Monitoring NetScaler HDX AppFlow

eG Innovations Product Documentation

www.eginnovations.com

Total Performance Visibility

Table of Contents

Table of Figures

Chapter 1: Introduction to NetScaler HDX AppFlow Monitoring

The NetScaler appliance is a central point of control for all application traffic in the data center. It collects flow and user-session level information valuable for application performance monitoring, analytics, and business intelligence applications. It also collects web page performance data and database information.

AppFlow is a new flow export standard for finding data pertaining to applications and transactions in the network infrastructure.

AppFlow records contain standard NetFlow or IPFIX information, such as time stamps for the beginning and end of a flow, packet count, and byte count. AppFlow records also contain application-level information (such as HTTP URLs, HTTP request methods and response-status codes, server response time, and latency).

You can configure AppFlow on a NetScaler device. This ensures that AppFlow records are transmitted to an IPv4 collector, using the Internet Protocol Flow Information eXport (IPFIX) format. The collector is where these real-time analytics can be aggregated for providing administrators with greater visibility into application traffic utilization and performance.

eG Enterprise supports AppFlow monitoring. You can configure eG Enterprise to serve as an AppFlow collector for an AppFlow-enabled NetScaler device in your environment. This way, you can have eG Enterprise receive , process, and publish in its console, the application and session-level statistics that AppFlow provides. With the help of these analytics, administrators can closely scrutinize application/desktop session traffic (via HDX virtual channels) and traffic to/from web applications. In the process, administrators can measure the bandwidth used and latency experienced by user accesses to XenApp servers, XenDesktop virtual desktops, and web applications, and determine the root-cause of poor user experience.

For this purpose, eG Enterprise provides two specialized monitoring models - the *NetScaler HDX AppFlow* model and the *NetScaler Web AppFlow* model.

This document elaborates on the *NetScaler HDX AppFlow* model only. For details regarding the *NetScaler Web AppFlow* model, refer to the *Monitoring NetScaler Web AppFlow document* .

The *NetScaler HDX AppFlow* model aggregates the AppFlow metrics on session traffic that flows through HDX virtual channels, and provides useful insights into session performance. Using this document , you can understand how to configure eG Enterprise as an AppFlow collector, what metrics it collects and reports for HDX virtual channel traffic, and how to interpret these analytics.

1.1 How does eG Enterprise Perform AppFlow Monitoring?

To intercept and process AppFlow records exported by NetScaler, eG Enterprise offers an eG AppFlow Collector. The collector runs as a Windows service. Once started, the collector service starts listening for AppFlow records on UDP port 9996 (by default). The NetScaler device in the target infrastructure should be configured to export AppFlow records to the collector. The collector receives these records, processes them, and stores the processed data in binary files.

To analyze and aggregate the data stored in the binary files, the eG remote agent monitoring the NetScaler device serves as the eG AppFlow Analyzer/Aggregator. This agent periodically reads these binary files, pulls statistics on appflow, processes/aggregates these statistics on the basis of HDX channels, users, applications, application/desktop sessions etc., and reports the aggregated data to the eG manager. The eG manager then stores this information in the eG database.

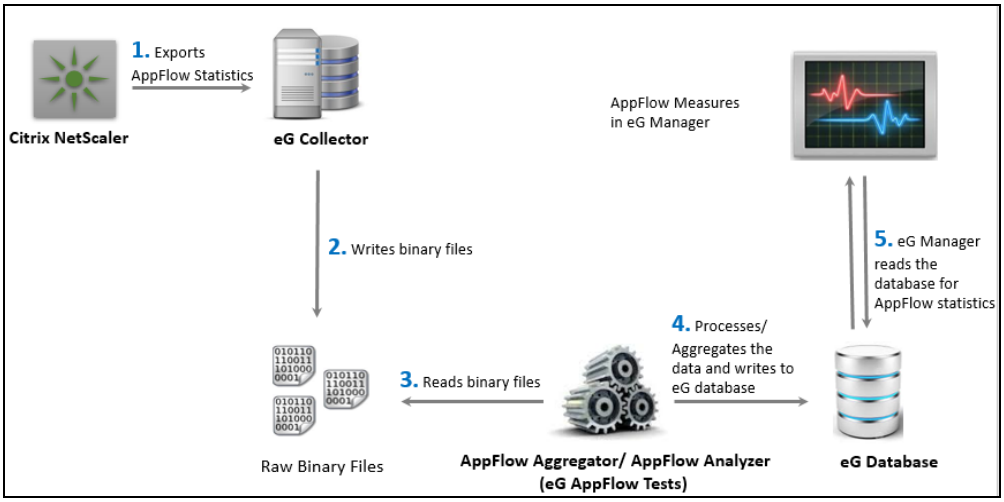


Figure 1.1: How the eG NetFlow Collector Works

For a NetScaler device, the eG manager also presents real-time metrics on traffic and bandwidth in the eG monitoring console using a specialized NetScaler HDX AppFlow monitoring model. If abnormalities are spotted during flow analysis, alerts are generated on this model.

1.2 Licensing

AppFlow monitoring by eG Enterprise is licensed by the number of eG external Agents used for collecting flow data from AppFlow-enabled devices. Licensing is NOT restricted by the number of devices/interfaces exporting flow data.

Each eG external Agent includes one AppFlow Collector, which supports the collection of up to 20,000 flows/second.

Chapter 2: Setting Up AppFlow Monitoring

The broad steps towards setting up AppFlow Monitoring are as follows:

1. Using the eG management console, manage the NetScaler device that will be generating AppFlow records, and assign a remote agent to it.
2. Deploy the eG AppFlow Collector on the same system that hosts the remote agent assigned to the NetScaler device at step 1 .
3. Open the UDP port (9996, by default) on the collector host, so that the collector can intercept and process AppFlow records; then, staeach of rt the collector.

This chapter discusses each of these steps in detail.

Section **2.1**

Section **2.2**

Section **2.3**

2.1 Deploying the eG AppFlow Collector

The next step is to deploy the eG AppFlow Collector. As mentioned already, the collector has to be created as a Windows service. Note that the **collector service has to be created on the same Windows host on which the remote agent monitoring the target NetScaler device has been deployed**.

To create the collector service on the remote agent host, follow the steps below:

1. Login to the Windows system hosting the eG remote agent.
2. Run the command prompt in elevated mode.
3. Go to the <EG_ INSTALL_ DIR>\Netflow\bin directory and execute the **CreateNetflowService.bat** batch file.
4. Successful execution of the batch file will result in the creation of a Windows service named **eGNetFlowAgent**. To confirm the creation of this service, open the **Services** window on the collector host. If you find the **eGNetflowAgent** service displayed therein (see Figure 2.1), it denotes that the collector has been successfully deployed.

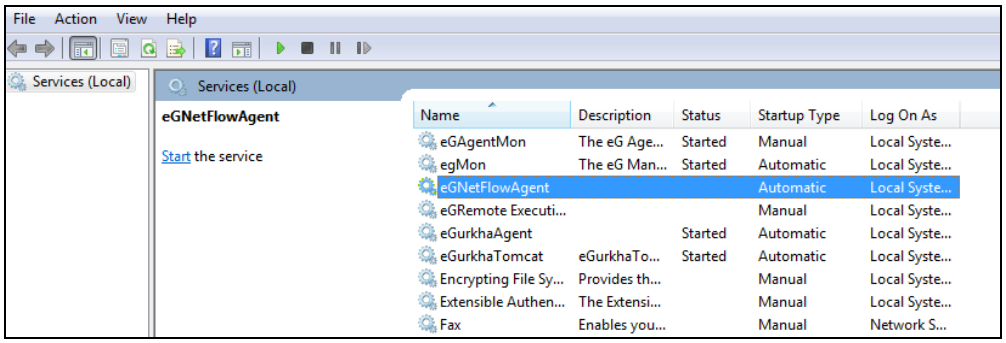


Figure 2.1: The eGNetflowAgent service displayed in the Services window

5. By default, the collector listens on UDP port 9996 for AppFlow records from the managed NetScaler device. To make sure that the AppFlow device communicates with the collector via this port, make sure you open this port on the firewall.
6. Once this is done, proceed to start the collector service. For this, right- click on the **eGNetFlowAgent** service in Figure 2.1 and select the **Start** option from the shortcut menu that appears.
7. On the other hand, if the default UDP port 9996 is already in use, then you will first have to change the listening port of the collector. In this case therefore, skip step 6 above; instead, follow the steps detailed in the Configuring the eG AppFlow Collector to Receive AppFlow Records topic.

2.2 Managing an HDX AppFlow Device in eG Enterprise

To do this, follow the steps below:

1. Login to the eG administrative interface.
2. Follow the Infrastructure -> Components -> Add/Modify menu sequence. In the page that appears next, select *NetScaler HDX AppFlow* as the **Component type** and click the **Add New Component** button to add a new component of that type. Figure 2.2 will then appear.

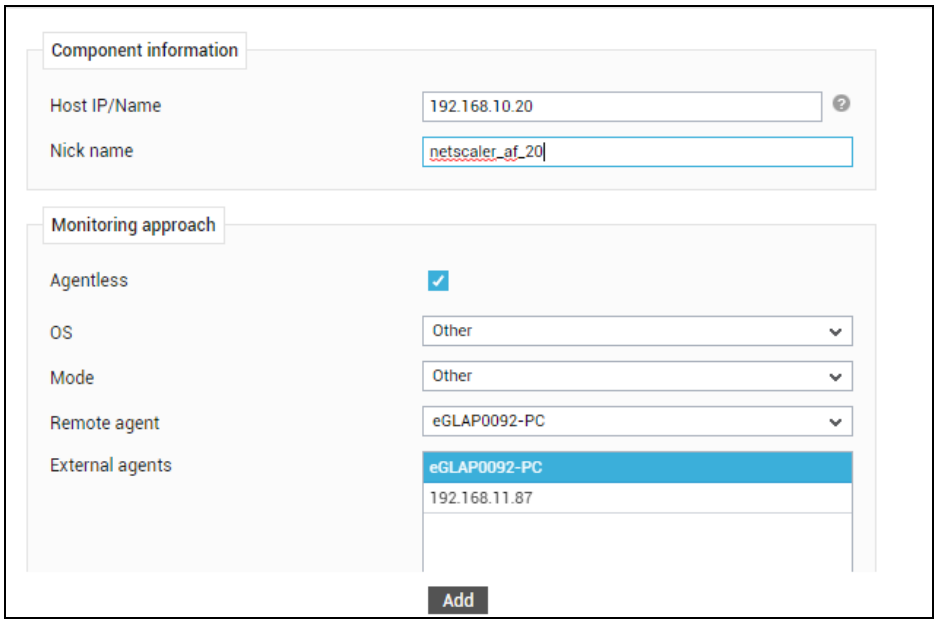


Figure 2.2: Adding an HDX AppFlow Device

3. Specify the IP address and **Nick name** of the AppFlow-enabled NetScaler device to be monitored in Figure 2.2. Then, pick a **Remote Agent** from Figure 2.2 to assign to that device.

   Note:

   Select a remote agent that has been deployed on a Windows host for the purpose of AppFlow monitoring.

4. Finally, click the **Add** button.

2.3 Configuring the eG AppFlow Collector to Receive AppFlow Records

By default, the collector listens for AppFlow records on UDP port 9996. **Make sure you open that port on the firewall,** so that the managed AppFlow-enabled NetScaler device is able to communicate with the collector via that port. If the default port is already in use in your environment, then you can change the listening port of the collector. For this, do the following:

1. Login to the system hosting the eG AppFlow Collector.
2. Edit the **NetFlow.properties** file in the <EG_INSTALL_DIR>\Netflow\config directory.
3. Look for the **net.bind.port** entry in that file. This will be set to *9996* by default. Change the port number against that entry and save the file.

4. Finally, start the collector service. For that open the Services window, right-click on the **eGNetFlowAgent** service therein, and select the **Start** option from the shortcut menu that appears.

Once the collector service is started, the AppFlow collector will begin receiving AppFlow records from the managed NetScaler device. These records are then processed and stored by the collector in binary files. At configured frequencies, each HDX AppFlow test that the eG remote agent runs, will read these binary files to pull and aggregate AppFlow statistics. The statistics so aggregated are then reported to the eG manager, which publishes these metrics in the eG monitoring console. To know what tests the remote agent runs, what metrics it collects, and how these metrics are grouped and presented in the eG monitoring console, refer to the Monitoring Citrix NetScaler HDX AppFlow topic.

Chapter 3: Monitoring Citrix NetScaler HDX AppFlow

eG Enterprise provides a specialized *Citrix HDX AppFlow* monitoring model for monitoring HDX AppFlow.

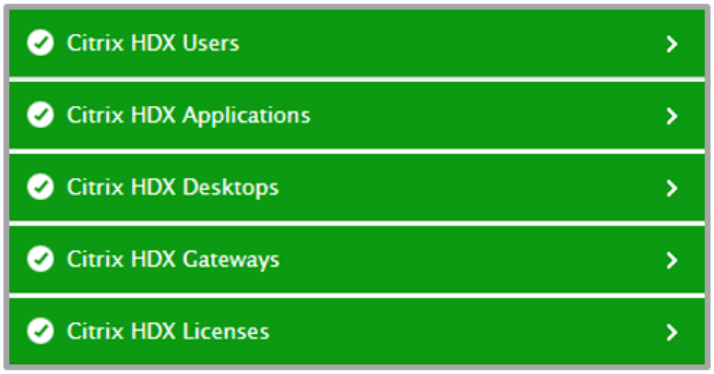


Figure 3.1: The layer model of Citrix HDX AppFlow

Each layer of Figure 3.1 above is mapped to tests that intercept AppFlow records to report a wide variety of useful metrics. Using these metrics, administrators can find quick and accurate answers for the following performance queries:

- Is any HDX virtual channel hogging bandwidth? If so, which one?
- What is the current session load on the XenApp/XenDesktop environment? Is the workload increasing consistently?
- Are any application/desktop session logging out suddenly? If so, which ones?
- Is any application user experiencing a time lag when accessing the application? If so, which user is affected and what could be delaying his/her access – the client side network? the server side network? the server hosting the applications? Or the NetScaler appliance?
- Is any desktop user experiencing slowness during desktop accesses? If so, which user has been impacted and what could be impacting his/her experience with the virtual desktop? In which ICA session was this slowness observed?
- Is any application session consuming bandwidth excessively? If so, Which applications were accessed by the user over that session?
- Which applications were launched by a user?
- Were any applications terminated by a user? If so, which ones?

- Is any desktop user consuming bandwidth excessively? If so, over which session did the abnormal bandwidth consumption occur? Which desktop was accessed by the user over that session?
- Is any application/desktop session inactive? How long is that session being inactive?
- Is any application taking too long to launch? Which application is it? Which user is impacted by the launch delay? Which server is the application running on?
- Has NetScaler been configured with adequate SSL VPN licenses?

This chapter discusses each layer of Figure 3.1 and the tests mapped to every layer.

3.1 The Citrix HDX Licenses Layer

Use the test mapped to this layer to be proactively alerted to license shortages on the NetScaler appliance.

Figure 3.2: The test mapped to the Citrix HD Licenses layer

3.1.1 Citrix HDX Licenses Test

Using this test, administrators can understand how the NetScaler appliance utilizes different types of licneses, and accordingly plan license requirements for the future. Moreover, the test also promptly alerts administrators if the NetScaler appliance is about to run out of any type of licenses.

