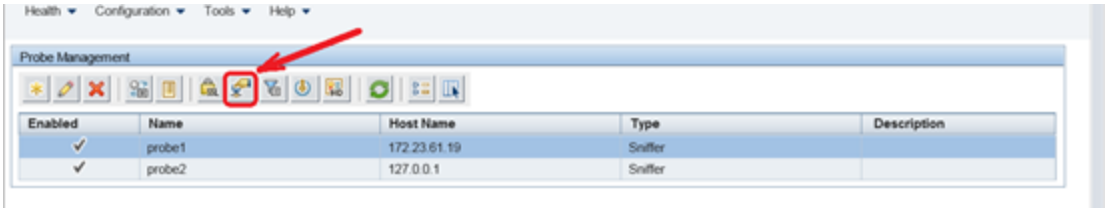
However, in this scenario, be aware of the following:

- Since the Interface Manager of the engine's web console does not work properly with the RUM Probe's remote interfaces, you need to manually edit **<RUM>\conf\configurationmanager\Beatbox_<Probe>_Const_Configuration.xml** for the corresponding RUM Probe.

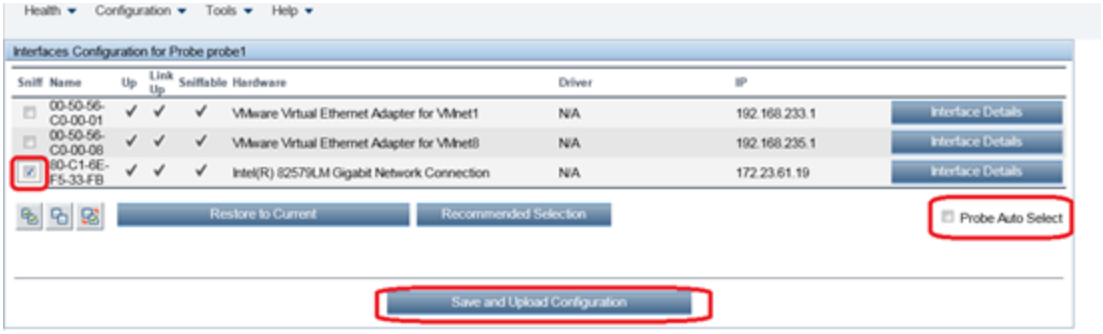
  For example, if the Engine is connected with two RUM Probes and you want to connect the first Probe to the RUM Server Collector running on one machine and monitor application traffic on NIC eth1. You also want to connect the RUM Proble to a different Server Collector that is listening to all network interfaces.

  a. From the Engine web console, click **Configuration > Probe Management** and select the corresponding RUM Probe.

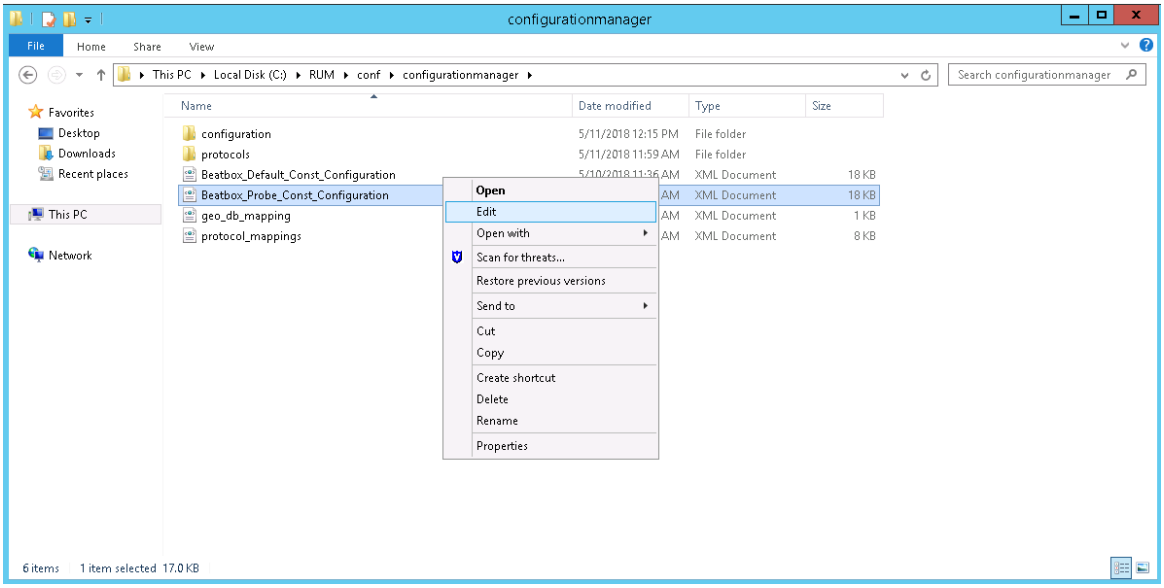
  b. Click the **Interface Configuration** button on the toolbar.



  c. Select any single interface and click **Save and Upload Configuration**.



  d. Under **<RUM>/conf/configurationmanager/** find the new file **c:\RUM\conf\configurationmanager\Beatbox_<PROBE_NAME>_Const_Configuration.xml** which corresponds to probe1, edit it, and replace the existing local interface with remote interfaces.




```
<?xml version="1.0" encoding="UTF-8"?><consts>

<collector><![CDATA[

[collector]

device rpcap://[IP address 1]:2002/eth1

device rpcap://[IP address 2]:2002/

device \Device\NPF_{A5EE5282-058B-4DF1-97D9-92ECB3A4637B}

]

]></collector>

</consts>
```

e. From the Engine's web console, synchronize the configuration. Probe1 gets its own Server Collector configuration. Probe2 uses the default/local network interfaces for capturing.

