



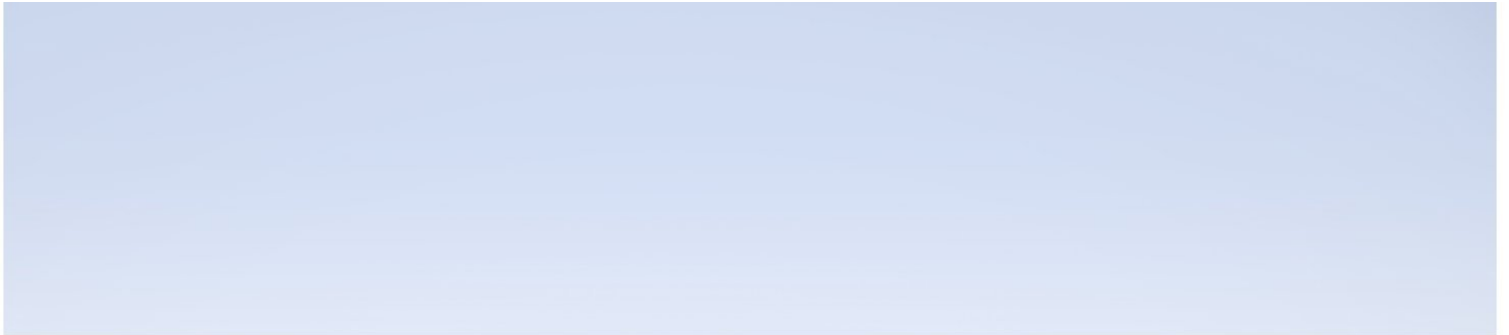
Hewlett Packard
Enterprise

Application Performance Management

Version 9.40, Released August 2017

Getting Started With RUM - Best Practices

Published August 2017



Legal Notices

Warranty

The only warranties for Hewlett Packard Enterprise 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. HPE 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 HPE 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 2005 - 2017 Hewlett Packard Enterprise Development LP

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.

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:
<https://softwaresupport.hpe.com/group/softwaresupport/search-result?keyword=>.

This site requires an HPE Passport account. If you do not have one, click the **Create an account** button on the HPE Passport Sign in page.

Support

Visit the HPE Software Support website at: <https://softwaresupport.hpe.com>

This website provides contact information and details about the products, services, and support that HPE Software offers.

HPE Software 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 website to:

- Search for knowledge documents of interest
- Submit and track support cases and enhancement requests
- Download software patches
- Manage support contracts
- Look up HPE 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 HPE Passport user and sign in. Many also require a support contract. To register for an HPE Passport ID, go to <https://softwaresupport.hpe.com> and click **Register**.

To find more information about access levels, go to:

<https://softwaresupport.hpe.com/web/softwaresupport/access-levels>

HPE Software Integrations and Solutions

Visit the Integrations and Solutions Catalog at <https://softwaresupport.hpe.com/km/KM01702731> to explore how the products in the HPE Software catalog work together, exchange information, and solve business needs.

Contents

Chapter 1: Introduction	5
Scope and Motivation	5
Why Use RUM?	5
Why Use BPM?	5
Chapter 2: Planning for RUM	7
Methods for Obtaining Monitored Traffic to the Probe	8
Chapter 3: RUM Deployment Architecture Design Considerations	9
Chapter 4: Installation, Configuration, and Validation	11
Chapter 5: Basic RUM Application Example	15
Chapter 6: Monitoring Applications Hosted on Virtual Infrastructure	20
Chapter 7: Advanced Reports Configuration	21
Reports Grouped by Event Extracted Data	21
Adding Session Properties for Mobile Applications	22
Chapter 8: Integrating with Other HPE Software Products	27
BPM	27
Diagnostics	27
Data Flow Probe	27
Chapter 9: Basic Troubleshooting	29
No New Application Data Displayed	29
Partial Data is Displayed	29
No Raw Data Reports	30
Creating a Debug Channel on the Probe for Analyzing Data	30
Appendix A – Support Matrix, Hardware Specifications, and Supported Protocols ..	33
Support Matrix	33
Hardware Specifications	33
RUM Supported Protocols for RUM Version 9.30	33
Appendix B – Advanced User Name Resolving	35
Appendix C - Network Examples	36
Send Documentation Feedback	39

Chapter 1: Introduction

This section introduces the Getting Started With RUM - Best Practices and includes the following topics:

- ["Scope and Motivation" below](#)
- ["Why Use RUM?" below](#)
- ["Why Use BPM?" below](#)

Scope and Motivation

RUM provides ongoing monitoring of user experience, application availability, and performance behavior, based on passive sniffing of network traffic. RUM can also collect data on traffic using JavaScript instrumentation of a Web application or Android native application instrumentation. The purpose of this document is to provide RUM users with an understanding of how to deploy RUM quickly and correctly, and how to realize value in a short time.

The target audience for this guide is customers who are new implementers of RUM, or HPE partners who already have Application Performance Management (APM) knowledge, but are new to RUM. While no deep networking/encryption knowledge is needed, you will need to know the protocol and encryption used by the monitored application.

Why Use RUM?

RUM provides you a window to realize the real experience of your end users, over and above the data collected by other APM data collectors.

SiteScope keeps you aware of infrastructure problems (and provides some details on business impact). Infrastructure monitoring can alert you when you have infrastructure problems, even though the business and the users may not be affected at the time. Also, there may be a situation when the infrastructure is normal, but there is an applicative problem in a specific use case that is not monitored by SiteScope.

Business Process Monitor (BPM) has a more narrow focus on the application, and cannot represent all user flows from all locations.

RUM gives you the entire picture - of all users, from all locations. It is common to see an application monitored by RUM and BPM, where the BPM transactions are OK and the RUM data is in a critical status. This can convey that the BPM scripts are not really monitoring what real users are doing, or that the problem may be in specific locations.

This is why a holistic approach of having a variety of data collectors is needed to cover all scenarios.

Why Use BPM?

BPM should also be used to gather data during inactive hours of the users, so you will be able to detect problems before a real user encounters them. In addition, BPM is the basic monitor for building SLAs,

because it has the same transaction monitored at the same time intervals.

Chapter 2: Planning for RUM

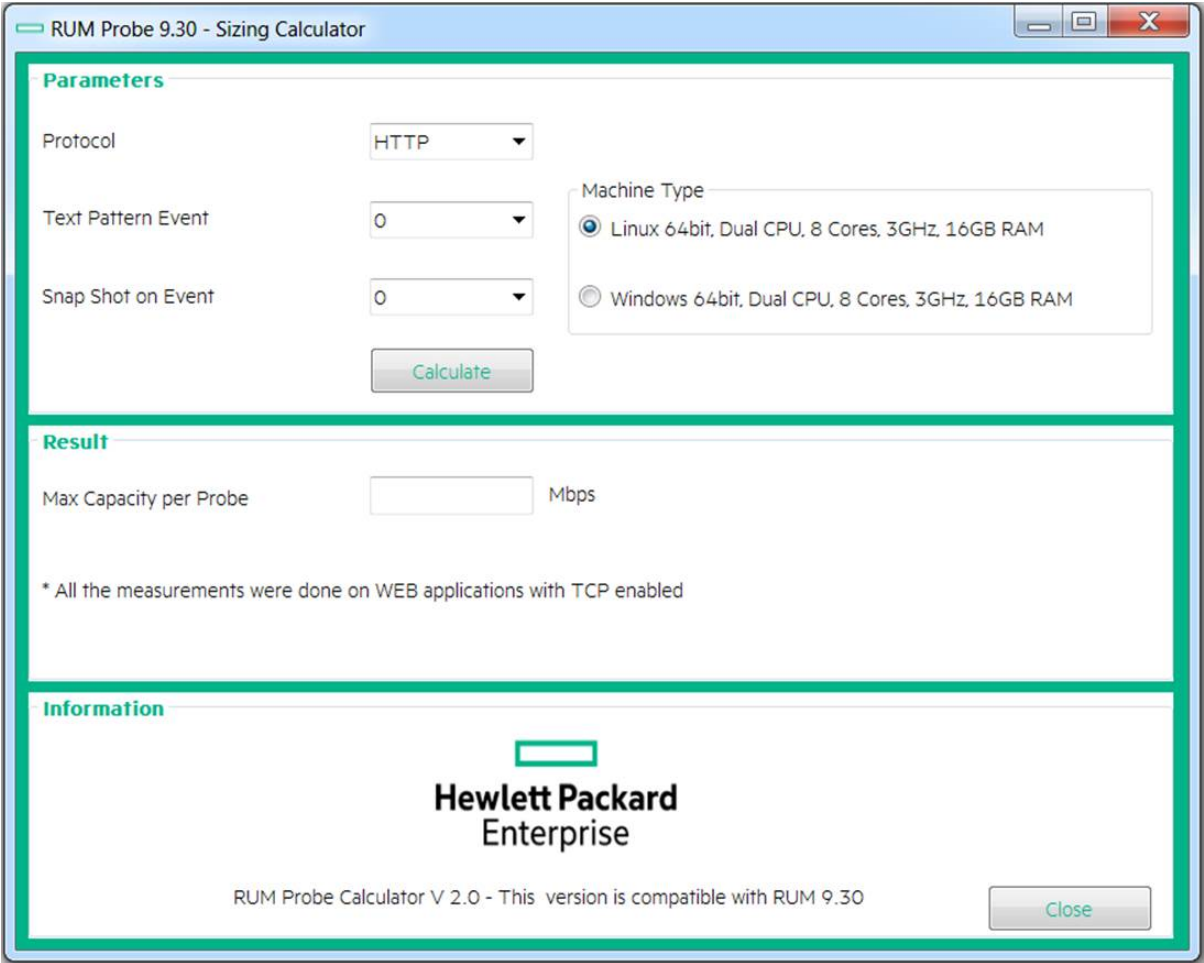
For optimal implementation of RUM, we recommend that you obtain in advance the information you will need for the implementation phases:

- The network diagram of the applications.
- The estimated amount of traffic to be monitored and its type (HTTP/HTTPS/TCP), for each segment (for multi-tier application monitoring). The RUM pricing model (except for the 360 APM license) is per probe.

You can use the traffic discovery tool (described in chapter 4) in the RUM web console to get the exact amount of traffic, to determine how many probes will be needed to monitor the overall traffic.