Target of the test : An AppFlow-enabled NetScaler Appliance

Agent deploying the test : A remote agent

Outputs of the test : One set of results for each type of license managed by the NetScaler appliance

Configurable parameters for the test

Parameter	Description
Test period	How often should the test be executed. **It is recommended that you set the test period to 5 minutes. This is because, the eG AppFlow Collector is capable of capturing and aggregating AppFlow data related to the last 5 minutes only.**
Host	The host for which the test is to be configured.
Cluster IPs	**This parameter applies only if the NetScaler appliance being monitored is part of a NetScaler cluster.** In this case, configure this parameter with a comma-separated list of IP addresses of all other nodes in that cluster. If the monitored NetScaler appliance is down/unreachable, then the eG AppFlow Collector uses the **Cluster IPs** configuration to figure out which other node in the cluster it should connect to for pulling AppFlow statistics. Typically, the collector attempts to connect to every IP address that is configured against **Cluster IPs**, in the same sequence in which they are specified. Metrics are pulled from the first cluster node that the collector successfully establishes a connection with.
Enable Logs	This flag is set to **No** by default. This means that, by default, the eG agent does not create AppFlow logs. You can set this flag to **Yes** to enable AppFlow logging. If this is done, then the eG agent automatically writes the raw AppFlow records it reads from the collector into individual CSV files. These CSV files are stored in the *<EG_ AGENT_INSTALL_DIR>\NetFlow\data\<IP_of_Monitored_ NetScaler>\hdxappflow\actual_csv* folder on the eG agent host. These CSV files provide administrators with granular insights into the HDX appflows, thereby enabling effective troubleshooting. **Note:** By default, the eG agent creates a maximum of 10 CSV files in the **actual_csv** folder. Beyond this point, the older CSV files will be automatically deleted by the eG agent to accommodate new files with current data. Likewise, a single CSV file can by default contain a maximum of 99999 records only. If the records to be written exceed this default value, then the eG agent automatically creates another CSV file to write the data. If required, you can overwrite these default settings . For this, do the following: 1. Login to the eG agent host. 2. Edit the **Netflow.Properties** file in the <EG_AGENT_INSTALL_ DIR>\NetFlow\config directory. 3. In the file, look for the parameter, **csv_file_retention_count**.

Parameter	Description
	4. This is the parameter that governs the maximum number of CSV files that can be created in the **auto_csv** folder. By default, this parameter is set to 10. If you want to retain more number of CSV files at any given point in time, you can increase the value of this parameter. If you want to retain only a few CSV files, then decrease the value of this parameter. 5. Next, look for the parameter, **csv_max_flow_record_per_file**. 6. This is the parameter that governs the number of flow records that can be written to a single CSV. By default, this parameter is set to 99999. If you want a single file to accommodate more records, so that the creation of new CSVs is delayed, then increase the value of this parameter. On the other hand, if you want to reduce the capacity of a CSV file, so that new CSVs are quickly created, then decrease the value of this parameter. 7. Finally, save the file.

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
Total licenses	Indicates the total number of licenses of this type configured on the NetScaler appliance.	Number	
Licenses in use	Indicates the number of licenses of this type currently utilized by the ICA sessions.	Number	If the number of Licenses in use is equal to or close to the Total licenses count for any license type, it is a cause for concern, as it indicates that the environment is running out of a specific type of license.
License usage	Indicates the percentage of licenses of this type in use.	Percent	A value close to 100% is a cause for concern, as it indicates that the environment is running out of licenses of this type.

Measurement	Description	Measurement Unit	Interpretation
Available licenses	Indicates the number of licenses of this type that are still be used.	Number	A high value is desired for this measure. A value close to 0 implies that not many licenses are available for the use of subsequent ICA sessions. You may want to buy more licenses to pre-empt this unpleasant outcome.

3.2 The Citrix HDX Gateways Layer

The test mapped to this layer tracks the session, user, and application/desktop load handled by the target NetScaler appliance, and also reports the bandwidth consumed by the appliance to process this load.

Figure 3.3: The test mapped to the Citrix HDX Gateways layer

3.2.1 Citrix HDX Gateways Test

To understand the workload of a NetScaler appliance and to assess whether/not the appliance has the adequate bandwidth resources to service the load, administrators should track the ICA sessions, users, and the application/desktop launches handled by that NetScaler appliance, and also measure the bandwidth usage by the appliance. This is what the **Citrix HDX Gateways** test helps administrators achieve. This test measures the session, user, and application/desktop load handled by the target NetScaler appliance, and also reports the bandwidth consumed by the appliance to process this load. This way, the test reveals potential bandwidth contentions, thus prompting the administrator to resize the appliance commensurate to its load.

Target of the test : An AppFlow-enabled NetScaler Appliance

Agent deploying the test : A remote agent

Outputs of the test : One set of results for the NetScaler appliance configured in the environment

Configurable parameters for the test

Parameter	Description
Test period	How often should the test be executed. **It is recommended that you set the test period to 5 minutes. This is because, the eG AppFlow Collector is capable of capturing and aggregating AppFlow data related to the last 5 minutes only.**
Host	The host for which the test is to be configured.
Cluster IPs	**This parameter applies only if the NetScaler appliance being monitored is part of a NetScaler cluster.** In this case, configure this parameter with a comma-separated list of IP addresses of all other nodes in that cluster. If the monitored NetScaler appliance is down/unreachable, then the eG AppFlow Collector uses the **Cluster IPs** configuration to figure out which other node in the cluster it should connect to for pulling AppFlow statistics. Typically, the collector attempts to connect to every IP address that is configured against **Cluster IPs**, in the same sequence in which they are specified. Metrics are pulled from the first cluster node that the collector successfully establishes a connection with.
Enable Logs	This flag is set to **No** by default. This means that, by default, the eG agent does not create AppFlow logs. You can set this flag to **Yes** to enable AppFlow logging. If this is done, then the eG agent automatically writes the raw AppFlow records it reads from the collector into individual CSV files. These CSV files are stored in the *<EG_AGENT_INSTALL_DIR>\NetFlow\data\<IP_of_Monitored_NetScaler>\hdxappflow\actual_csv* folder on the eG agent host. These CSV files provide administrators with granular insights into the HDX appflows, thereby enabling effective troubleshooting. **Note:** By default, the eG agent creates a maximum of 10 CSV files in the **actual_csv** folder. Beyond this point, the older CSV files will be automatically deleted by the eG agent to accommodate new files with current data. Likewise, a single CSV file can by default contain a maximum of 99999 records only. If the records to be written exceed this default value, then the eG agent automatically creates another CSV file to write the data. If required, you can overwrite these default settings . For this, do the following: 1. Login to the eG agent host. 2. Edit the **Netflow.Properties** file in the <EG_AGENT_INSTALL_DIR>\NetFlow\config directory. 3. In the file, look for the parameter, **csv_file_retention_count**.

Parameter	Description
	4. This is the parameter that governs the maximum number of CSV files that can be created in the **auto_csv** folder. By default, this parameter is set to 10. If you want to retain more number of CSV files at any given point in time, you can increase the value of this parameter. If you want to retain only a few CSV files, then decrease the value of this parameter. 5. Next, look for the parameter, **csv_max_flow_record_per_file**. 6. This is the parameter that governs the number of flow records that can be written to a single CSV. By default, this parameter is set to 99999. If you want a single file to accommodate more records, so that the creation of new CSVs is delayed, then increase the value of this parameter. On the other hand, if you want to reduce the capacity of a CSV file, so that new CSVs are quickly created, then decrease the value of this parameter. 7. Finally, save the file.

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
Unique users	Indicates the count of distinct users whose requests were processed by the NetScaler appliance.	Number	This is a good indicator of the user load on the NetScaler appliance.
Avg bandwidth	Indicates the bandwidth used by the NetScaler appliance	Kbps	An abnormally high value for this measure is indicative of excessive bandwdith usage by the appliance. If the value of this measure is consistently high, it could indicate that the appliance requires more processing power to service its workload. You may want to consider resizing the appliance to prevent this problem.
Active sessions:	Indicates the number of application/desktop	Number	This is a good indicator of the session load on the appliance.

Measurement	Description	Measurement Unit	Interpretation
	sessions that are currently active on the appliance.		In the event of a session overload, check the value of the Active application sessions measure to figure out what contributed to the overload - application sessions? or desktop sessions?
New session launches	Indicates the count of sessions launched on XenApp servers and XenDesktop desktops via the appliance during the current measurement period.	Number	
Session terminates	Indicates the number of sessions that were terminated on XenDesktop desktops via the appliance.	Number	
Active application sessions	Indicates the number of application sessions currently active on XenApp servers managed by the appliance.	Number	
New application sessions	Indicates the number of application sessions handled by the NetScaler appliance for XenApp servers, during the current measurement period.	Number	
Active applications	Indicates the number of applications currently accessed on XenApp servers via this appliance.	Number	In the event of a session overload, check the value of this measure to figure out what contributed to the overload - application sessions? or desktop sessions?
New application launches	Indicates the number of applications that were launched via the appliance, during the current	Number	

Measurement	Description	Measurement Unit	Interpretation
	measurement period.		
Application terminates	Indicates the number of applications that were terminated via the appliance, during the current measurement period.	Number	
Active desktops :	Indicates the number of desktops into which users have currently logged in via this NetScaler appliance.	Number	In the event of an overload, compare the value of this measure with that of the Active applications measure to know what caused the overload - too many open applications? or open desktops?
New desktop launches	Indicates the number of desktops that were launched via the NetScaler appliance, during the current measurement period.	Number	
Desktop terminates	Indicates the number of desktops that were terminated via the appliance during the current measurement period.	Number	

3.3 The Citrix HDX Desktops Layer

The tests mapped to this layer tracks desktop sessions on the NetScaler appliance and measures the experience of each desktop user/session.

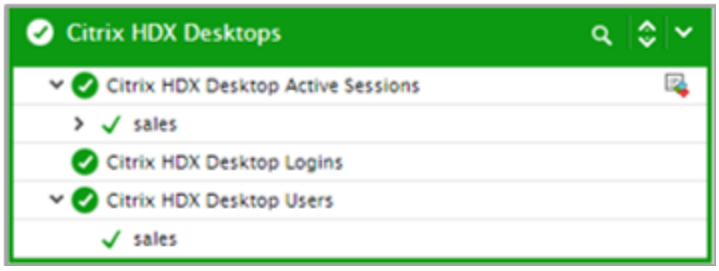


Figure 3.4: The tests mapped to the Citrix HDX Desktops layer

3.3.1 Citrix HDX Desktop Active Sessions Test

In order to ensure that the user experience with desktops deployed on a XenDesktop environment remains 'superlative' at all times, administrators should be able to proactively detect potential slowdowns when accessing desktops, precisely pinpoint the user session affected by the slowdown, accurately isolate the root-cause of such slowness, and rapidly initiate measures to eliminate the root-cause. The **Citrix HDX Desktop Active Sessions** test facilitates all the above, and thus assures users of uninterrupted desktop access!

For a user session that is currently active on a XenDesktop virtual desktop, this test measures session latencies and leads you to the probable cause of session slowness (if any) - is it the network? the server hosting the desktops? If a latent network is causing the slowness, then the test provides administrators with detailed insights into network performance and enables them to rapidly figure out where the bottleneck lies - on the client-side network? or on the server-side network? This way, the test promptly leads administrators to slow user sessions, and also reveals what is causing the slowness, so that administrators can initiate the right steps to enhance user experience with desktops.

Target of the test : An AppFlow-enabled NetScaler Appliance

Agent deploying the test : A remote agent

Outputs of the test : One set of results for each session for every user to a XenDesktop virtual desktop

First level descriptor: User name

Second level descriptor: Session GUID

A desktop session is identified by a separate session GUID - one each for every desktop that is accessed

Configurable parameters for the test

Parameter	Description
Test period	How often should the test be executed. **It is recommended that you set the test period to 5 minutes. This is because, the eG AppFlow Collector is capable of capturing and aggregating AppFlow data related to the last 5 minutes only.**
Host	The host for which the test is to be configured.
Cluster IPs	**This parameter applies only if the NetScaler appliance being monitored is part**

Parameter	Description
	of a NetScaler cluster. In this case, configure this parameter with a comma-separated list of IP addresses of all other nodes in that cluster. If the monitored NetScaler appliance is down/unreachable, then the eG AppFlow Collector uses the **Cluster IPs** configuration to figure out which other node in the cluster it should connect to for pulling AppFlow statistics. Typically, the collector attempts to connect to every IP address that is configured against **Cluster IPs**, in the same sequence in which they are specified. Metrics are pulled from the first cluster node that the collector successfully establishes a connection with.
Enable Logs	This flag is set to **No** by default. This means that, by default, the eG agent does not create AppFlow logs. You can set this flag to **Yes** to enable AppFlow logging. If this is done, then the eG agent automatically writes the raw AppFlow records it reads from the collector into individual CSV files. These CSV files are stored in the *<EG_AGENT_INSTALL_DIR>\NetFlow\data\<IP_of_Monitored_NetScaler>\hdxappflow\actual_csv* folder on the eG agent host. These CSV files provide administrators with granular insights into the HDX appflows, thereby enabling effective troubleshooting. **Note:** By default, the eG agent creates a maximum of 10 CSV files in the **actual_csv** folder. Beyond this point, the older CSV files will be automatically deleted by the eG agent to accommodate new files with current data. Likewise, a single CSV file can by default contain a maximum of 99999 records only. If the records to be written exceed this default value, then the eG agent automatically creates another CSV file to write the data. If required, you can overwrite these default settings . For this, do the following: 1. Login to the eG agent host. 2. Edit the **Netflow.Properties** file in the <EG_AGENT_INSTALL_DIR>\NetFlow\config directory. 3. In the file, look for the parameter, **csv_file_retention_count**. 4. This is the parameter that governs the maximum number of CSV files that can be created in the **auto_csv** folder. By default, this parameter is set to 10. If you want to retain more number of CSV files at any given point in time, you can increase the value of this parameter. If you want to retain only a few CSV files, then decrease the value of this parameter.

Parameter	Description
	5. Next, look for the parameter, **csv_max_flow_record_per_file**. 6. This is the parameter that governs the number of flow records that can be written to a single CSV. By default, this parameter is set to 99999. If you want a single file to accommodate more records, so that the creation of new CSVs is delayed, then increase the value of this parameter. On the other hand, if you want to reduce the capacity of a CSV file, so that new CSVs are quickly created, then decrease the value of this parameter. 7. Finally, save the file.
DD Frequency	Refers to the frequency with which detailed diagnosis measures are to be generated for this test. The default is *1:1*. This indicates that, by default, detailed measures will be generated every time this test runs, and also every time the test detects a problem. You can modify this frequency, if you so desire. Also, if you intend to disable the detailed diagnosis capability for this test, you can do so by specifying *none* against **DD Frequency**.
Detailed Diagnosis	To make diagnosis more efficient and accurate, the eG Enterprise suite embeds an optional detailed diagnostic capability. With this capability, the eG agents can be configured to run detailed, more elaborate tests as and when specific problems are detected. To enable the detailed diagnosis capability of this test for a particular server, choose the **On** option. To disable the capability, click on the **Off** option. The option to selectively enable/disable the detailed diagnosis capability will be available only if the following conditions are fulfilled: • The eG manager license should allow the detailed diagnosis capability • Both the normal and abnormal frequencies configured for the detailed diagnosis measures should not be 0.

