Application Note

Isolating Network Problems with the cCloud™ Visibility Suite

How to Identify and Isolate Network Problems in the Cloud to Reduce Service Outages

www.cpacket.com

Highlights

- Learn how to acquire packet data in Cloud environments
- Learn about network Key Performance Indicators (KPIs)
- Learn how to isolate network problems in Hybrid-Cloud infrastructures

Isolating Network Problems

This Application Note is intended for network, security, application, and operational teams to understand the process of efficiently isolating network problems in today's cloud environments, including hybrid infrastructures. This is important because when issues arise, IT team members often spend time trying to isolate the problem to know who is best suited to troubleshoot and resolve it. Isolating network issues with various teams and exonerating the network infrastructure is always very frustrating!

Network Packet data and Key Performance Indicators (KPIs) provide the foundational network data and insights that facilitate root cause analysis (RCA) that can be challenging depending on which operational team owns the infrastructure layer under scrutiny. This can lead to time delays with stressful finger-pointing between infrastructure owners and Cloud Service Providers.

This application note will step you through instrumenting public cloud infrastructure for visibility, optimizing RCA and the overall Operational Efficiency (OE) of the IT team.

Network Visibility for Cloud Infrastructure

The NetOps team often takes the lead in isolating a problem to determine whether the network is at fault and communicating that result. The NetOps investigation also often provides clues to the areas of concern and which IT domain owns the problem. Therefore, network visibility is vital to efficiently isolating and solving IT problems.

Network visibility fabric, Network Packet Brokers (NPB) specifically, ensure that packets are replicated and delivered to the correct personnel, dashboards and performance management tools are key to effective troubleshooting. Every troubleshooting effort begins with RCA that relies on network visibility, especially since a network failure is often blamed first.

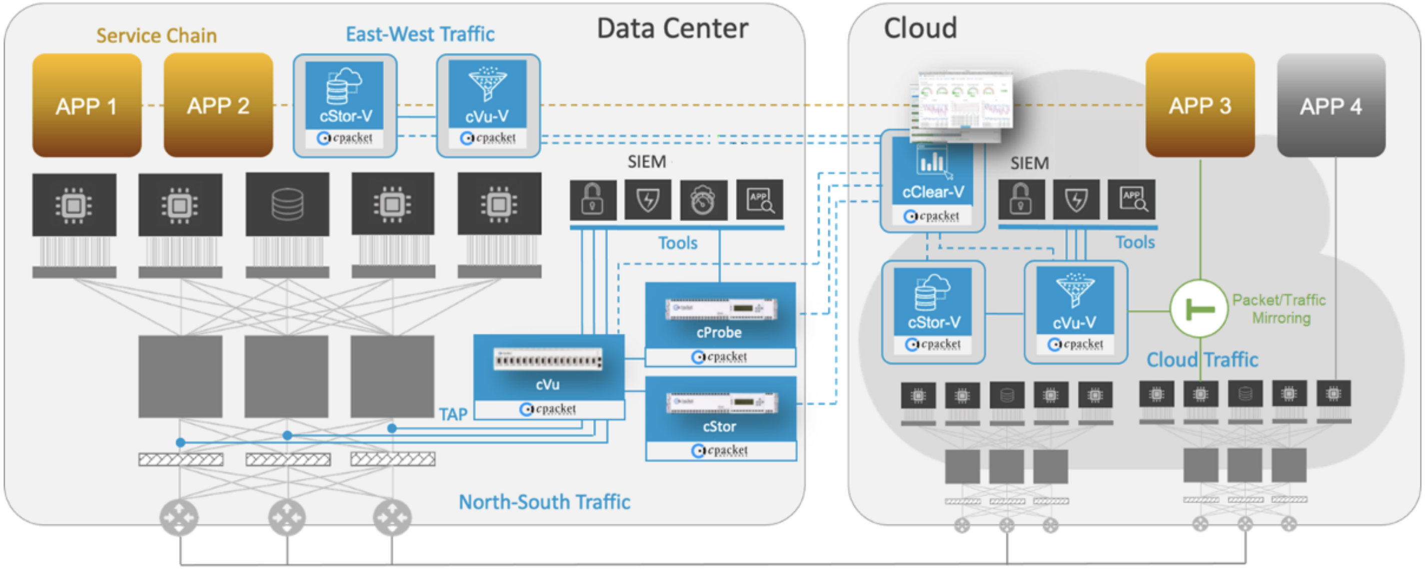


Figure 1: Hybrid-Cloud Observability Architecture

If your IT infrastructure is single-cloud, multi-cloud, or hybrid, you will need to gain visibility from streamed and stored network packet data throughout your entire environment. You'll want your cloud visibility fabric and network packet data to be holistic and uniform for hybrid infrastructure to minimize tool sprawl, manual correlation, and RCA.

cCloud Visibility Suite Appliances

The cPacket Networks cCloud Visibility Suite has several components that perform packet acquisition, replication, filtering, forwarding, storage, and analytics. Altogether it provides vital visibility for cloud infrastructure without placing agents or probes into the production workload host, virtual machine (VM), or Application Layer. Analytics applied to acquired network packets provide KPIs, that along with the raw data, can be exported via API, presented in dashboards, and trigger actionable alerts. Figure 1 below shows an example of hybrid infrastructure, including physical and virtual appliances for network visibility.

At cPacket, we know the importance of the "4Ws" of pinpointing the root cause of problems, especially complex and hard to diagnose problems. The "4Ws" are: What, Where, When, and Why? Organizations deploy instances of the cVu®-V Virtualized Network Packet Broker appliance into the infrastructure to provide lossless monitors to acquire, filter, replicate, and forward packets from native and custom vantage points to multiple targets and tools. Network monitors strategically located in the network infrastructure forward traffic to security, forensics, NDR, performance, and packet capture tools. cPacket cStor® Packet Capture appliance provides network packet storage and archiving for forensic investigation, and the cClear® Analytics Engine appliance provides the KPI visualizations through a single pane of glass.

Network metrics data and telemetry increasingly provide actionable insights into network flow behavior and anomalies. The deluge of data is overwhelming, including noise from noncritical events, logs, and metrics. The shared responsibilities in the cloud and layered network ownership create increasingly challenging incident ownership, troubleshooting, and dealing with the SLA process for today's hybrid operational teams. Incident Response optimization and reducing Mean Time to Resolution (MTTR) become the focus when dealing with multiple support teams and organizations quickly getting to the data! Not only do we deal with the operational ping pong with our server, platform, or DevOps teams, now we must factor in Cloud Service Provider support teams requesting PCAP data and analysis increasing time to RCA.

Key Performance Indicators (KPIs)

Let's first review a few definitions that are listed in Table 1 that will facilitate the use-cases and examples that follow.