Use the Probe Sizing calculator (which is located on the HPE Software Support site) to see how many probes you will need to use. Note that Text Pattern Events have a high impact on the probe capacity to monitor traffic. If using them, it is recommended to create Text Pattern Events on selective pages and not globally on all the traffic.

Example of the Probe Sizing Calculator:



- The protocols that will be monitored. Verify in Appendix A that they are supported.
- What the user considers are the important Business Services and Business transactions to monitor.

- Who the users of the product are (IT/application owners/CIO). What is the data each user wants to receive and how will he consume it (report, email ...).
- If SSL traffic is to be monitored, obtain the private keys (not the certificates) in the correct format (PEM, DER, PKCS8, and PKCS12 private key types, as well as Java Keystore) before implementation. Note that for the Asymmetrical encryption method only RSA is supported (and not Diffie-Hellman due to its cryptographic nature that is designed to make it very difficult to sniff and understand the traffic). This is not a real limitation as web servers can easily be configured to request only RSA-based connections.
- We recommend that you contact the network administrator at an early stage to arrange the span port, to save time later on.

Methods for Obtaining Monitored Traffic to the Probe

It is important to understand the differences between network tapping, port mirroring, and the RUM Server Collector. A tap is the most reliable way to get all of the traffic with a guarantee that none of it is lost. The downside is that a tap is an expensive device and it requires changing physical wiring. Port mirroring requires the assistance of network administrators, but is usually simple to accomplish, although it should be arranged in advance due to the time it takes to set up. However, when the switch is loaded, the traffic to the RUM Probe will be the first to lose packets.

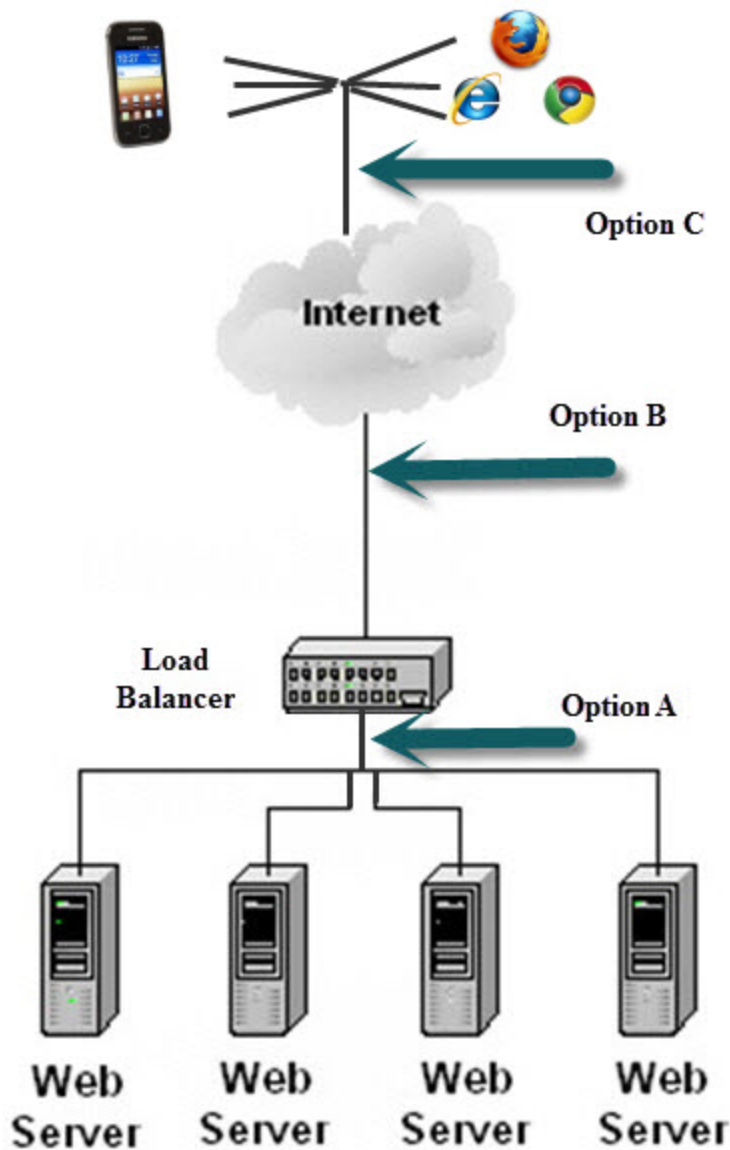
The Server Collector offers an alternative solution, by installing (with root permissions during the installation) a light-weight component directly on the server machine. This component captures all traffic from the local network card and sends it to a RUM Probe, thereby eliminating the need of network personnel support. However, its bandwidth is limited to 140 Mbps on a Windows machine and 250 Mbps on a Linux machine, and it duplicates the traffic on the server.

Note: The RUM Server Collector was not certified for production use and should be used only for proof of concept (POC) purposes.

Chapter 3: RUM Deployment Architecture Design Considerations

It is important to understand where to connect the RUM Probe to your environment. The following are some options of where to place the probe in order to listen to traffic.

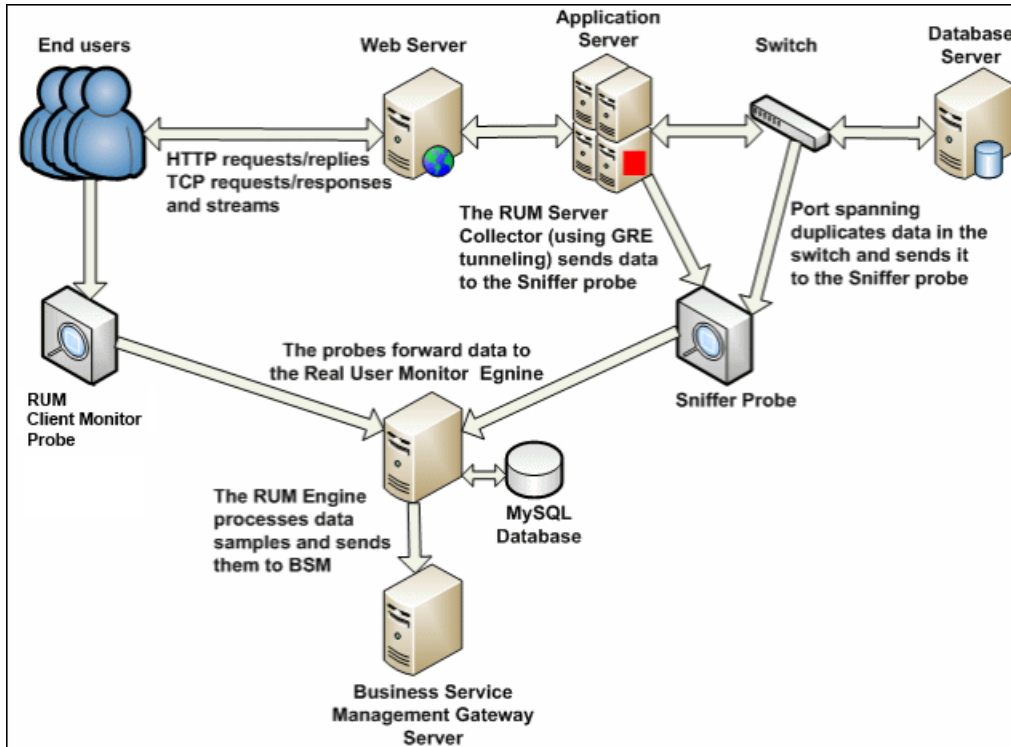
Diagram of architecture options:



- If option A is selected, the probe will identify the web servers, but all the traffic will appear as originating from the LB and there is no breakdown information about the real end user.

- Option **B** is to place the probe between the LB and the WAN. In this case, the probe sees the breakdown of the users, but the server breakdown is masked by the LB, so that all the traffic looks as if it is designated to the LB and there is no information about the real Web/App Servers.
- Another approach is to use the RUM Client Monitor (Option **C**) to catch the end user experience of an application from the client-side point of view, including the actual time the user experienced due to CDNs, external content, and proxies.

The optimal solution is a combination of getting the traffic from all ends, based on network feasibility and the amount of traffic the probes are capable of monitoring. This is called the RUM multi-tier approach, and it gives you both end-user experience and back-end drilldown. This is illustrated in the following diagram:



Note:

- If only one probe can be installed, we recommend that you install the probe as illustrated in option A and set the x-forwarder-for (XFF) header on the LB to get the real IP by RUM and not the LB IP.
- In cases where the traffic before the load balancer is encrypted and after the LB is not, it makes sense to listen to the traffic both before and after the LB (options **A+B**) so that you also get the SSL breakdown and can find out if problems reside there.

Chapter 4: Installation, Configuration, and Validation

1. Always install the latest RUM Engine and Probe versions that match your APM installation, according to the support matrix (see Appendix B). This ensures that you get the best performance and the latest fixes available.
2. Make sure you have the correct hardware (according to the Real User Monitor Installation and Upgrade guide). Also make sure you have enough disk space for your MySQL server to store RUM's saved data, which by default is 2 weeks back (all the aggregated data resides on the APM DB, so only the last 2 weeks of raw data is saved by RUM).
3. Prepare a list of all of the URLs and IPs (since server ports are very important for kernel level traffic filtering on the probe side), for all of the application tiers you intend to monitor with RUM.
4. In the RUM Engine web console, configure your APM and RUM Sniffer or Client Monitor probes (under the **Configuration** tab). Make sure you are able to sync configuration in the engine with no errors and that the RUM web console does not show any red statuses. A common problem is when the RTSM integration password was changed from the default (admin) in the APM installation, but not updated properly in RUM. You will need the new password for the rum_integration_user. In APM 9.30, integration users are regular users with SDK permission, so the rum_integration_user management is performed using the JMX for User Management.