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
Session status	Indicates the current status of this session.		The values that this measure can take and the numeric values that correspond to each measure value are listed in the table below:

<table>
<tr><th>Measurement</th><th>Description</th><th>Measurement Unit</th><th>Interpretation</th></tr>
<tr><td></td><td></td><td></td><td>
<table>
<tr><th>Measure Value</th><th>Numeric Value</th></tr>
<tr><td>Active</td><td>0</td></tr>
<tr><td>SR successful</td><td>1000</td></tr>
<tr><td>Existing ICA session got terminated</td><td>1001</td></tr>
<tr><td>Existing ICA connection got terminated and SR failed</td><td>1002</td></tr>
<tr><td>Existing ICA connection terminated and SR failed and client is trying to do ACR and is successful</td><td>1003</td></tr>
</table>
Note:
Typically, this test reports the Measure Values in the table above to indicate session status. In the graph of this measure however, the same is indicated using the numeric equivalents only.
</td></tr>
<tr><td>ICA client type</td><td>Indicates the type of client on which the ICA client/receiver used by this session is running.</td><td></td><td>The values that this measure can take and the numeric values that correspond to each measure value are listed in the table below:
<table>
<tr><th>Measure Value</th><th>Numeric Value</th></tr>
<tr><td>Windows</td><td>1</td></tr>
<tr><td>Console</td><td>3</td></tr>
<tr><td>Text Terminals</td><td>4</td></tr>
<tr><td>MVGA Terminals</td><td>7</td></tr>
<tr><td>Java</td><td>8</td></tr>
<tr><td>EPOC</td><td>10</td></tr>
</table>
</td></tr>
</table>

<table>
<tr><th>Measurement</th><th>Description</th><th>Measurement Unit</th><th>Interpretation</th></tr>
<tr><td></td><td></td><td></td><td>
<table>
<tr><td>OS/2</td><td>11</td></tr>
<tr><td>DOS 32</td><td>14</td></tr>
<tr><td>UNIX/LINUX</td><td>81</td></tr>
<tr><td>MAC</td><td>82</td></tr>
<tr><td>iPhone</td><td>83</td></tr>
<tr><td>Android</td><td>84</td></tr>
<tr><td>Blackberry</td><td>85</td></tr>
<tr><td>Windows Metro</td><td>86</td></tr>
<tr><td>Windows Mobile</td><td>87</td></tr>
<tr><td>Blackberry Playbook</td><td>88</td></tr>
<tr><td>OEM</td><td>100</td></tr>
<tr><td>HTML5</td><td>257</td></tr>
<tr><td>Java</td><td>261</td></tr>
<tr><td>Windows CE</td><td>7945</td></tr>
<tr><td>Terminal Client</td><td>32768</td></tr>
<tr><td>Windows CE - Wyse</td><td>32940</td></tr>
<tr><td>Thin OS - Wyse</td><td>32993</td></tr>
<tr><td>Client requires license number</td><td>16384</td></tr>
</table>
Note:
Typically, this test reports the Measure Values in the table above to indicate ICA client type. In the graph of this measure however, the same is indicated using the numeric equivalents only.
</td></tr>
<tr><td>RTT</td><td>Indicates the screen lag experienced by this session while interacting with desktops.</td><td>Msecs</td><td>A high value for this measure is indicative of the poor quality of a user’s experience with desktops.
To know the reason for this below-par UX, compare the value of the WAN latency,</td></tr>
</table>

Measurement	Description	Measurement Unit	Interpretation
			DC latency, and Host delay measures of that session.
WAN latency	Indicates the average latency experienced by this user session due to problems with the client side network.	Msecs	A high value for this measure indicates that the client side network is slow. If the value of the *RTT* measure is abnormally high for a session, you can compare the value of this measure with that of the *DC latency and Host delay, and* measures of that user session to know what is causing the slowness – is it the client side network? the server side network? or the server hosting the desktops?
DC latency	Indicates the average latency experienced by this session due to problems with the server side network.	Msecs	A high value for this measure indicates that the server side network is slow. If the value of the *RTT* measure is abnormally high for a session, you can compare the value of this measure with that of the *WAN latency and Host delay, cy* measures of that session to know what is causing the slowness – is it the client side network? the server side network? or the server hosting the desktops?
Host delay	Indicates the delay that this session experienced when waiting for the host to process the packets.	Msecs	A high value for this measure indicates a processing bottleneck with the server hosting the applications. If the value of the *RTT* measure is abnormally high for a session, you can compare the value of this measure with that of the *WAN latency and DC latency,* measures to know what is causing the slowness – is it the client side network? the server side network? or the server hosting the desktops?
Bandwidth	Indicates the rate at which data is transferred over this ICA session.	Kbps	Ideally, the value of this measure should be low.

Measurement	Description	Measurement Unit	Interpretation
			A high value indicates excessive bandwidth usage by the session. Compare the value of this measure across sessions to know which session is consuming bandwidth excessively.
Bytes	Indicates the total bytes consumed by this session.	Bytes	Compare the value of this measure across sessions to know which session has the maximum throughput and which has the least.
Client side retransmits	Indicates the number of packets retransmitted on the client side connection during the last measurement period.	Number	Ideally, the value of these measures should be 0.
Server side retransmits	Indicates the number of packets retransmitted on the server side connection during the last measurement period.	Number	
ACR counts	Indicates the total number of times the client automatically reconnected the user to this session.	Number	The Automatic Client Reconnect (ACR) policy setting, when enabled, allows automatic reconnection by the same client after a connection has been interrupted. Allowing automatic client reconnect allows users to resume working where they were interrupted when a connection was broken. Automatic reconnection detects broken connections and then reconnects the users to their sessions.
Session reconnects	Indicates the number of times this session reconnected.	Number	This measure includes only those times a user reconnected to a disconnected session by mechanisms other than the ACR setting.
Client side NS delay	Indicates the average latency experienced by	Msecs	A high value for these measures indicates a processing bottleneck with the NetScaler appliance.

Measurement	Description	Measurement Unit	Interpretation
	this session, which was caused by the NetScaler appliance when ICA traffic flowed from client network to server. network.		If the value of the *WAN latency* measure is abnormally high for a session, you can check the value of the *Client side NS delay* measure to determine if the network delays noticed on the client side were caused by the NetScaler's lethargy in processing traffic from the client. If the value of the *DC latency* measure is abnormally high for a session, you can check the value of the *Server side NS delay* measure to determine if the network delays noticed on the server side were caused by the NetScaler's lethargy in processing traffic from the server.
Server side NS delay	Indicates the average latency experienced by this session, which was caused by the NetScaler appliance when ICA traffic flowed from server network to client network.	Msecs	

The detailed diagnosis of the *Session status* measure provides additional details of a user session. If the status of a session is abnormal, you can use these details to know from which client the user is connecting, the client type and version, which desktop the user is connecting to, the start time, and the uptime of the session. This will help in troubleshooting the abnormal session status.

Details of desktop user

USER NAME	VDI IMAGE NAME	SESSION GUID	CONNECTION ID	SOURCE ADDRESS	DESTINATION ADDRESS	DESKTOP STARTUP TIME	SESSION DURATION (MINUTES)	CLIENT VERSION	DOMAIN
Feb 26, 2016 11:32:03									
sales	XENAPP7V6	0b7dd572-d9f3-4683-b203-d21ebdaff2e5	2497567	192.168.9.168	192.168.8.185	26/02/2016 11:16:25 IST	15	14.0.0.91	citrix

Figure 3.5: The detailed diagnosis of the Session status measure reported by the Citrix HDX Desktop Active Sessions test

3.3.2 Citrix HDX Desktop Logins Test

Use this test to determine the current session load on the virtual desktops in your XenDekstop infrastructure. The detailed diagnosis of this test also reveals who launched the sessions, when, from which client, to which server, the session duration, and the overall quality of each user's session. This way, administrators can quickly identify which user's experience with virtual desktops is below-par and what is causing it – latent client-side network connection? a slow server-side

network connection? or a bandwidth-intensive communication over the network? The test also tracks session logouts, and points to abnormal logouts.

Target of the test : An AppFlow-enabled NetScaler appliance

Agent deploying the test : A remote agent

Outputs of the test : One set of results for the NetScaler appliance being monitored

Configurable parameters for the test

Parameter	Description
Test period	How often should the test be executed. **It is recommended that you set the test period to 5 minutes. This is because, the eG AppFlow Collector is capable of capturing and aggregating AppFlow data related to the last 5 minutes only.**
Host	The host for which the test is to be configured.
Cluster IPs	**This parameter applies only if the NetScaler appliance being monitored is part of a NetScaler cluster.** In this case, configure this parameter with a comma-separated list of IP addresses of all other nodes in that cluster. If the monitored NetScaler appliance is down/unreachable, then the eG AppFlow Collector uses the **Cluster IPs** configuration to figure out which other node in the cluster it should connect to for pulling AppFlow statistics. Typically, the collector attempts to connect to every IP address that is configured against **Cluster IPs**, in the same sequence in which they are specified. Metrics are pulled from the first cluster node that the collector successfully establishes a connection with.
Enable Logs	This flag is set to **No** by default. This means that, by default, the eG agent does not create AppFlow logs. You can set this flag to **Yes** to enable AppFlow logging. If this is done, then the eG agent automatically writes the raw AppFlow records it reads from the collector into individual CSV files. These CSV files are stored in the *<EG_AGENT_INSTALL_DIR>\NetFlow\data\<IP_of_Monitored_NetScaler>\hdxappflow\actual_csv* folder on the eG agent host. These CSV files provide administrators with granular insights into the HDX appflows, thereby enabling effective troubleshooting. **Note:** By default, the eG agent creates a maximum of 10 CSV files in the **actual_csv** folder. Beyond this point, the older CSV files will be automatically deleted by the eG agent to accommodate new files with current data. Likewise, a single CSV file can by default contain a maximum of 99999 records only. If the records to be written exceed this default value, then the eG agent automatically creates another CSV file to write the

Parameter	Description
	data. If required, you can overwrite these default settings . For this, do the following: 1. Login to the eG agent host. 2. Edit the **Netflow.Properties** file in the <EG_AGENT_INSTALL_ DIR>\NetFlow\config directory. 3. In the file, look for the parameter, **csv_file_retention_count**. 4. This is the parameter that governs the maximum number of CSV files that can be created in the **auto_csv** folder. By default, this parameter is set to 10. If you want to retain more number of CSV files at any given point in time, you can increase the value of this parameter. If you want to retain only a few CSV files, then decrease the value of this parameter. 5. Next, look for the parameter, **csv_max_flow_record_per_file**. 6. This is the parameter that governs the number of flow records that can be written to a single CSV. By default, this parameter is set to 99999. If you want a single file to accommodate more records, so that the creation of new CSVs is delayed, then increase the value of this parameter. On the other hand, if you want to reduce the capacity of a CSV file, so that new CSVs are quickly created, then decrease the value of this parameter. 7. Finally, save the file.
DD Frequency	Refers to the frequency with which detailed diagnosis measures are to be generated for this test. The default is *1:1*. This indicates that, by default, detailed measures will be generated every time this test runs, and also every time the test detects a problem. You can modify this frequency, if you so desire. Also, if you intend to disable the detailed diagnosis capability for this test, you can do so by specifying *none* against **DD Frequency**.
Detailed Diagnosis	To make diagnosis more efficient and accurate, the eG Enterprise suite embeds an optional detailed diagnostic capability. With this capability, the eG agents can be configured to run detailed, more elaborate tests as and when specific problems are detected. To enable the detailed diagnosis capability of this test for a particular server, choose the **On** option. To disable the capability, click on the **Off** option. The option to selectively enable/disable the detailed diagnosis capability will be

Parameter	Description
	available only if the following conditions are fulfilled: • The eG manager license should allow the detailed diagnosis capability • Both the normal and abnormal frequencies configured for the detailed diagnosis measures should not be 0.

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
Current sessions	Indicates the number of desktop user sessions that are currently active on the NetScaler appliance.	Number	This is a good indicator of the session load on the virtual desktops managed by the NetScaler appliance. To know which users are contributing to the load, use the detailed diagnosis of this measure. The detailed diagnosis additionally reveals when each user logged in, from which client, to which desktop, the session duration, and the quality of the user's session. This way, administrators can quickly identify which application user's experience is below-par and what is causing it - latent client-side network connection? a slow server-side network connection? or a bandwidth-intensive communication over the network?
New sessions	Indicates the number of desktop user logins since the last measurement period.	Number	A consistent zero value could indicate a connection issue. If this measure reports a non-zero value, use the detailed diagnosis of the measure to know which user logged into which virtual desktop, when.
Percent of new sessions	Indicates the percentage of current desktop sessions that logged in during the	Percent	

Measurement	Description	Measurement Unit	Interpretation
	last measurement period.		
Sessions logging out	Indicates the number of desktop sessions that logged out.	Number	If all the current sessions suddenly log out, it indicates a problem condition that requires investigation. The detailed diagnosis of this measure lists the sessions that logged out. The quality of each user's session is also revealed. This way, administrators can quickly identify which application user's experience is below-par and what is causing it – latent client-side network connection? a slow server-side network connection? or a bandwidth-intensive communication over the network?

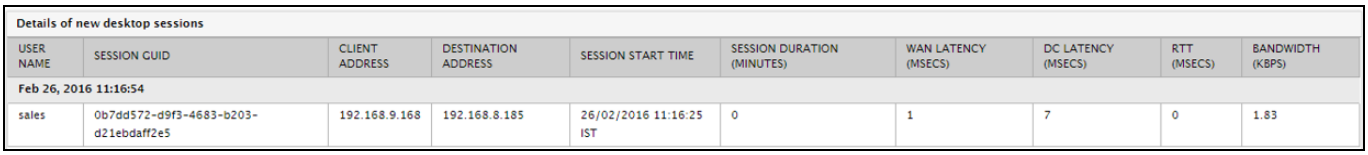
To know which users are contributing to the load on XenDesktop, use the detailed diagnosis of the *Current sessions* measure. The detailed diagnosis additionally reveals when each user logged in, from which client, to which desktop, the session duration, and the quality of the user's session. This way, administrators can quickly identify which desktop user's experience is below-par and what is causing it - a latent client-side network connection? a slow server-side network connection? or a bandwidth-intensive communication over the network?

Details of connected desktop sessions

USER NAME	SESSION GUID	CLIENT ADDRESS	DESTINATION ADDRESS	SESSION START TIME	SESSION DURATION (MINUTES)	WAN LATENCY (MSECS)	DC LATENCY (MSECS)	RTT (MSECS)	BANDWIDTH (KBPS)
Feb 26, 2016 11:36:54									
sales	0b7dd572-d9f3-4683-b203-d21ebdaff2e5	192.168.9.168	192.168.8.185	26/02/2016 11:16:25 IST	20	164	41	0	0.02

Figure 3.6: The detailed diagnosis of the Current sessions measure of the Citrix HDX Desktop Logins Test

The detailed diagnosis of the *New sessions* measure lists the desktop users who logged in during the last measurement period. The quality of each user's session is also revealed. This way, administrators can quickly identify which desktop user's experience is below-par and what is causing it - latent client-side network connection? a slow server-side network connection? or a bandwidth-intensive communication over the network?



Details of new desktop sessions

USER NAME	SESSION GUID	CLIENT ADDRESS	DESTINATION ADDRESS	SESSION START TIME	SESSION DURATION (MINUTES)	WAN LATENCY (MSECS)	DC LATENCY (MSECS)	RTT (MSECS)	BANDWIDTH (KBPS)
Feb 26, 2016 11:16:54									
sales	0b7dd572-d9f3-4683-b203-d21ebdaff2e5	192.168.9.168	192.168.8.185	26/02/2016 11:16:25 IST	0	1	7	0	1.83

Figure 3.7: The detailed diagnosis of the New sessions measure of the Citrix HDX Desktop Logins test

3.3.3 Citrix HDX Desktop Users Test

To ensure that users are able to access desktops on-demand, administrators must closely track that user's accesses, promptly detect probable access latencies, diagnose its root-cause, and take steps to avert it, well before that user notices and complains. To achieve this, administrators can use the **Citrix HDX Desktop Users** test. This test automatically discovers the users who are currently accessing virtual desktops in a XenDesktop infrastructure, and for each user, reports the latencies that user experienced when interacting with the desktops. This way, the test quickly and accurately points administrators to those desktop users who are experiencing slowness, and also leads them to what is causing the slowness – the network? or the server hosting the desktops? If a latent network is causing the slowness, then the test provides administrators with detailed insights into network performance and enables them to rapidly figure out where the bottleneck lies - on the client-side network? or on the server-side network?