Parameters	Definition
Session 5 tuple	Identify the session with Client IP, Server IP, Client Port, Server Port, IP Protocol (always TCP)
Client	dpoint that sent the SYN packet
Server	The endpoint that received the SYN packet and sent the SYN-ACK
RTT From Server	Round trip time as measured between the packet sent by the client to the acknowledgment sent by the server
RTT To Server	Round trip time as measured between the packet sent by the server to the acknowledgment sent by the client
Retransmissions Server	The total number of retransmissions from the server-side. Retransmission is considered when the packet has a sequence number that goes backward
Retransmissions Client	The total number of retransmissions from the client-side. Retransmission is considered when the packet has a sequence number that goes backward
SYN Failure	The session had SYN packet but without SYN-ACK, indicate a security issue or server issue or network issue
Missing SYN	The session had a SYN-ACK packet, but no SYN packet might indicate security threat or asymmetric routing
Z-Win Server count	The total count of packets with TCP window set to zero from the server
Z-Win Client count	The total count of packets with TCP window set to zero from the client
[TCP] Sessions	Sessions represent a TCP connection. TCP sessions are two-sided and have a "client" and a "server". The client is the endpoint that sent the SYN message, and the server is the endpoint that responded with a SYN-ACK. This relationship has to be maintained during the session. TCP sessions are identified by the duration of the session, the number of packets and bytes of the session, retransmission, window-size (MinWin), response-time (RSP-Time), and round-trip time (RTT).
TCP Retransmission	TCP retransmissions occur in many ways, and different tools, including Wireshark, have different methods to identify and report them. cPacket uses a simple algorithm by which a retransmission is counted every time the sequence number in a TCP packet goes backward

Table 1: Relevant KPI Definitions

Figure 2 below shows a typical client-server TCP connection flow. A cVu-V Virtual Network Packet Broker is used as an agentless monitor in the conversation path that reports network KPIs to help understand the health of the connection flow and latency metrics.