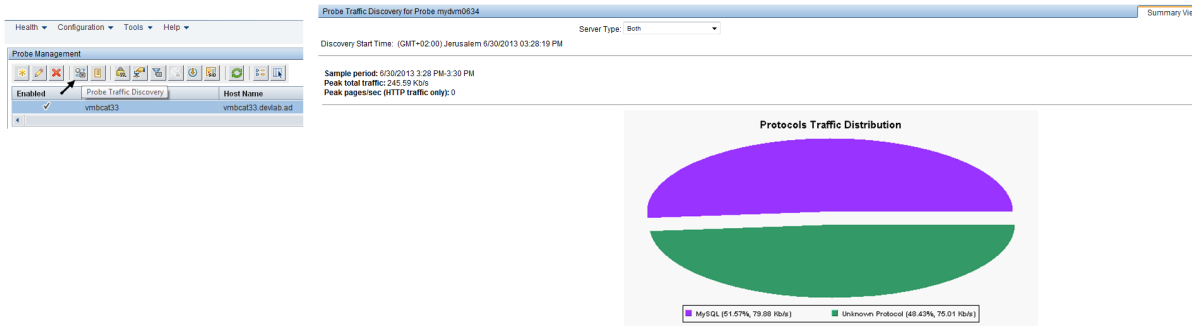
You can modify the rum_integration_user password using the CMDB JMX:

http:// <APM Data Process Server machine>:21212/jmx-console/HtmlAdaptor?action=inspectMBean&name=UCMDB:service=Authorization%20Services#resetPassword

Use the following parameters:

- **customerID** = 1
- **userName** = rum_integration_user
- **password** = The new password that should now be used in the RUM web console

- Run Probe Traffic Discovery from the RUM web console for a while to validate that the traffic you intend to monitor is being received by the probe and that you are not exceeding the traffic capacity limit.



Probe Traffic Discovery for Probe vmammd35.devalab.ad

Discovery Start Time: (GMT+02:00) Jerusalem 8/1/2013 01:10:52 PM

Sample period: 8/1/2013 1:10 PM-1:14 PM
Peak total traffic: 1.13 Mb/s
Peak pages/sec (HTTP traffic only): 2

	% Throughput	Throughput (Kb/s)	Peak Traffic (Kb/s)	Peak Pages/sec	Compressed	Encrypted	Server Info	More Details
HTTP	68.17	214.77	1,137.07	2				
16.55.244.192	32.13	101.25	1,137.07					
9090	32.13	101.25	1,137.07					
16.60.185.253	36.03	113.52	211.44	1				
2006	36.03	113.52	211.44	1				
mydvm1771								
MySQL	25.93	81.7	245.98					
16.59.63.17	25.93	81.7	245.98					
3306	25.93	81.7	245.98					
Unknown Protocol	5.58	17.58	56.7					
Unrecognized Servers	5.58	17.58	56.7					

- If the monitored traffic is encrypted, upload the relevant private keys to the RUM web console.
- Configure the application you want to monitor in APM, using IP and port if applicable. Note that the traffic is filtered in the kernel on the probe by TCP port, so including the port in the configuration (and not just the URL) improves the probe's performance.
- Deploy RUM monitored applications one at a time so that you will be able to analyze problems and realize benefit easily.
- For web applications, we recommend that you run the Session ID detection from the RUM web console and insert the top values into the application definition.

Session ID Detection Report for probe vmammd35.devalab.ad

Detection run at (GMT+00:00)Greenwich Mean Time 1/9/1970 01:38:42 PM

Filter By Application Name: All

App ID	Application Name	Application Components	Application Pages	Application Clients	Application Connections	All Session IDs Coverage	Session ID Key	Type	Regular Expression	Specific Session ID Coverage	Specific Session ID Correctness
64	Online Banking	148	21	1	12	100.0 %	SESSIONID	COOKIE		100.0 %	100.0 %
64	Online Banking	148	21	1	12	100.0 %	unique_id	COOKIE		86.5 %	100.0 %
64	Online Banking	148	21	1	12	100.0 %	RT	COOKIE		83.8 %	100.0 %

- In the **Data Collection** tab in End User Management (EUM) Administration, you can mask sensitive parameters of the URL, or sensitive content of the requests/response when displaying snapshots.

To collect more valuable data, you should also configure the following in EUM Administration:

- User Names (where the probe can find the user name in the traffic).
- Transactions (for getting Business Data). Also, if you have also BPM monitoring this application, try to have the same transaction monitored by both RUM and BPM. This will enhance your data collection to be

consolidated in Service Health and EUM reports. In addition the infrastructure model will unite with BPM transactions and not just with RUM transactions.

- End User Groups.

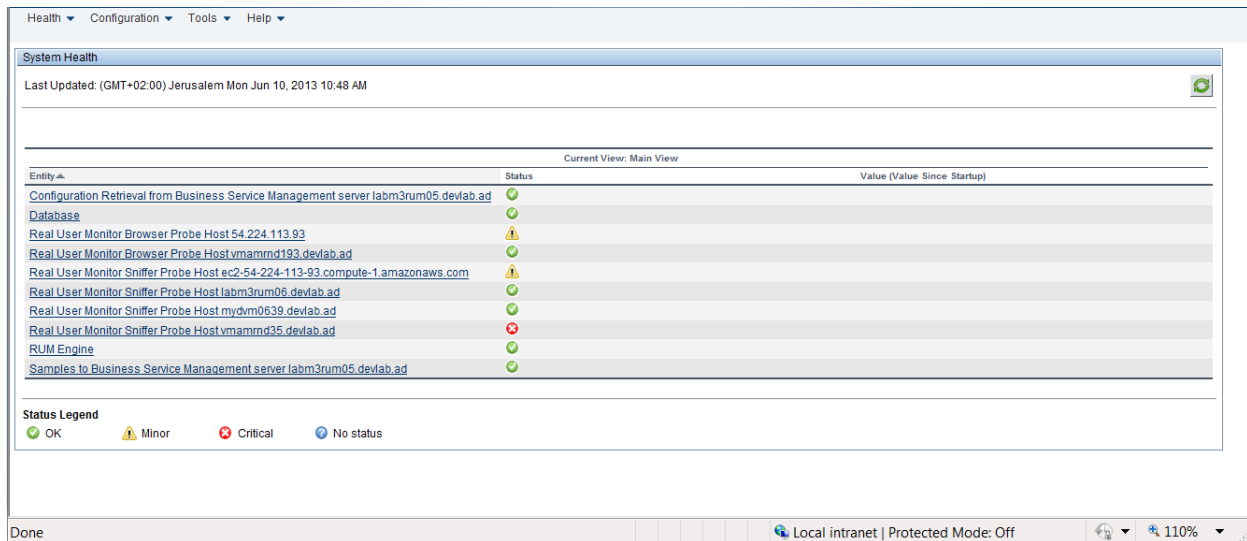
Using the RUM Client Monitor Probe requires you to instrument your web pages with a JavaScript snippet before being sent to the browser. This should be done in one static place for ease of deployment, and not in each page. For details, see "Installing the JavaScript Snippet" in the Real User Monitor Installation and Upgrade Guide

There is also a simple way to get real user traffic without the need of the sniffer probe (and network personnel) and without instrumenting the application with the JavaScript snippet. You can download a RUM Client Monitor plugin for Chrome which will perform the same JavaScript instrumentation, but only on your Chrome browser. After configuring the plugin to connect to your RUM Client Monitor Probe, you will be able to see all the traffic coming from the Chrome browser.

On a physical network the probe has to be physically close to the network tap or switch from which the traffic is mirrored.

The throughput between the RUM Engine and Probe is relatively high, so it is recommended to place them in such a way that they have a good network connection between them. The throughput between the RUM Engine and the APM Gateway server is much less than the throughput between the RUM Engine and Probe. Therefore, if necessary, the RUM Engine can be placed on a geographically remote location, close to the Probe and communicating with APM over the WAN. You may even consider installing several RUM Engines near each Probe that will communicate with one APM. However, since the communication is two-sided, ports have to be opened in both directions.

The RUM web console health page gives you a simple way of learning about problems at a glance, before looking in the logs.



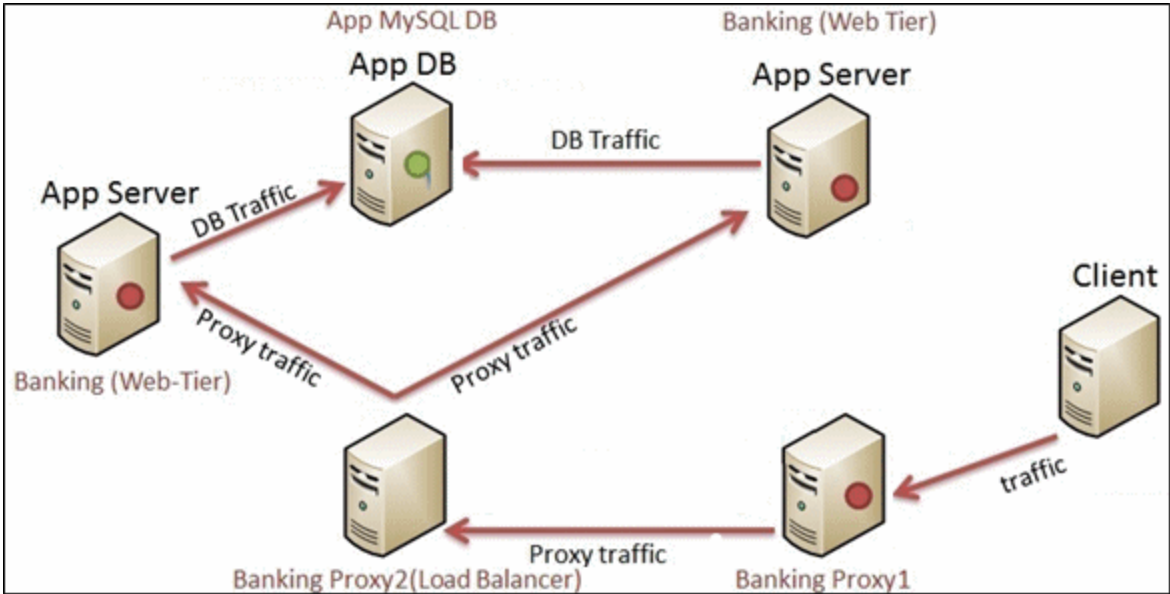
When you have finished your configuration and data is flowing from RUM to APM, apart from the RUM data you will also be able to see the topology that RUM detected 'on the fly' (Running Software, Nodes, IPs and Ports) and the locations that RUM detected automatically by its internal GEO map of IPs.

Below is an example of the CIs that are auto discovered by RUM. Apart from the Business Application and the Business Transaction, that are configured by the user (in End User Management Administration), all the other CIs (node, runningsoftware, ip, ip_service_endpoint) and relationships are created by RUM.