Target of the test : An AppFlow-enabled NetScaler Appliance

Agent deploying the test : A remote agent

Outputs of the test : One set of results for every user who is currently connected to an virtual desktop

Configurable parameters for the test

Parameter	Description
Test period	How often should the test be executed. **It is recommended that you set the test period to 5 minutes. This is because, the eG AppFlow Collector is capable of capturing and aggregating AppFlow data related to the last 5 minutes only.**
Host	The host for which the test is to be configured.
Cluster IPs	**This parameter applies only if the NetScaler appliance being monitored is part of a NetScaler cluster.** In this case, configure this parameter with a comma-separated list of IP addresses of all other nodes in that cluster. If the monitored NetScaler appliance is down/unreachable, then the eG AppFlow Collector uses the **Cluster IPs** configuration to figure out which other node in the

Parameter	Description
	cluster it should connect to for pulling AppFlow statistics. Typically, the collector attempts to connect to every IP address that is configured against **Cluster IPs**, in the same sequence in which they are specified. Metrics are pulled from the first cluster node that the collector successfully establishes a connection with.
Enable Logs	This flag is set to **No** by default. This means that, by default, the eG agent does not create AppFlow logs. You can set this flag to **Yes** to enable AppFlow logging. If this is done, then the eG agent automatically writes the raw AppFlow records it reads from the collector into individual CSV files. These CSV files are stored in the *<EG_AGENT_INSTALL_DIR>\NetFlow\data\<IP_of_Monitored_NetScaler>\hdxappflow\actual_csv* folder on the eG agent host. These CSV files provide administrators with granular insights into the HDX appflows, thereby enabling effective troubleshooting. **Note:** By default, the eG agent creates a maximum of 10 CSV files in the **actual_csv** folder. Beyond this point, the older CSV files will be automatically deleted by the eG agent to accommodate new files with current data. Likewise, a single CSV file can by default contain a maximum of 99999 records only. If the records to be written exceed this default value, then the eG agent automatically creates another CSV file to write the data. If required, you can overwrite these default settings . For this, do the following: 1. Login to the eG agent host. 2. Edit the **Netflow.Properties** file in the <EG_AGENT_INSTALL_DIR>\NetFlow\config directory. 3. In the file, look for the parameter, **csv_file_retention_count**. 4. This is the parameter that governs the maximum number of CSV files that can be created in the **auto_csv** folder. By default, this parameter is set to 10. If you want to retain more number of CSV files at any given point in time, you can increase the value of this parameter. If you want to retain only a few CSV files, then decrease the value of this parameter. 5. Next, look for the parameter, **csv_max_flow_record_per_file**. 6. This is the parameter that governs the number of flow records that can be written to a single CSV. By default, this parameter is set to 99999. If you want a

Parameter	Description
	single file to accommodate more records, so that the creation of new CSVs is delayed, then increase the value of this parameter. On the other hand, if you want to reduce the capacity of a CSV file, so that new CSVs are quickly created, then decrease the value of this parameter. 7. Finally, save the file.
DD Frequency	Refers to the frequency with which detailed diagnosis measures are to be generated for this test. The default is *1:1*. This indicates that, by default, detailed measures will be generated every time this test runs, and also every time the test detects a problem. You can modify this frequency, if you so desire. Also, if you intend to disable the detailed diagnosis capability for this test, you can do so by specifying *none* against **DD Frequency**.
Detailed Diagnosis	To make diagnosis more efficient and accurate, the eG Enterprise suite embeds an optional detailed diagnostic capability. With this capability, the eG agents can be configured to run detailed, more elaborate tests as and when specific problems are detected. To enable the detailed diagnosis capability of this test for a particular server, choose the **On** option. To disable the capability, click on the **Off** option. The option to selectively enable/disable the detailed diagnosis capability will be available only if the following conditions are fulfilled: • The eG manager license should allow the detailed diagnosis capability • Both the normal and abnormal frequencies configured for the detailed diagnosis measures should not be 0.

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
Desktop launches	Indicates the number of desktops launched by this user.	Number	To know which desktops were launched by this user, use the detailed diagnosis of this measure.
Session status	Indicates the current status of this user's sessions.		The values that this measure can take and the numeric values that correspond to each measure value are listed in the table below:

<table>
<tr><th>Measurement</th><th>Description</th><th>Measurement Unit</th><th>Interpretation</th></tr>
<tr><td></td><td></td><td></td><td>
<table>
<tr><th>Measure Value</th><th>Numeric Value</th></tr>
<tr><td>Active</td><td>0</td></tr>
<tr><td>SR successful</td><td>1000</td></tr>
<tr><td>Existing ICA session got terminated</td><td>1001</td></tr>
<tr><td>Existing ICA connection got terminated and SR failed</td><td>1002</td></tr>
<tr><td>Existing ICA connection terminated and SR failed and client is trying to do ACR and is successful</td><td>1003</td></tr>
</table>
Note:
Typically, this test reports the Measure Values in the table above to indicate session status. In the graph of this measure however, the same is indicated using the numeric equivalents only.
</td></tr>
<tr><td>RTT</td><td>Indicates the screen lag experienced by this user while interacting with desktops.</td><td>Msecs</td><td>A high value for this measure is indicative of the poor quality of a user’s experience with desktops.
To know the reason for this below-par UX, compare the value of the WAN latency, DC latency, and Host delay measures of that user.</td></tr>
<tr><td>WAN latency</td><td>Indicates the average latency experienced by this user due to problems with the client side network.</td><td>Msecs</td><td>A high value for this measure indicates that the client side network is slow.
If the value of the RTT measure is abnormally high for a user, you can compare the value of this measure with that of the DC latency and Host delay, and measures of that user to know what is</td></tr>
</table>

Measurement	Description	Measurement Unit	Interpretation
			causing the slowness – is it the client side network? the server side network? or the server hosting the desktops?
DC latency	Indicates the average latency experienced by this user due to problems with the server side network.	Msecs	A high value for this measure indicates that the server side network is slow. If the value of the *RTT* measure is abnormally high for a user, you can compare the value of this measure with that of the *WAN latency and Host delay* measures of that user to know what is causing the slowness – is it the client side network? the server side network? or the server hosting the desktops?
Host delay	Indicates the delay that this user experienced when waiting for the host to process the packets.	Msecs	A high value for this measure indicates a processing bottleneck with the server hosting the desktops. If the value of the *RTT* measure is abnormally high for a user, you can compare the value of this measure with that of the *WAN latency and DC latency,* measures to know what is causing the slowness – is it the client side network? the server side network? or the server hosting the desktops?
Bandwidth	Indicates the bandwidth used by this user.	Kbps	Ideally, the value of this measure should be low. A high value indicates excessive bandwidth usage by the user. Compare the value of this measure across users to know which user is consuming bandwidth excessively.
Bytes	Indicates the total bytes consumed by this user's sessions.	Bytes	Compare the value of this measure across users to know which user has the maximum throughput and which has the least.

Measurement	Description	Measurement Unit	Interpretation
Client side retransmits	Indicates the number of packets retransmitted on the client side connection during the last measurement period.	Number	Ideally, the value of these measures should be 0.
Server side retransmits	Indicates the number of packets retransmitted on the server side connection during the last measurement period.	Number	
ACR counts	Indicates the total number of times the client automatically reconnected this user to sessions.	Number	The Automatic Client Reconnect (ACR) policy setting, when enabled, allows automatic reconnection by the same client after a connection has been interrupted. Allowing automatic client reconnect allows users to resume working where they were interrupted when a connection was broken. Automatic reconnection detects broken connections and then reconnects the users to their sessions.
Session reconnects	Indicates the number of times this user's sessions reconnected.	Number	This measure includes only those times a user reconnected to a disconnected session by mechanisms other than the ACR setting.
Client side NS delay	Indicates the average latency experienced by this user, which was caused by the NetScaler appliance when ICA traffic flowed from client network to server network.	Msecs	A high value for these measures indicates a processing bottleneck with the NetScaler appliance. If the value of the *WAN latency* measure is abnormally high for a desktop user, you can check the value of the *Client side NS delay* measure to determine if the network delays noticed on the client side were caused by the NetScaler's lethargy in processing traffic from the client.

Measurement	Description	Measurement Unit	Interpretation
Server side NS delay	Indicates the average latency experienced by this session, which was caused by the NetScaler appliance when ICA traffic flowed from server network to client network.	Msecs	If the value of the *DC latency* measure is abnormally high for a desktop user, you can check the value of the *Server side NS delay* measure to determine if the network delays noticed on the server side were caused by the NetScaler's lethargy in processing traffic from the server.

The detailed diagnosis of the *Session status* measure provides additional details of a user. If the status of a user's sessions is abnormal, you can use these details to know from which client the user is connecting, the client type and version, which desktop the user is connecting to, the start time, and the uptime of the session. This will help in troubleshooting the abnormal session status.

Session details of desktop user

VDI IMAGE NAME	SESSION GUID	CONNECTION ID	SOURCE ADDRESS	DESTINATION ADDRESS	DESKTOP STARTUP TIME	UP TIME (MINUTES)	CLIENT TYPE	CLIENT VERSION	DOMAIN
Feb 26, 2016 11:37:36									
XENAPP7V6	0b7dd572-d9f3-4683-b203-d21ebdaff2e5	2497567	192.168.9.168	192.168.8.185	26/02/2016 11:16:25 IST	21	Windows	14.0.0.91	citrix

Figure 3.8: The detailed diagnosis of the Session status measure reported by the Citrix HDX Desktop Users test

Use the detailed diagnosis of the *Desktop launches* measure to know which desktop(s) was recently launched by the user.

Details of desktops launched by this user

VDI IMAGE NAME	SESSION GUID	CONNECTION ID	SOURCE ADDRESS	DESTINATION ADDRESS	DESKTOP STARTUP TIME	UP TIME (MINUTES)	CLIENT TYPE	CLIENT VERSION	DOMAIN
Feb 26, 2016 11:17:36									
XENAPP7V6	0b7dd572-d9f3-4683-b203-d21ebdaff2e5	2497567	192.168.9.168	192.168.8.185	26/02/2016 11:16:25 IST	1	Windows	14.0.0.91	citrix

Figure 3.9: The detailed diagnosis of the Desktop launches measure of the Citrix HDX Desktop Users test

3.4 The Citrix HDX Applications Layer

Use the test mapped to this layer to know which application is experiencing launching delays.

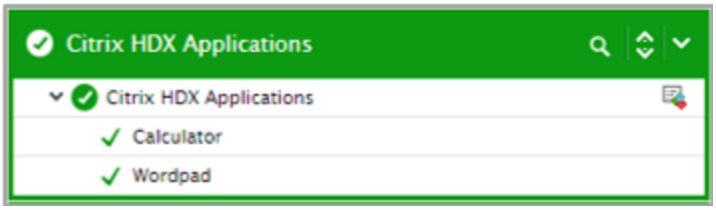


Figure 3.10: The test mapped to the Citrix HDX Applications layer

3.4.1 Citrix HDX Applications Test

How much time an application takes to be launched via ICA affects not just the user experience with that application, but also user productivity, as key business transactions may be delayed if business-critical applications launch slowly. It is hence good practice to periodically run the **Citrix HDX Applications** test and check on the launch time of individual applications. This test auto-discovers applications that are deployed on XenApp, reports the number of instances of each application that have been launched currently, and the time every application took to launch; this way, the test sheds light on applications that are experiencing delays when launching. Using the detailed diagnosis of this test, you can also figure out which user sessions have been impacted by such applications.

Target of the test : An AppFlow-enabled Citrix NetScaler appliance

Agent deploying the test : A remote agent

Outputs of the test : One set of results for every application running on XenApp

Configurable parameters for the test

Parameter	Description
Test period	How often should the test be executed. **It is recommended that you set the test period to 5 minutes. This is because, the eG AppFlow Collector is capable of capturing and aggregating AppFlow data related to the last 5 minutes only.**
Host	The host for which the test is to be configured.
Cluster IPs	**This parameter applies only if the NetScaler appliance being monitored is part of a NetScaler cluster.** In this case, configure this parameter with a comma-separated list of IP addresses of all other nodes in that cluster. If the monitored NetScaler appliance is down/unreachable, then the eG AppFlow Collector uses the **Cluster IPs** configuration to figure out which other node in the cluster it should connect to for pulling AppFlow statistics. Typically, the collector attempts to connect to every IP address that is configured against **Cluster IPs**, in the same sequence in which they are specified. Metrics are pulled from the first cluster node that the collector successfully establishes a connection with.
Enable Logs	This flag is set to **No** by default. This means that, by default, the eG agent does not create AppFlow logs. You can set this flag to **Yes** to enable AppFlow logging. If this is done, then the eG agent automatically writes the raw AppFlow records it reads from the collector into individual CSV files. These CSV files are stored in the *<EG_ AGENT_INSTALL_DIR>\NetFlow\data\<IP_of_Monitored_ NetScaler>\hdxappflow\actual_csv* folder on the eG agent host. These CSV files

Parameter	Description
	provide administrators with granular insights into the HDX appflows, thereby enabling effective troubleshooting. **Note:** By default, the eG agent creates a maximum of 10 CSV files in the **actual_csv** folder. Beyond this point, the older CSV files will be automatically deleted by the eG agent to accommodate new files with current data. Likewise, a single CSV file can by default contain a maximum of 99999 records only. If the records to be written exceed this default value, then the eG agent automatically creates another CSV file to write the data. If required, you can overwrite these default settings . For this, do the following: 1. Login to the eG agent host. 2. Edit the **Netflow.Properties** file in the <EG_AGENT_INSTALL_ DIR>\NetFlow\config directory. 3. In the file, look for the parameter, **csv_file_retention_count**. 4. This is the parameter that governs the maximum number of CSV files that can be created in the **auto_csv** folder. By default, this parameter is set to 10. If you want to retain more number of CSV files at any given point in time, you can increase the value of this parameter. If you want to retain only a few CSV files, then decrease the value of this parameter. 5. Next, look for the parameter, **csv_max_flow_record_per_file**. 6. This is the parameter that governs the number of flow records that can be written to a single CSV. By default, this parameter is set to 99999. If you want a single file to accommodate more records, so that the creation of new CSVs is delayed, then increase the value of this parameter. On the other hand, if you want to reduce the capacity of a CSV file, so that new CSVs are quickly created, then decrease the value of this parameter. 7. Finally, save the file.
DD Frequency	Refers to the frequency with which detailed diagnosis measures are to be generated for this test. The default is *1:1*. This indicates that, by default, detailed measures will be generated every time this test runs, and also every time the test detects a problem. You can modify this frequency, if you so desire. Also, if you intend to disable the

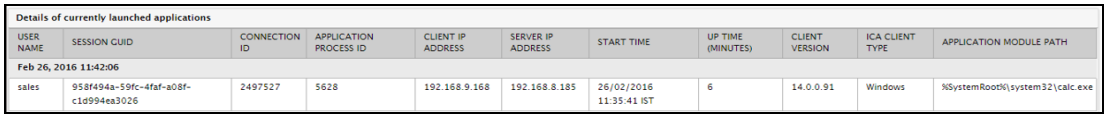
Parameter	Description
	detailed diagnosis capability for this test, you can do so by specifying *none* against **DD Frequency**.
Detailed Diagnosis	To make diagnosis more efficient and accurate, the eG Enterprise suite embeds an optional detailed diagnostic capability. With this capability, the eG agents can be configured to run detailed, more elaborate tests as and when specific problems are detected. To enable the detailed diagnosis capability of this test for a particular server, choose the **On** option. To disable the capability, click on the **Off** option. The option to selectively enable/disable the detailed diagnosis capability will be available only if the following conditions are fulfilled: • The eG manager license should allow the detailed diagnosis capability • Both the normal and abnormal frequencies configured for the detailed diagnosis measures should not be 0.