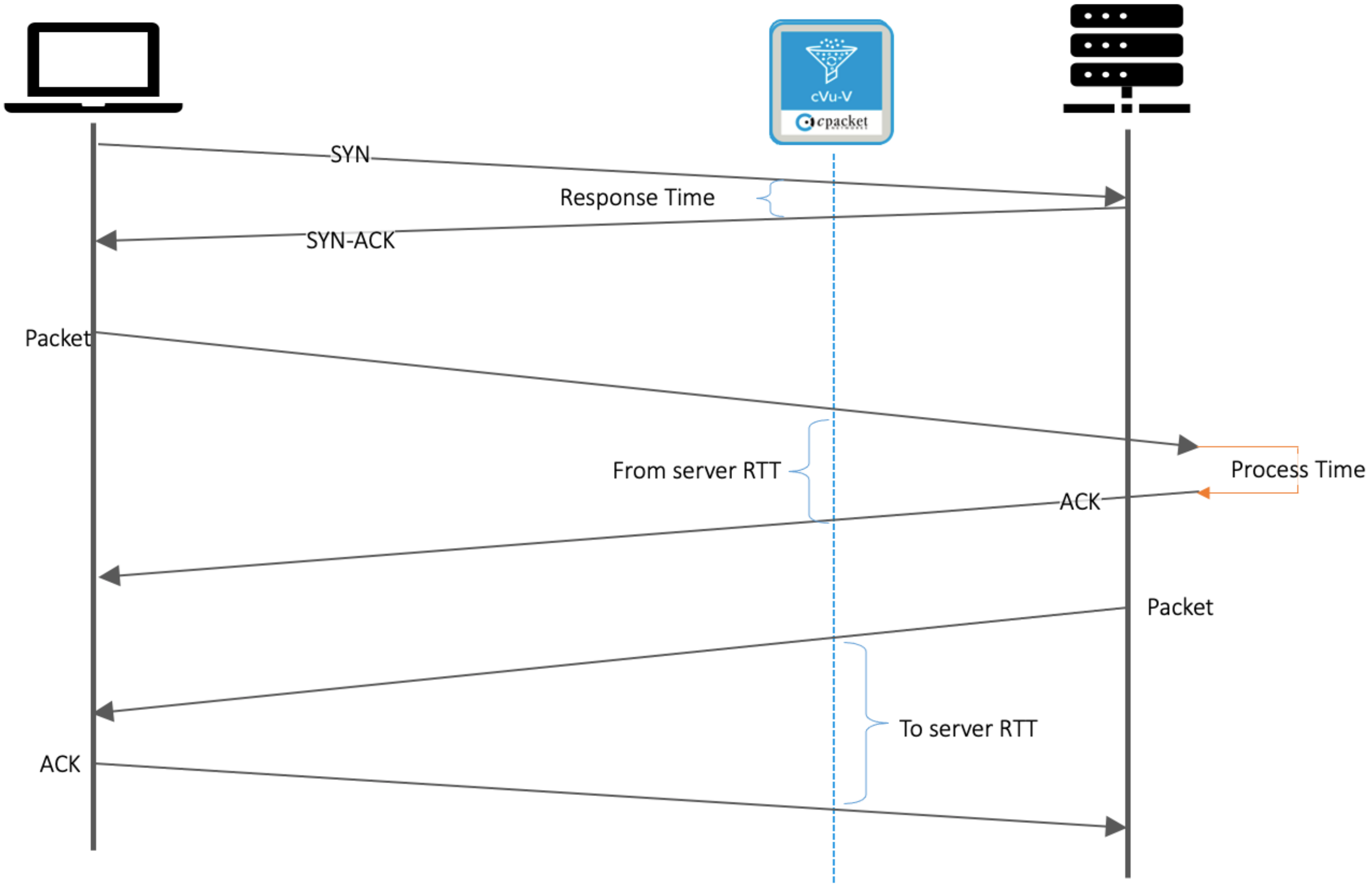


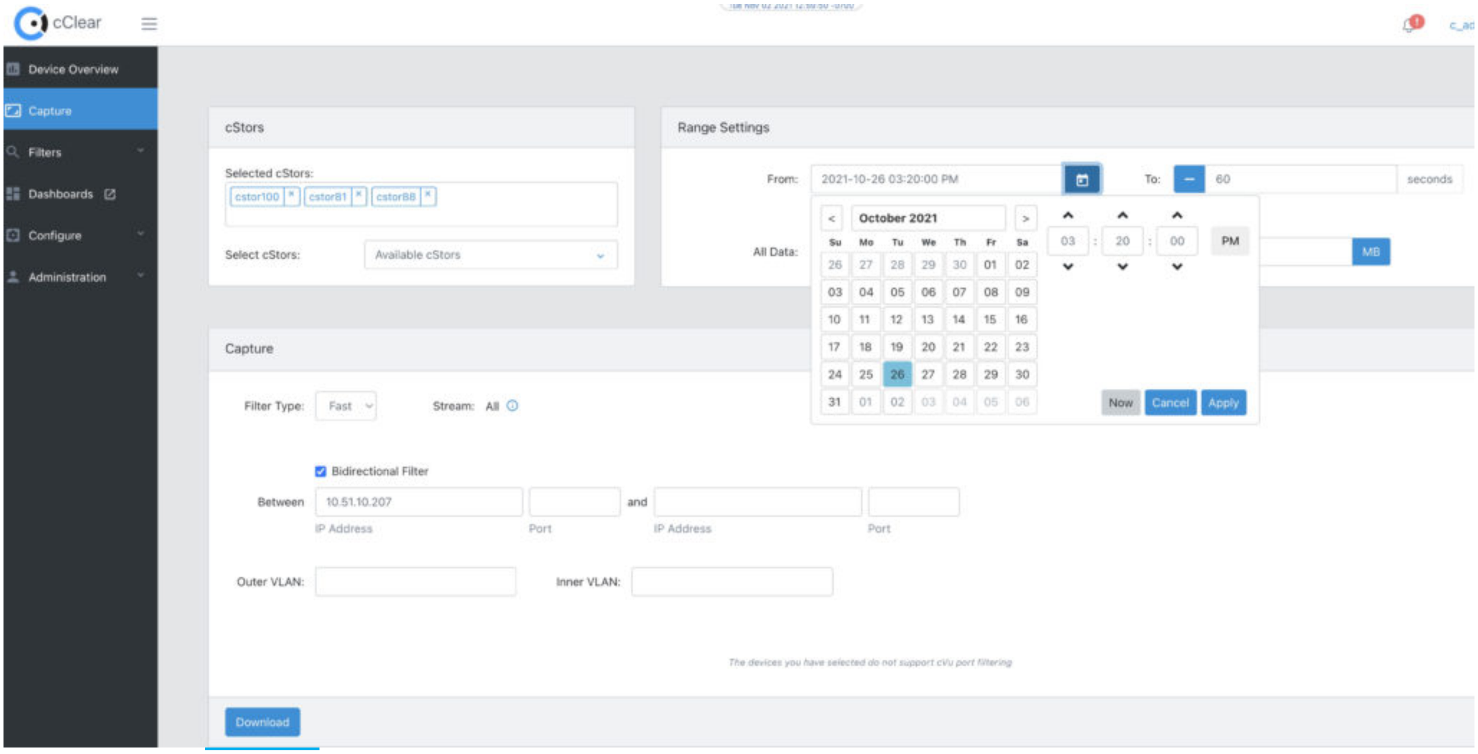
Figure 2 – TCP Connection Flow

Isolating Network Issues

D When problems arise, the Help Desk receives an incident ticket that kicks off an RCA effort to identify the root cause of the problem and understand its impact. Once the root cause is known, the problem can be isolated, typically one of these domains: client device(s), servers, virtual machines, application (including underlying services), or network services. Determining where the problem is, such as a VM, instance, application, service, or network, enables the IT team to respond appropriately.

If the problem is due to a VM or cloud instance, knowing the IP address of what has failed or malfunctioned completes the isolation of the problem so that remediation can proceed. Let's take an example of a reported service down, or the user experience is impacted. Once the IP address of the problematic service is known, you can use the cPacket cClear®-V Analytics Engine appliance to download the PCAP file for the time of the incident. To do so, you'll want to select **cClear> Capture**

