

Metric Interpretation Table

Metric Name	Metric Definition	Metric Use and Interpretation
95th Percentile Throughput (Inbound/Outbound)	The 95th percentile is a commonly used statistic for evaluating a network's peak utilization. Specifically, the 95th percentile says that 95 percent of the time, usage is below a given amount. The 95th Percentile Throughput rate is determined by discarding the top 5 percent of all throughput rate measurements in the selected time interval. The highest remaining throughput rate is the 95th percentile. All calculations are done on 5-minute data granularity.	95 th Percentile Throughput is the usual basis of billing by telcos and ISPs. This metric is useful for validating telco/ISP bills, and for "rightsizing" WAN links.
AS Description	AS Description is the description field of the autonomous system of the group from the WHOIS database (page 57). Information metrics can be added and removed from tables, but cannot be displayed in charts.	This identifies the owner of a given Autonomous System, and can be important when troubleshooting Internet connectivity and topology.
AS Name	AS Name is the name of the autonomous system to which the group belongs. This information is only available for AS groups (ISP AS, Peer AS or Dest AS) and IP addresses. Information metrics can be added and removed from tables, but cannot be displayed in charts.	See above.
AS Number	AS Number is the autonomous system number of the group. This information is only available for AS groups (ISP AS, Peer AS or Dest AS) and IP addresses. Information metrics can be added and removed from tables, but cannot be displayed in charts.	See above.

<p>Client Reset Rate (TCP Clients/Servers)</p>	<p>Client Reset Rate is the number of TCP connections per second that are terminated with a TCP reset from a client over the selected time interval.</p>	<p>This metric helps you monitor abnormal connection terminations. For instance, some servers, (e.g., ISA) will send a reset when a connection is attempted on a restricted port or IP address. Client Reset Rate can be used in conjunction with the Failed Connections metric to look for abnormal behavior that may indicate malicious activity.</p> <p>Some other causes of changes in Client Reset Rate:</p> <ul style="list-style-type: none"> • Client-side aborts <ul style="list-style-type: none"> ○ This may be due to user impatience due to slow response <ul style="list-style-type: none"> ▪ Correlate against RTT and/or Retransmission Delay • Loss of connectivity to the server <ul style="list-style-type: none"> ○ Correlate against Connections Failed • Application is written to use Client Resets to terminate a connection
<p>Connection Duration (TCP Clients/Servers)</p>	<p>Connection Duration is the average duration of all TCP connections that terminated in the selected time interval. Under regular network conditions, Connection Duration is measured from the beginning of the TCP three-way handshake (SYN sent by the client) to the end of the TCP connection (client ACK of the server FIN). Under asymmetric traffic conditions where client traffic is not observed, Connection Duration is measured from the server SYN-ACK to the server FIN. In the event that the connection is reset without the standard TCP termination sequence, the end of the connection is considered to be the time at which the TCP reset is observed. Table averages include TCP connections that time out: those for which neither a TCP SYN packet or a TCP FIN packet is observed within a 30-second interval.</p>	<p>This is a good baseline metric: i.e., it can be useful to monitor it over time for a specific application. It can be used, for instance, as a tip-off to excessive Web browsing.</p> <p>Note that Connection Duration has little meaning for applications that use persistent connections.</p>

<p>Connection Rate (TCP Clients/Servers)</p>	<p>Connection Rate is the number of successful TCP connections per second. The TCP three-way handshake must be completed for the connection to be counted as a connection.</p>	<p>All connection metrics are indicative of server load.</p> <p>Connection Rate should correlate with Connection Request Rate and Connection Failed Rate as follows:</p> <p>Connection Rate = Connection Request Rate – Connection Failed Rate</p> <p>If this correlation is not observed, it implies either a spanning problem (a network problem) or asymmetric traffic (an appliance placement problem).</p> <p>If you chart Connection Rate against Connection Request Rate, and the lines track each other closely, this is a good indication of smooth network and server operation: e.g., correct routing, properly loaded servers, and good client/server communication.</p> <p>This metric may also be useful for application/server discovery: high Connection Rate (Server) in a Business Group indicates the presence in that group of a busy server.</p> <p>Correlating Connection Rate against Turn Rate or Throughput can be useful in discovering chatty applications or persistent connections.</p> <p>Connection Rate can also be used to assist application tuning by baselining concurrent connections.</p>
--	--	---

