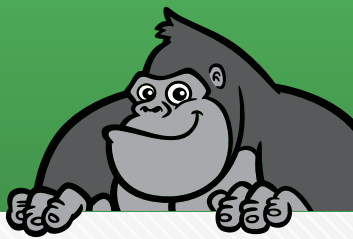
ActualTech Media

riverbed

The Gorilla Guide to...®

Express Edition



Packet Capture and Analysis in Hybrid Environments

Lawrence Miller

Inside the Guide

- Discover why packet capture and analysis of on-premises and cloud applications in modern hybrid environments is so challenging
- Learn how to analyze packets at the Transport and Application Layers to uncover the root cause of issues
- Explore how putting the right tools in the hands of the right people can help make your entire IT team more effective

THE GORILLA GUIDE TO...

Packet Capture and Analysis in Hybrid Environments

Express Edition

By Lawrence Miller

ACTUALTECH MEDIA

6650 Rivers Ave Ste 105 #22489
North Charleston, SC 29406-4829
www.actualtechmedia.com

PUBLISHER'S ACKNOWLEDGEMENTS

EDITOR
Keith Ward, ActualTech Media

PROJECT MANAGER
Wendy Hernandez, ActualTech Media

EXECUTIVE EDITOR
James Green, ActualTech Media

LAYOUT AND DESIGN
Olivia Thomson, ActualTech Media

WITH SPECIAL CONTRIBUTIONS FROM
Paul R. Dietz, Riverbed
Stephen Creel, Riverbed
Heidi Gabrielson, Riverbed
Kowshik Bhat, Riverbed

TABLE OF CONTENTS

Introduction 7

Chapter 1: A Primer on Packets 8

Looking at Packets and Packet-Switched Networks 8

Understanding the Open Systems Interconnection (OSI) Model and Packet Encapsulation 11

Addressing Modern Packet Capture and Analysis Challenges 13

Chapter 2: A Packet's Tale: The Case of the Slow Web App 16

The Game Is Afoot 16

Examining Packets at the Transport Layer 17

Analyzing Packets at the Application Layer 24

Putting It All Together: Time to Find Out Who the Real Villain Is 33

Chapter 3: Force Multipliers: Making Your Smart People More Effective 35

World-Class Packet Analysis 35

Beyond Capturing Packets 40

Performing Magic with AppResponse 40

Delivering Better Business Outcomes 43

Get More Production and Less Frustration 45

CALLOUTS USED IN THIS BOOK

In the School House callout, you'll gain insight into topics that may be outside the main subject but are still important.

This is a special place where you can learn a bit more about ancillary topics presented in the book.

When we have a great thought, we express them through a series of grunts in the Bright Idea section.

Takes you into the deep, dark depths of a particular topic.

Discusses items of strategic interest to business leaders.

ICONS USED IN THIS BOOK

DEFINITION

Defines a word, phrase, or concept.

KNOWLEDGE CHECK

Tests your knowledge of what you've read.

PAY ATTENTION

We want to make sure you see this!

GPS

We'll help you navigate your knowledge to the right place.

WATCH OUT!

Make sure you read this so you don't make a critical error!

TIP

A helpful piece of advice based on what you've read.

INTRODUCTION

Today's complex application architectures span hybrid environments consisting of multiple public and private clouds, on-premises data centers hosting hundreds (and even thousands) of microservices, containers, and virtual machines—all connected across disparate wide-area network (WAN) and internet links. Identifying and troubleshooting performance issues without complete visibility across the entire architecture and using siloed tools and data causes longer resolution times, breached service-level agreements, and miscommunication and finger pointing across different IT teams. Ultimately, this untenable situation leads to lost user productivity and frustration that negatively impacts the business and the bottom line.

The Gorilla Guide To...® (Express Edition) Packet Capture and Analysis will help you understand why identifying and resolving application and network performance issues has become so challenging for even the best and brightest of your IT teams. You'll learn how to take a more evidence-based approach to diagnosing and analyzing performance issues across the entire stack and how the right packet capture and analysis tools can help your IT teams work together more effectively to resolve performance issues faster.

CHAPTER 1

A Primer on Packets

Software may be "eating the world," but it's the network that connects software applications to services, data, and users. In this chapter, we'll look at the challenges of packet capture and analysis in a hybrid world consisting of on-premises data centers, public/private clouds, and remote and mobile users. Later chapters will take us through a sample exercise in troubleshooting a network, and describe some standard tools in packet and transaction analysis through services offered by Riverbed.

Looking at Packets and Packet-Switched Networks

Before the advent of packet-switched networks, circuit-switched networks provided connectivity, literally, from point A to point B. The principle of a circuit-switched network was captured in Mayberry's switchboard operator Sarah on *The Andy Griffith Show*, manually connecting calls, say, between Sheriff Andy Taylor and Floyd's barbershop. Each phone line (circuit)

connected exactly two endpoints (phones). To connect to a different endpoint (for example, if Andy wanted to call Aunt Bee), Sarah would need to break Andy's connection to Floyd and switch it to Aunt Bee. Although this is a greatly simplified example of circuit-switched networks, it illustrates the basic concept and some of the challenges associated with circuit-switched networks, including high costs (to maintain multiple point-to-point connections) and limited resiliency (each circuit is a single point of failure).