Enter the reported Server IP address and time period under investigation and add any filtering to reduce the noise in the PCAP file. Then click **Select> Download**



cStors

Selected cStors: cstor100 × cstor81 × cstor88 ×

Select cStors: Available cStors

Range Settings

From: 2021-10-26 03:20:00 PM

To: 60 seconds

All Data: MB

Su	Mo	Tu	We	Th	Fr	Sa
26	27	28	29	30	01	02
03	04	05	06	07	08	09
10	11	12	13	14	15	16
17	18	19	20	21	22	23
24	25	26	27	28	29	30
31	01	02	03	04	05	06

Capture

Filter Type: Fast

Stream: All

☑ Bidirectional Filter

Between 10.51.10.207 (IP Address) Port and IP Address Port

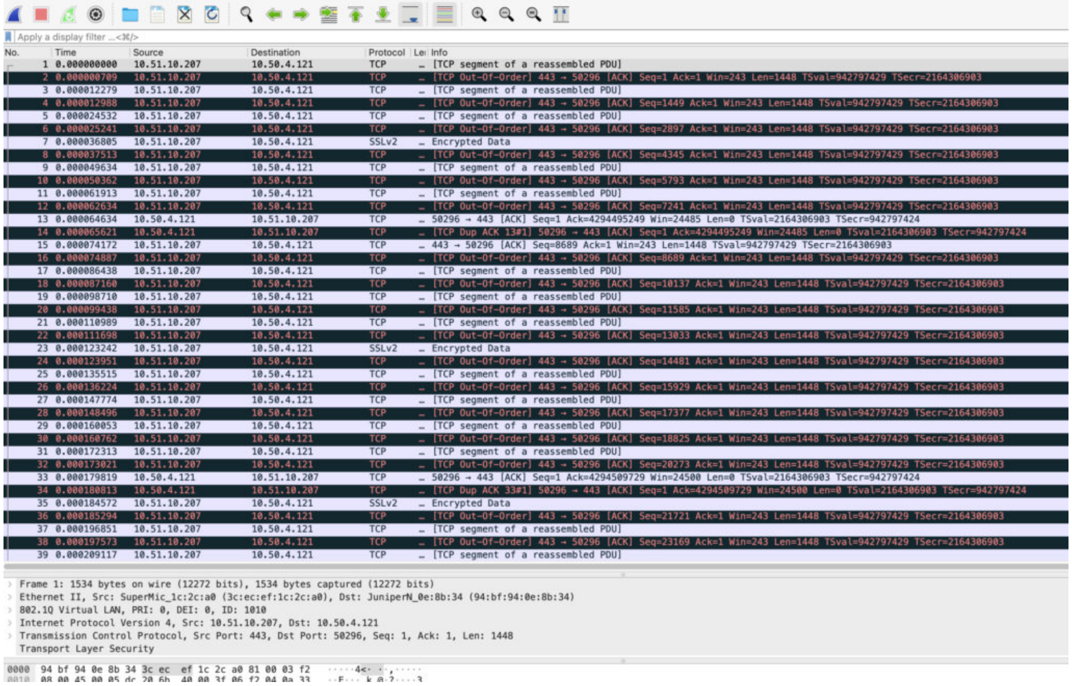
Outer VLAN:

Inner VLAN:

The devices you have selected do not support cVu port filtering

Figure 3 – Select Download PCAP for 10.51.10.207

It is that easy to group packets from across multiple vantage points in the network and download them as a single PCAP file. Select the “Range Settings” for the incident time period under investigation to filter packet details on the archived forensic data. Figure 4 shows an example PCAP file with out-of-order packets.