<p>Connection Request Rate (TCP Clients/Servers)</p>	<p>Connection Request Rate is the number of attempted TCP connections per second over the selected time interval. An attempted connection occurs when the client sends a TCP SYN request to the server, regardless of whether or not the server responds.</p>	<p>All connection metrics are indicative of server load.</p> <p>Useful when correlated with Connection Rate, above, or Connection Failed rate, below.</p> <p>This metric indicates attempts by one machine to connect to another. It is useful for the following tasks:</p> <ul style="list-style-type: none"> • Worm Detection • Firewall Validation • Detecting active communication applications • Correlation with Connections Failed Rate to detect problematic applications or hardware • Mapping application usage (TCP Clients vs. TCP Servers) • Server and Client usage • Behavioral detection: heartbeat communications • Detecting SYN Floods/DOS attacks • Identifying key servers
<p>Connection Requests (TCP Clients/Servers)</p>	<p>Connection Requests is the number of attempted TCP connections over the selected time interval. An attempted connection occurs when the client sends a TCP SYN request to the server, regardless of whether or not the server responds.</p>	<p>All connection metrics are indicative of server load.</p> <p>Similar uses to Connection Request Rate, above.</p>

<p>Connection Setup Time (TCP Clients/Servers)</p>	<p>Connection Setup Time is the average time required to establish the three-way handshake at the beginning of all TCP connections over the selected time interval. In other words, it is the time between the first TCP SYN packet and the TCP ACK sent by the client.</p>	<p>Connection Setup Time is a combination of network RTT and the TCP stack response time in setting up a connection. A high Connection Setup Time when RTT is low, or a variable Connection Setup Time when RTT is constant, may indicate a TCP stack or other “front-end” problem. Another cause of high Connection Setup Time can be too many connection requests.</p> <p>Because the three-way handshake uses small packets, and because stack issues are rare, it is sometimes instructive to use CST as a check on RTT, since RTT involves data packets, which has the effect of averaging in Payload Transfer Time. In other words, if you know that the hardware is performing correctly, CST can be more accurate than RTT, especially for “non-chatty” applications.</p>
<p>Connections (TCP Clients/Servers)</p>	<p>Connections is the number of successful TCP connections over the selected time interval. The TCP three-way handshake must be completed for the connection to be counted as a connection.</p>	<p>All connection metrics are indicative of server load.</p> <p>Similar uses to Connection Rate, above.</p> <p>As with Connection Rate, Connections should correlate with Connection Requests and Connections Failed as follows:</p> <p>Connections= Connection Requests – Connections Failed</p> <p>If this correlation is not observed, it implies either a spanning problem (a network problem) or asymmetric traffic (an appliance placement problem).</p>

Connections Failed (TCP Clients/Servers)	Connections Failed is the number of Connection Requests to which servers in the group do not respond over the selected time interval.	<p>This is one of the most important metrics in NetSensory, and is the first one to check when looking for the following problems:</p> <ul style="list-style-type: none"> • Worms • Decommissioned machines • Web-based spyware detection • Improperly working applications (both client and server side) • Unauthorized web-based programs (e.g., Webshots) • Expired Subscription Update Servers (e.g., NAV) • Failed links • Improper configuration(e.g., TCP, routing)
Connections Failed Rate (TCP Clients/Servers)	Connections Failed Rate is the number of failed connections per second over the selected time interval.	See Connections Rate, above.
Data Transfer Time (Clients/Servers)	Data Transfer Time—a measure of the time between the first byte sent from the turn Server in response to a turn Client request to the ACK from the turn Client of the last response packet sent—is the average time required to transfer data for all TCP turns that terminated in the selected time interval.	<p>Data Transfer Time is a “user-centric” metric in that it includes retransmissions, so it represents the actual average time required to deliver the average payload to the user, in contrast to Payload Transfer time, which measures only the time to deliver useful (i.e., application data) payload. Changes in DTT can arise from many causes, among which are:</p> <ul style="list-style-type: none"> • More data being transferred <ul style="list-style-type: none"> ○ Correlate against Payload Transfer Time • RTT increases <ul style="list-style-type: none"> ○ Correlate against RTT • Packet Loss <ul style="list-style-type: none"> ○ Correlate against Packet Loss or Retransmission Delay <p>Payload Transfer Time (see below) is more generally useful than Data Transfer Time because it is a component of User Response Time and the Response Time Composition Chart (RTCC).</p>