The first packet-switched network, ARPANET, was originally developed for the U.S. Department of Defense in 1969, specifically to address the issue of resiliency in circuit-switched networks. In a packet-switched network, endpoints (such as servers and client workstations) can communicate over numerous communications links, typically forming a mesh of available links. This provides numerous ways to connect point A to point B. Depending on your business and technical requirements, different links may be dynamically selected for individual transmissions based on factors such as:

- Speed
- Latency
- Cost

- Security
- Utilization
- Availability

Better still, in a packet-switched network the traffic is segmented into packets (consisting of a header and payload) that don't necessarily have to take the same network path from point A to point B. The header information in the individual packets allows the endpoints to assemble the packets in the correct order when they're received. An internet Protocol (IP) header contains important information used in packet capture and analysis, including:

- Version
- IP Header Length
- Type of Service
- Total Length
- 16-bit Identification
- Flags
- Fragment Offset
- Time to Live
- Protocol

- Header Checksum
- Source and Destination IP Address
- Options (if any)

Understanding the Open Systems Interconnection (OSI) Model and Packet Encapsulation

The OSI model helps network engineers understand and troubleshoot complex networking issues by separating the network into simpler functional components. This model consists of seven layers that describe how systems and applications communicate and interoperate on a packet-switched network (see **Figure 1**).

Data packets in the OSI model are broadly referred to as *protocol data units* (PDUs). The OSI model promotes interoperability by only requiring a layer to be able to communicate with the layer directly above and below it. This is accomplished through a process known as encapsulation. Each layer adds a header (and a footer, too, just in the case of Layer 2) with delivery instructions and information about the data, which is called the *payload*. A PDU at the Physical Layer (Layer 1) is known as a *bit* or *symbol*. When the Layer 1 PDU is passed up to the Datalink Layer (Layer 2), the Datalink Layer treats the entire PDU (header and payload) as its own payload.

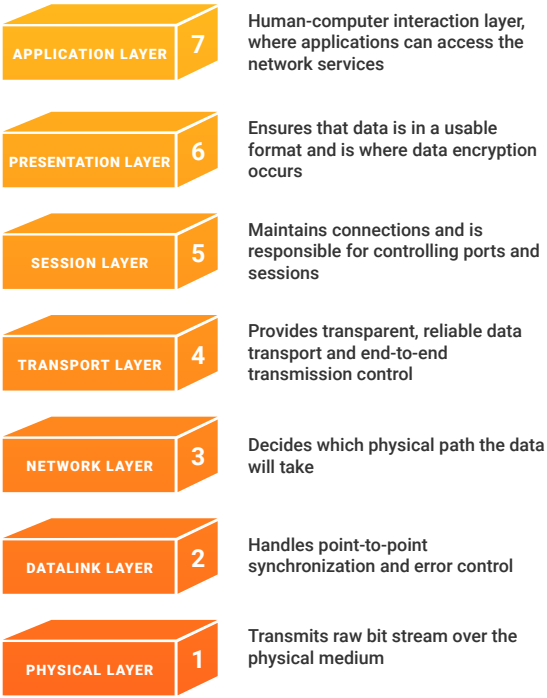


Figure 1: The OSI model

Layer 2 adds its own header and layer, thus creating a Datalink Layer PDU, known as a *frame*. This process is repeated at the Network Layer (Layer 3), which adds a header to create the Network Layer PDU, known as a *packet*. The encapsulation process continues at the

Transport Layer (Layer 4), where the PDU is known as a *segment* or *datagram*, all the way to the Application Layer (Layer 7). In the upper layers of the OSI model (Layers 5 through 7), PDUs are simply known as PDUs. On the receiving end, the individual headers are stripped off, one by one, at each layer (effectively reversing the process) as a PDU passes down the application stack.

In this Guide, we primarily focus on the following layers:

- Layer 3: The internet Protocol (IP)
- Layer 4: Mostly TCP and UDP
- Layer 7: Applications, especially Web apps

Addressing Modern Packet Capture and Analysis Challenges

To gain visibility into the raw data (packets) traversing enterprises' systems and networks, network engineers, infrastructure and operations (I&O) teams, and application owners have traditionally used network performance management (NPM) and application performance management (APM) tools. The analyses these tools provide can help optimize performance and troubleshoot issues. However, these enterprise IT teams must now address challenges associated with a whole host of new architectures and technologies that are being adopted within the enterprise.

According to the "Flexera 2020 State of the Cloud Report," 98% of organizations now use at least one public or private cloud, and 93% have adopted a multi-cloud strategy. These environments call for packet capture and analysis tools that enable a unified view across a hybrid world consisting of:

- On-premises data centers
- Public and private clouds
- Hosted Software-as-a-Service (SaaS) applications
- Modern cloud-native applications comprised of hundreds (if not thousands) of ephemeral container-based microservices deployed across multiple clouds
- Users spread across headquarters, branch, remote, and home office locations
- Mobile platforms

Deploying packet analysis across this varied landscape is daunting, to say the least. Challenges for your various IT teams include:

- **Where and how do you acquire packets?** In today's hybrid environment, where packets may traverse multiple on-premises data centers and public clouds, you can't simply deploy a physical appliance to capture packets.

- **How big are the captures you need to work with?** When you download large packet files, are they time consuming to analyze? Does the download introduce network issues of its own?

- **Can you transport packets to your workstation securely and efficiently?** Network performance problems can sometimes be an indicator of malicious activity on the network. Protecting the confidentiality and integrity of your raw packet data in motion is critical.

- **How do you see and understand the interaction and behavior of applications and the network?** If different application and network teams use complex, disparate tools that are not integrated and do not provide a single source of truth to analyze packet data, they receive an incomplete—or even conflicting—view of the environment and have trouble reaching conclusions.

- **How do you communicate the results to the business unit/stakeholders?** TCP acknowledgements? Connection resets? Cache misses? To the business unit and stakeholders, you're speaking a foreign language. They just want to know what it's going to take to fix the issue.

These are the challenges addressed in the rest of this Guide.