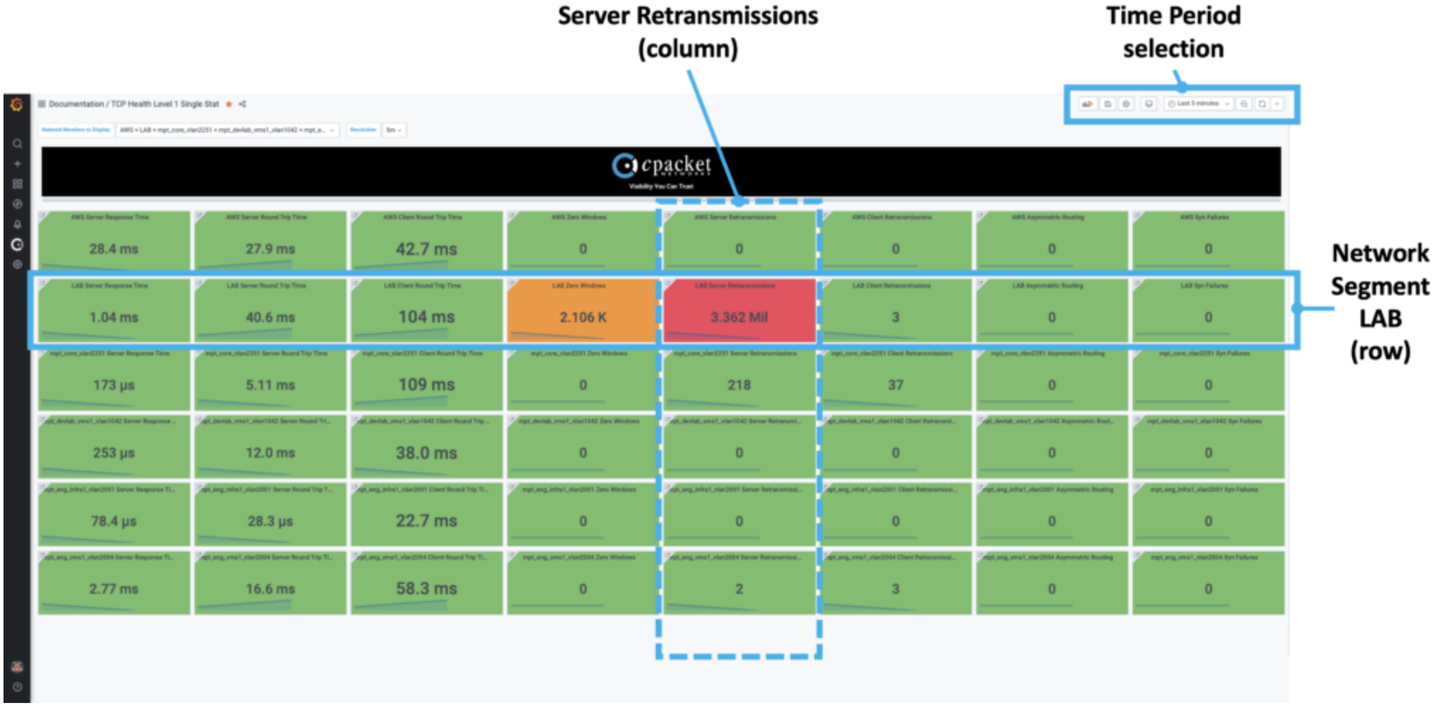


```
No.  Time         Source        Destination   Protocol Le Info
   1 0.000000000  10.51.10.207  10.50.4.121   TCP      … [TCP segment of a reassembled PDU]
   2 0.000000709  10.51.10.207  10.50.4.121   TCP      … [TCP Out-Of-Order] 443 → 50296 [ACK] Seq=1 Ack=1 Win=243 Len=1448 TSval=942797429 TSecr=2164306903
   3 0.000012279  10.51.10.207  10.50.4.121   TCP      … [TCP segment of a reassembled PDU]
   4 0.000012988  10.51.10.207  10.50.4.121   TCP      … [TCP Out-Of-Order] 443 → 50296 [ACK] Seq=1449 Ack=1 Win=243 Len=1448 TSval=942797429 TSecr=2164306903
   5 0.000024532  10.51.10.207  10.50.4.121   TCP      … [TCP segment of a reassembled PDU]
   6 0.000025241  10.51.10.207  10.50.4.121   TCP      … [TCP Out-Of-Order] 443 → 50296 [ACK] Seq=2897 Ack=1 Win=243 Len=1448 TSval=942797429 TSecr=2164306903
   7 0.000036805  10.51.10.207  10.50.4.121   SSLv2    … Encrypted Data
   8 0.000037513  10.51.10.207  10.50.4.121   TCP      … [TCP Out-Of-Order] 443 → 50296 [ACK] Seq=4345 Ack=1 Win=243 Len=1448 TSval=942797429 TSecr=2164306903
   9 0.000049634  10.51.10.207  10.50.4.121   TCP      … [TCP segment of a reassembled PDU]
  10 0.000050362  10.51.10.207  10.50.4.121   TCP      … [TCP Out-Of-Order] 443 → 50296 [ACK] Seq=5793 Ack=1 Win=243 Len=1448 TSval=942797429 TSecr=2164306903
  11 0.000061913  10.51.10.207  10.50.4.121   TCP      … [TCP segment of a reassembled PDU]
  12 0.000062634  10.51.10.207  10.50.4.121   TCP      … [TCP Out-Of-Order] 443 → 50296 [ACK] Seq=7241 Ack=1 Win=243 Len=1448 TSval=942797429 TSecr=2164306903
  13 0.000064634  10.50.4.121   10.51.10.207  TCP      … 50296 → 443 [ACK] Seq=1 Ack=4294495249 Win=24485 Len=0 TSval=2164306903 TSecr=942797424
  14 0.000065621  10.50.4.121   10.51.10.207  TCP      … [TCP Dup ACK 13#1] 50296 → 443 [ACK] Seq=1 Ack=4294495249 Win=24485 Len=0 TSval=2164306903 TSecr=942797424
  15 0.000074172  10.51.10.207  10.50.4.121   TCP      … 443 → 50296 [ACK] Seq=8689 Ack=1 Win=243 Len=1448 TSval=942797429 TSecr=2164306903
  16 0.000074887  10.51.10.207  10.50.4.121   TCP      … [TCP Out-Of-Order] 443 → 50296 [ACK] Seq=8689 Ack=1 Win=243 Len=1448 TSval=942797429 TSecr=2164306903
  17 0.000086438  10.51.10.207  10.50.4.121   TCP      … [TCP segment of a reassembled PDU]
  18 0.000087160  10.51.10.207  10.50.4.121   TCP      … [TCP Out-Of-Order] 443 → 50296 [ACK] Seq=10137 Ack=1 Win=243 Len=1448 TSval=942797429 TSecr=2164306903
  19 0.000098710  10.51.10.207  10.50.4.121   TCP      … [TCP segment of a reassembled PDU]
  20 0.000099438  10.51.10.207  10.50.4.121   TCP      … [TCP Out-Of-Order] 443 → 50296 [ACK] Seq=11585 Ack=1 Win=243 Len=1448 TSval=942797429 TSecr=2164306903
  21 0.000110989  10.51.10.207  10.50.4.121   TCP      … [TCP segment of a reassembled PDU]
  22 0.000111698  10.51.10.207  10.50.4.121   TCP      … [TCP Out-Of-Order] 443 → 50296 [ACK] Seq=13033 Ack=1 Win=243 Len=1448 TSval=942797429 TSecr=2164306903
  23 0.000123242  10.51.10.207  10.50.4.121   SSLv2    … Encrypted Data
  24 0.000123951  10.51.10.207  10.50.4.121   TCP      … [TCP Out-Of-Order] 443 → 50296 [ACK] Seq=14481 Ack=1 Win=243 Len=1448 TSval=942797429 TSecr=2164306903
  25 0.000135515  10.51.10.207  10.50.4.121   TCP      … [TCP segment of a reassembled PDU]
  26 0.000136224  10.51.10.207  10.50.4.121   TCP      … [TCP Out-Of-Order] 443 → 50296 [ACK] Seq=15929 Ack=1 Win=243 Len=1448 TSval=942797429 TSecr=2164306903
  27 0.000147774  10.51.10.207  10.50.4.121   TCP      … [TCP segment of a reassembled PDU]
  28 0.000148496  10.51.10.207  10.50.4.121   TCP      … [TCP Out-Of-Order] 443 → 50296 [ACK] Seq=17377 Ack=1 Win=243 Len=1448 TSval=942797429 TSecr=2164306903
  29 0.000160053  10.51.10.207  10.50.4.121   TCP      … [TCP segment of a reassembled PDU]
  30 0.000160762  10.51.10.207  10.50.4.121   TCP      … [TCP Out-Of-Order] 443 → 50296 [ACK] Seq=18825 Ack=1 Win=243 Len=1448 TSval=942797429 TSecr=2164306903
  31 0.000172313  10.51.10.207  10.50.4.121   TCP      … [TCP segment of a reassembled PDU]
  32 0.000173021  10.51.10.207  10.50.4.121   TCP      … [TCP Out-Of-Order] 443 → 50296 [ACK] Seq=20273 Ack=1 Win=243 Len=1448 TSval=942797429 TSecr=2164306903
  33 0.000179819  10.50.4.121   10.51.10.207  TCP      … 50296 → 443 [ACK] Seq=1 Ack=4294509729 Win=24500 Len=0 TSval=2164306903 TSecr=942797424
  34 0.000180813  10.50.4.121   10.51.10.207  TCP      … [TCP Dup ACK 33#1] 50296 → 443 [ACK] Seq=1 Ack=4294509729 Win=24500 Len=0 TSval=2164306903 TSecr=942797424
  35 0.000184572  10.51.10.207  10.50.4.121   SSLv2    … Encrypted Data
  36 0.000185294  10.51.10.207  10.50.4.121   TCP      … [TCP Out-Of-Order] 443 → 50296 [ACK] Seq=21721 Ack=1 Win=243 Len=1448 TSval=942797429 TSecr=2164306903
  37 0.000196851  10.51.10.207  10.50.4.121   TCP      … [TCP segment of a reassembled PDU]
  38 0.000197573  10.51.10.207  10.50.4.121   TCP      … [TCP Out-Of-Order] 443 → 50296 [ACK] Seq=23169 Ack=1 Win=243 Len=1448 TSval=942797429 TSecr=2164306903
  39 0.000209117  10.51.10.207  10.50.4.121   TCP      … [TCP segment of a reassembled PDU]

> Frame 1: 1534 bytes on wire (12272 bits), 1534 bytes captured (12272 bits)
> Ethernet II, Src: SuperMic_1c:2c:a0 (3c:ec:ef:1c:2c:a0), Dst: JuniperN_0e:8b:34 (94:bf:94:0e:8b:34)
> 802.1Q Virtual LAN, PRI: 0, DEI: 0, ID: 1010
> Internet Protocol Version 4, Src: 10.51.10.207, Dst: 10.50.4.121
> Transmission Control Protocol, Src Port: 443, Dst Port: 50296, Seq: 1, Ack: 1, Len: 1448
  Transport Layer Security

0000  94 bf 94 0e 8b 34 3c ec  ef 1c 2c a0 81 00 03 f2   ·····4<· ··,·····
0010  08 00 45 00 05 dc 20 6b  40 00 3f 06 f2 04 0a 33   ··E··· k @·?····3
```