<p>Goodput (Inbound/Outbound)</p>	<p>Goodput is the effective Payload transfer rate for all TCP connections over the selected time interval. Goodput measures the rate of data transferred within the TCP packet payload. It does not include packets that are retransmitted or TCP packets that do not contain data (e.g., TCP SYN, TCP ACK).</p>	<p>Answers the question: “What fraction of my ‘pipe’ is being used for real data?” May be useful for application tuning/design.</p> <p>If Goodput is low while retransmissions are low, this may indicate a “chatty” application.</p>
<p>Group</p>	<p>The Group column identifies the group for a row within a table. Information metrics can be added and removed from tables, but cannot be displayed in charts.</p>	<p>This is the assigned name of a Business Group or Application. It cannot be changed without eliminating all stored data about the group.</p>
<p>Group Definition</p>	<p>The Group Definition column displays the definition for a group displayed within a table. For instance, in rows showing business groups the group definition metric will display the IP addresses within the business group. In rows showing applications, the group definition metric will display the IP Protocol and Port(s) assigned to the application. Information metrics can be added and removed from tables, but cannot be displayed in charts.</p>	<p>A quick reference to the meaning of the group.</p>
<p>Group Membership</p>	<p>The Group Membership column displays all of the Business Groups to which an IP address belongs, or all of the Business Group Containers a Business Group belongs to.</p>	<p>May be useful when configuring a distributed NetSensory system.</p>

Information	<p>The Information column provides additional information for the group of a row within a table. The Information column displays the DNS hostname for IP addresses and shows the autonomous system name for AS groups (ISP AS, Peer AS, Dest AS). Information metrics can be added and removed from tables, but cannot be displayed in charts.</p>	<p>For Business Groups, this can be used in conjunction with the Search function in NetSensory Insights to focus on a particular business-significant set of Business Groups. For instance, if you put the string “RO” in the Information field for all Business Groups that are remote offices, searching on “RO” will give you data only on remote offices. You could extend this by adding a string to identify the remote office’s region, and so forth.</p> <p>NOTE: the Information field may be changed at any time without affecting data collection. This is important because although Business Group Containers offer a similar functionality in some cases, they can only represent data collected since their creation, and cannot be changed without data loss; whereas using the Information field allows a form of hierarchy that is retroactive to the creation of the string and which can be changed without data loss.</p>
ISP Peering Point Round Trip Time	<p>ISP Peering Point RTT is the average round trip time (in milliseconds) between the ISP border router and the border router of the ISP’s Peer AS (see “Built-in Groups” on page 77 for a definition of ISP AS and Peer AS). This value represents the latency across the ISP peering point. Each hop in the route to the destination IP address is mapped to an AS using the appliance WHOIS database (page 57). The ISP peering point is identified in the traceroute results as the hop between the ISP AS and Peer AS.</p>	<p>This metric helps determine network latency between an ISP and its peers, and may be useful in diagnosing routing problems or poor ISP performance.</p> <p>However, because it depends on ICMP/TCP traceroutes, which are usually handled differently by routers than actual data traffic, it cannot be guaranteed to be accurate. (See Round Trip Time, below.) It should be treated as just one more data point in diagnosis., and, if Round Trip Time is not excessive, should be disregarded.</p>