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
Active applications	Indicates the number of instances of this application that users are currently accessing on XenApp.	Number	
Application launches	Indicates the number of instances of this application that are currently launched by all XenApp users.	Number	Compare the value of this measure across applications to identify the most popular application. Use the detailed diagnosis of this measure to know in which sessions this application has been launched.
Session launches	Indicates the number of sessions in which this application was launched.	Number	The detailed diagnosis of the Session launches measure lists the users launching the application currently, and the sessions in which the application is launched. With the help of this information,

Measurement	Description	Measurement Unit	Interpretation
			you can identify the exact user session that is impacted by the slow application launch (if any).
Average launch duration	Indicates the average time it took to launch this application.	Msecs	A high value for this measure indicates that the application is slow in launching. You may want to compare the value of this measure across applications to isolate the slowest application.

Use the detailed diagnosis of the *Application launches* measure to know which users launched the application and in which sessions. The details of each session such as the client from which the session was launched, the client version, and the server on which the application was launched are also reported. The start time and uptime of the application are also revealed so that you can promptly detect unsheduled application reboots.

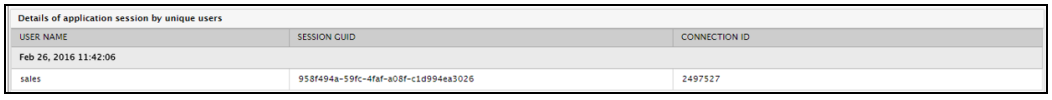


Details of currently launched applications

USER NAME	SESSION GUID	CONNECTION ID	APPLICATION PROCESS ID	CLIENT IP ADDRESS	SERVER IP ADDRESS	START TIME	UP TIME (MINUTES)	CLIENT VERSION	ICA CLIENT TYPE	APPLICATION MODULE PATH
Feb 26, 2016 11:42:06										
sales	958f494a-59fc-4faf-a08f-c1d994ea3026	2497527	5628	192.168.9.168	192.168.8.185	26/02/2016 11:35:41 IST	6	14.0.0.91	Windows	%SystemRoot%\system32\calc.exe

Figure 3.11: The detailed diagnosis of the Application launches measure of the HDX Applications test

The detailed diagnosis of the *Session launches* measure lists the users launching the application currently, and the sessions in which the application is launched.



Details of application session by unique users

USER NAME	SESSION GUID	CONNECTION ID
Feb 26, 2016 11:42:06		
sales	958f494a-59fc-4faf-a08f-c1d994ea3026	2497527

Figure 3.12: The detailed diagnosis of the Session launches measure

3.5 The Citrix HDX Users Layer

The tests mapped to this layer tracks user sessions on the XenApp servers managed by the target NetScaler appliance, and measures the experience of each user on the server. The layer also monitors the bandwidth usage of each ICA channel and each device type (user agent).

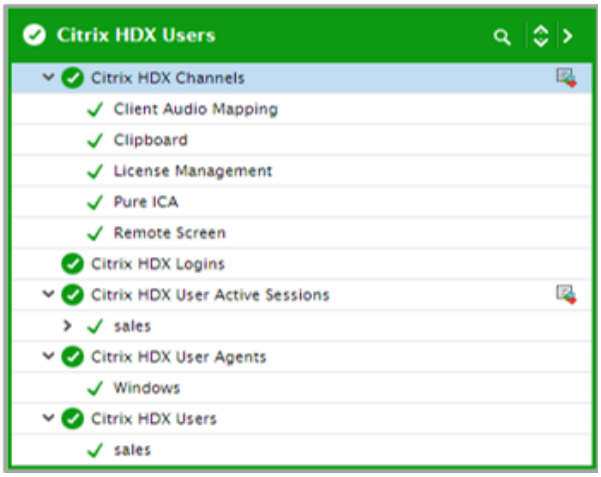


Figure 3.13: The tests mapped to the Citrix HDX Users layer

3.5.1 Citrix HDX Channels Test

ICA traffic is composed of virtual channels. A virtual channel is a bidirectional, error-free connection used for the exchange of generalized packet data between a Citrix host for online delivery (XenApp or XenDesktop) and the Citrix Receiver online plug-in. Connections for sound, graphics, client drive mapping, printing, and end user experience monitoring are a few examples of the virtual channels. One of the important motives behind ICA traffic monitoring is to understand how much bandwidth is being consumed by each channel and in the process identify the most bandwidth-intensive channel. The **Citrix HDX Channels** test helps administrators with this. This test monitors the ICA traffic handled by each channel and leads administrators to that channel which is consuming bandwidth excessively. The insights provided by this test help administrators in understanding how the NetScaler appliance needs to be fine-tuned to optimize bandwidth consumption.

Target of the test : An AppFlow-enabled Citrix NetScaler

Agent deploying the test : A remote agent

Outputs of the test : One set of results for every virtual channel

Configurable parameters for the test

Parameter	Description
Test period	How often should the test be executed. **It is recommended that you set the test period to 5 minutes. This is because, the eG AppFlow Collector is capable of capturing and aggregating AppFlow data related to the last 5 minutes only.**
Host	The host for which the test is to be configured.

Parameter	Description
Cluster IPs	**This parameter applies only if the NetScaler appliance being monitored is part of a NetScaler cluster.** In this case, configure this parameter with a comma-separated list of IP addresses of all other nodes in that cluster. If the monitored NetScaler appliance is down/unreachable, then the eG AppFlow Collector uses the **Cluster IPs** configuration to figure out which other node in the cluster it should connect to for pulling AppFlow statistics. Typically, the collector attempts to connect to every IP address that is configured against **Cluster IPs**, in the same sequence in which they are specified. Metrics are pulled from the first cluster node that the collector successfully establishes a connection with.
Enable Logs	This flag is set to **No** by default. This means that, by default, the eG agent does not create AppFlow logs. You can set this flag to **Yes** to enable AppFlow logging. If this is done, then the eG agent automatically writes the raw AppFlow records it reads from the collector into individual CSV files. These CSV files are stored in the *<EG_AGENT_INSTALL_DIR>\NetFlow\data\<IP_of_Monitored_NetScaler>\hdxappflow\actual_csv* folder on the eG agent host. These CSV files provide administrators with granular insights into the HDX appflows, thereby enabling effective troubleshooting. **Note:** By default, the eG agent creates a maximum of 10 CSV files in the **actual_csv** folder. Beyond this point, the older CSV files will be automatically deleted by the eG agent to accommodate new files with current data. Likewise, a single CSV file can by default contain a maximum of 99999 records only. If the records to be written exceed this default value, then the eG agent automatically creates another CSV file to write the data. If required, you can overwrite these default settings . For this, do the following: 1. Login to the eG agent host. 2. Edit the **Netflow.Properties** file in the <EG_AGENT_INSTALL_DIR>\NetFlow\config directory. 3. In the file, look for the parameter, **csv_file_retention_count**. 4. This is the parameter that governs the maximum number of CSV files that can be created in the **auto_csv** folder. By default, this parameter is set to 10. If you want to retain more number of CSV files at any given point in time, you can increase the value of this parameter. If you want to retain only a few CSV files,

Parameter	Description
	then decrease the value of this parameter. 5. Next, look for the parameter, **csv_max_flow_record_per_file**. 6. This is the parameter that governs the number of flow records that can be written to a single CSV. By default, this parameter is set to 99999. If you want a single file to accommodate more records, so that the creation of new CSVs is delayed, then increase the value of this parameter. On the other hand, if you want to reduce the capacity of a CSV file, so that new CSVs are quickly created, then decrease the value of this parameter. 7. Finally, save the file.

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
Total bytes	Indicates the total amount of data (in bytes) that passed through this channel.	Bytes	
Avg bandwidth	Indicates the rate at which data is transferred over this virtual channel.	Kbps	Ideally, the value of this measure should be low. A high value indicates excessive bandwidth usage by a virtual channel. Compare the value of this measure across channels to know which channel is consuming bandwidth excessively
Percentage of traffic by this channel	Indicates what percentage of total traffic flowing through the target NetScaler appliance was handled by this virtual channel.	Percent	Compare the value of this measure across virtual channels to identify the virtual channel that is consuming bandwidth excessively. A value close to 100% indicates a channel that is hogging the bandwidth resources.

3.5.2 Citrix HDX Logins Test

Use this test to determine the current session load on the XenApp servers in your environment. The detailed diagnosis of this test also reveals who launched the sessions, when, from which client, to which server, the session duration, and the overall quality of each user's session. This way, administrators can quickly identify which application user's experience is below-par and what is causing it – a latent client-side network connection? a slow server-side network connection? or a bandwidth-intensive communication over the network? The test also tracks session logouts, and points to abnormal logouts.

Target of the test : An AppFlow-enabled NetScaler Appliance

Agent deploying the test : A remote agent

Outputs of the test : One set of results for the monitored NetScaler appliance

Configurable parameters for the test

Parameter	Description
Test period	How often should the test be executed. **It is recommended that you set the test period to 5 minutes. This is because, the eG AppFlow Collector is capable of capturing and aggregating AppFlow data related to the last 5 minutes only.**
Host	The host for which the test is to be configured.
Cluster IPs	**This parameter applies only if the NetScaler appliance being monitored is part of a NetScaler cluster.** In this case, configure this parameter with a comma-separated list of IP addresses of all other nodes in that cluster. If the monitored NetScaler appliance is down/unreachable, then the eG AppFlow Collector uses the **Cluster IPs** configuration to figure out which other node in the cluster it should connect to for pulling AppFlow statistics. Typically, the collector attempts to connect to every IP address that is configured against **Cluster IPs**, in the same sequence in which they are specified. Metrics are pulled from the first cluster node that the collector successfully establishes a connection with.
Enable Logs	This flag is set to **No** by default. This means that, by default, the eG agent does not create AppFlow logs. You can set this flag to **Yes** to enable AppFlow logging. If this is done, then the eG agent automatically writes the raw AppFlow records it reads from the collector into individual CSV files. These CSV files are stored in the *<EG_ AGENT_INSTALL_DIR>\NetFlow\data\<IP_of_Monitored_ NetScaler>\hdxappflow\actual_csv* folder on the eG agent host. These CSV files provide administrators with granular insights into the HDX appflows, thereby enabling effective troubleshooting.

Parameter	Description
	Note: By default, the eG agent creates a maximum of 10 CSV files in the **actual_csv** folder. Beyond this point, the older CSV files will be automatically deleted by the eG agent to accommodate new files with current data. Likewise, a single CSV file can by default contain a maximum of 99999 records only. If the records to be written exceed this default value, then the eG agent automatically creates another CSV file to write the data. If required, you can overwrite these default settings . For this, do the following: 1. Login to the eG agent host. 2. Edit the **Netflow.Properties** file in the <EG_AGENT_INSTALL_ DIR>\NetFlow\config directory. 3. In the file, look for the parameter, **csv_file_retention_count**. 4. This is the parameter that governs the maximum number of CSV files that can be created in the **auto_csv** folder. By default, this parameter is set to 10. If you want to retain more number of CSV files at any given point in time, you can increase the value of this parameter. If you want to retain only a few CSV files, then decrease the value of this parameter. 5. Next, look for the parameter, **csv_max_flow_record_per_file**. 6. This is the parameter that governs the number of flow records that can be written to a single CSV. By default, this parameter is set to 99999. If you want a single file to accommodate more records, so that the creation of new CSVs is delayed, then increase the value of this parameter. On the other hand, if you want to reduce the capacity of a CSV file, so that new CSVs are quickly created, then decrease the value of this parameter. 7. Finally, save the file.
DD Frequency	Refers to the frequency with which detailed diagnosis measures are to be generated for this test. The default is *1:1*. This indicates that, by default, detailed measures will be generated every time this test runs, and also every time the test detects a problem. You can modify this frequency, if you so desire. Also, if you intend to disable the detailed diagnosis capability for this test, you can do so by specifying *none* against **DD Frequency**.

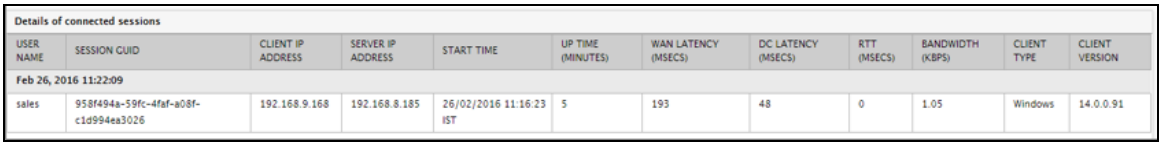
Parameter	Description
Detailed Diagnosis	To make diagnosis more efficient and accurate, the eG Enterprise suite embeds an optional detailed diagnostic capability. With this capability, the eG agents can be configured to run detailed, more elaborate tests as and when specific problems are detected. To enable the detailed diagnosis capability of this test for a particular server, choose the **On** option. To disable the capability, click on the **Off** option. The option to selectively enable/disable the detailed diagnosis capability will be available only if the following conditions are fulfilled: • The eG manager license should allow the detailed diagnosis capability • Both the normal and abnormal frequencies configured for the detailed diagnosis measures should not be 0.

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
Current sessions	Indicates the number of user sessions that are currently active across all XenApp servers.	Number	This is a good indicator of the session load on the XenApp servers. To know which users are contributing to the load, use the detailed diagnosis of this measure. The detailed diagnosis additionally reveals when each user logged in, from which client, to which server, the session duration, and the quality of the user's session. This way, administrators can quickly identify which application user's experience is below-par and what is causing it - a flaky network connection? a bandwidth-intensive communication over the network? or a delay at the server-end?
New logins	Indicates the number of new logins to the XenApp servers.	Number	A consistent zero value could indicate a connection issue. If this measure reports a non-zero value, use the detailed diagnosis of the measure to know which user logged

Measurement	Description	Measurement Unit	Interpretation
			into which XenApp server, when.
Percent new logins	Indicates the percentage of current sessions that logged in during the last measurement period.	Percent	
Sessions logging out	Indicates the number of sessions that logged out.	Number	If all the current sessions suddenly log out, it indicates a problem condition that requires investigation. The detailed diagnosis of this measure lists the sessions that logged out. The quality of each user's session is also revealed. This way, administrators can quickly identify which application user's experience is below-par and what is causing it –This way, administrators can quickly identify which application user's experience is below-par and what is causing it – a latent client-side network connection? a slow server-side network connection? or a bandwidth-intensive communication over the network?

To know which users are contributing to the load on XenApp, use the detailed diagnosis of the *Current sessions* measure. The detailed diagnosis additionally reveals when each user logged in, from which client, to which server, the session duration, and the quality of the user's session. This way, administrators can quickly identify which application user's experience is below-par and what is causing it - a latent client-side network connection? a slow server-side network connection? or a bandwidth-intensive communication over the network?