CHAPTER 2

A Packet's Tale: The Case of the Slow Web App

With all the complexity and challenges in today's hybrid IT environments, troubleshooting an application or network issue can feel a bit like solving a diabolical mystery. However, with an understanding of packets, packet-switched networks, and the OSI model—all of which you gained in Chapter 1—and the right packet capture and analysis tools, you can master the skill of application and network performance analysis. In this chapter, you'll spend a day in the life of your application owners and network engineers.

The Game Is Afoot

Of all the phones, in all the offices, in all the world, she called mine. I hadn't even had a chance to finish my first cup of coffee before an urgent trouble ticket got escalated to me. It was the help desk technician on the other end of the line. She said the application development team had released a critical new business application over the weekend and users were complaining

that is was loading slowly. Really slowly. Users in several remote offices are complaining that it's taking 2 to 3 minutes for the application to load on their computers. She continues, "in my opinion the users are just being impatient, and they should be more understanding, and ...." I politely interrupt, "Just the facts, ma'am. Just the facts."

My first call was to the application development team. They said they tested it before they released it and it loaded in just 2 seconds—well under the 8-second service-level agreement (SLA) requirement. So, it "must be a network issue." Time to get to work.

Examining Packets at the Transport Layer

To solve this mystery, I have several tools at my disposal (see **Figure 2**). For this particular case, I'll use AppResponse, Packet Analyzer Plus, and Transaction Analyzer. Each solution provides more granular detail than the one before, allowing me to inspect the clues further until I have the information I need.

You can think of AppResponse as digital binoculars that are capturing detailed information 24/7/365. With AppResponse, I can survey the terrain and seek out problem areas, or I can choose to let AppResponse alert

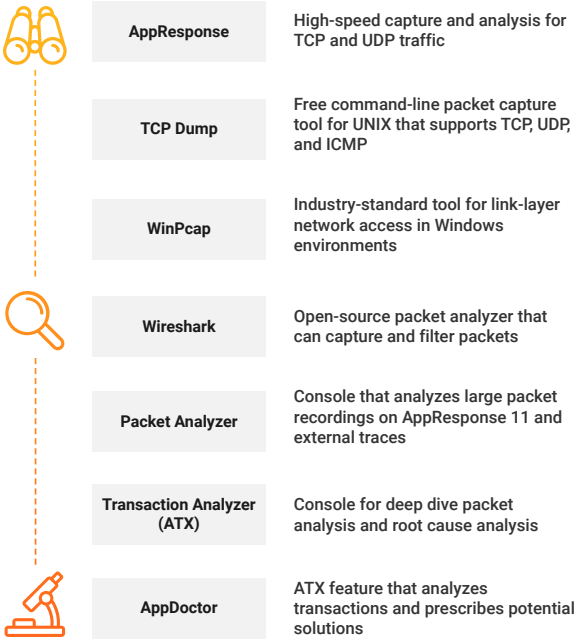


Figure 2: The full suite of Riverbed packet capture and analysis tools

me to potential issues. Continuing with this analogy, Packet Analyzer Plus would be a very powerful magnifying glass allowing for much closer inspection of potential clues. Finally, when the situation calls for the use of a microscope (to inspect DNA evidence, if you will),

I would use Transaction Analyzer. Each tool addresses a specific need throughout my investigation. Although Transaction Analyzer gives me the most granularity, it is not practical to start looking for a fire by inspecting an entire forest with a microscope when I can easily see the smoke with my binoculars, then begin to narrow my focus to a specific area of the forest.

Since the users are complaining about a web application, I will use the AppResponse Web Transaction Analysis module. Leveraging the **Page Views HD** screen and looking at the **User IPs** tab for the IP address in question, I can already see pages that are taking more than 24 seconds to load (see **Figure 3**)!

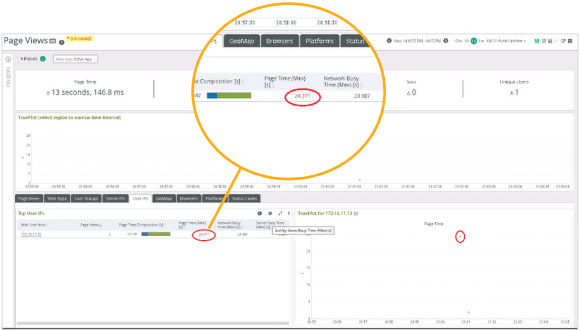


Figure 3: The Page Views HD screen in AppResponse

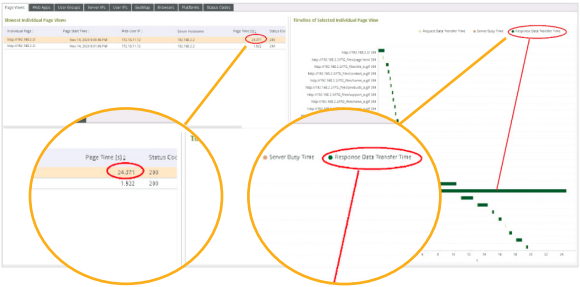


Figure 4: The Page View tab in AppResponse

Moving to the **Page Views** tab (see **Figure 4**), I can again see that it is taking 24 seconds for the page to load, but I can also now see which components (Request Data Transfer Time, Server Busy Time, Response Data Transfer Time) are causing the long delays. Here, I can clearly see the issue is Response Data Transfer Time and I can see how long it's taking to transfer each object that makes up the web page.

I've now identified the main cause of slowness impacting my end users, but let's take a closer look with the other tools in my tool bag to further validate my hypothesis.