<p>Packet Loss (Inbound/Outbound)</p>	<p>Packet Loss is the average percentage of packets lost (resulting in a retransmission) transmitted from a server to a client for TCP connections over the selected time interval. Packet Loss represents the total percentage of packets lost in end-to-end communications. It should not be compared to packet loss reported on a router, which only accounts for loss on a single link in the end-to-end communication.</p>	<p>Packet loss degrades network and application performance and is a common problem on public networks in particular.</p> <p>Because Packet Loss is a ratio, it can be deceiving, and should be checked against Packet Retransmission Rate, which is a scalar. For example, if a user sends two packets and one is lost, the Packet Loss would be 50%, but if the Packet Retransmission Rate is 1/hour, it's unlikely to represent a real problem.</p> <p>It should be noted that the Packet Loss metric is based on the assumption that a retransmission indicates packet loss. In reality, it may also represent a retransmission due to a TCP or application time out, but as the effect on an application is the same, this is not really a problem. Nonetheless, it is useful to correlate Packet Loss with RTT: if RTT is very high, this metric may not actually represent packet loss.</p> <p>Packet Loss can point to many of the same problems as RTT (see below).</p> <ul style="list-style-type: none"> • Router/route problems <ul style="list-style-type: none"> ○ Differential diagnosis against multiple BGs using the same application/server ○ Correlate against traceroute metrics, may not be a reliable indication • Hardware problems (e.g. firewall, hub, switch, load balancer, proxy, etc.). <ul style="list-style-type: none"> ○ May require packet capture and analysis for diagnosis • Internet "pipe" saturation <ul style="list-style-type: none"> ○ Correlate against Throughput • Large file transfers swamping a link <ul style="list-style-type: none"> ○ Correlate against Throughput ○ Use Bandwidth Hog Insight • Rogue applications swamping link
---	---	--

<p>Packet Loss (cont.)</p>		<ul style="list-style-type: none"> ○ Correlate against Throughput ○ Use Bandwidth Hog Insight • Malware <ul style="list-style-type: none"> ○ Correlate against Throughput ○ Use Bandwidth Hog Insight ○ Correlate against Connection Failed metrics • TCP stack issue <ul style="list-style-type: none"> ○ May require packet capture and analysis for diagnosis <p>Packet loss is useful for differential diagnosis to distinguish between network and application problems. If packet loss is observed for all traffic between two groups, then it's likely to be a network problem; if it is observed only for a given application or server, it's likely to be an application or server problem.</p> <p>The meaning of Inbound and Outbound is dependent on the location of the appliance with respect to the Business Group for which Packet Loss is being measured. For a typical installation where the BG is local and the Connected Group is "outside" (e.g., a remote office), Packet Loss (Inbound) = LAN problems, while RTT (Outbound) = WAN problems. This relationship is, of course, reversed for remote Business Groups.</p>
<p>Packet Payload (Clients/Servers)</p>	<p>Packet Payload is the average number of TCP data packets required to transfer data in turns from the turn Servers in the group to turn Clients over the selected time interval. Packet Payload does not include TCP handshake, termination, retransmissions or ACK packets that do not include data.</p>	<p>Packet Payload is a rough measure of an application's TCP efficiency: the lower the number for a given application, the "chattier" that application is and the more sensitive it is to network conditions. Such an application may benefit from application tuning or acceleration, or from WAN optimization.</p> <p>$\text{Goodput/Packet Payload} \cong \text{Average Payload}$</p>
<p>Packet Retransmission Rate (Inbound/Outbound)</p>	<p>Packet Retransmission Rate is the number of retransmitted TCP packets per second over the selected time interval.</p>	<p>See Packet Loss, above.</p>