Figure 4 – PCAP File for 10.51.10.207

If you do not have the specific client/server details, then you will need to identify them using the cPacket cClear®-V Analytics Engine, which is done by selecting **Dashboards> TCP Health Dashboard**

Figure 5 below is a high-level TCP Health dashboard that displays the network segments horizontally (i.e., DMZ, AWS, LAB) with relevant KPIs listed in columns, which is an excellent high-level starting point. This tells you which network service or key performance indicators are signaling problems versus operating normally. This gives the operator a high-level view of the network segments and a general indication of health. The TCP Health dashboard very quickly isolates the incident to the LAB segment by showing the health and performance for the past five minutes.



Documentation / TCP Health Level 1 Single Stat

Network Monitors to Display: AWS + LAB + mpt_core_vlan2251 + mpt_devlab_vms1_vlan1042 + mpt_a...

Resolution: 5m

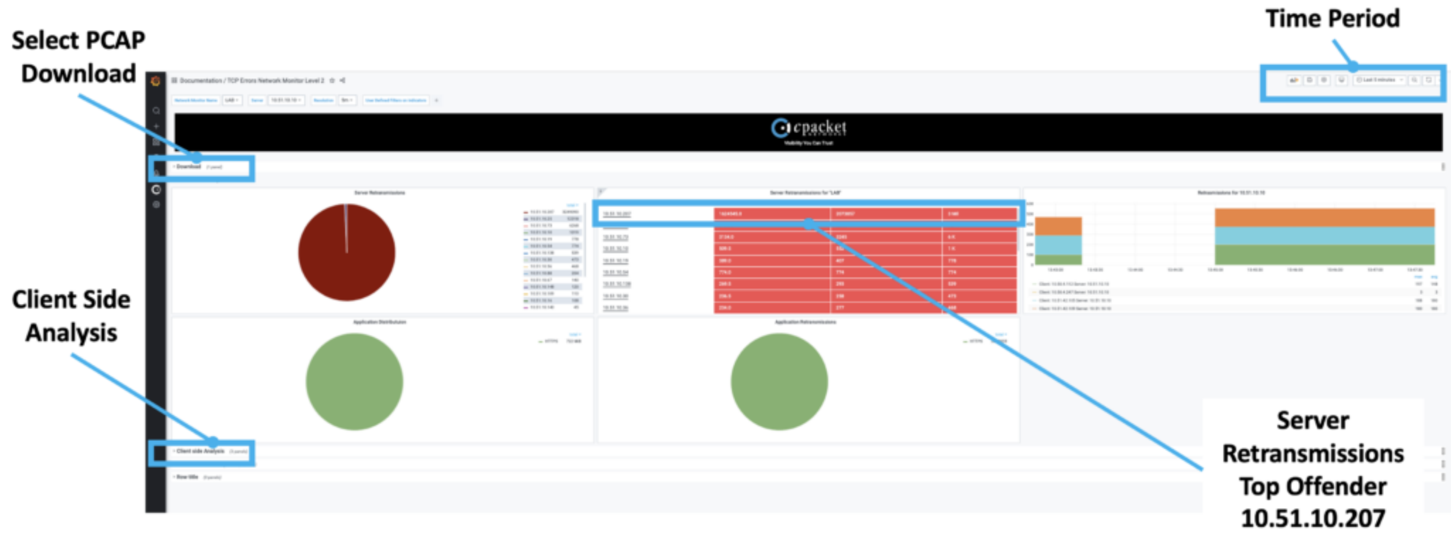
Last 5 minutes

AWS Server Response Time 28.4 ms	AWS Server Round Trip Time 27.9 ms	AWS Client Round Trip Time 42.7 ms	AWS Zero Windows 0	AWS Server Retransmissions 0	AWS Client Retransmissions 0	AWS Asymmetric Routing 0	AWS Syn Failures 0
LAB Server Response Time 1.04 ms	LAB Server Round Trip Time 40.6 ms	LAB Client Round Trip Time 104 ms	LAB Zero Windows 2.106 K	LAB Server Retransmissions 3.362 Mil	LAB Client Retransmissions 3	LAB Asymmetric Routing 0	LAB Syn Failures 0
mpt_core_vlan2251 Server Response Time 173 µs	mpt_core_vlan2251 Server Round Trip Time 5.11 ms	mpt_core_vlan2251 Client Round Trip Time 109 ms	mpt_core_vlan2251 Zero Windows 0	mpt_core_vlan2251 Server Retransmissions 218	mpt_core_vlan2251 Client Retransmissions 37	mpt_core_vlan2251 Asymmetric Routing 0	mpt_core_vlan2251 Syn Failures 0
mpt_devlab_vms1_vlan1042 Server Response ... 253 µs	mpt_devlab_vms1_vlan1042 Server Round Tri... 12.0 ms	mpt_devlab_vms1_vlan1042 Client Round Trip ... 38.0 ms	mpt_devlab_vms1_vlan1042 Zero Windows 0	mpt_devlab_vms1_vlan1042 Server Retransmi... 0	mpt_devlab_vms1_vlan1042 Client Retransmi... 0	mpt_devlab_vms1_vlan1042 Asymmetric Rout... 0	mpt_devlab_vms1_vlan1042 Syn Failures 0
mpt_eng_infra1_vlan2001 Server Response Ti... 78.4 µs	mpt_eng_infra1_vlan2001 Server Round Trip T... 28.3 µs	mpt_eng_infra1_vlan2001 Client Round Trip Ti... 22.7 ms	mpt_eng_infra1_vlan2001 Zero Windows 0	mpt_eng_infra1_vlan2001 Server Retransmissi... 0	mpt_eng_infra1_vlan2001 Client Retransmissi... 0	mpt_eng_infra1_vlan2001 Asymmetric Routing 0	mpt_eng_infra1_vlan2001 Syn Failures 0
mpt_eng_vms1_vlan2004 Server Response Ti... 2.77 ms	mpt_eng_vms1_vlan2004 Server Round Trip Ti... 16.6 ms	mpt_eng_vms1_vlan2004 Client Round Trip Ti... 58.3 ms	mpt_eng_vms1_vlan2004 Zero Windows 0	mpt_eng_vms1_vlan2004 Server Retransmissi... 2	mpt_eng_vms1_vlan2004 Client Retransmissi... 3	mpt_eng_vms1_vlan2004 Asymmetric Routing 0	mpt_eng_vms1_vlan2004 Syn Failures 0

Figure 5 – TCP Health Dashboard

At this point, you and your team have valuable insights - the What, Where, and When that isolate the problem and determine the root cause.

By clicking on the LAB Server Retransmissions KPI (red box), this will take you to a drill-down visualization showing the IP addresses in the flow for the last 5-minutes (Figure 6). This view will show you both the client and server involved in the Server Retransmissions.



Select PCAP Download

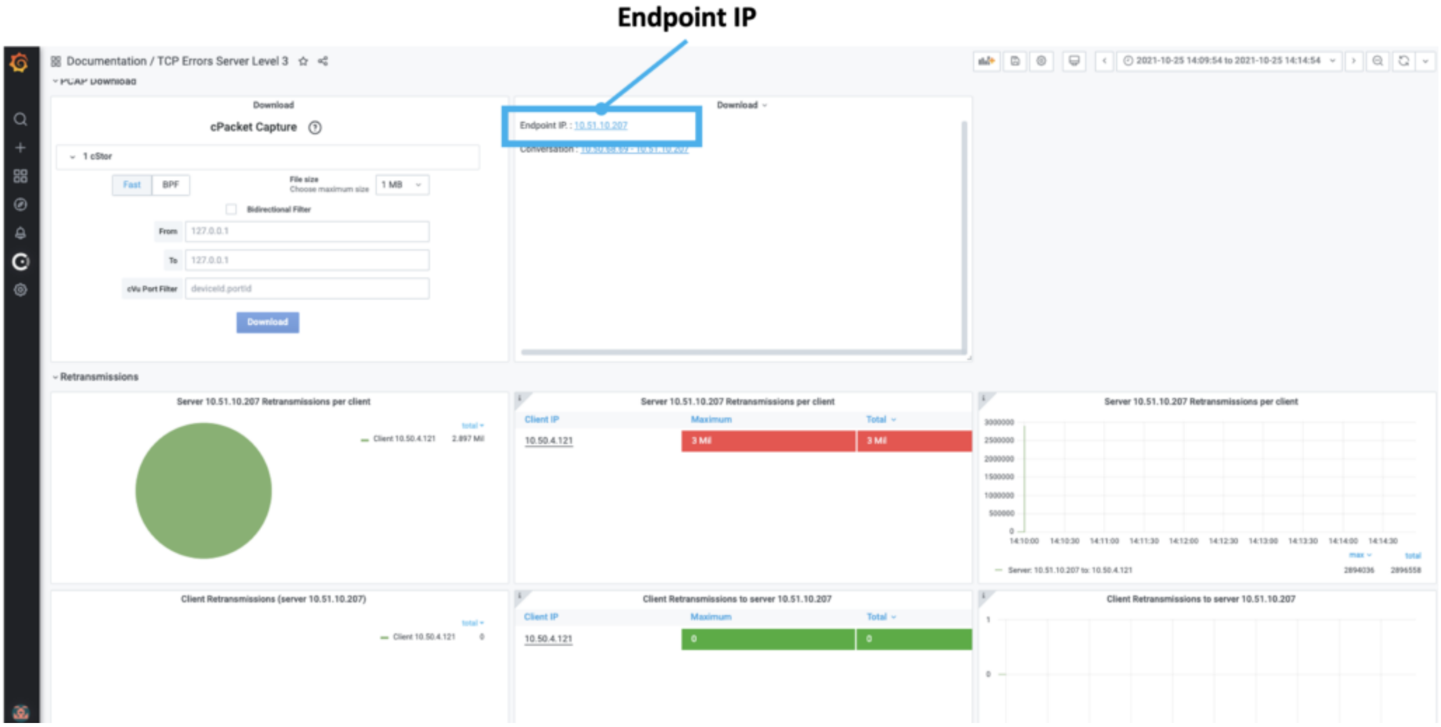
Time Period

Client Side Analysis

Server Retransmissions Top Offender 10.51.10.207

Figure 6 – TCP Errors Level 2 – Server-Side Analysis

Now you have the IP addresses we are interested in, selecting the download is very simple, as shown in Figure 7. There are options for filtering, including Berkley Packet Filtering (BPF), that allow you to home in on the data of interest.



Endpoint IP

Documentation / TCP Errors Server Level 3
PCAP Download
2021-10-25 14:09:54 to 2021-10-25 14:14:54
Download
cPacket Capture
1 cStor
Fast
BPF
Download
Retransmissions

File size: Choose maximum size: 1 MB

☐ Bidirectional Filter

From: 127.0.0.1

To: 127.0.0.1

cVu Port Filter: deviceId.portId

Endpoint IP: : 10.51.10.207

Conversation : [illegible]

Server 10.51.10.207 Retransmissions per client

	total
Client 10.50.4.121	2.897 Mil

Server 10.51.10.207 Retransmissions per client

Client IP	Maximum	Total
10.50.4.121	3 Mil	3 Mil

Server 10.51.10.207 Retransmissions per client

	max	total
Server: 10.51.10.207 to: 10.50.4.121	2894036	2896558

Client Retransmissions (server 10.51.10.207)

