One major application of Netrounds is the assurance of Layer 2 and 3 VPN services for various QoS classes. This exercise is vital for communications service providers (CSPs), but it is also a challenging and resource-demanding process.

Netrounds offers an approach to L2/L3 VPN service assurance which helps to overcome many of the limitations of traditional hardware-based tools. For service activation testing and troubleshooting, these limitations include the need for truck rolls and shipping of HW test devices, manual and resource-intensive processes, and costly equipment. For ongoing assurance, there is a need to improve the KPIs for specific customer connections beyond time delayed five-minute counters in order to better understand customers’ perception of the service.

The following features of Netrounds allow CSPs to perform VPN service assurance in an agile and cost-efficient manner: it is fully software-based, it is comprehensive and easy to use, and it has support for API-driven automation.

Netrounds in the OSS landscape

A service activation test will be initiated and completed through the Netrounds Control Center API by a service orchestrator or other component handling service fulfillment.
During active monitoring, real-time performance KPIs are provided to Service Quality (SQM) or Performance Management (PM) systems for correlation and analysis, while alarms are sent to existing Fault Management (FM) systems.

**Y.1731 and TWAMP testing**

Netrounds can automate Y.1731 and TWAMP tests by sending traffic to network equipment with built-in reflector functionality, which then bounce the traffic back to its point of origin (a Netrounds Test Agent). Reflectors are often built-in in equipment such as CPEs, routers and switches, microwave links, and eNodeBs or radio base stations, and turning the reflector function on in this equipment is a simple operation.

Both of these testing protocols measure round-trip performance in a network. Y.1731 is a Layer 2 protocol, while TWAMP operates at Layer 3. **Data rate**, **packet loss**, and **round-trip delay** KPIs are obtained through both protocols.

Y.1731 and TWAMP tests of large scope, targeting thousands of CPEs, are easily set up and run from Netrounds Control Center and can be **completely automated** using the Netrounds API.

**Deployment examples**

The figure below shows a Netrounds Test Agent placed as test head at the ENNI (External Network to Network Interface) between a service provider and an operator from whom it leases a Layer 2 network. A service activation test (SAT) needs to be conducted, using Y.1731, so that a “birth certificate” can be issued for the service. The same setup can subsequently be used for performance monitoring.
The second figure shows Layer 3 performance monitoring with TWAMP. A Netrounds Test Agent is placed somewhere in the operator’s IP core network, enabling testing of the connection to various CPEs (customer premise equipment).

Test Agents bootable from USB

Truck rolls should be avoided whenever possible due to the time and cost resources they consume. When more advanced service activation testing or troubleshooting is called for, the Netrounds approach can streamline the process in several ways.

One way is to install Netrounds Test Agents as a **bootable image on a USB memory stick**. This makes it possible to boot a regular PC (laptop) from the USB, temporarily transforming that PC into a powerful measurement device. In essence, field testers could carry a Test Agent on their keychain. For examples of practical deployment, see the test setup diagrams above.

Utilizing Netrounds’ USB Test Agents enables a **unified and cohesive** way of working and provides many advantages over field test tools traditionally used. Some of these advantages are laid out in the table below.

<table>
<thead>
<tr>
<th>Netrounds solution</th>
<th>Traditional, hardware-based field test tools</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Cost efficient and flexible licensing</strong> (see also Licensing section) – no hardware cost</td>
<td>Expensive – built on custom hardware</td>
</tr>
<tr>
<td><strong>Swiss army knife</strong> supporting both carrier Ethernet and IP network services</td>
<td>Multiple tools required to test different types of networks and services</td>
</tr>
<tr>
<td><strong>Always available</strong> – plug a USB stick into any laptop to turn it into a professional test tool</td>
<td>Right tools must be on hand at customer site – shipping and multiple tools often needed</td>
</tr>
</tbody>
</table>
VPN Service Lifecycle Assurance

SOLUTION BRIEF

© 2016 Netrounds. All rights reserved.