<p>Packet Throughput (Inbound/Outbound)</p>	<p>Packet Throughput is the number of packets transmitted per second over the selected time interval.</p>	<p>This useful for:</p> <ul style="list-style-type: none"> • “Rightsizing” a NetSensory installation <ul style="list-style-type: none"> ○ For instance, an NP-2000 appliance can handle about 120,000 pps (depending on average packet size and other parameters). If Packet Throughput is consistently near this limit, it may be time to add another appliance (using a Gigamon switch) and a Director to make sure no traffic statistics are lost. • Diagnosing DOS attacks <ul style="list-style-type: none"> ○ Especially for non-TCP traffic where connections metrics (e.g., Connections Failed) are not available • Calculating average packet size <ul style="list-style-type: none"> ○ $Average\ Packet\ Size = \frac{Throughput}{Packet\ Throughput}$
<p>Packet Traffic (Inbound/Outbound)</p>	<p>Packet Traffic is the number of packets observed over the selected time interval.</p>	<p>See Packet Throughput, above.</p>
<p>Payload (Clients/Servers)</p>	<p>Payload is the average size of data transfers from turn Servers in the group to turn Clients for all TCP turns that terminated in the selected time interval. Payload does not include retransmissions or ACK packets that do not include data.</p>	<p>Payload is a measure of the amount of useful data the network is carrying. It is an “application-centric” metric that can be useful, in combination with the turn metrics, in application tuning.</p> <p>$Payload = \frac{Average\ payload}{turn} - headers$</p>
<p>Payload Transfer Time</p>	<p>Payload Transfer Time represents the average time it takes to deliver a good payload per Turn. This new metric is a subset of the existing Data Transfer Time where $Payload\ Transfer\ Time = Data\ Transfer\ Time - Retransmission\ Delay$.</p>	<p>As with Payload, this is an “application-centric” metric useful for application tuning. (Contrast with Data Transfer Time, which includes retransmitted packets, and so is a more useful metric for gauging user experience.)</p> <p>Payload Transfer Time is used in the Response Time Composition Chart to avoid counting the impact of packet retransmissions twice, as would happen if Data Transfer Time were used instead.</p>

<p>Retransmission Delay (Inbound/Outbound)</p>	<p>Retransmission Delay is the average latency added to TCP turns due to packet retransmissions over the selected time interval. Retransmission Delay is calculated as Data Transfer Time multiplied by an empirically-derived function of packet loss.</p>	<p>Retransmission Delay is an essential metric for the Response Time Composition Chart and User Response Time, as it translates packet loss from a ratio to a temporal metric using an empirical correlation between Data Transfer Time and Packet Loss. It is an excellent measure of the user impact of packet loss and retransmissions.</p> <p>Retransmission Delay is often more useful than Packet Loss, as it represents the impact of packet loss on the user experience.</p> <p>It is often useful to correlate this with Server Reset Rate (see below) to see if there are server issues masquerading as packet loss.</p>
<p>Retransmission Rate (Inbound/Outbound)</p>	<p>Retransmission Rate is the amount (bits) of retransmitted TCP data per second over the selected time interval.</p>	<p>Useful as a check on Packet Loss (see above).</p> <p>Retransmission Rate can also be useful when managing WAN links. Ranking remote destinations by Retransmission Rate can help judge how which of them are losing the most bandwidth to retransmissions, regardless of what they're due to.</p> <p>Goodput + Retransmission Rate = Total TCP Payload/Second = Throughput – (packet headers)</p>
<p>Round Trip Time (Inbound/Outbound)</p>	<p>Round Trip Time is the average round trip time between the NetSensory Appliance and the client for all TCP connections over the selected time interval. Round Trip Time is measured as the time elapsed between an outgoing TCP data packet from a server and the receipt of the corresponding incoming TCP ACK from a client.</p>	<p>Round Trip Time is one of the most important metrics for judging the impact of network conditions on application performance. Since it measures the actual RTT for data, it is much more accurate than traceroute measurements. RTT can point to many different issues:</p> <ul style="list-style-type: none"> • Router/route problems <ul style="list-style-type: none"> ○ Differential diagnosis against multiple BGs using the same application/server ○ Correlate against traceroute metrics, may not be a reliable indication