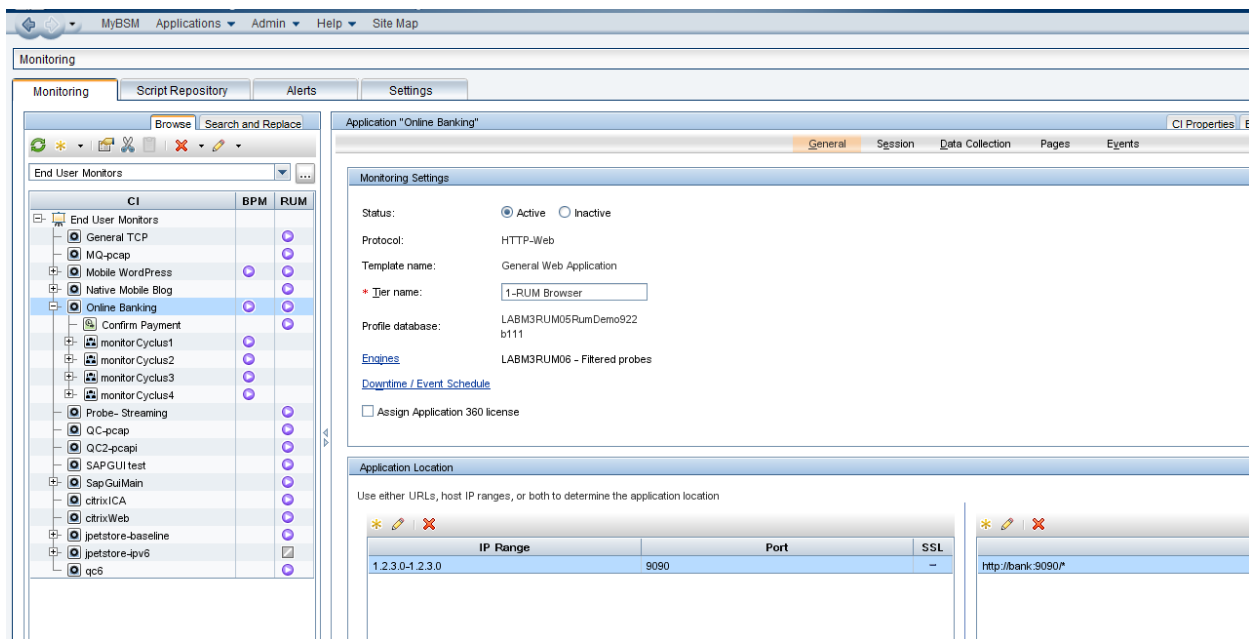
Chapter 5: Basic RUM Application Example

Below is an example of an online banking application:

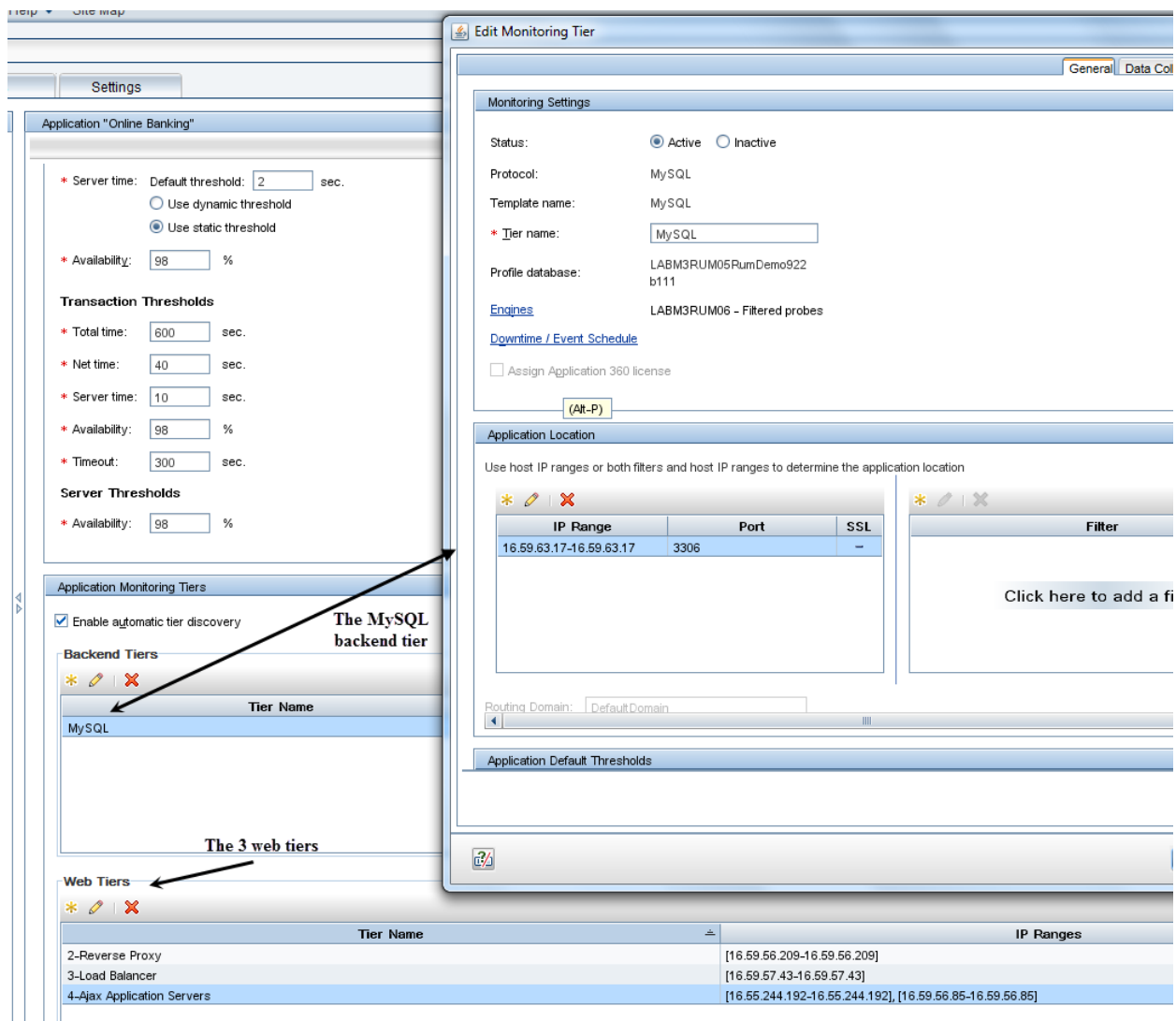
- Front End tier
 - Reverse Proxy
- Backend Tier
 - MySQL
- Web Tiers
 - Load Balancer
 - Application Servers



The following graphics show the basic configuration for correctly capturing the traffic:



The application has a MySQL back-end tier and three web tiers (Reverse Proxy, LB, and an Ajax application server):



For all these five applications, you only need to configure the IP and port, and the URL if applicable (for the front-end).

When configured, you will start to see the data RUM has monitored.

After you have some pages (auto discovered or configured manually by you) you can simply create a RUM business transaction. It is important to create transactions for key flows, so that you can easily see the impact on your business.

Business Transaction "Confirm Payment"

Monitoring Settings

Status: Active Inactive

Thresholds:

* Total time (seconds):

* Availability (%):

* Timeout (seconds):

Determine thresholds from included pages

Manual thresholds

* Net time (seconds):

* Server time (seconds):

[Location Aware Thresholds...](#) [Suggested Thresholds Wizard...](#)

Transaction Page Settings

Include intermediate pages in measurements

Ignore page order

Define the pages included in the transaction and their order.

*

Page Name	Optional
Confirm Payment	<input type="checkbox"/>

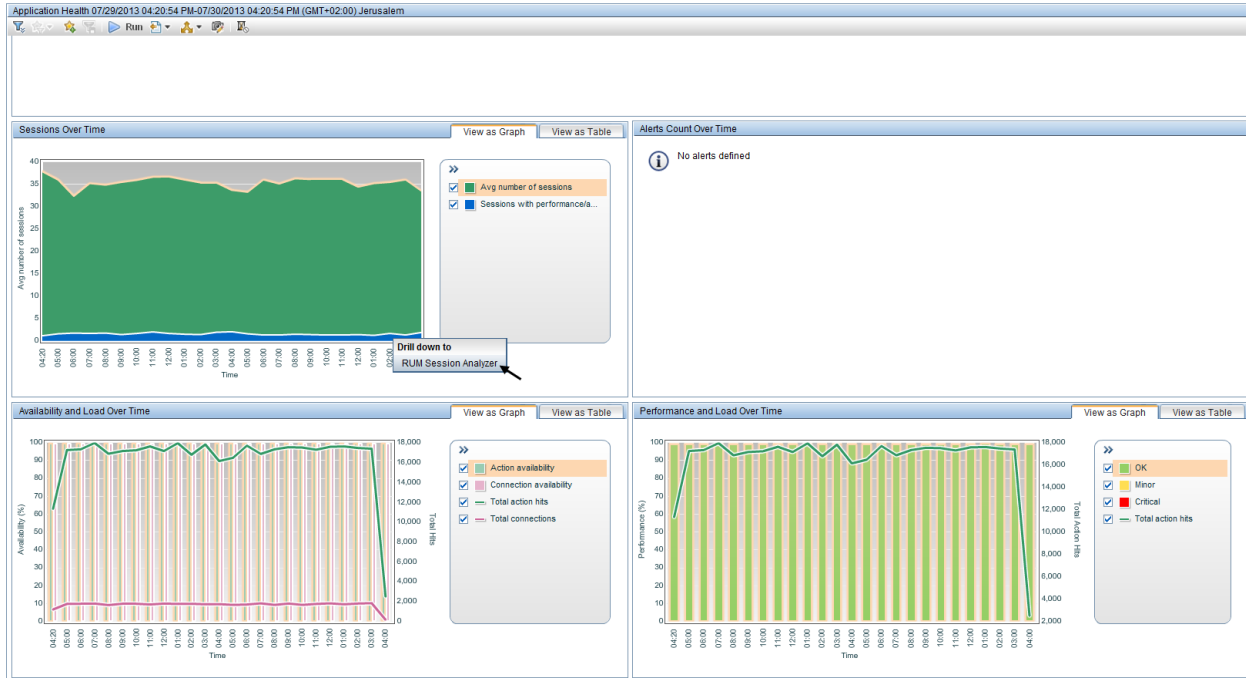
Apart from Service Health, where you can see the status of your application and drill down to the relevant data:

The screenshot displays the HPE APM interface. At the top, there's a navigation bar with 'MyBSM Applications Admin Help Site Map'. Below it, a 'Hierarchy' pane shows 'End User Monitors' with 'Online Banking' selected. The main area shows a table with columns: Name, Business Impact, Status, Acknowledge, System Unassigned Events, Unresolved Events, Business Volume, Application Performance, and Application Availability. The 'Online Banking' entry shows a status of 'Warning' and a volume of 35. Below this, there's a detailed performance view for 'Online Banking' with columns for Availability (%), Past day (Beginning to end), Locations, Performance (%), Past day (Beginning to end), Locations, and Volume. The data shows 99.92% availability and 99.53% performance across various locations.