	total
Client 10.50.4.121	0

Client Retransmissions to server 10.51.10.207

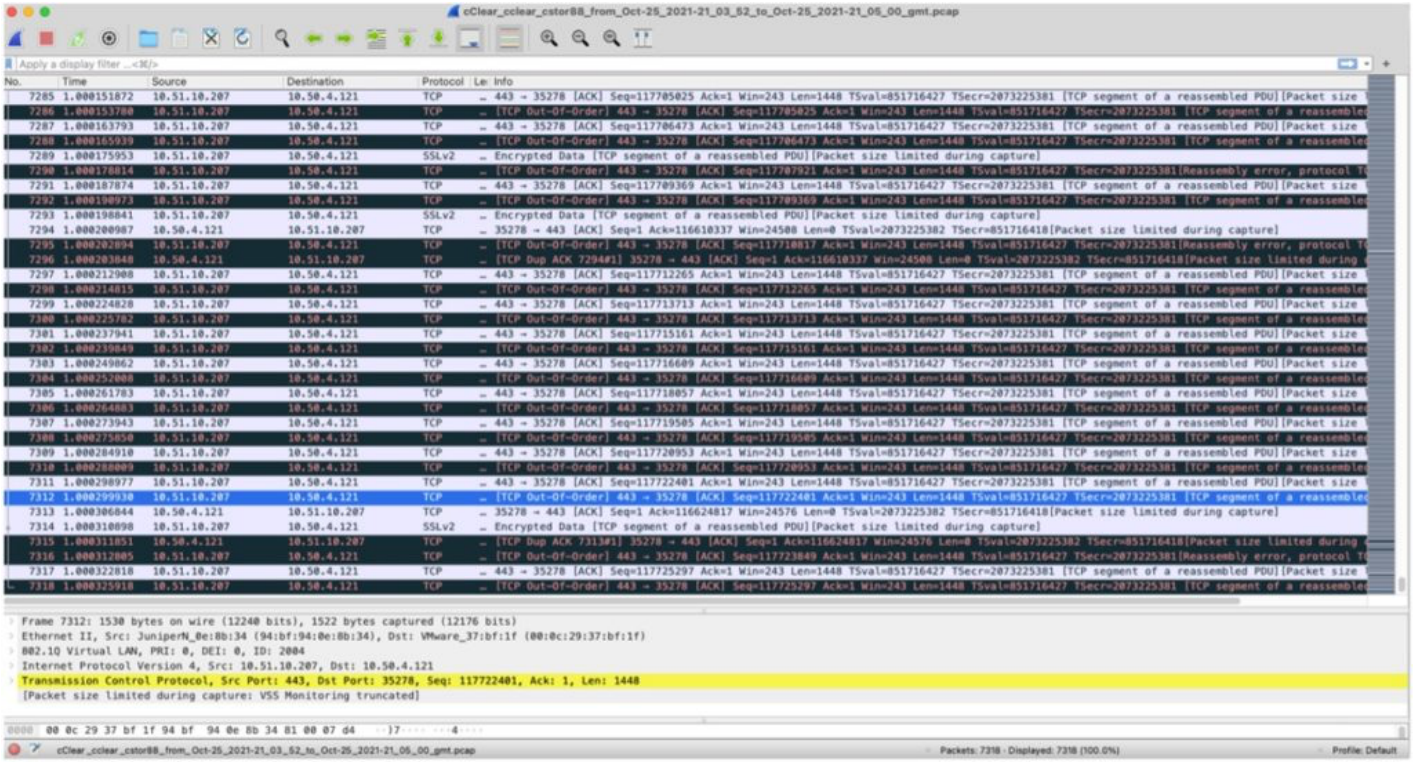
Client IP	Maximum	Total
10.50.4.121	0	0

Client Retransmissions to server 10.51.10.207

Figure 7 – Select Download PCAP 10.51.10.207

In this incident example, we discovered the network was operating as expected. The connectivity between the two offending hosts was generating out-of-order TCP sequence packets. This is the time to engage with the server and/or application team to let them know further investigation of the two nodes in the LAB network requires detailed inspection. This enables them to efficiently isolate the problem by giving them access to the network packet data and KPIs or sending a PCAP file.

The team discovered the port 443 connection was coming from a development vSphere VM instance to an engineering server in a hung state. The system was no longer responding to user inputs, but its IP address was still responding.