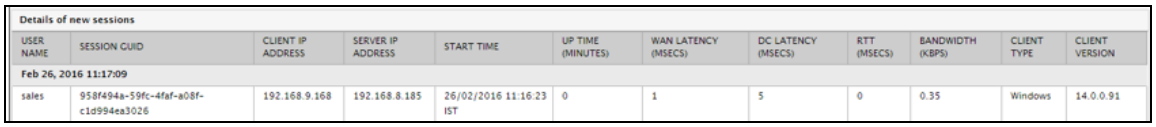
Details of connected sessions

USER NAME	SESSION GUID	CLIENT IP ADDRESS	SERVER IP ADDRESS	START TIME	UP TIME (MINUTES)	WAN LATENCY (MSECS)	DC LATENCY (MSECS)	RTT (MSECS)	BANDWIDTH (KBPS)	CLIENT TYPE	CLIENT VERSION
Feb 26, 2016 11:22:09											
sales	958f494a-59fc-4faf-a08f-c1d994ea3026	192.168.9.168	192.168.8.185	26/02/2016 11:16:23 IST	5	193	48	0	1.05	Windows	14.0.0.91

Figure 3.14: The detailed diagnosis of the Current sessions measure of the Citrix HDX Logins Test

The detailed diagnosis of the *New sessions* measure lists the users who logged in during the last measurement period. The quality of each user's session is also revealed. This way, administrators

can quickly identify which application user's experience is below-par and what is causing it - latent client-side network connection? a slow server-side network connection? or a bandwidth-intensive communication over the network?

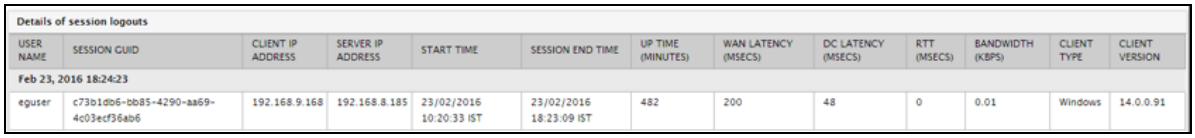


Details of new sessions

USER NAME	SESSION GUID	CLIENT IP ADDRESS	SERVER IP ADDRESS	START TIME	UP TIME (MINUTES)	WAN LATENCY (MSECS)	DC LATENCY (MSECS)	RTT (MSECS)	BANDWIDTH (KBPS)	CLIENT TYPE	CLIENT VERSION
Feb 26, 2016 11:17:09											
sales	958f494a-59fc-4faf-a08f-c1d994ea3026	192.168.9.168	192.168.8.185	26/02/2016 11:16:23 IST	0	1	5	0	0.35	Windows	14.0.0.91

Figure 3.15: The detailed diagnosis of the New sessions measure of the Citrix HDX Logins test

The detailed diagnosis of the *Sessions logging out* measure lists the sessions that logged out. The quality of each user's session is also revealed. This way, administrators can quickly identify which application user's experience is below-par and what is causing it - latent client-side network connection? a slow server-side network connection? or a bandwidth-intensive communication over the network?



Details of session logouts

USER NAME	SESSION GUID	CLIENT IP ADDRESS	SERVER IP ADDRESS	START TIME	SESSION END TIME	UP TIME (MINUTES)	WAN LATENCY (MSECS)	DC LATENCY (MSECS)	RTT (MSECS)	BANDWIDTH (KBPS)	CLIENT TYPE	CLIENT VERSION
Feb 23, 2016 18:24:23												
eguser	c73b1db6-bb85-4290-aa69-4c03ecf36ab6	192.168.9.168	192.168.8.185	23/02/2016 10:20:33 IST	23/02/2016 18:23:09 IST	482	200	48	0	0.01	Windows	14.0.0.91

Figure 3.16: The detailed diagnosis of the Sessions logging out measure of the Citrix HDX Logins test

3.5.3 Citrix HDX User Active Sessions Test

In order to ensure that the user experience with applications/desktops deployed on a XenApp/XenDesktop environment remains 'superlative' at all times, administrators should be able to proactively detect potential slowdowns when accessing applications/desktops, precisely pinpoint the user session affected by the slowdown, accurately isolate the root-cause of such slowness, and rapidly initiate measures to eliminate the root-cause. The **Citrix HDX User Active Sessions** test facilitates all the above, and thus assures users of uninterrupted application/desktop access!

For a user session that is currently active on a XenApp server or a XenDesktop virtual desktop, this test measures session latencies and leads you to the probable cause of session slowness (if any) - is it the network? the server hosting the applications/desktops? or are the applications (in the case of sessions to a XenApp server) taking too long to startup? If a latent network is causing the slowness, then the test provides administrators with detailed insights into network performance and enables them to rapidly figure out where the bottleneck lies - on the client-side network? or on the server-side network? This way, the test promptly leads administrators to slow user sessions, and also reveals what is causing the slowness, so that administrators can initiate the right steps to enhance user experience with applications/desktops.

Target of the test : An AppFlow-enabled NetScaler Appliance

Agent deploying the test : A remote agent

Outputs of the test : One set of results for each session for every user to a XenApp server / XenDesktop virtual desktop

First level descriptor: User name

Second level descriptor: Session GUID

An application session is identified by a single session GUID, regardless of the number of applications accessed by a user during that session.

A desktop session is identified by a separate session GUID - one each for every desktop that is accessed

Configurable parameters for the test

Parameter	Description
Test period	How often should the test be executed. **It is recommended that you set the test period to 5 minutes. This is because, the eG AppFlow Collector is capable of capturing and aggregating AppFlow data related to the last 5 minutes only.**
Host	The host for which the test is to be configured.
Cluster IPs	**This parameter applies only if the NetScaler appliance being monitored is part of a NetScaler cluster.** In this case, configure this parameter with a comma-separated list of IP addresses of all other nodes in that cluster. If the monitored NetScaler appliance is down/unreachable, then the eG AppFlow Collector uses the **Cluster IPs** configuration to figure out which other node in the cluster it should connect to for pulling AppFlow statistics. Typically, the collector attempts to connect to every IP address that is configured against **Cluster IPs**, in the same sequence in which they are specified. Metrics are pulled from the first cluster node that the collector successfully establishes a connection with.
Enable Logs	This flag is set to **No** by default. This means that, by default, the eG agent does not create AppFlow logs. You can set this flag to **Yes** to enable AppFlow logging. If this is done, then the eG agent automatically writes the raw AppFlow records it reads from the collector into individual CSV files. These CSV files are stored in the *<EG_ AGENT_INSTALL_DIR>\NetFlow\data\<IP_of_Monitored_ NetScaler>\hdxappflow\actual_csv* folder on the eG agent host. These CSV files provide administrators with granular insights into the HDX appflows, thereby enabling effective troubleshooting.

Parameter	Description
	Note: By default, the eG agent creates a maximum of 10 CSV files in the **actual_csv** folder. Beyond this point, the older CSV files will be automatically deleted by the eG agent to accommodate new files with current data. Likewise, a single CSV file can by default contain a maximum of 99999 records only. If the records to be written exceed this default value, then the eG agent automatically creates another CSV file to write the data. If required, you can overwrite these default settings . For this, do the following: 1. Login to the eG agent host. 2. Edit the **Netflow.Properties** file in the <EG_AGENT_INSTALL_ DIR>\NetFlow\config directory. 3. In the file, look for the parameter, **csv_file_retention_count**. 4. This is the parameter that governs the maximum number of CSV files that can be created in the **auto_csv** folder. By default, this parameter is set to 10. If you want to retain more number of CSV files at any given point in time, you can increase the value of this parameter. If you want to retain only a few CSV files, then decrease the value of this parameter. 5. Next, look for the parameter, **csv_max_flow_record_per_file**. 6. This is the parameter that governs the number of flow records that can be written to a single CSV. By default, this parameter is set to 99999. If you want a single file to accommodate more records, so that the creation of new CSVs is delayed, then increase the value of this parameter. On the other hand, if you want to reduce the capacity of a CSV file, so that new CSVs are quickly created, then decrease the value of this parameter. 7. Finally, save the file.
DD Frequency	Refers to the frequency with which detailed diagnosis measures are to be generated for this test. The default is *1:1*. This indicates that, by default, detailed measures will be generated every time this test runs, and also every time the test detects a problem. You can modify this frequency, if you so desire. Also, if you intend to disable the detailed diagnosis capability for this test, you can do so by specifying *none* against **DD Frequency**.

Parameter	Description
Detailed Diagnosis	To make diagnosis more efficient and accurate, the eG Enterprise suite embeds an optional detailed diagnostic capability. With this capability, the eG agents can be configured to run detailed, more elaborate tests as and when specific problems are detected. To enable the detailed diagnosis capability of this test for a particular server, choose the **On** option. To disable the capability, click on the **Off** option. The option to selectively enable/disable the detailed diagnosis capability will be available only if the following conditions are fulfilled: • The eG manager license should allow the detailed diagnosis capability • Both the normal and abnormal frequencies configured for the detailed diagnosis measures should not be 0.

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
Active applications	Indicates the number of applications currently accessed by this user session.	Number	To know which applications are active in this user session, use the detailed diagnosis of this measure. **This measure is reported only for application sessions, and not desktop sessions.**
Application launches	Indicates the number of applications that were launched by this user session.	Number	To know which applications were launched by this user session, use the detailed diagnosis of this measure. **This measure is reported only for application sessions, and not desktop sessions.**
Application terminates	Indicates the number of applications terminated by this user session.	Number	To know which applications were terminated in this user session, use the detailed diagnosis of this measure. **This measure is reported only for application sessions, and not desktop sessions.**
Session status	Indicates the current status of this session.		The values that this measure can take and the numeric values that correspond

<table>
<tr><th>Measurement</th><th>Description</th><th>Measurement Unit</th><th>Interpretation</th></tr>
<tr><td></td><td></td><td></td><td>to each measure value are listed in the table below:

<table><tr><th>Measure Value</th><th>Numeric Value</th></tr><tr><td>Active</td><td>0</td></tr><tr><td>SR successful</td><td>1000</td></tr><tr><td>Existing ICA session got terminated</td><td>1001</td></tr><tr><td>Existing ICA connection got terminated and SR failed</td><td>1002</td></tr><tr><td>Existing ICA connection terminated and SR failed and client is trying to do ACR and is successful</td><td>1003</td></tr></table>
Note:

Typically, this test reports the Measure Values in the table above to indicate session status. In the graph of this measure however, the same is indicated using the numeric equivalents only.</td></tr>
<tr><td>Average application startup duration</td><td>Indicates the average time that elapsed between when an application accessed by this session was launched and when it started running.</td><td>Msecs</td><td>A high value for this measure indicates that one/more applications are starting up slowly on the server. In this case, use the detailed diagnosis of the Active applications measure to know which application is the slowest in starting up.

This measure is reported only for application sessions, and not desktop sessions.</td></tr>
<tr><td>RTT</td><td>Indicates the screen lag experienced by this</td><td>Msecs</td><td>A high value for this measure is indicative of the poor quality of a user’s</td></tr>
</table>

Measurement	Description	Measurement Unit	Interpretation
	session while interacting with applications/desktops.		experience with applications/desktops. To know the reason for this below-par UX, compare the value of the *WAN latency, DC latency, and Host delay* measures of that session.
WAN latency	Indicates the average latency experienced by this user session due to problems with the client side network.	Msecs	A high value for this measure indicates that the client side network is slow. If the value of the *RTT* measure is abnormally high for a session, you can compare the value of this measure with that of the *DC latency and Host delay, and* measures of that user session to know what is causing the slowness – is it the client side network? the server side network? or the server hosting the applications/desktops?
DC latency	Indicates the average latency experienced by this session due to problems with the server side network.	Msecs	A high value for this measure indicates that the server side network is slow. If the value of the *RTT* measure is abnormally high for a session, you can compare the value of this measure with that of the *WAN latency and Host delay, cy* measures of that session to know what is causing the slowness – is it the client side network? the server side network? or the server hosting the applications/desktops?
Host delay	Indicates the delay that this session experienced when waiting for the host to process the packets.	Msecs	A high value for this measure indicates a processing bottleneck with the server hosting the applications. If the value of the *RTT* measure is abnormally high for a session, you can compare the value of this measure with that of the *WAN latency and DC latency,* measures to know what is

Measurement	Description	Measurement Unit	Interpretation
			causing the slowness – is it the client side network? the server side network? or the server hosting the applications/desktops?
Bandwidth	Indicates the rate at which data is transferred over this ICA session.	Kbps	Ideally, the value of this measure should be low. A high value indicates excessive bandwidth usage by the session. Compare the value of this measure across sessions to know which session is consuming bandwidth excessively.
Bytes	Indicates the total bytes consumed by this session.	Bytes	Compare the value of this measure across sessions to know which session has the maximum throughput and which has the least.
Client side retransmits	Indicates the number of packets retransmitted on the client side connection during the last measurement period.	Number	Ideally, the value of these measures should be 0.
Server side retransmits	Indicates the number of packets retransmitted on the server side connection during the last measurement period.	Number	
Client side 0 window count	Indicates how many times in this session the client advertised a zero TCP window during the last measurement period.	Number	TCP Zero Window is when the Window size in a machine remains at zero for a specified amount of time. TCP Window size is the amount of information that a machine can receive during a TCP session and still be able to process the data. Think of it like a TCP receive buffer. When a machine initiates a TCP connection to a server, it will let the server know how much data it can

Measurement	Description	Measurement Unit	Interpretation
			receive by the Window Size. In many Windows machines, this value is around 64512 bytes. As the TCP session is initiated and the server begins sending data, the client will decrement it's Window Size as this buffer fills. At the same time, the client is processing the data in the buffer, and is emptying it, making room for more data. Through TCP ACK frames, the client informs the server of how much room is in this buffer. If the TCP Window Size goes down to 0, the client will not be able to receive any more data until it processes and opens the buffer up again.
Server side 0 window count	Indicates how many times in this session the server advertised a zero TCP window during the last measurement period.	Number	The machine (client/server) alerting the Zero Window will not receive any more data from the host. This is why, ideally, the value of these measures should be 0. A non-zero value warrants an immediate investigation to determine the reason for the Zero Window. It could be that the client/server was running too many processes at that moment, and its processor is maxed. Or it could be that there is an error in the TCP receiver, like a Windows registry misconfiguration. Try to determine what the client was doing when the TCP Zero Window happened. **These measures are reported only for application sessions, and not desktop sessions.**
Client RTO	Indicates how many times during the last	Number	An RTO occurs when the sender is missing too many acknowledgments

Measurement	Description	Measurement Unit	Interpretation
	measurement period the retransmit timeout got invoked in this session on the client side connection.		and decides to take a time out and stop sending altogether. After some amount of time, usually at least one second, the sender cautiously starts sending again, testing the waters with just one packet at first, then two packets, and so on. As a result, an RTO causes, at minimum, a one-second delay on your network. A low value is hence desired for these measures. **These measures are reported only for application sessions, and not desktop sessions.**
Server RTO	Indicates how many times during the last measurement period the retransmit timeout got invoked in this session on the server side connection.	Number	
ACR counts	Indicates the total number of times the client automatically reconnected the user to this session.	Number	The Automatic Client Reconnect (ACR) policy setting, when enabled, allows automatic reconnection by the same client after a connection has been interrupted. Allowing automatic client reconnect allows users to resume working where they were interrupted when a connection was broken. Automatic reconnection detects broken connections and then reconnects the users to their sessions.
Session reconnects	Indicates the number of times this session reconnected.	Number	This measure includes only those times a user reconnected to a disconnected session by mechanisms other than the ACR setting.
Client SRTT	Indicates the RTT (round-trip time or screen lag time) of this session smoothed over the client side connection.	MSecs	TCP implementations attempt to predict future round-trip times by sampling the behavior of packets sent over a connection and averaging those samples into a ‘‘smoothed’’ round-trip time estimate, SRTT. When a packet is sent over a TCP connection, the sender times how long it takes for it to be