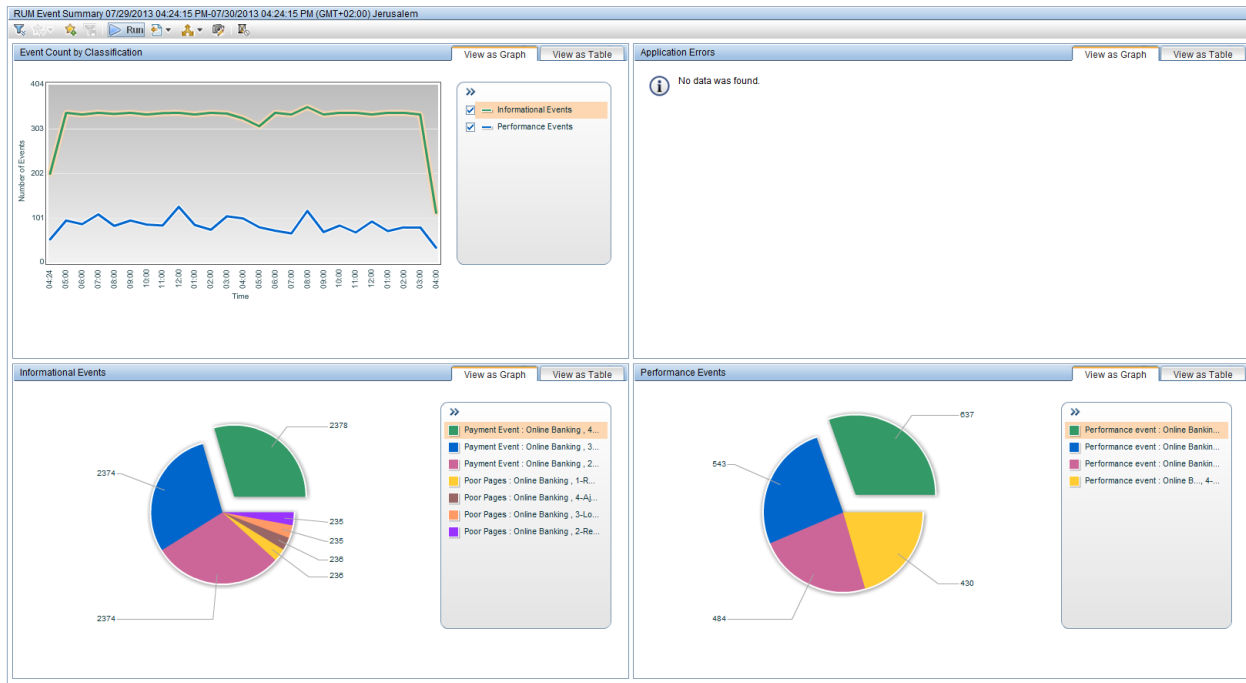
Application	Availability (%)	Past day (Beginning to end)	Locations	Performance (%)	Past day (Beginning to end)	Locations	Volume
Online Banking	99.92	[Progress Bar]	[Progress Bar]	99.53	[Progress Bar]	[Progress Bar]	3,815
Real User Session Data	100.00	[Progress Bar]	[Progress Bar]	99.63	[Progress Bar]	[Progress Bar]	1,933
Real User Transaction Data	100.00	[Progress Bar]	[Progress Bar]	99.53	[Progress Bar]	[Progress Bar]	475
Synthetic User Data	99.92	[Progress Bar]	[Progress Bar]	100.00	[Progress Bar]	[Progress Bar]	3,815

There are also a variety of RUM reports that reflect an application's status.

Among the important reports you can view is the Application Health report, in which you can see an over time report of the sessions on your application, with the performance and errors that occurred on it (with the ability to drill down and triage the problems):



Also, the RUM Event Summary report that shows you the top errors on your application on which your development team should focus (also with the ability to drill down and triage the problems):



Chapter 6: Monitoring Applications Hosted on Virtual Infrastructure

- To monitor traffic when Vmotion is disabled:
 - Deploy the RUM Probe on the same ESX as the monitored application.
 - Allow promiscuous mode on the virtual switch or portgroup.
- To monitor traffic when Vmotion is enabled, perform one of the following:
 - Use GRE Tunneling.
 - Use Virtual Dedicated Taps.
 - Install the RUM Probe on the same ESX as the monitored application, and keep virtual machines together (VMware option).
 - Use the RUM Server Collector.

Note: The RUM Server Collector was not certified for production use and should be used only for proof of concept (POC) purposes.

- In ESX 5.x there is an option to define a mirror port between two ESXs.
- In ESX 4.x you need ERSPAN (Cisco Nexus switches).

Chapter 7: Advanced Reports Configuration

This section provides information about configuring advanced reports and includes the following topics:

- ["Reports Grouped by Event Extracted Data" below](#)
- ["Adding Session Properties for Mobile Applications" on the next page](#)

Reports Grouped by Event Extracted Data

Text pattern events consume probe CPU and therefore, reduces the probe's performance.

Instead of configuring many similar text pattern events that are only differentiated by the captured text, it is more efficient to configure one text pattern event and extract the important data in it. APM stores your important data in the event's extracted data field in the APM database.

Regular reports do not display the data field. You can only see this value in an out of the box configuration in the Event Log report, with no grouping. However, you can create a simple Custom Query report that will display these values.

For example, to see how many different items of a certain event occurred in the last month, you can create the following report:

The screenshot shows the 'Custom Query Builder' interface. At the top, there are tabs for 'Report Manager' and 'Custom Query Builder'. Below the tabs is a toolbar with icons for 'Run', 'Save', and 'Refresh'. The main configuration area includes:

- View: Past month
- From: 6/30/13 3:58 PM To: 7/30/13 3:58 PM (GMT+02:00)...
- Sample type: RUM Events (rum_event_t) Return data per specified granularity
- Field name: Action Description Function: None
- Filter: application_name = (petstore)

Below the configuration is a section titled 'Fields to Return:' with a table:

Field Alias	Field Formula
Field alias: Count	Field formula: COUNT(action_id)
Field alias: Event Extra Data	Field formula: event_data

At the bottom of the configuration area is a button labeled 'Edit Presentation'.

Below the configuration is a table of results:

Count	Event Extra Data
22.00	Signing Up valign="middle" > Signing Up
207.00	Rattlesnake
221.00	Amazon Parrot
225.00	Iguana
226.00	Persian
229.00	Poodle

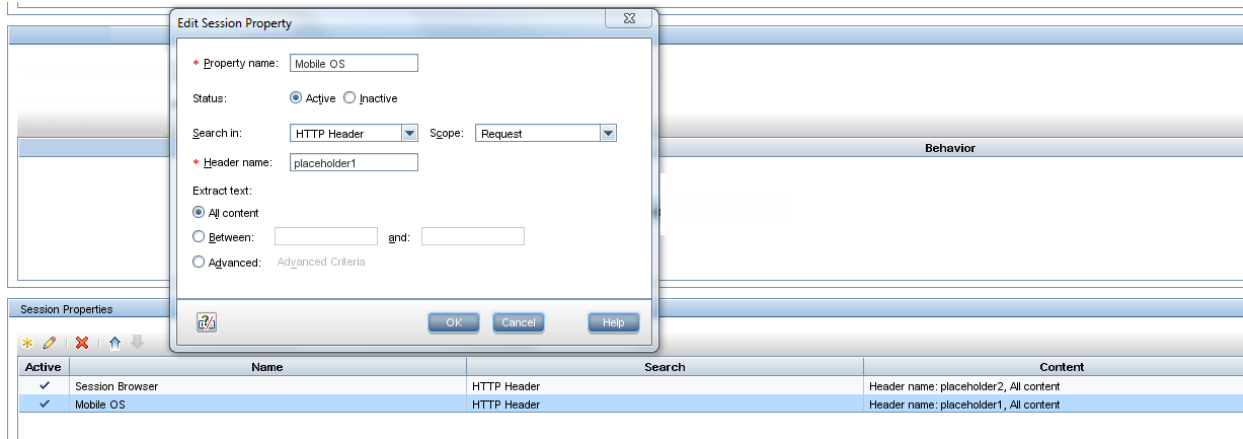
Note: If the text you are looking for only appears on specific pages, set the event to search only those pages, since this will have a major impact on the probe. There is a big difference when searching for text on specific pages rather than on all the application pages. The probe calculator takes into consideration text pattern events that are configured globally (and therefore have a high penalty).

Adding Session Properties for Mobile Applications

You can configure session properties to be tagged by RUM. The tags can be used to see data grouped by the properties in the Session Summary report, to filter sessions included in the RUM Session Analyzer report, and to view the tagged properties for a session when viewing the session details. You can also create end user groups based on session properties.

You can use this technique for mobile applications. To see RUM reports filtered by the type of mobile users:

1. Create two session properties - Mobile OS and Session Browser - in End User Management Administration.



2. Edit the file **HPRUM\conf\configurationmanager\Beatbox_Default_Const_Configuration.xml**:

- The **os-name** definition should be as follows. If it is different, replace the definition with the following:

```
[field os-name]
granularity hits
extract cs(User-Agent) iPhone rumos_iPhone
extract cs(User-Agent) Android rumos_Android
extract cs(User-Agent) BlackBerry rumos_BlackBerry
extract cs(User-Agent) webOS.*Pre rumos_WebOS
extract cs(User-Agent) Linux rumos_Linux
extract cs(User-Agent) SunOS rumos_Solaris
extract cs(User-Agent) (Windows|MSIE|Win32|msnbot|MSPProxy) rumos_Windows
extract cs(User-Agent) Mac rumos_Macintosh
extract cs(User-Agent) !(^rumos)
extract cs(User-Agent) rumos_(.*) $1
content cs(User-Agent)
```

Note: The delimiter between fields must be a tab and not a space.