With my trusty Riverbed Packet Analyzer Plus (you'll learn more about Packet Analyzer Plus in Chapter 3), I got started. First, I took a look at the packet trace for the new application using the Wireshark tool that is

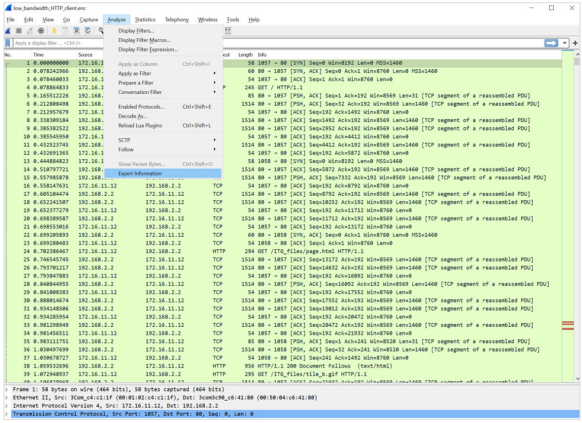


Figure 5: A Wireshark packet trace in Packet Analyzer Plus

integrated with Packet Analyzer Plus (see **Figure 5**). It is a capital mistake to theorize before one has packet data. Our network is set up to continuously capture packets with Riverbed AppResponse, so we never miss a clue. The Wireshark trace shows communications between the client and web server. Other than a high number of Transmission Control Protocol (TCP) acknowledgements (ACKs), there doesn't appear to be a smoking gun here. I left-click **Analyze** in the menu bar and select **Expert Information** in the dropdown menu.

Riverbed and Wireshark

Gerald Combs is an innovator in the field of packet capture and analysis. He created Wireshark (originally known as Ethereal) in 1997. In 2010, he joined Riverbed and Riverbed is now the official corporate sponsor of Wireshark.

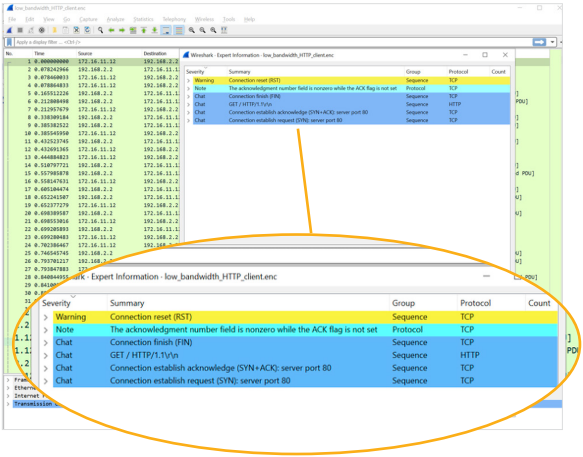


Figure 6: The Expert Information window displays details about any Warnings, Notes, Errors, and other relevant communications in the packet trace

In the **Expert Information** window, I look for any errors, warnings, or crash landings (see **Figure 6**) in the packet trace. I notice a **Warning** and **Note**, so I expand both lines to see the details. Other than some connection resets, there isn't much in the warning. Notes generally shouldn't affect performance, and this particular note simply confirms that there was some ACK traffic associated with the connection resets.

Next, I follow the TCP stream (see **Figure 7**). In the **Analyze** menu, I select **Follow and** then **TCP Stream**. The TCP stream shows that a webpage is being created, but nothing troubling stands out.

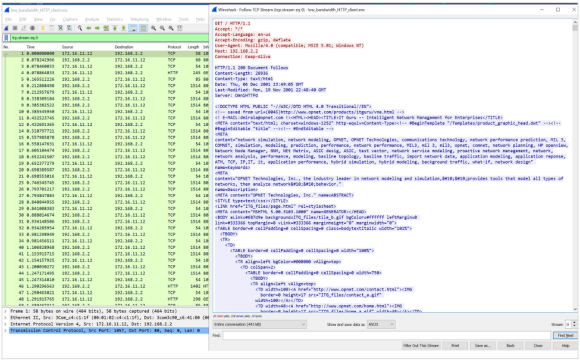


Figure 7: Nothing troubling stands out in the Hypertext Markup Language (HTML) in the TCP Stream

Analyzing Packets at the Application Layer

I switch from the WireShark tool to Riverbed Packet Analyzer Plus and select a preconfigured view in the bottom-left pane: **Service Response Time by Web Object – Advanced** (see **Figure 8**). To run the pre-configured view, I drag and drop the view from the bottom-left pane onto the trace in the top-left pane, and my analysis is ready in just a few seconds. The **Resource** column in the analysis shows all the parts of the webpage. The **Server Response** column doesn't

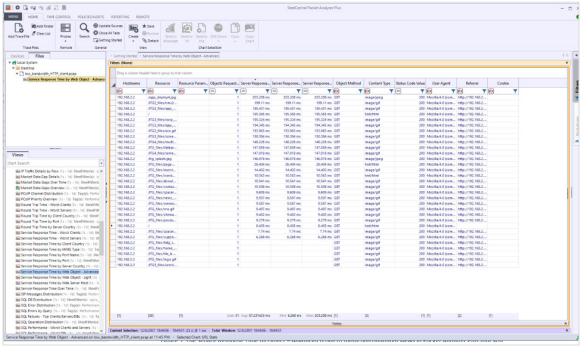


Figure 8: The Service Response Time by Object – Advanced is one of many preconfigured views in Packet Analyzer Plus that you can use to quickly run different queries and analyze the results

show anything particularly egregious. All the webpage parts appear to be loading in milliseconds, and even when I add up all the individual parts, it doesn't come anywhere close to the 2 to 3 minutes that my users are reporting. Elementary math, my dear Watson, elementary math.

Now I take a look from the network standpoint. I select another preconfigured view: **Web Bandwidth by Object – Advanced** (**Figure 9**). This view shows that several large objects are being loaded by the application, totaling nearly 6MB of traffic. The plot thickens.