Measurement	Description	Measurement Unit	Interpretation
			acknowledged, producing a sequence, S, of round-trip time samples: s1, s2, s3.... With each new sample, si, the new SRTT is computed from the formula: $SRTTi+1 = (\alpha \times SRTTi) + (1 - \alpha) \times si$ Here, SRTTi is the current estimate of the round-trip time, SRTTi+1 is the new computed value, and α is a constant between 0 and 1 that controls how rapidly the SRTT adapts to change. The retransmission time-out (RTOi), the amount of time the sender will wait for a given packet to be acknowledged, is computed from SRTTi. The formula is: $RTOi = \beta \times SRTTi$ Here, β is a constant, greater than 1, chosen such that there is an acceptably small probability that the round-trip time for the packet will exceed RTOi. **These measures are reported only for application sessions, and not desktop sessions.**
Server SRTT	Indicates the RTT (round-trip time or screen lag time) of this session, smoothed over the server side connection.	MSecs	
Client side NS delay	Indicates the average latency experienced by this session, which was caused by the NetScaler appliance when ICA traffic flowed from client network to server network.	Msecs	A high value for these measures indicates a processing bottleneck with the NetScaler appliance. If the value of the *WAN latency* measure is abnormally high for an application session, you can compare the value of the *Client side NS delay* measure with the value of the *Client jitter* measure for that session to determine what could have caused network delays on the client side - a NetScaler appliance that was slow in

Measurement	Description	Measurement Unit	Interpretation
			processing traffic from the client? or a traffic congestion on the client side? If the value of the *DC latency* measure is abnormally high for an application session, you can compare the value of the *Server side NS delay* measure with the value of the *Server jitter* measure for that session to determine what could have caused network delays on the server side - a NetScaler appliance that was slow in processing traffic from the server? or a traffic congestion on the server network?
Server side NS delay	Indicates the average latency experienced by this session, which was caused by the NetScaler appliance when ICA traffic flowed from server network to client network.	Msecs	
Client jitter	Indicates the client side jitter.	Msecs	Jitter is defined as a variation in the delay of received packets. At the sending side, packets are sent in a continuous stream with the packets spaced evenly apart. Due to network congestion, improper queuing, or configuration errors, this steady stream can become lumpy, or the delay between each packet can vary instead of remaining constant. A high value for these measures therefore is indicative of a long time gap between ICA packets. To know where the delay is longer – whether on the client side or on the server side - compare the value of the Client jitter measure with that of the Server jitter measure. Also, if the value of the *Round trip time – RTT* measure is abnormally high for a user, then you can compare the values of these measures with that of the *WAN latency* and *DC latency* measures to know what is causing the problem – the

Measurement	Description	Measurement Unit	Interpretation
			client side network? or the server side network?
Server jitter	Indicates the server side jitter.	Msecs	**These measures are reported only for application sessions, and not desktop sessions.**

Use the detailed diagnosis of the *Active applications* measure to know which applications are being actively used by a user session. The application startup time, startup duration, application uptime, and module path are displayed for each active application. From this, you can quickly identify applications that took too long to startup and applications that restarted recently, and initiate investigations to find the reasons for the same.

Figure 3.17: The detailed diagnosis of the Active applications measure reported by the Citrix HDX User Active Sessions test

Use the detailed diagnosis of the Application launches measure to know which applications were launched during a user session.

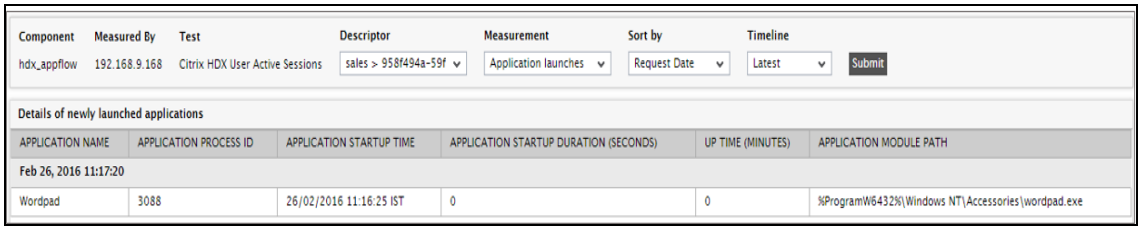
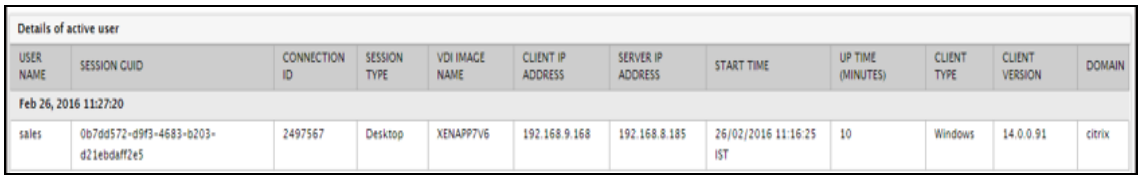


Figure 3.18: The detailed diagnosis of the Application launches measure reported by the Citrix HDX User Active Sessions test

The detailed diagnosis of the Session status measure provides additional details of a user session. If the status of a session is abnormal, you can use these details to know from which client the user is connecting, the client type and version, which server the user is connecting to, the start time, and the uptime of the session. This will help in troubleshooting the abnormal session status.



Details of active user

USER NAME	SESSION GUID	CONNECTION ID	SESSION TYPE	VDI IMAGE NAME	CLIENT IP ADDRESS	SERVER IP ADDRESS	START TIME	UP TIME (MINUTES)	CLIENT TYPE	CLIENT VERSION	DOMAIN
Feb 26, 2016 11:27:20											
sales	0b7dd572-d9f3-4683-b203-d21ebdaff2e5	2497567	Desktop	XENAPP7V6	192.168.9.168	192.168.8.185	26/02/2016 11:16:25 IST	10	Windows	14.0.0.91	citrix

Figure 3.19: The detailed diagnosis of the Session status measure reported by the Citrix HDX User Active Sessions test

3.5.4 Citrix HDX User Agents Test

Users can connect to applications/virtual desktops using may client devices – the ICA client, web browsers, mobile phones, tablets, etc. Each such client device is called a user agent. Users using certain types of client devices/user agents may be engaged in bandwidth-intensive communication over ICA, scarring the experience of other users to the XenApp/XenDesktop environment. To capture such problem conditions quickly, it would be best to know what types of client devices users are connecting from and how much bandwidth each of these device types are currently consuming. This is exactly what the **HDX User Agents** test reveals! This test auto-discovers the types of client devices that are interacting with the applications/virtual desktops, and reports the bandwidth usage of each device type. This way, the test leads administrators to those device types that are consuming bandwidth excessively.

Target of the test : An AppFlow-enabled Citrix NetScaler appliace

Agent deploying the test : A remote agent

Outputs of the test : One set of results for each type of client device/user agent users are connecting from

Configurable parameters for the test

Parameter	Description
Test period	How often should the test be executed. **It is recommended that you set the test period to 5 minutes. This is because, the eG AppFlow Collector is capable of capturing and aggregating AppFlow data related to the last 5 minutes only.**
Host	The host for which the test is to be configured.
Cluster IPs	**This parameter applies only if the NetScaler appliance being monitored is part of a NetScaler cluster.** In this case, configure this parameter with a comma-separated list of IP addresses of all other nodes in that cluster. If the monitored NetScaler appliance is down/unreachable, then the eG AppFlow Collector uses the **Cluster IPs** configuration to figure out which other node in the

Parameter	Description
	cluster it should connect to for pulling AppFlow statistics. Typically, the collector attempts to connect to every IP address that is configured against **Cluster IPs**, in the same sequence in which they are specified. Metrics are pulled from the first cluster node that the collector successfully establishes a connection with.
Enable Logs	This flag is set to **No** by default. This means that, by default, the eG agent does not create AppFlow logs. You can set this flag to **Yes** to enable AppFlow logging. If this is done, then the eG agent automatically writes the raw AppFlow records it reads from the collector into individual CSV files. These CSV files are stored in the *<EG_AGENT_INSTALL_DIR>\NetFlow\data\<IP_of_Monitored_NetScaler>\hdxappflow\actual_csv* folder on the eG agent host. These CSV files provide administrators with granular insights into the HDX appflows, thereby enabling effective troubleshooting. **Note:** By default, the eG agent creates a maximum of 10 CSV files in the **actual_csv** folder. Beyond this point, the older CSV files will be automatically deleted by the eG agent to accommodate new files with current data. Likewise, a single CSV file can by default contain a maximum of 99999 records only. If the records to be written exceed this default value, then the eG agent automatically creates another CSV file to write the data. If required, you can overwrite these default settings . For this, do the following: 1. Login to the eG agent host. 2. Edit the **Netflow.Properties** file in the <EG_AGENT_INSTALL_DIR>\NetFlow\config directory. 3. In the file, look for the parameter, **csv_file_retention_count**. 4. This is the parameter that governs the maximum number of CSV files that can be created in the **auto_csv** folder. By default, this parameter is set to 10. If you want to retain more number of CSV files at any given point in time, you can increase the value of this parameter. If you want to retain only a few CSV files, then decrease the value of this parameter. 5. Next, look for the parameter, **csv_max_flow_record_per_file**. 6. This is the parameter that governs the number of flow records that can be written to a single CSV. By default, this parameter is set to 99999. If you want a

Parameter	Description
	single file to accommodate more records, so that the creation of new CSVs is delayed, then increase the value of this parameter. On the other hand, if you want to reduce the capacity of a CSV file, so that new CSVs are quickly created, then decrease the value of this parameter. 7. Finally, save the file.

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
Total bytes	Indicates the total amount of data (in bytes) transmitted/received by this user agent.	Bytes	
Avg bandwidth	Indicates the rate at which data is transferred over the ICA sessions launched from this device type.	Mbps	Ideally, the value of this measure should be low. A high value indicates excessive bandwidth usage by users connecting from the device type. Compare the value of this measure across device types to know users using which type of device are engaged in bandwidth-intensive communication over ICA.
Percentage of traffic by this user agent	Indicates what percentage of total traffic flowing through the target NetScaler appliance was generated by this user agent.	Percent	Compare the value of this measure across user agents to identify the user agent that is consuming bandwidth excessively. A value close to 100% indicates a user agent/device that is hogging the bandwidth resources.

3.5.5 Citrix HDX Users Test

To ensure that users are able to access applications/desktops on-demand, administrators must closely track that user's accesses, promptly detect probable access latencies, diagnose its root-cause, and take steps to avert it, well before that user notices and complains. To achieve this, administrators can use the **Citrix HDX Users** test. This test automatically discovers the users who are currently accessing applications and virtual desktops in a XenApp/XenDesktop infrastructure, and for each user, reports the latencies that user experienced when interacting with the applications/desktops. This way, the test quickly and accurately points administrators to those users who are experiencing slowness, and also leads them to what is causing the slowness – the network? or the server hosting the applications/desktops? If a latent network is causing the slowness, then the test provides administrators with detailed insights into network performance and enables them to rapidly figure out where the bottleneck lies - on the client-side network? or on the server-side network?

Target of the test : An AppFlow-enabled NetScaler Appliance

Agent deploying the test : A remote agent

Outputs of the test : One set of results for every user who is currently connected to an application/virtual desktop

Configurable parameters for the test

Parameter	Description
Test period	How often should the test be executed. **It is recommended that you set the test period to 5 minutes. This is because, the eG AppFlow Collector is capable of capturing and aggregating AppFlow data related to the last 5 minutes only.**
Host	The host for which the test is to be configured.
Cluster IPs	**This parameter applies only if the NetScaler appliance being monitored is part of a NetScaler cluster.** In this case, configure this parameter with a comma-separated list of IP addresses of all other nodes in that cluster. If the monitored NetScaler appliance is down/unreachable, then the eG AppFlow Collector uses the **Cluster IPs** configuration to figure out which other node in the cluster it should connect to for pulling AppFlow statistics. Typically, the collector attempts to connect to every IP address that is configured against **Cluster IPs**, in the same sequence in which they are specified. Metrics are pulled from the first cluster node that the collector successfully establishes a connection with.
Enable Logs	This flag is set to **No** by default. This means that, by default, the eG agent does not

Parameter	Description
	create AppFlow logs. You can set this flag to **Yes** to enable AppFlow logging. If this is done, then the eG agent automatically writes the raw AppFlow records it reads from the collector into individual CSV files. These CSV files are stored in the *<EG_ AGENT_INSTALL_DIR>\NetFlow\data\<IP_of_Monitored_ NetScaler>\hdxappflow\actual_csv* folder on the eG agent host. These CSV files provide administrators with granular insights into the HDX appflows, thereby enabling effective troubleshooting. **Note:** By default, the eG agent creates a maximum of 10 CSV files in the **actual_csv** folder. Beyond this point, the older CSV files will be automatically deleted by the eG agent to accommodate new files with current data. Likewise, a single CSV file can by default contain a maximum of 99999 records only. If the records to be written exceed this default value, then the eG agent automatically creates another CSV file to write the data. If required, you can overwrite these default settings . For this, do the following: 1. Login to the eG agent host. 2. Edit the **Netflow.Properties** file in the <EG_AGENT_INSTALL_ DIR>\NetFlow\config directory. 3. In the file, look for the parameter, **csv_file_retention_count**. 4. This is the parameter that governs the maximum number of CSV files that can be created in the **auto_csv** folder. By default, this parameter is set to 10. If you want to retain more number of CSV files at any given point in time, you can increase the value of this parameter. If you want to retain only a few CSV files, then decrease the value of this parameter. 5. Next, look for the parameter, **csv_max_flow_record_per_file**. 6. This is the parameter that governs the number of flow records that can be written to a single CSV. By default, this parameter is set to 99999. If you want a single file to accommodate more records, so that the creation of new CSVs is delayed, then increase the value of this parameter. On the other hand, if you want to reduce the capacity of a CSV file, so that new CSVs are quickly created, then decrease the value of this parameter. 7. Finally, save the file.