- Replace the definition of the **browser-name** with the following:

```
[field browser-name]
granularity hits
extract cs(User-Agent) Chrome/([0-9]+\.[0-9]) Chrome $1
extract cs(User-Agent) Safari/([0-9]+\.[0-9]) Safari $1
extract cs(User-Agent) Firefox/([0-9]+\.[0-9]) FireFox $1
extract cs(User-Agent) MSIE.([0-9]+\.[0-9]) Internet Explorer $1
extract cs(User-Agent) Mozilla/([0-9]+\.[0-9]) Netscape $1
extrac cs(User-Agent) ([^s]+)/([0-9]+\.[0-9]) $1 $2
```

```
extract    cs(User-Agent) ([^\s]+)/([0-9]+)    $1 $2
content    cs(User-Agent)
```

Note: The delimiter between fields must be a tab and not a space.

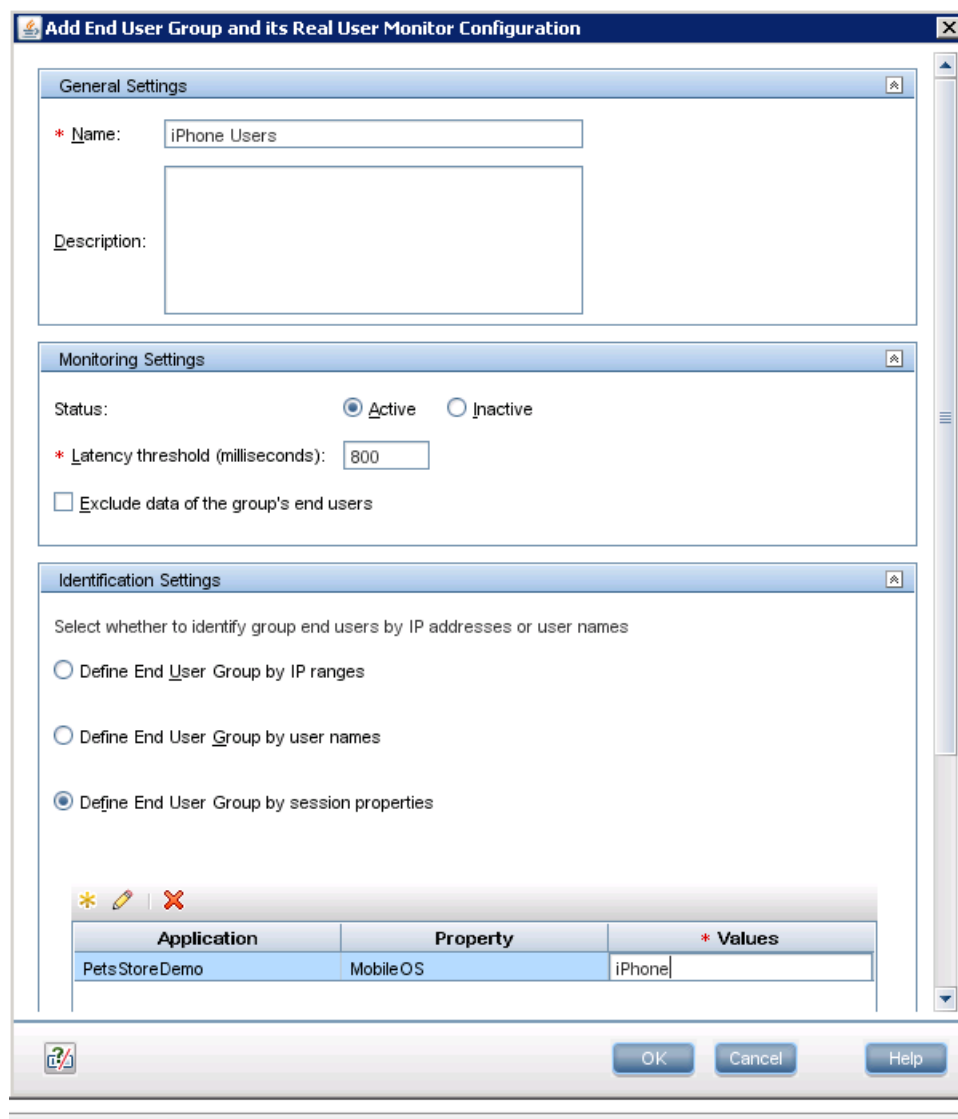
- At the end of file, replace the **extra-setting** section with the following:

```
<extra_settings>
<![CDATA[
[field    cs-placeholder2]
content browser-name
[field    cs-placeholder1]
content os-name
]]>
</extra_settings>
```


3. You can also create end user groups for important devices, in EUM Administration. The definition of the end user group is based on a session property called MobileOS, which you must predefine. Define a separate end user group for each mobile device to be able to filter the device in reports based on the end user group. For example:

- iPhone
- Android
- BlackBerry

The configuration is illustrated in the following image:



You can see the value of this configuration in the Session Summary report data.

The screenshot shows the 'RUM Session Summary' report for the period 27/4/2011 10:27-11:27 (GMT+2:00)ART. The report is filtered by 'MobileOS'. The main table displays session statistics for different operating systems.

Value	Number of Sessions	Number of unique users	Error Events	Info Events	Performance Events	Total Actions	Unavailable Actions	Latency	Total Traffic(KB)
Android	2	1	0	0	0	8	0	24.20	278.4
iPhone	2	1	0	0	0	7	0	40.08	232.5
Linux	2	1	0	0	0	18	0	22.42	152.9
Macintosh	1	1	0	0	0	6	0	22.28	210.2
WebOS	2	1	0	0	0	10	0	48.93	321.6
Windows	2	1	0	2	0	18	0	23.13	129.3
Total	11	6	0	2	0	65	0	29.67	1,724.8

Below the main table, there is a 'Detailed Sub Groups' section with a table showing 'Other Value' statistics:

Value	Number of Sessions	Error Events	Info Events	Performance Events	Total Actions	Unavailable Actions	Latency	Total Traffic(KB)
Other Value	2	0	0	0	8	0	24.20	278.4
Total	2	0	0	0	8	0	24.20	278.4

After configuring end user groups by this property, you will also see it in other reports (End User Summary and Breakdown by End Users reports).

The screenshot shows the 'RUM End User Group Summary' report for 12/6/2011 4:28 PM-5:28 PM (GMT+02:00) Jerusalem. The report is filtered by 'End User Group'. The main table displays performance metrics for different user groups.

End User Group	Network				Applications	
	Latency (ms)	Connection Availability (%)	Packets with Network Errors (%)	Packets with Client Errors (%)	Actions Availability (%)	Actions with Performance Events (%)
Android users	0.00	100.00	0.00	0.00	100.00	7.14
Android users	0.00	100.00	0.00	0.00	100.00	7.14
BlackBerry users	0.00	100.00	0.00	0.00	100.00	9.52
BlackBerry users	0.00	100.00	0.00	0.00	100.00	9.52
iPhone users	0.00	100.00	0.00	0.00	100.00	8.33
iPhone users	0.00	100.00	0.00	0.00	100.00	8.33
Others	69.36	100.00	0.09	0.00	97.85	5.96
Others [0.0.0.0-255.255.255.255]	69.36	100.00	0.09	0.00	97.85	5.96

Legend: Critical (Red), OK (Green), Inor (Yellow), No data (Blue).

Below the main table, there is an 'End User Group Highlights by Location' section with a table showing performance metrics by location:

Location	Network				Applications	
	Latency (ms)	Connection Availability (%)	Packets with Network Errors (%)	Packets with Client Errors (%)	Actions Availability (%)	Actions with Performance Events (%)
Burpee	0.00	100.00	0.00	0.00	100.00	7.14
AT&T Mobility	0.00	100.00	0.00	0.00	100.00	7.14
USA	0.00	100.00	0.00	0.00	100.00	7.14
T-Mobile USA	0.00	100.00	0.00	0.00	100.00	7.14

Chapter 8: Integrating with Other HPE Software Products

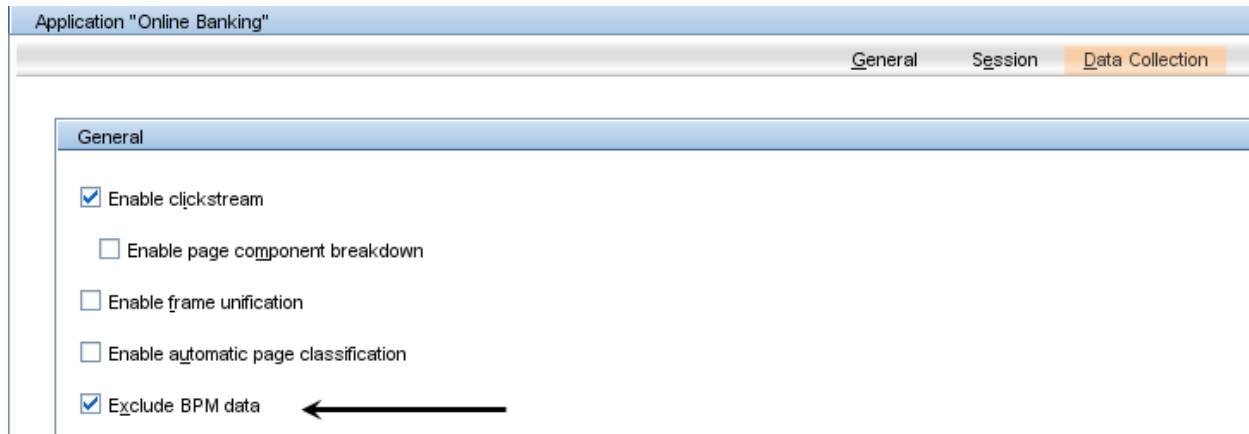
For a new monitored application in APM, it may be advisable to start monitoring it using BPM with synthetic scripts. Later add RUM to provide the real end user experience. Finally add Diagnostics on the application to create a way to deeply investigate and diagnose the problems users encounter.

This section includes the following HPE software products:

- "BPM" below
- "Diagnostics" below
- "Data Flow Probe" below

BPM

You can configure RUM to ignore BPM generated traffic for an application, so RUM reports will only show real end user data (and not synthetic data as well).



Diagnostics

If you have registered and enabled Diagnostics in APM, and RUM has detected Diagnostics data for the requested action, you will have a drill-down from relevant RUM reports to Diagnostics reports. For a list of these RUM reports, refer to "Viewing Diagnostics Data from End User Management Reports" in the APM User Guide.

Data Flow Probe

RUM is also able to act as a passive probe, sniffing the network topology and reporting to RTSM.

Register the uCmdb Data Flow Probe Connection Settings in the RUM web console. In the Data Flow Probe, configure the IPs and ports to which you want RUM to listen.