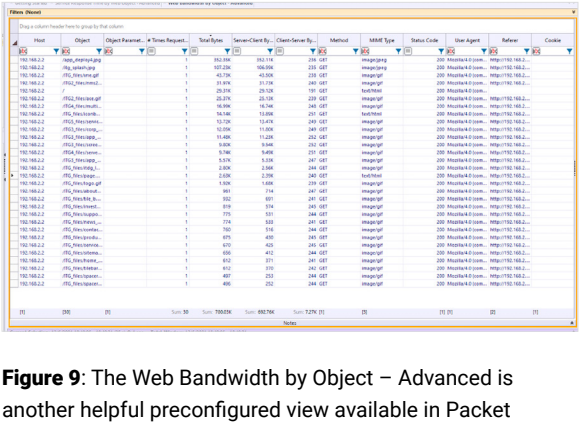


Figure 9: The Web Bandwidth by Object – Advanced is another helpful preconfigured view available in Packet Analyzer Plus

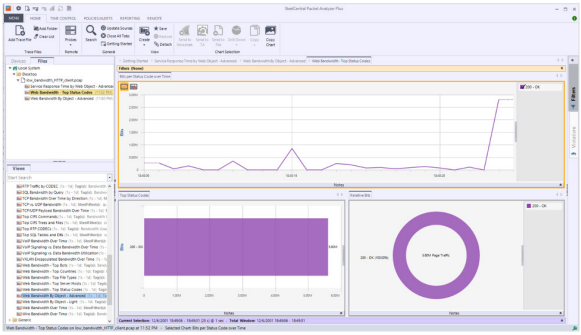


Figure 10: The Web Bandwidth – Top Status Codes view in Packet Analyzer Plus

Next, I load the **Web Bandwidth – Top Status Codes** (see **Figure 10**) preconfigured view and see that there are no issues loading the images.

Next, my analysis shifts to the Application Layer (Layer 7) with a transaction-level analysis in Transaction Analyzer. I right-click the packet trace in the upper-left pane and select **Send to** and **Transaction Analyzer** in the dropdown menu that appears (see **Figure 11**) to send the packet trace to the Transaction Analyzer tool.

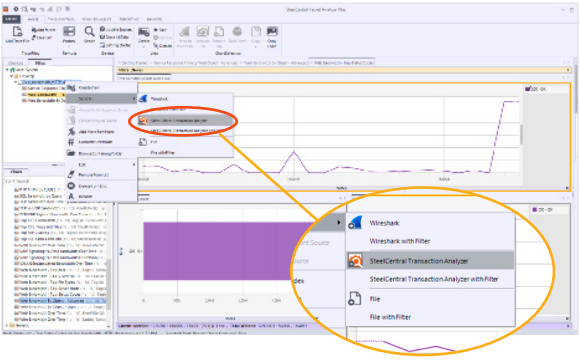


Figure 11: Launching the Transaction Analyzer tool in Packet Analyzer Plus

The Transaction Analyzer tool (see **Figure 12**) shows the total time it took for the application to load—approximately 25 seconds. Jinkies!

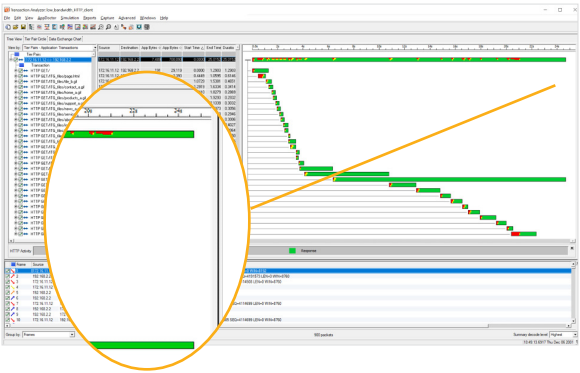


Figure 12: The Transaction Analyzer tool

In **Figure 12**, the Transaction Analyzer tool shows the first 18 web objects loading relatively quickly. However, the next 5 objects—particularly the 21st object—appear to be taking an inordinately long time to load. Hmm, an important clue perhaps. Clicking on the object in the right pane allows me to view additional details, such as window and segment size, for any TCP issues that might stand out.

Now I invoke the AppDoctor by clicking **AppDoctor** in the menu bar of the Transaction Analyzer and selecting **Summary of Delays (AppDoctor Analysis)** in the drop-down menu that appears (see **Figure 13**).

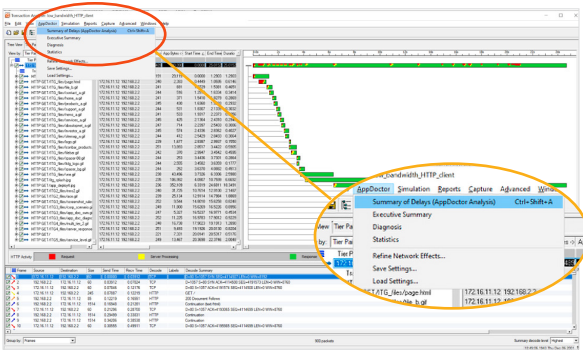


Figure 13: Launching AppDoctor

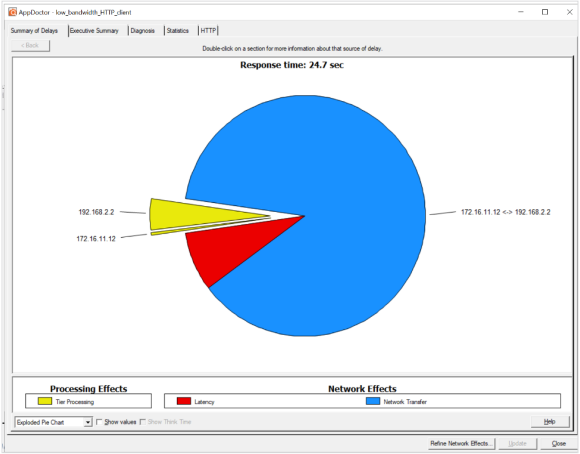


Figure 14: The Summary of Delays tab in AppDoctor

The AppDoctor displays several tabs with helpful information, including the **Summary of Delays.** In **Figure 14**, the blue area of the pie chart represents the network transfer time (approximately 88% of the total 24.7 seconds response time). Network latency (red) accounts for approximately 8%, and web server and client processing (yellow) accounts for approximately 4%.