Parameter	Description
DD Frequency	Refers to the frequency with which detailed diagnosis measures are to be generated for this test. The default is *1:1*. This indicates that, by default, detailed measures will be generated every time this test runs, and also every time the test detects a problem. You can modify this frequency, if you so desire. Also, if you intend to disable the detailed diagnosis capability for this test, you can do so by specifying *none* against **DD Frequency**.
Detailed Diagnosis	

Measurements made by the test

Measurement	Description	Measurement Unit	Interpretation
Active applications	Indicates the number of applications currently accessed by this user.	Number	To know which applications are actively used by this user, use the detailed diagnosis of this measure. **This measure is reported only for application users, and not desktop users.**
Application launches	Indicates the number of applications that were launched by this user.	Number	To know which applications were launched by this user, use the detailed diagnosis of this measure. **This measure is reported only for application users, and not desktop users.**
Application terminates	Indicates the number of applications terminated by this user.	Number	To know which applications were terminated in this user, use the detailed diagnosis of this measure. **This measure is reported only for application users, and not desktop users.**
Active desktops	Indicates the number of desktops currently accessed by this user.	Number	To know which adesktops are actively used by this user, use the detailed diagnosis of this measure. **This measure is reported only for desktop users, and not application**

<table>
<tr><th>Measurement</th><th>Description</th><th>Measurement Unit</th><th>Interpretation</th></tr>
<tr><td></td><td></td><td></td><td>users.</td></tr>
<tr><td>Desktop launches</td><td>Indicates the number of desktops launched by this user.</td><td>Number</td><td>To know which desktops were launched by this user, use the detailed diagnosis of this measure.
This measure is reported only for desktop users, and not application users.</td></tr>
<tr><td>Session status</td><td>Indicates the current status of this user.</td><td></td><td>The values that this measure can take and the numeric values that correspond to each measure value are listed in the table below:
<table><tr><th>Measure Value</th><th>Numeric Value</th></tr><tr><td>Active</td><td>0</td></tr><tr><td>SR successful</td><td>1000</td></tr><tr><td>Existing ICA session got terminated</td><td>1001</td></tr><tr><td>Existing ICA connection got terminated and SR failed</td><td>1002</td></tr><tr><td>Existing ICA connection terminated and SR failed and client is trying to do ACR and is successful</td><td>1003</td></tr></table>Note:
Typically, this test reports the Measure Values in the table above to indicate session status. In the graph of this measure however, the same is indicated using the numeric equivalents only.</td></tr>
<tr><td>Average</td><td>Indicates the average time</td><td>Msecs</td><td>A high value for this measure indicates</td></tr>
</table>

Measurement	Description	Measurement Unit	Interpretation
application startup duration	that elapsed between when an application accessed by this user was launched and when it started running.		that one/more applications are starting up slowly on the server. In this case, use the detailed diagnosis of the Active applications measure to know which application is the slowest in starting up. **This measure is reported only for application users, and not desktop users.**
RTT	Indicates the screen lag experienced by this user while interacting with applications/desktops.	Msecs	A high value for this measure is indicative of the poor quality of a user's experience with applications/desktops. To know the reason for this below-par UX, compare the value of the *WAN latency, DC latency, and Host delay* measures of that user.
WAN latency	Indicates the average latency experienced by this user due to problems with the client side network.	Msecs	A high value for this measure indicates that the client side network is slow. If the value of the *RTT* measure is abnormally high for a user, you can compare the value of this measure with that of the *DC latency and Host delay, and* measures of that user to know what is causing the slowness – is it the client side network? the server side network? or the server hosting the applications/desktops?
DC latency	Indicates the average latency experienced by this user due to problems with the server side network.	Msecs	A high value for this measure indicates that the server side network is slow. If the value of the *RTT* measure is abnormally high for a user, you can compare the value of this measure with that of the *WAN latency and Host delay* measures of that user to know what is causing the slowness – is it the client side network? the server side

Measurement	Description	Measurement Unit	Interpretation
			network? or the server hosting the applications/desktops?
Host delay	Indicates the delay that this user experienced when waiting for the host to process the packets.	Msecs	A high value for this measure indicates a processing bottleneck with the server hosting the applications. If the value of the *RTT* measure is abnormally high for a user, you can compare the value of this measure with that of the *WAN latency and DC latency,* measures to know what is causing the slowness – is it the client side network? the server side network? or the server hosting the applications/desktops?
Bandwidth	Indicates the bandwidth used by this user.	Kbps	Ideally, the value of this measure should be low. A high value indicates excessive bandwidth usage by the user. Compare the value of this measure across users to know which user is consuming bandwidth excessively.
Bytes	Indicates the total bytes consumed by this user's sessions.	Bytes	Compare the value of this measure across users to know which user has the maximum throughput and which has the least.
Client side retransmits	Indicates the number of packets retransmitted on the client side connection during the last measurement period.	Number	Ideally, the value of these measures should be 0.
Server side retransmits	Indicates the number of packets retransmitted on the server side connection during the last measurement period.	Number	

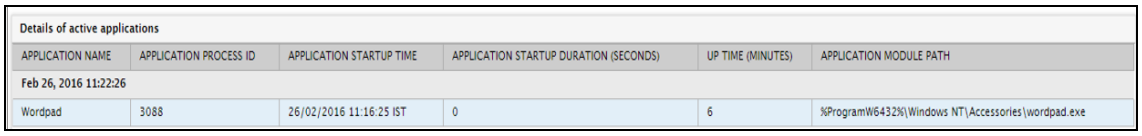
Measurement	Description	Measurement Unit	Interpretation
Client side 0 win count	Indicates how many times this user's client advertised a zero TCP window during the last measurement period.	Number	TCP Zero Window is when the Window size in a machine remains at zero for a specified amount of time. TCP Window size is the amount of information that a machine can receive during a TCP session and still be able to process the data. Think of it like a TCP receive buffer. When a machine initiates a TCP connection to a server, it will let the server know how much data it can receive by the Window Size. In many Windows machines, this value is around 64512 bytes. As the TCP session is initiated and the server begins sending data, the client will decrement it's Window Size as this buffer fills. At the same time, the client is processing the data in the buffer, and is emptying it, making room for more data. Through TCP ACK frames, the client informs the server of how much room is in this buffer. If the TCP Window Size goes down to 0, the client will not be able to receive any more data until it processes and opens the buffer up again. The machine (client/server) alerting the Zero Window will not receive any more data from the host. This is why, ideally, the value of these measures should be 0. A non-zero value warrants an immediate investigation to determine the reason for the Zero Window. It could be that the client/server was running too many processes at that moment, and its processor is maxed. Or it could be that

Measurement	Description	Measurement Unit	Interpretation
			there is an error in the TCP receiver, like a Windows registry misconfiguration. Try to determine what the client was doing when the TCP Zero Window happened. **These measures are reported only for application users, and not desktop users.**
Server side 0 win count	Indicates how many times during this user's sessions the server advertised a zero TCP window during the last measurement period.	Number	
Client RTO	Indicates how many times during the last measurement period the retransmit timeout got invoked on this user's client side connection.	Number	An RTO occurs when the sender is missing too many acknowledgments and decides to take a time out and stop sending altogether. After some amount of time, usually at least one second, the sender cautiously starts sending again, testing the waters with just one packet at first, then two packets, and so on. As a result, an RTO causes, at minimum, a one-second delay on your network. A low value is hence desired for these measures. **These measures are reported only for application sessions, and not desktop sessions.**
Server RTO	Indicates how many times during the last measurement period the retransmit timeout got invoked on this user's server side connection.	Number	
ACR counts	Indicates the total number of times the client automatically reconnected this user to sessions.	Number	The Automatic Client Reconnect (ACR) policy setting, when enabled, allows automatic reconnection by the same client after a connection has been interrupted. Allowing automatic client reconnect allows users to resume working where they were interrupted when a connection was broken. Automatic reconnection detects broken connections and then reconnects the users to their sessions.
Session reconnects	Indicates the number of times this user's sessions	Number	This measure includes only those times a user reconnected to a disconnected

Measurement	Description	Measurement Unit	Interpretation
	reconnected.		session by mechanisms other than the ACR setting.
Client SRTT	Indicates the RTT (round-trip time or screen lag time) for this user smoothed over the client side connection.	MSecs	TCP implementations attempt to predict future round-trip times by sampling the behavior of packets sent over a connection and averaging those samples into a ‘‘smoothed’’ round-trip time estimate, SRTT. When a packet is sent over a TCP connection, the sender times how long it takes for it to be acknowledged, producing a sequence, S, of round-trip time samples: s1, s2, s3.... With each new sample, si, the new SRTT is computed from the formula: $SRTTi+1 = (\alpha \times SRTTi) + (1 - \alpha) xsi$ Here, SRTTi is the current estimate of the round-trip time, SRTTi+1 is the new computed value, and α is a constant between 0 and 1 that controls how rapidly the SRTT adapts to change. The retransmission time-out (RTOi), the amount of time the sender will wait for a given packet to be acknowledged, is computed from SRTTi. The formula is: $RTOi = \beta \times SRTTi$ Here, β is a constant, greater than 1, chosen such that there is an acceptably small probability that the round-trip time for the packet will exceed RTOi. **These measures are reported only for application sessions, and not desktop sessions.**
Server SRTT	Indicates the RTT (round-trip time or screen lag time) of this session, smoothed over the server side connection.	MSecs	
Client jitter	Indicates the client side jitter.	Msecs	Jitter is defined as a variation in the delay of received packets. At the

Measurement	Description	Measurement Unit	Interpretation
			sending side, packets are sent in a continuous stream with the packets spaced evenly apart. Due to network congestion, improper queuing, or configuration errors, this steady stream can become lumpy, or the delay between each packet can vary instead of remaining constant. A high value for these measures therefore is indicative of a long time gap between ICA packets. To know where the delay is longer – whether on the client side or on the server side - compare the value of the Client jitter measure with that of the Server jitter measure. Also, if the value of the *Round trip time – RTT* measure is abnormally high for a user, then you can compare the values of these measures with that of the *WAN latency* and *DC latency* measures to know what is causing the problem – the client side network? or the server side network? **These measures are reported only for application sessions, and not desktop sessions.**
Server jitter	Indicates the server side jitter.	Msecs	

Use the detailed diagnosis of the *Active applications* measure to know which applications are being actively used by a user. The application startup time, startup duration, application uptime, and module path are displayed for each active application. From this, you can quickly identify applications that took too long to startup and applications that restarted recently, and initiate investigations to find the reasons for the same.

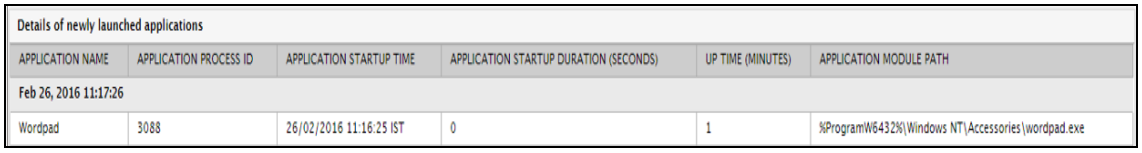


Details of active applications

APPLICATION NAME	APPLICATION PROCESS ID	APPLICATION STARTUP TIME	APPLICATION STARTUP DURATION (SECONDS)	UP TIME (MINUTES)	APPLICATION MODULE PATH
Feb 26, 2016 11:22:26					
Wordpad	3088	26/02/2016 11:16:25 IST	0	6	%ProgramW6432%\Windows NT\Accessories\wordpad.exe

Figure 3.20: The detailed diagnosis of the Active applications measure reported by the Citrix HDX Users test

Use the detailed diagnosis of the Application launches measure to know which applications were launched by a user.

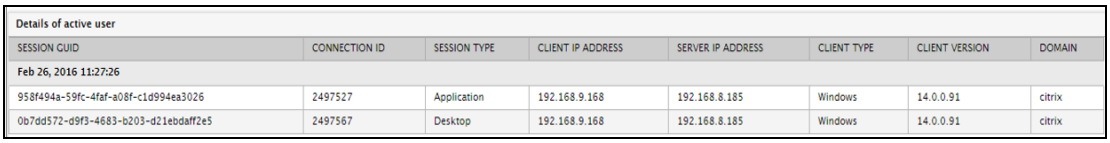


Details of newly launched applications

APPLICATION NAME	APPLICATION PROCESS ID	APPLICATION STARTUP TIME	APPLICATION STARTUP DURATION (SECONDS)	UP TIME (MINUTES)	APPLICATION MODULE PATH
Feb 26, 2016 11:17:26					
Wordpad	3088	26/02/2016 11:16:25 IST	0	1	%ProgramW6432%\Windows NT\Accessories\wordpad.exe

Figure 3.21: The detailed diagnosis of the Application launches measure reported by the Citrix HDX Users test

The detailed diagnosis of the *Session status* measure provides additional details of a user. If the status of a session is abnormal, you can use these details to know from which client the user is connecting, the client type and version, which server the user is connecting to, the start time, and the uptime of the session. This will help in troubleshooting the abnormal session status.

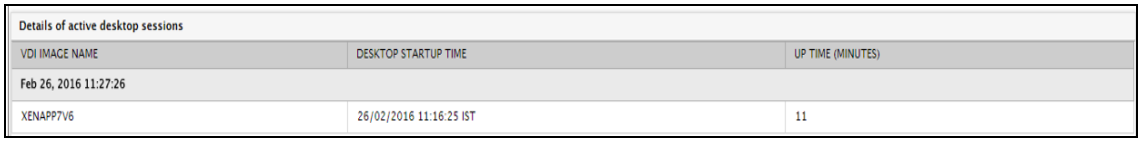


Details of active user

SESSION GUID	CONNECTION ID	SESSION TYPE	CLIENT IP ADDRESS	SERVER IP ADDRESS	CLIENT TYPE	CLIENT VERSION	DOMAIN
Feb 26, 2016 11:27:26							
958f494a-59fc-4faf-a08f-c1d994ea3026	2497527	Application	192.168.9.168	192.168.8.185	Windows	14.0.0.91	citrix
0b7dd572-d9f3-4683-b203-d21ebdaff2e5	2497567	Desktop	192.168.9.168	192.168.8.185	Windows	14.0.0.91	citrix

Figure 3.22: The detailed diagnosis of the Session status measure reported by the Citrix HDX Users test

For a desktop user, you can know which desktop that user is currently logged into using the detailed diagnosis of the Active desktops measure. The time at which the desktop started up and the uptime of the desktop are revealed, so that you can instantly figure out whether the desktop experienced any unusual/unscheduled reboot.

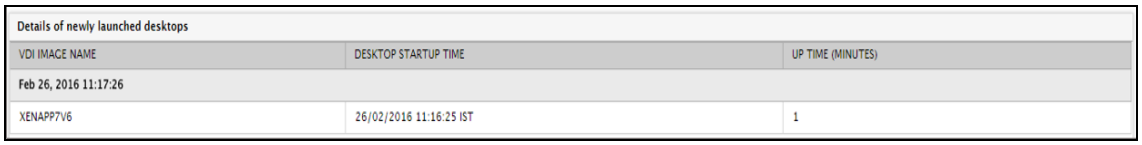


Details of active desktop sessions

VDI IMAGE NAME	DESKTOP STARTUP TIME	UP TIME (MINUTES)
Feb 26, 2016 11:27:26		
XENAPP7V6	26/02/2016 11:16:25 IST	11

Figure 3.23: The detailed diagnosis of the Active desktops measure

Use the detailed diagnosis of the Desktop launches measure to know which desktop(s) was recently launched by the user.



Details of newly launched desktops

VDI IMAGE NAME	DESKTOP STARTUP TIME	UP TIME (MINUTES)
Feb 26, 2016 11:17:26		
XENAPP7V6	26/02/2016 11:16:25 IST	1

Figure 3.24: The detailed diagnosis of the Desktop launches measure of the Citrix HDX Users test

About eG Innovations

eG Innovations provides intelligent performance management solutions that automate and dramatically accelerate the discovery, diagnosis, and resolution of IT performance issues in on-premises, cloud and hybrid environments. Where traditional monitoring tools often fail to provide insight into the performance drivers of business services and user experience, eG Innovations provides total performance visibility across every layer and every tier of the IT infrastructure that supports the business service chain. From desktops to applications, from servers to network and storage, from virtualization to cloud, eG Innovations helps companies proactively discover, instantly diagnose, and rapidly resolve even the most challenging performance and user experience issues.

eG Innovations is dedicated to helping businesses across the globe transform IT service delivery into a competitive advantage and a center for productivity, growth and profit. Many of the world's largest businesses use eG Enterprise to enhance IT service performance, increase operational efficiency, ensure IT effectiveness and deliver on the ROI promise of transformational IT investments across physical, virtual and cloud environments.

To learn more visit www.eginnovations.com.

Contact Us

For support queries, email support@eginnovations.com.

To contact eG Innovations sales team, email sales@eginnovations.com.