- Using the RUM Server Collector with a RUM Probe increases resource consumption. The RUM Probe wastes time and resources connecting, unwrapping, encoding, and handling traffic from the RUM Server Collector. The impact of using one RUM Server Collector is not significant. The RUM Probe capacity and resource consumption remains almost the same. The more RUM Server Collector instances used, the more the resource consumption on the RUM Probe increases.

Note: The flow of one RUM Probe connected to 50 RUM Server Collectors was evaluated. This scenario worked, but the RUM Probe performed very slow. We strongly do not recommend such extreme work flows.

Starting the RUM Probe and RUM Server Collector

It is important that you start the RUM Server Collector service before the RUM Probe. In general, the RUM Server Collector should be running and accessible from the RUM Probe each time the RUM Probe's

configuration is updated, except for cases when the configuration update does not require reestablishing or capturing devices.

Reconnecting to a RUM Server Collector

If an established connection to the RUM Server Collector is broken, the RUM Probe attempts to reconnect to the RUM Server Collector for a preconfigured time period. This time period is configured in **rpsecurity.conf: [servercollector]/maximum_reconnect_duration**. The default time period is one hour.

```
[servercollector]

…

# threshold for reconnecting attempts, in seconds

maximum_reconnect_duration     3600
```

Note: The RUM Probe cannot reconnect when it first starts or after a configuration update.

If you encounter an error when updating the RUM configuration, it might be caused by a connectivity issue between the RUM Probe and the RUM Server Collector.

1. Inspect the RUM Probe's logs to verify that there is no error message regarding a failure to connect to the RUM Server Collector.

   ERROR sniffer.device <..\..\..\collector\sniffer\DeviceSniffer.cpp:428> - Could not obtain NIC list from remote host rpcap://[172.23.61.19]:2002/: Is the server properly installed on 172.23.61.19? connect() failed: No connection could be made because the target machine actively refused it. (code 10061)

   Or

   ERROR collector <..\..\collector\MasterConfig.cpp:1006> - Cannot initialize PacketSniffer socket. Please check the configuration of following device: rpcap://[172.23.61.19]:2002/eth1 :[collector] WARN collector <..\..\collector\main.cpp:356> - Errors occurred during startup (RUM(r) probe Capture v9.2.3.150.0273 Win64 r0)

2. Verify that the RUM Server Collector runs and is accessible from the RUM Probe machine.
3. Restart the RUM Probe.

TCP Offloading

Note: This issue affects only RUM Probe versions prior to 9.23. Starting from version 9.23, the RUM Probe sets the global_skip_checksum to **true** when a remote network interface is recognized in the configuration.

Since the RUM Server Collector runs on the same machine as the Web/AppServer, the captured traffic could contain zero TCP checksums, depending on the TCP offload engine (TOE) configured on that machine. If the TOE is enabled, the TCP checksums of the network packets are calculated on a NIC before the packets are

sent to network. Therefore, when the packets are captured by the RUM Server Collector, or any other sniffing software, the TCP checksums have not been calculated yet.

By default, the RUM Probe rejects packets with the wrong TCP checksums. Therefore, we need to tell the RUM Probe that the TCP checksums should not be checked. This can be done by editing the **[global]** section of the **\RUM\conf\configurationmanager\Beatbox_<Sniffer Probe name>_Const_Configuration.xml** file:

```
[global]

…

global_skip_checksum true

…
```

Chapter 4: Performance Probe Packet Loss

Problem: When running RUM on a Linux operation system in a high load situation, packet loss occurs in the Sniffer probe due to a bad checksum.

To determine if there is packet loss:

1. Go to **rumwebconsole > health > System health**.
2. Click the Real User Monitor Sniffer Probe Host **hostname.fqdn** and determine the value of the **Packets lost total sum** parameter.

Solution:

On the Probe side:

1. Stop the probe service.
2. Access the following file: **/etc/rum_probe/capture.conf**.
3. Update the **global_skip_checksum false** parameter to **global_skip_checksum true**.
4. Save the **/etc/rum_probe/capture.conf** file.
5. Restart the probe service.

On the hardware side:

1. Make sure that you have root permissions.
2. Edit the **/etc/sysctl.conf** file and update the values of the following parameters:
   net.core.optmem_max =1073741824
   net.core.rmem_default =1073741824
   net.core.rmem_max =1073741824
   net.core.wmem_max =1073741824

Chapter 5: High CPU Utilization

Problem: High CPU utilization during peak load on the sniffer probe.

Solution: In case of high CPU utilization by the sniffer probe, make the following change under global section in RUM Engine beatbox configuration file and sync the configuration:

Change the **processor_threads** value based on your hardware capabilities. By Default, the **processor_ threads** value will be 4.

Send Documentation Feedback

If you have comments about this document, you can contact the documentation team by email. If an email client is configured on this system, click the link above and an email window opens with the following information in the subject line:

Feedback on RUM Troubleshooting (Real User Monitor 9.51)

Just add your feedback to the email and click send.

If no email client is available, copy the information above to a new message in a web mail client, and send your feedback to docs.feedback@microfocus.com.

We appreciate your feedback!