Chapter 9: Basic Troubleshooting

This section provides basic troubleshooting and includes the following:

- ["No New Application Data Displayed" below](#)
- ["Partial Data is Displayed" below](#)
- ["No Raw Data Reports" on the next page](#)
- ["Creating a Debug Channel on the Probe for Analyzing Data" on the next page](#)

No New Application Data Displayed

It is easier to isolate a problem if other applications are disabled. Therefore, if possible, disable other applications in EUM Administration in APM.

In RUM version 9.22 there is a new JMX called FlowMonitoring (<http://<Engine>:8180/jmx-console/HtmlAdaptor?action=inspectMBean&name=RUM.modules%3Aservice%3DFlowMonitoring>), which displays the main flow points of RUM. You can use this to locate where data is not being received, in order to continue further investigation using engine logs or in the probe.

If you are not able to see traffic arriving in RUM:

1. Check your application configuration. A typical mistake is not adding an asterisk at the end of the URL, or not assigning the application to the engine.
2. If the configuration is correct, check that the probe sees the desired traffic, either with traffic discovery, or using tcpdump on the probe machine.
3. Make sure that the probe sees both sides of the traffic – requests and responses. If the probe only sees one side, all the traffic is ignored. You can determine this using the Incomplete Transactions indicator in the RUM web console, under the probe section. If you see just one side of the conversation, you will need to contact network assistance to fix it. To document the problem, record a pcap file on the probe machine (using wireshark) and run the following command:
Wireshark > statistics > conversations > IP Tab > check packet from A > B and from B > A
4. If you see traffic in the engine but it is not published, check further in the logs. Use the JMX mentioned above (**FlowMonitoring**) to focus your search on the correct module.

Partial Data is Displayed

If partial data is displayed:

1. Make sure your application is configured correctly.
2. Check that there is no data loss on the probe side. Use the System Health option in the RUM web console, under the probe section, to determine if this is the situation. If you are monitoring an SSL application, and the beginning of the conversation is missed, you may lose the entire conversation, so even a small amount of packet loss or transaction dropped may result in loss of significant data.

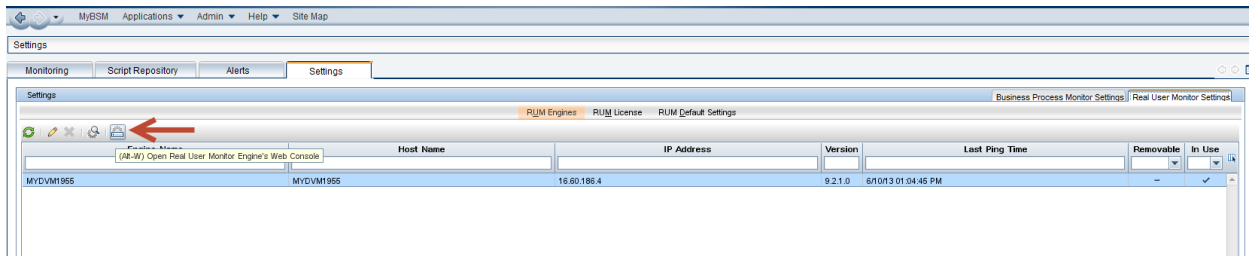
Lost SSL Requests	✓	0 %
Missing Mirrored Data	✓	
Network Captures Retriever Queue Size	✓	0
Orphan Application Hits	✓	
Packet Queue Work	✓	160
Packets Filtered IPv4	✓	0 Packets
Packets Filtered IPv6	✓	0 Packets
Packets Filtered Sum	✓	0 Packets
Packets Lost Per Second IPv4	✓	0 %/Second
Packets Lost Per Second IPv6	✓	0 %/Second
Packets Lost Per Second Sum	✓	0 %/Second
Packets Lost Total IPv4	✓	0 %
Packets Lost Total IPv6	✓	0 %
Packets Lost Total Sum	✓	0 %

3. Check the overall health of RUM.
4. Use the FlowMonitoring JMX as described in ["No New Application Data Displayed"](#) on the previous page.

No Raw Data Reports

If you see data in some RUM reports but not in others (such as Session Analyzer, Session Summary, or other reports with special filters), it is possible that the RUM Gateway URL credentials are not correct.

To check this, click the **Open the Real User Monitor Web Console** button on your APM machine (see image below), and verify that you can see the RUM web console and that the RUM hostname/IP or user and password is correct. The user and password should be the RUM JMX user and password.



Creating a Debug Channel on the Probe for Analyzing Data

If you do not know where the user name or session ID is located in the data, you can look for it using Wireshark. If you cannot install Wireshark on the probe machine, you can create a temporary debug channel in the probe that will capture the suspected part of the data, which will enable you to correctly configure the user name/session ID.

Note: This solution requires a change in the configuration that influences the probe performance. Use it carefully and for a limited time. Be sure to remove the added configuration after the test is finished.

As an example, suppose you know that the user name is stored in the header but you are not sure exactly where.

To get all the headers for every hit:

1. Locate the `\conf\configmanager\BeatBox_Default_Const_Config.xml` file in the RUM Engine.
2. Make a backup copy and open the original file for editing.

3. Find the **[channel rum-pages]** section and add the following right before `</channel_rum_pages>`:

```
[content my-req-headers]
extract_raw request headers
[content my-res-headers]
extract_raw response headers
[channel debug-channel]
granularity hits
type web
rotate 40000
cache false
require x-is-pageview 1
format date-time cs-uri-stem x-content-my-req-headers x-content-my-res-headers
```

4. Synchronize the configuration.
5. Remove these configuration changes when done.
6. Delete the debug-channel you created as the probe will not delete the files under this folder.

To get just the cookie, the section will look like this:

```
[channel debug-channel]
granularity hits
type web
rotate 40000
cache false
require x-is-pageview 1
format date-time cs-uri-stem "cs(Cookie)"
```

Another possibility is to get all parameters (including post parameters) in case john-doe is found in one of the parameters.

This is an example of the configuration (all the other steps are the same):

```
[field search-params]
granularity hits
content all-parameters
extract_filter all-parameters (john-doe) $1
[filter search-not-found]
granularity hits
require search-params eq
[channel debug-channel]
```

```
granularity hits
type web
rotate 40000
cache false
include !search-not-found
require x-is-pageview 1
#require x-application-name 318
format x-application-name c-port "cs-uri-stem" all-parameters
```

Remember to remove the configuration changes when done and delete the debug-channel you created, as the probe will not delete the files under this folder.

Appendix A - Support Matrix, Hardware Specifications, and Supported Protocols

Support Matrix

To view the support matrix, see "RUM Compatibility Matrix" in the Real User Monitor Administration guide.

Hardware Specifications

For hardware specifications, check the [HPE Software Support site](#) for the latest Real User Monitor Installation and Upgrade guide.

RUM Supported Protocols for RUM Version 9.30

The following is the list of RUM Supported Protocols for version 9.30. See "Parsing Supported Protocols" in the Real User Monitor Administration guide (for version 9.30).

HTTP Protocols

- HTTP
- Flash/ActionScript AMF - HTTP Based

SOA Protocols

- SOA
- WCF - HTTP Based

Database Protocols

- Microsoft SQL Server
- Oracle DB [+]
- MySQL
- DB2

Application Servers

- Citrix XenApp (ICA) [+]
- XenApp Application configured as a VDI tier under the main General Web Application
- Oracle Forms NCA [+]
- WMQ
- SAPGUI

Mail Protocols

- IMAP (Internet Message Access Protocol)
- SMTP (Simple Mail Transfer Protocol)
- POP3 (Post Office Protocol)

Generically Supported Protocols

- DNS – Generic UDP
- Microsoft Terminals Services (RDP) – Generic Streaming TCP
- RMI Registry – Generic TCP
- SSH – Generic Streaming TCP

Financial Protocols

- NDC

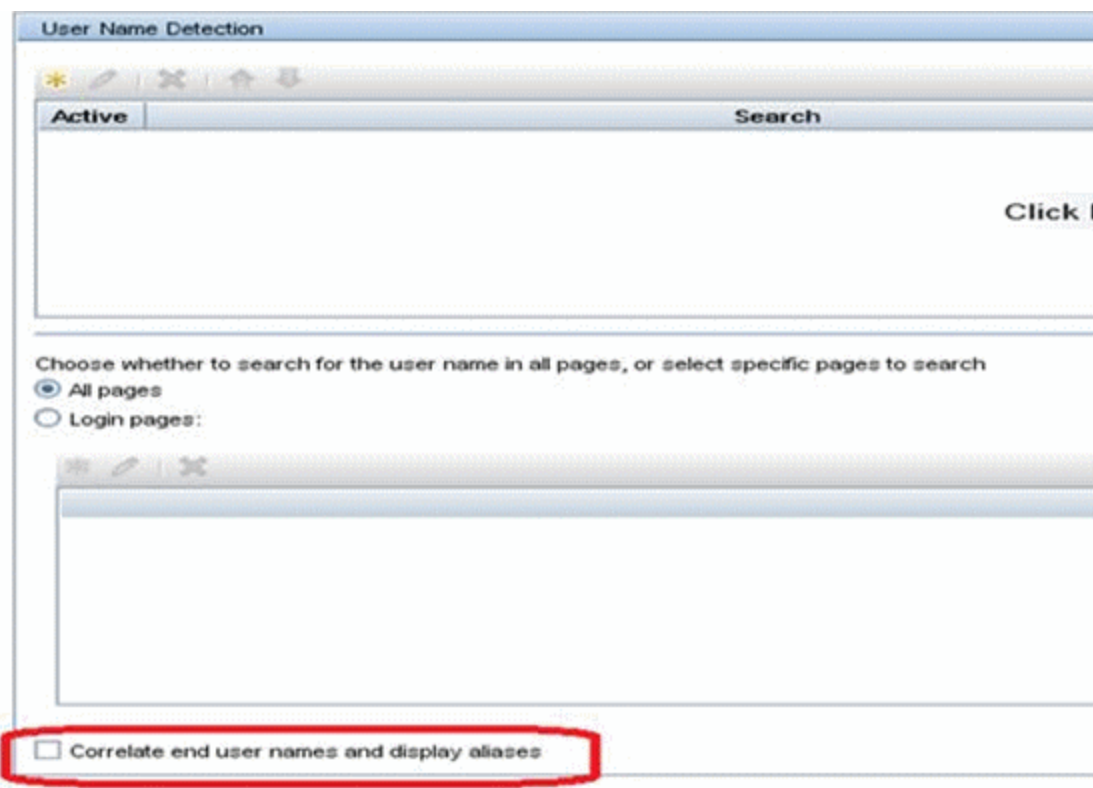
Additional Applications

- FTP (File Transfer Protocol)
- LDAP (Lightweight Directory Access Protocol)
- ISO 8583 (financial transaction card originated messages for Visa and Mastercard)
- SHVA
- AMF
- UDP