<p>Round Trip Time (cont.)</p>		<ul style="list-style-type: none"> • Hardware problems (e.g. firewall, hub, switch, load balancer, proxy, etc.). <ul style="list-style-type: none"> ○ For instance, variable round trip time on a VLAN in the presence of spiking traffic (correlate against Throughput) may indicate the presence of an unauthorized hub. • Internet “pipe” saturation <ul style="list-style-type: none"> ○ Correlate against Throughput • Large file transfers swamping a link <ul style="list-style-type: none"> ○ Correlate against Throughput ○ Use Bandwidth Hog Insight • Rogue applications swamping link <ul style="list-style-type: none"> ○ Correlate against Throughput ○ Use Bandwidth Hog Insight • Malware <ul style="list-style-type: none"> ○ Correlate against Throughput ○ Use Bandwidth Hog Insight ○ Correlate against Connection Failed metrics <p>The meaning of Inbound and Outbound is dependent on the location of the appliance with respect to the Business Group for which RTT is being measured. For a typical installation where the BG is local and the Connected Group is “outside” (e.g., a remote office), RTT (Inbound) = LAN latency, while RTT (Outbound) = WAN latency. This relationship is, of course, reversed for remote Business Groups.</p> <p>Note that Connection Setup Time (see above) can be used as a check on the accuracy of RTT.</p>
--------------------------------	--	---

<p>Server Reset Rate (TCP Clients/Servers)</p>	<p>Server Reset Rate is the number of TCP connections that are terminated with a TCP reset by a server per second over the selected time interval.</p>	<p>Server Reset Rate helps determine if the server is having problems maintaining connections. It can usefully be correlated with retransmission metrics to reveal the impact of packet loss or timeouts on server behavior.</p> <p>Other causes of changes in Server Reset Rate include:</p> <ul style="list-style-type: none"> • Loss of connectivity to the client <ul style="list-style-type: none"> ○ Correlate against Connections Failed • Application is written to use server resets to terminate the connection
<p>Server Response Time (Clients/Servers)</p>	<p>Server Response Time is the average turn Server latency in responding to a turn Client request for all TCP turns over the selected time interval. It is calculated as the time between the last packet of the turn request and the first data packet of the response.</p>	<p>This is one of the most important NetSensory metrics, as it can be used to distinguish between server and network problems. An increase in Server Response Time should be correlated with one of the Connection metrics (see above), e.g.:</p> <p>High SRT and high Connection Rate = too many users, or possible DOS attack.</p> <p>High SRT and low Connection Rate = server issues. This correlation is performed automatically by the Increased Application Response time Adaptive Alert. With proper spanning of the appliance, SRT and Connection Rate can be used to diagnose problems with one tier of multi-tier application. Further diagnosis can be accomplished via Packet Capture and a dump to a third-party program such as Ethereal.</p> <p>Some server issues, such as a backup in progress or a user doing a large application query, can be diagnosed by further correlation with other metrics (e.g., Payload or Payload Transfer Time or Turn Rate) or a topology chart to detect an unexpected communication link.</p>

<p>Throughput (Inbound/Outbound, Inbound & Outbound)</p>	<p>Throughput is the rate at which data is transmitted over the selected time interval. Throughput is the rate of raw bytes transferred by the group.</p>	<p>Throughput is a key metric for network sizing and for diagnosing application performance problems that are due to an increase in traffic volume. It is often thought of in terms of “utilization.”</p> <p>Causes of changes in Throughput include:</p> <ul style="list-style-type: none"> • Router/route problems <ul style="list-style-type: none"> ○ Differential diagnosis against multiple BGs using the same application/server ○ Correlate against traceroute metrics, may not be a reliable indication • Hardware problems (e.g. firewall, hub, switch, load balancer, proxy, etc.). <ul style="list-style-type: none"> ○ May require packet capture and analysis for diagnosis • Internet “pipe” saturation <ul style="list-style-type: none"> ○ Correlate against Retransmission Rate and/or Packet Loss • Large file transfers swamping a link <ul style="list-style-type: none"> ○ Use Bandwidth Hog Insight • Rogue applications swamping link <ul style="list-style-type: none"> ○ Use Bandwidth Hog Insight • Malware <ul style="list-style-type: none"> ○ Use Bandwidth Hog Insight ○ Correlate against Connection Failed metrics
<p>Time To First Byte</p>	<p>Time to First Byte is the average latency for the servers within the group to transmit the first data byte for all TCP connections over the selected time interval. In other words, it is the time between the beginning of the TCP three-way handshake and the first data packet sent from the server.</p>	<p>Time to First Byte is a combination of Connection Setup Time and the Server Response Time for the first data request of a connection. It is useful as the basis of an alert that can be further diagnosed by checking Connection Setup Time (an increase here indicating a TCP stack or network RTT problem, or overload by too many connection requests) and Server Response Time (an increase here indicating a server or application problem).</p>