cClear_cclear_cstor88_from_Oct-25_2021-21_03_52_to_Oct-25_2021-21_05_00_gmt.pcap
Apply a display filter ...<⌘/>

```
No.   Time        Source        Destination   Protocol Le Info
 7285 1.000151872 10.51.10.207  10.50.4.121   TCP      … 443 → 35278 [ACK] Seq=117705025 Ack=1 Win=243 Len=1448 TSval=851716427 TSecr=2073225381 [TCP segment of a reassembled PDU][Packet size
 7286 1.000153780 10.51.10.207  10.50.4.121   TCP      … [TCP Out-Of-Order] 443 → 35278 [ACK] Seq=117705025 Ack=1 Win=243 Len=1448 TSval=851716427 TSecr=2073225381 [TCP segment of a reassemble
 7287 1.000163793 10.51.10.207  10.50.4.121   TCP      … 443 → 35278 [ACK] Seq=117706473 Ack=1 Win=243 Len=1448 TSval=851716427 TSecr=2073225381 [TCP segment of a reassembled PDU][Packet size
 7288 1.000165939 10.51.10.207  10.50.4.121   TCP      … [TCP Out-Of-Order] 443 → 35278 [ACK] Seq=117706473 Ack=1 Win=243 Len=1448 TSval=851716427 TSecr=2073225381 [TCP segment of a reassemble
 7289 1.000175953 10.51.10.207  10.50.4.121   SSLv2    … Encrypted Data [TCP segment of a reassembled PDU][Packet size limited during capture]
 7290 1.000178814 10.51.10.207  10.50.4.121   TCP      … [TCP Out-Of-Order] 443 → 35278 [ACK] Seq=117707921 Ack=1 Win=243 Len=1448 TSval=851716427 TSecr=2073225381[Reassembly error, protocol T
 7291 1.000187874 10.51.10.207  10.50.4.121   TCP      … 443 → 35278 [ACK] Seq=117709369 Ack=1 Win=243 Len=1448 TSval=851716427 TSecr=2073225381 [TCP segment of a reassembled PDU][Packet size
 7292 1.000190973 10.51.10.207  10.50.4.121   TCP      … [TCP Out-Of-Order] 443 → 35278 [ACK] Seq=117709369 Ack=1 Win=243 Len=1448 TSval=851716427 TSecr=2073225381 [TCP segment of a reassemble
 7293 1.000198841 10.51.10.207  10.50.4.121   SSLv2    … Encrypted Data [TCP segment of a reassembled PDU][Packet size limited during capture]
 7294 1.000200987 10.50.4.121   10.51.10.207  TCP      … 35278 → 443 [ACK] Seq=1 Ack=116610337 Win=24508 Len=0 TSval=2073225382 TSecr=851716418[Packet size limited during capture]
 7295 1.000202894 10.51.10.207  10.50.4.121   TCP      … [TCP Out-Of-Order] 443 → 35278 [ACK] Seq=117710817 Ack=1 Win=243 Len=1448 TSval=851716427 TSecr=2073225381[Reassembly error, protocol T
 7296 1.000203848 10.50.4.121   10.51.10.207  TCP      … [TCP Dup ACK 7294#1] 35278 → 443 [ACK] Seq=1 Ack=116610337 Win=24508 Len=0 TSval=2073225382 TSecr=851716418[Packet size limited during
 7297 1.000212908 10.51.10.207  10.50.4.121   TCP      … 443 → 35278 [ACK] Seq=117712265 Ack=1 Win=243 Len=1448 TSval=851716427 TSecr=2073225381 [TCP segment of a reassembled PDU][Packet size
 7298 1.000214815 10.51.10.207  10.50.4.121   TCP      … [TCP Out-Of-Order] 443 → 35278 [ACK] Seq=117712265 Ack=1 Win=243 Len=1448 TSval=851716427 TSecr=2073225381 [TCP segment of a reassemble
 7299 1.000224828 10.51.10.207  10.50.4.121   TCP      … 443 → 35278 [ACK] Seq=117713713 Ack=1 Win=243 Len=1448 TSval=851716427 TSecr=2073225381 [TCP segment of a reassembled PDU][Packet size
 7300 1.000225782 10.51.10.207  10.50.4.121   TCP      … [TCP Out-Of-Order] 443 → 35278 [ACK] Seq=117713713 Ack=1 Win=243 Len=1448 TSval=851716427 TSecr=2073225381 [TCP segment of a reassemble
 7301 1.000237941 10.51.10.207  10.50.4.121   TCP      … 443 → 35278 [ACK] Seq=117715161 Ack=1 Win=243 Len=1448 TSval=851716427 TSecr=2073225381 [TCP segment of a reassembled PDU][Packet size
 7302 1.000239849 10.51.10.207  10.50.4.121   TCP      … [TCP Out-Of-Order] 443 → 35278 [ACK] Seq=117715161 Ack=1 Win=243 Len=1448 TSval=851716427 TSecr=2073225381 [TCP segment of a reassemble
 7303 1.000249862 10.51.10.207  10.50.4.121   TCP      … 443 → 35278 [ACK] Seq=117716609 Ack=1 Win=243 Len=1448 TSval=851716427 TSecr=2073225381 [TCP segment of a reassembled PDU][Packet size
 7304 1.000252008 10.51.10.207  10.50.4.121   TCP      … [TCP Out-Of-Order] 443 → 35278 [ACK] Seq=117716609 Ack=1 Win=243 Len=1448 TSval=851716427 TSecr=2073225381 [TCP segment of a reassemble
 7305 1.000261783 10.51.10.207  10.50.4.121   TCP      … 443 → 35278 [ACK] Seq=117718057 Ack=1 Win=243 Len=1448 TSval=851716427 TSecr=2073225381 [TCP segment of a reassembled PDU][Packet size
 7306 1.000264883 10.51.10.207  10.50.4.121   TCP      … [TCP Out-Of-Order] 443 → 35278 [ACK] Seq=117718057 Ack=1 Win=243 Len=1448 TSval=851716427 TSecr=2073225381 [TCP segment of a reassemble
 7307 1.000273943 10.51.10.207  10.50.4.121   TCP      … 443 → 35278 [ACK] Seq=117719505 Ack=1 Win=243 Len=1448 TSval=851716427 TSecr=2073225381 [TCP segment of a reassembled PDU][Packet size
 7308 1.000275850 10.51.10.207  10.50.4.121   TCP      … [TCP Out-Of-Order] 443 → 35278 [ACK] Seq=117719505 Ack=1 Win=243 Len=1448 TSval=851716427 TSecr=2073225381 [TCP segment of a reassemble
 7309 1.000284910 10.51.10.207  10.50.4.121   TCP      … 443 → 35278 [ACK] Seq=117720953 Ack=1 Win=243 Len=1448 TSval=851716427 TSecr=2073225381 [TCP segment of a reassembled PDU][Packet size
 7310 1.000288009 10.51.10.207  10.50.4.121   TCP      … [TCP Out-Of-Order] 443 → 35278 [ACK] Seq=117720953 Ack=1 Win=243 Len=1448 TSval=851716427 TSecr=2073225381 [TCP segment of a reassemble
 7311 1.000298977 10.51.10.207  10.50.4.121   TCP      … 443 → 35278 [ACK] Seq=117722401 Ack=1 Win=243 Len=1448 TSval=851716427 TSecr=2073225381 [TCP segment of a reassembled PDU][Packet size
 7312 1.000299930 10.51.10.207  10.50.4.121   TCP      … [TCP Out-Of-Order] 443 → 35278 [ACK] Seq=117722401 Ack=1 Win=243 Len=1448 TSval=851716427 TSecr=2073225381 [TCP segment of a reassemble
 7313 1.000306844 10.50.4.121   10.51.10.207  TCP      … 35278 → 443 [ACK] Seq=1 Ack=116624817 Win=24576 Len=0 TSval=2073225382 TSecr=851716418[Packet size limited during capture]
 7314 1.000310898 10.51.10.207  10.50.4.121   SSLv2    … Encrypted Data [TCP segment of a reassembled PDU][Packet size limited during capture]
 7315 1.000311851 10.50.4.121   10.51.10.207  TCP      … [TCP Dup ACK 7313#1] 35278 → 443 [ACK] Seq=1 Ack=116624817 Win=24576 Len=0 TSval=2073225382 TSecr=851716418[Packet size limited during
 7316 1.000312805 10.51.10.207  10.50.4.121   TCP      … [TCP Out-Of-Order] 443 → 35278 [ACK] Seq=117723849 Ack=1 Win=243 Len=1448 TSval=851716427 TSecr=2073225381[Reassembly error, protocol T
 7317 1.000322818 10.51.10.207  10.50.4.121   TCP      … 443 → 35278 [ACK] Seq=117725297 Ack=1 Win=243 Len=1448 TSval=851716427 TSecr=2073225381 [TCP segment of a reassembled PDU][Packet size
 7318 1.000325918 10.51.10.207  10.50.4.121   TCP      … [TCP Out-Of-Order] 443 → 35278 [ACK] Seq=117725297 Ack=1 Win=243 Len=1448 TSval=851716427 TSecr=2073225381 [TCP segment of a reassemble

Frame 7312: 1530 bytes on wire (12240 bits), 1522 bytes captured (12176 bits)
Ethernet II, Src: JuniperN_0e:8b:34 (94:bf:94:0e:8b:34), Dst: VMware_37:bf:1f (00:0c:29:37:bf:1f)
802.1Q Virtual LAN, PRI: 0, DEI: 0, ID: 2004
Internet Protocol Version 4, Src: 10.51.10.207, Dst: 10.50.4.121
Transmission Control Protocol, Src Port: 443, Dst Port: 35278, Seq: 117722401, Ack: 1, Len: 1448
 [Packet size limited during capture: VSS Monitoring truncated]

0000  00 0c 29 37 bf 1f 94 bf  94 0e 8b 34 81 00 07 d4   ··)7···· ···4····
```

cClear_cclear_cstor88_from_Oct-25_2021-21_03_52_to_Oct-25_2021-21_05_00_gmt.pcap
Packets: 7318 · Displayed: 7318 (100.0%)
Profile: Default

Figure 8 – Wireshark PCAP Forensic file for 10.51.10.207

Summary for Isolating Network Problems

This document helps you navigate the challenge of managing a large amount of network telemetry data to efficiently isolate problems and reduce the back and forth between operational teams. Faster isolation and root cause determination is essential to the NetOps team because the network is typically guilty until proven innocent (which is somewhat jokingly referred to as the meantime to innocence).

When you need to isolate and troubleshoot network, server, or application problems, you will need to leverage the output of the cCloud Visibility Suite, which are streamed and stored packet data plus KPIs from analytics for troubleshooting and isolating Network, Server, or Application problems. We used an example to show how network problems are isolated after too many Server Retransmissions occurred. This resulted in discovering the IP addresses involved in the connection flow and the PCAP file available for analysis.

This application note showed you how to use the cCloud Visibility Suite, the packet data it provides, and the steps to take to solve problems. Hence, giving much greater confidence when working on a high-priority incident during an enormously stressful time will ensure operational teams effectively avoid excessive outages.

Related Information:
cPacket Cloud Observability for AWS – Solution Brief
cPacket Intelligent Observability Platform for Azure – Solution Brief