Appendix B - Advanced User Name Resolving

To resolve user names, you can use two advanced name resolvers: **CSVIPUserNameResolver** (from client IP to user name) and **CSVLoginUserNameResolver** (from the user name detected by RUM to a logical user name that can be mapped from LDAP, for example). To use either one of them, do the following:

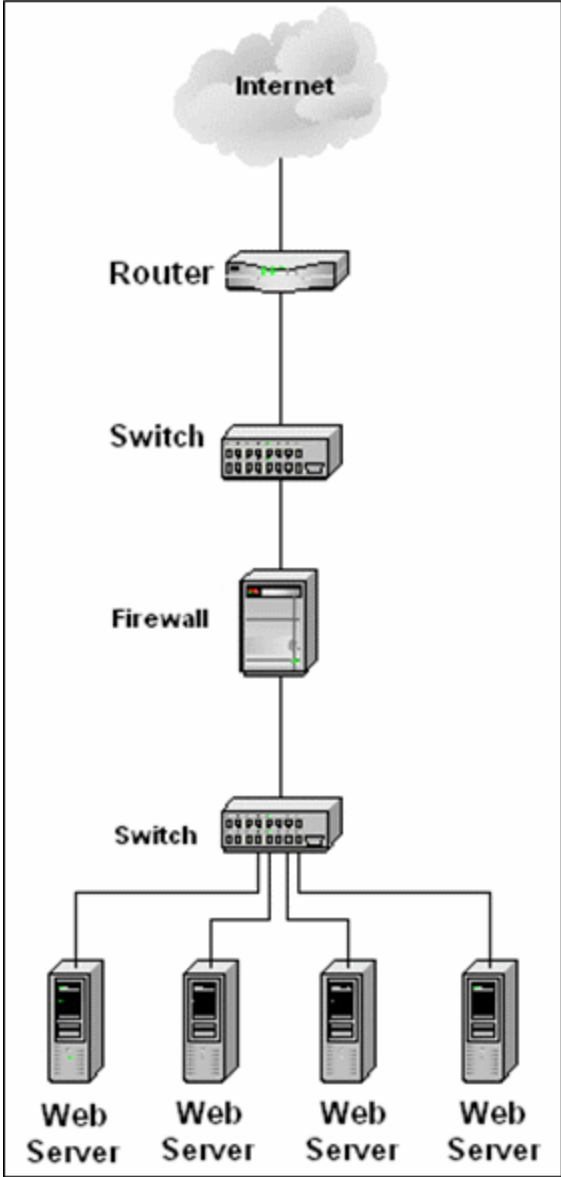
1. On the RUM Engine machine, edit the **confresolver\UserNameResolver.xml** file and change the class to the required value (**CSVIPUserNameResolver** or **CSVLoginUserNameResolver**).
2. Save a .csv file with the relevant data in the same folder (**confresolver**) and point the file **param** in the **UserNameResolver.xml** to the file. The .csv format is important and it has contain both the source and target (for example, columns for IP and name).
3. For each application for which you want to use this IP to name resolving, select **Correlate end user names and display aliases** in the User Name Detection dialog box in EUM Administration (see below).

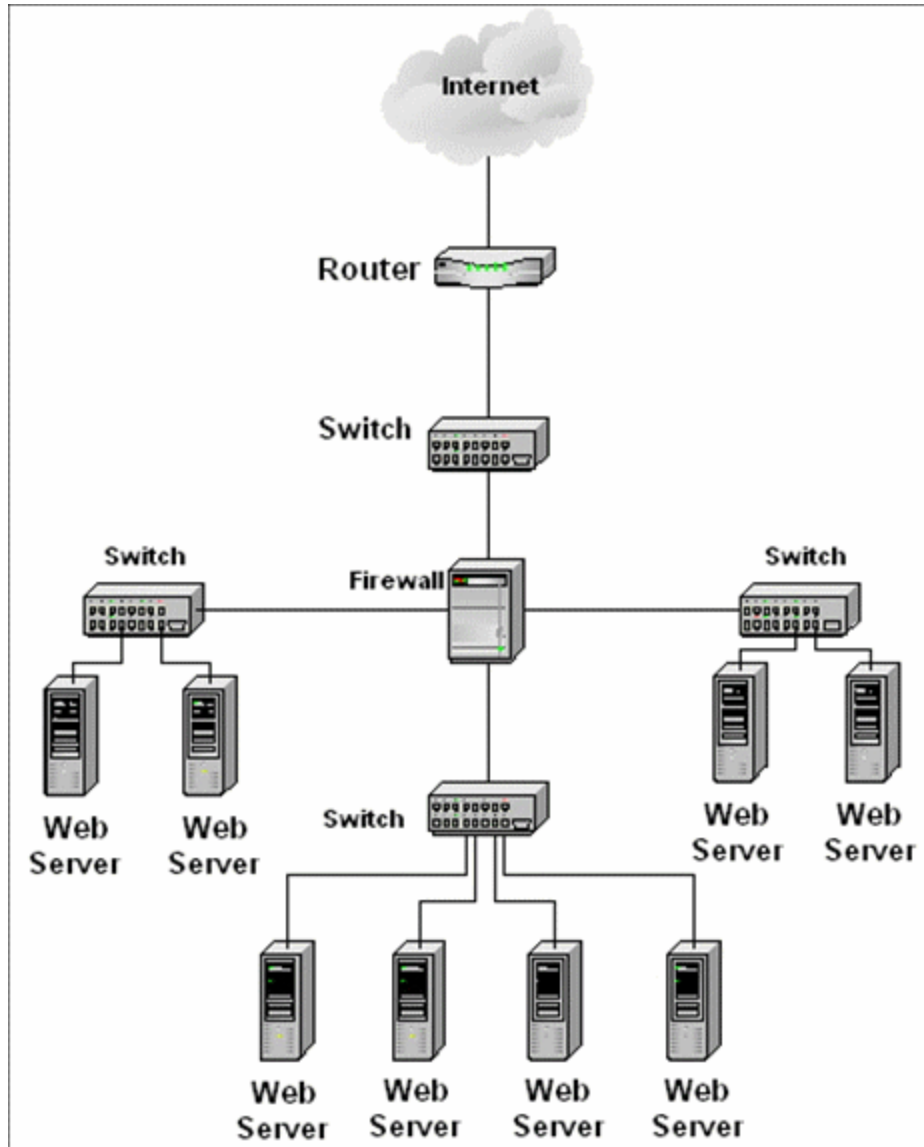


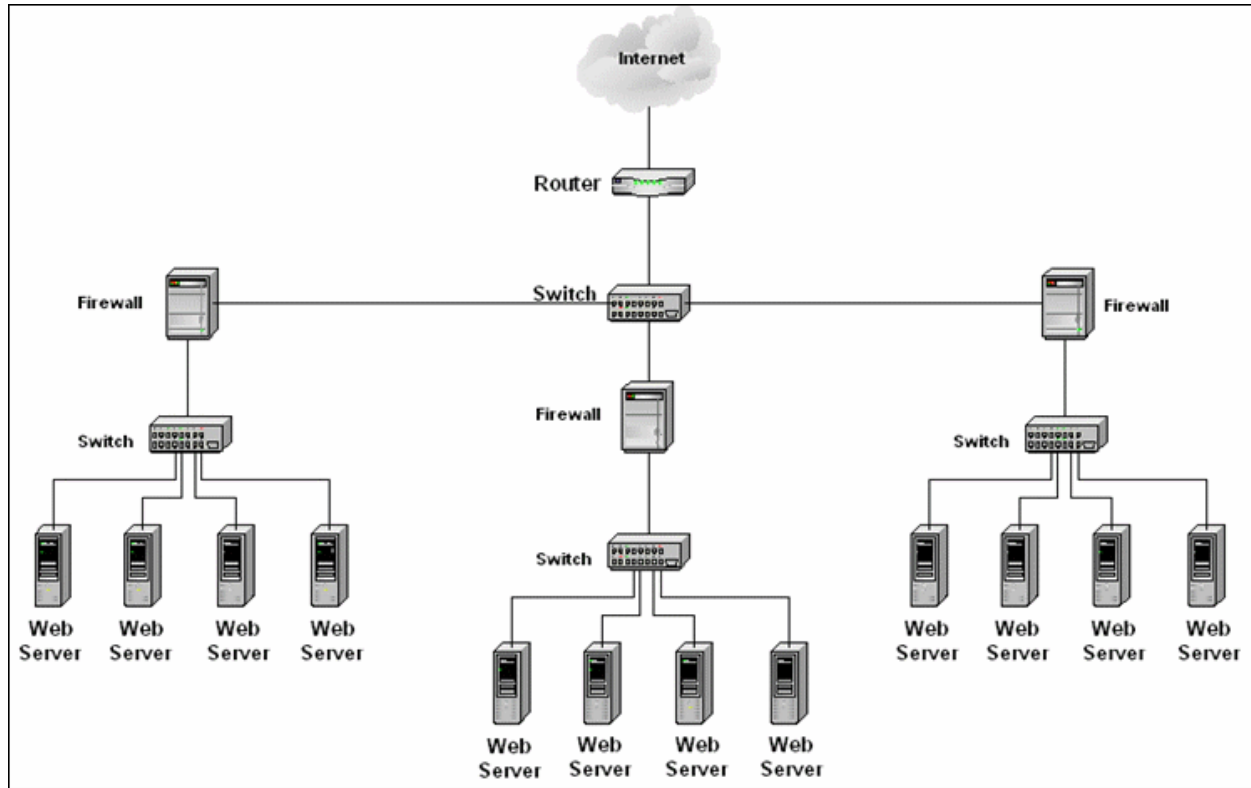
4. Restart the RUM Engine. In the **resolver.log** and **config.manager.log**, check that your configuration changes are correct.

If you do not have a network diagram, the examples in ["Appendix C - Network Examples" on page 36](#) may assist you in determining where to place the probes.

Appendix C - Network Examples







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 Getting Started With RUM - Best Practices (Application Performance Management 9.40)

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 docteam@hpe.com.

We appreciate your feedback!