Traceroute Round Trip Time	Traceroute Round Trip Time is a measure of the end-to-end round trip time for traceroute traffic. Under many circumstances, this metric value is lower than TCP Round Trip Time. The difference may be due to a variety of reasons. TCP data transfers naturally induce congestion delay due to large data packets sent to the destination, whereas traceroute packets do not have data payload. TCP Round Trip Time measurements may also show higher latency due to delays in the client.	<p>Although not as accurate an indication of network latency as the Round Trip Time metrics (see above) TRTT is nonetheless important for detecting and diagnosing network problems such as route flapping and router congestion. Correlation against Throughput and Packet Loss can be helpful.</p> <p>Given the NetSensory appliance's storage of historical traceroute information, it is a good idea to use Preferred IPs to make sure that critical links and resources are included in the traceroute history, which can greatly accelerate troubleshooting network issues, especially across networks where one has no other sources of information (.e.g., the Internet or a customer's or partner's network).</p>
Traffic (Inbound/Outbound, Inbound & Outbound)	Traffic is the amount of data transmitted over the selected time interval. Traffic is a count of the raw bytes transferred by the group.	The Traffic metric can be useful for distinguishing between the amount of data transferred reported by an application and the actual load on the network as reported by this metric. For instance, over-the-network data backup applications report only the amount of application data transferred, which, depending on the efficiency of the application, may be much lower than the actual amount of traffic generated. For this reason, it is sometimes more useful to set alerts on Traffic rather than Throughput.
Turn Rate (Clients/Servers)	Turn Rate is the number of turn responses from turn Servers in the group per second.	<p>Turn rate correlates closely with transaction rate, and so is an excellent "user-centric" metric for server activity and load. A high turn rate indicates a busy server. It is an important metric for auditing a network in preparation for application rollout or extension.</p> <p>Correlation against Payload measures the "chattiness" of an application as an indication of the possible need for application or network optimization. A low payload per turn makes the application more dependent on network latency.</p>

Turns (Clients/Servers)	Turns is the total number of responses from the turn Servers in the group in the time interval selected.	See Turn Rate, above.
Trans-ISP RTT	Trans-ISP Round Trip Time is a measure of the average time (in milliseconds) required for traceroute packets to cross the ISP AS and come back. In other words, it measures the round trip latency from the ISP AS ingress border router to the egress border router. See "Built-in Groups" on page 77 for a definition of ISP AS.	As with ISP Peering Point RTT (see above), this is a useful, albeit not always accurate, metric for judging ISP performance.
User Response Time	User Response Time is the average time it takes for an application to complete a TCP turn. The User Response Time = Client Set Up Time + Server Response Time + Payload Transfer Time + Retransmission Delay. When looking at a Response Time Composition Chart (see "Response Time Composition Chart" on page 211), this represents the peak values (if the two RTT metrics are removed from the chart) and is a good barometer of the end user response time.	<p>URT is a critical metric, as it correlates very closely with actual user experience. Its primary use is in the Response Time Composition Chart, and as the basis of alerts on degraded user experience.</p> <p>URT represents the average time from a data request packet to the end of the related turn. For the first turn it includes Connection Setup Time, so for very short connections it may be higher than the real average.</p>