<table>
<thead>
<tr>
<th>Netrounds solution</th>
<th>Traditional, hardware-based field test tools</th>
</tr>
</thead>
<tbody>
<tr>
<td>The entire Test Agent fleet can be <strong>managed from a central location</strong>, optionally using a mobile phone or tablet, via a Web GUI which adapts to screen size</td>
<td>Controlled through local GUI only</td>
</tr>
<tr>
<td><strong>Test automation and centralized cloud storage</strong> ensure that all tests are consistent and that no results are lost</td>
<td>Test results are stored locally in each tool and must be retrieved manually</td>
</tr>
<tr>
<td>All troubleshooting activities at customer sites are stored and <strong>available</strong> to central customer support staff <strong>for later reference</strong></td>
<td>No history of test activities stored; no easy way to enable changes in processes or drive continuous improvements</td>
</tr>
</tbody>
</table>

**Licensing of USB Test Agents**

USB Test Agents are offered according to a flexible license model where the customer purchases:

1. **Access to an inventory** of Test Agents (small amount charged per unit).
2. **Concurrent use** of a subset of Test Agents in that inventory (full Software Test Agent price charged per unit).

This model allows Test Agent deployment on a large scale while keeping costs limited to the scope of actual product usage.

**Layer 2 and Layer 3 service KPIs – output from Netrounds Y.1731 and TWAMP tests**

**Layer 2: Y.1731 ETH-LB (Ethernet Loopback)**
- Rate (Mbit/s)
- Number of frames sent and received
- Two-way loss (frames)
- Packet loss (%)
- Packet misorderings (no. of)
- Two-way delay: min, average, max, percentiles\(^*\) (ms)
- Two-way delay variation (ms)
- Errored seconds (%): total, due to loss, due to delay, due to delay variation
- Severely errored seconds (%)
- Unavailable seconds (%)

**Layer 2: Y.1731 ETH-DM (Dual-ended Delay Meas.)**
- Rate (Mbit/s)
- Number of frames sent and received
- Delay: min, average, max, percentiles\(^*\) (ms)
- Delay variation (ms)
- Errored seconds (%): total, due to delay/delay var.
- Severely errored seconds (%)
- Unavailable seconds (%)

**Layer 3: TWAMP Full, TWAMP Light**
- Rate (Mbit/s)
- Round-trip delay: min, avg, max, percentile\(^*\) (ms)
- Round-trip delay variation (ms)
- Packet loss (%)
- Delay: min, average, max, configurable percentiles\(^*\) (ms)
- Delay variation (ms)
- Errored seconds (%): total, due to delay, due to delay var.
- Severely errored seconds
- Unavailable seconds
Examples of Test Agent-to-Test Agent activation tests

<table>
<thead>
<tr>
<th>L2 control protocol handling</th>
<th>DSCP remapping</th>
</tr>
</thead>
<tbody>
<tr>
<td>RFC 6349</td>
<td>VLAN CoS ID preservation</td>
</tr>
<tr>
<td>QoS policy profiling</td>
<td>Unicast/Multicast/Broadcast frame delivery</td>
</tr>
<tr>
<td>Y.1564 service performance</td>
<td>Y.1564 bandwidth profile (CIR, EIR)</td>
</tr>
</tbody>
</table>

The exciting future: Running vTAs on vCPEs

This solution brief has focused on the use of reflector-based tests and USB-booted Test Agents. However, it is also interesting to consider a related scenario which will become increasingly common and important, namely one where hypervisors are deployed further out in the network on so called "distributed virtual CPEs" (vCPEs) or Mobile Edge Compute (MEC).

Netrounds Virtual Test Agents (vTAs) can then be automatically deployed on those vCPEs, and the full test functionality can be utilized in order to validate services (service chains) end-to-end. This kind of setup is illustrated in the diagram below.