The information in the **Summary of Delays** tab can be further refined, such as to account for different speeds on individual network links (see **Figure 15**).

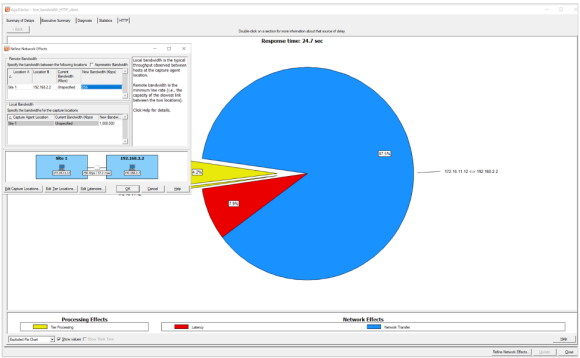


Figure 15: Refining network effects in AppDoctor

I can also display the information in AppDoctor in a tabular view. The green line in **Figure 16** shows the network bandwidth on this link: 256 kilobits per second (Kbps). The red line shows the bandwidth consumed on the server side in one-second intervals, and the blue line shows the bandwidth consumed on the client side. Thus, the application is maxing out the available bandwidth on the server side throughout most of the transaction. Aha! Another clue.

Further refining the view with a one-millisecond interval (see **Figure 17**) shows that the application is actually exceeding the available bandwidth. The application is sending more data than the link can handle which, in turn, causes lots of connection resets and further exacerbates the issue. Very interesting.

Figure 16: Tabular view in AppDoctor (one-second intervals)

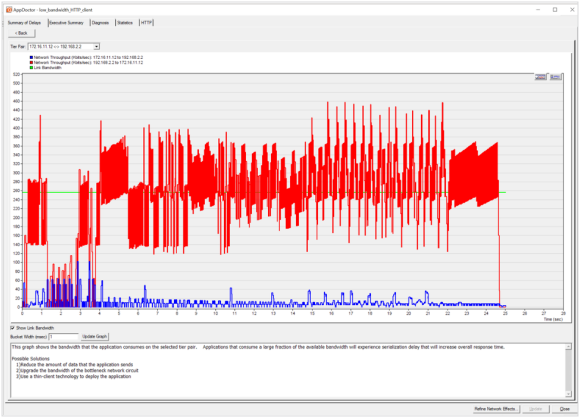


Figure 17: Tabular view in AppDoctor (one-millisecond intervals)

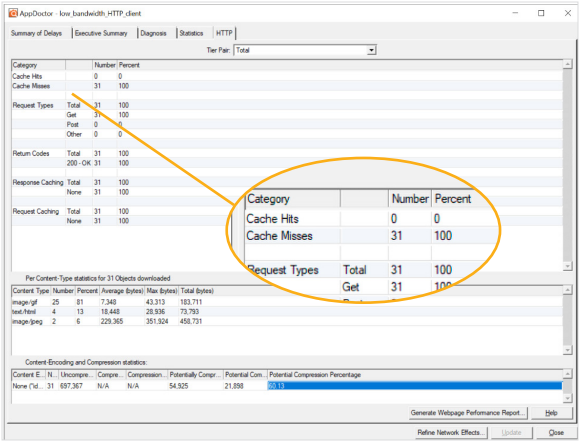


Figure 18: The HTTP tab in AppDoctor shows no cache hits for the web application objects

Finally, on the **HTTP** tab in AppDoctor (see **Figure 18**), I see that the web application had no cache hits. That is, none of the 31 objects were locally cached. This means either that the client launched the app for the first time and therefore had nothing cached, or that the application developers did not set cache persistence for the application. In the bottom pane of the tab, I also see that 60% of the 31 objects, which are primarily images, could have been compressed but were not.

Putting It All Together: Time to Find Out Who the Real Villain Is

Now that I've performed my packet capture and analysis on both the network and application sides, it's time to unmask this performance villain: Mister Willoughby?!

Actually, on the network side, I was able to determine that the application is saturating the 256 Kbps network links connecting several branch locations to the application in our on-premises data center. Further inquiries reveal that the quality assurance testing (QAT) on the application was performed in a virtual lab environment in the data center on one gigabit per second (Gbps) links, which did not accurately reflect the various WAN links to your branch locations. Although increasing our bandwidth to a full T-1 (1.544 Mbps) would provide the necessary performance, this isn't always feasible and doesn't really address the root issue causing the application's initial page to load slowly. Instead, we might consider enabling WAN optimization on the individual links to locally cache and compress as much traffic as possible.

On the application side, our developers can take several steps to optimize the application, including:

- Making use of browser caching

- Parallelizing downloads across hostnames
- Optimizing images
- Enabling compression
- Avoiding a character set in the meta tag
- Minifying HTML
- Specifying a character set

Transaction Analyzer provides a detailed report describing how to implement each recommendation and the impact on performance for each recommendation. And with that, our work here is done.

CHAPTER 3

Force Multipliers: Making Your Smart People More Effective

When businesses push their employees to "work smarter not harder" and "do more with less," they need to ensure they're providing the right tools for their people to succeed. Using the right tools for the task at hand can be a true force multiplier—it's far more effective to drive a nail with a hammer than your fist! Of course, you need something other than a hammer (though it's perhaps tempting at times) to resolve network and application issues. In this chapter, we'll help you identify the right tools to help you make your smart people more effective and deliver better business outcomes.

World-Class Packet Analysis

Packet analysis begins with capturing (or "sniffing") packets on a network. Riverbed Packet Analyzer Plus is a network packet sniffer that rapidly isolates the specific packets needed to diagnose and troubleshoot complex

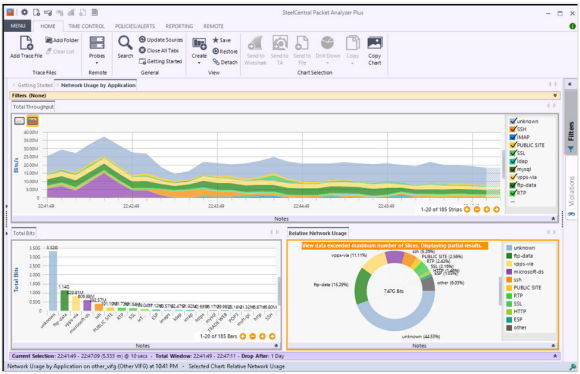


Figure 19: Preconfigured "views" in Packet Analyzer Plus allow quick analysis of local trace files or packets

performance issues. Packet data can be quickly analyzed where it is locally stored, rather than waiting to transfer large packet files across the network and potentially causing other network congestion issues. Packet Analyzer Plus has an intuitive graphical user interface that streamlines packet analysis. The tool offers preconfigured "views" (see **Figure 19**) that you simply drag and drop onto a group of virtual interfaces or a packet trace file to see results immediately.

Riverbed Transaction Analyzer provides powerful, detailed analysis of a single user-level transaction through powerful visualization and advanced analytical

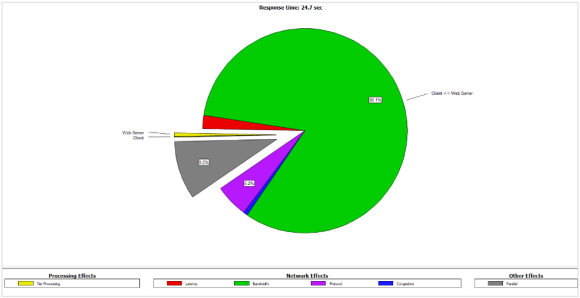


Figure 20: Transaction Analyzer's AppDoctor feature automatically diagnoses performance bottlenecks

techniques, to help you quickly and conclusively determine the root cause of a performance problem.

The analyzer can collate packet traces between multiple tiers of servers involved in individual user transactions, to build a single view of application and network message exchanges across multiple tiers. Then you can visualize how each pair of communicating systems contributed to the overall behavior and performance of that transaction, and easily communicate key findings to application stakeholders. This multi-tier view lets you picture application performance, including application message exchanges, turns, and system performance statistics, to accurately identify the primary sources of delay in the transaction (see **Figure 20**).

Customer Success Story: OneMain Financial

OneMain Financial has been a trusted provider of personal loans for more than 100 years. Offering fixed rates and fixed payments, the company helps its customers take care of the range of expenses that life can bring, from debt consolidation to vacations to medical costs. OneMain's 10,000 team members provide personalized service in nearly 1,600 locations in 44 states.

Challenges

- Attract customers by quickly approving loans
- Determine whether slow performance is caused by the network or the application
- Remediate performance issues quickly—ideally before users notice

Solution

Riverbed AppResponse provides fast packet capture and storage that feeds intelligent network and application analysis with fast troubleshooting workflows to speed problem diagnosis and resolution. It includes a license of Packet Analyzer Plus, for fast, easy packet analysis.

Benefits

- Deploy anywhere and everywhere you need—on-premises, private or public AWS or Azure cloud—AppResponse is designed to meet your hybrid monitoring needs with network forensics, application analytics, and end-user experience monitoring in a single solution.
- Modular in design, AppResponse lets you select the analysis capabilities you need, including network forensics, all TCP and UDP applications and their metrics, web application performance, database analysis, VoIP and video analysis, and Citrix analysis.
- Captures and stores all packets, all the time at one-minute granularity, so the details are always available when you need them. When required, explore the second- and micro-second level details.
- Built-in policies and adaptive thresholds automatically highlight brewing problems as they are occurring, so you can address them before they become full-blown incidents.

"The combination of AppResponse and Packet Analyzer Plus makes it easy to find the precise set of packets I need to see if a performance problem starts with the network or the application," says Richard Hurst, supervisor of Network Services, OneMain Financial.

Read the complete case study.

Beyond Capturing Packets

Raw packet capture and analysis is critical for troubleshooting day-to-day performance issues, but this data has significant historical value, as well. Historical packet data can be used to establish performance baselines and compare snapshots of application and network performance over time. This data can also be used by security teams for forensic analysis.

Locally storing packet captures, which can be several terabytes in total size, securely and efficiently requires physical and/or virtual devices that can be deployed where the data is collected—whether on-premises (for example, in a data center or branch location) or in a public cloud.

Performing Magic with AppResponse

Riverbed AppResponse delivers full stack application analysis—from packets to pages to end-user experience—letting you observe all network and application interactions as they cross the wire, whether they are encrypted or not (see **Figure 21**). Using powerful, flexible network and application analytics and workflows, AppResponse speeds problem diagnosis and resolution, helping you get answers fast.

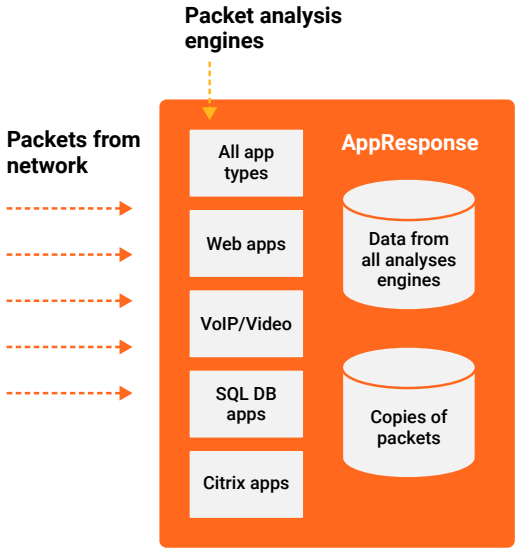


Figure 21: How AppResponse works

AppResponse offers a variety of optional modules that provide specialized analysis, including:

- **Application Stream Analysis (ASA)—**The ASA module provides real-time and historical network-based application analysis. It monitors all TCP and UDP-based applications and other Layer 4 protocol-based metrics. ASA also provides rich response time composition metrics so you can quickly determine where

to focus your troubleshooting efforts—the network, application, or client.

- **Web Transaction Analysis (WTA)—**The WTA module offers real-time web application performance analysis for monitoring business transactions. It automatically discovers all URLs accessed and all end-user activity to simplify monitoring. You can view end-user experience for webpages and detect page errors, page rates, unique users, and more. Geographic heat maps make it easy to focus triage efforts on critically affected users and sites.

- **Citrix Analysis (CXA)—**CXA correlates front-end user sessions to back-end transactions to help you understand where and why problems are occurring in Citrix Virtual App and Desktop (formerly XenApp and XenDesktop). CXA enables you to follow individual user transactions through the Citrix tier to troubleshoot performance issues and investigate utilization, latency, and Citrix Independent Computing Architecture (ICA) priorities.

- **Database Analysis (DBA)—**The DBA module identifies the impact of the database on end-to-end application performance. By monitoring database performance at the transaction level, you can identify the particular SQL statement or database call responsible

for application delay and equip your database team with actionable information. The module's agentless approach introduces zero overhead on database operation and does not require privileged access to database systems or database diagnostics logging.

- **Unified Communications Analysis (UCA)—**The UCA module provides real-time and historical analysis of voice and video performance calls. Drill down to the underlying problem to understand the interaction of voice and data traffic and proactively monitor voice call quality, allowing you to resolve issues before they affect users.

Delivering Better Business Outcomes

In today's complex application architectures, a single transaction can involve many server tiers and containers, in multiple data center and cloud environments, exchanging thousands of messages between them. Finding the sources of delay for an individual transaction in these hybrid environments can be like finding a needle in a haystack. As a result, IT operations staff spend significant time analyzing packet traces and performance metrics. They often arrive at different and conflicting conclusions about the cause of performance problems, which further delays problem resolution.

Deploying the right tools to enable unified packet capture with AI-based analytics leads to better business outcomes by enabling:

- **Ubiquitous visibility—**You can deploy the tools anywhere and everywhere you need on-premises, virtual, or cloud visibility.
- **On-demand scalability**—Large data sets need intelligent tools that can handle big data without downloading the data to a central collection point and bogging down network responsiveness.
- **Actionable insight—**Packets are the ultimate source of network truth. AppResponse captures and stores all packets all the time. Using Packet Analyzer Plus and Transaction Analyzer, your application and network teams can get detailed analysis and details of your packet captures when you need them.
- **Fast answers—**Built-in policies and adaptive thresholds highlight emerging issues on the network, allowing you to get ahead of them before they become full-blown incidents. Streamlined troubleshooting workflows and graphical data help you get answers quickly—typically in minutes.

Get More Production and Less Frustration

Throughout this Gorilla Guide, you've learned how much things have changed—hybrid cloud environments and distributed microservices architectures certainly don't make troubleshooting performance issues any easier—and how much things have stayed the same—packet-based networks still rule the day and the fundamentals of troubleshooting are still, well, fundamental.

The challenges lie in ensuring complete, end-to-end visibility across the entire application architecture, collecting and analyzing terabytes of packet information in near real-time without negatively impacting network performance due to massive downloads, and enabling effective collaboration across IT teams with a fully integrated suite of tools that enables everyone to "speak the same language."

Riverbed AppResponse, Packet Analyzer Plus, and Transaction Analyzer provide network and application performance solutions with advanced features and capabilities that can be a real force multiplier, helping to make your smart people more effective, your networks and applications faster and more efficient, and your

users more productive—and less frustrated (after all, no one ever complains about their network or apps running "too fast!").

To learn more about Riverbed AppResponse, Packet Analyzer Plus, and Transaction Analyzer, visit https://riverbed.com/npm.

ABOUT RIVERBED

riverbed

Riverbed enables organizations to maximize performance and visibility for networks and applications, so they can overcome complexity and fully capitalize on their digital and cloud investments. The Riverbed Network and Application Performance Platform enables organizations to visualize, optimize, remediate and accelerate the performance of any network for any application. The platform addresses performance and visibility holistically with best-in-class WAN optimization, network performance management (NPM), application acceleration (including Office 365, SaaS, client and cloud acceleration), and enterprise-grade SD-WAN. Riverbed's 30,000+ customers include 99% of the Fortune 100. Learn more at riverbed.com.

ABOUT ACTUALTECH MEDIA

ActualTech Media is a B2B tech marketing company that connects enterprise IT vendors with IT buyers through innovative lead generation programs and compelling custom content services.

ActualTech Media's team speaks to the enterprise IT audience because we've been the enterprise IT audience.

Our leadership team is stacked with former CIOs, IT managers, architects, subject matter experts and marketing professionals that help our clients spend less time explaining what their technology does and more time creating strategies that drive results.

For more information, visit www.actualtechmedia.com