



Detecting, Troubleshooting, and Preventing Congestion in Storage Networks

ciscopress.com

PARESH GUPTA, CCIE® NO. 36645
EDWARD MAZUREK, CCIE® NO. 6448

FREE SAMPLE CHAPTER |



Contents

Introduction xxxii

Chapter 1 Introduction to Congestion in Storage Networks 1

Types of Storage in a Data Center	1
Storage Type—By Location	2
<i>Local Storage</i>	2
<i>Remote Storage</i>	2
Storage Type—By Access Level	3
<i>Block Storage</i>	3
<i>File Storage</i>	4
<i>Object Storage</i>	4
<i>Storage for Clustered and Distributed File Systems</i>	5
<i>SDS, HCI, and Everything Else</i>	5
Storage Protocols, Transports, and Networks	6
Network Type—By Framing and Encoding	6
<i>Ethernet</i>	6
<i>Fibre Channel (FC)</i>	7
<i>InfiniBand (IB)</i>	7
Network Type—By Use of Flow Control	8
<i>Lossy Networks</i>	8
<i>Lossless Networks</i>	9
<i>Converged Ethernet Networks</i>	11
Crossing the Boundaries of Network Types	11
<i>Fibre Channel over Ethernet (FCoE)</i>	11
<i>RDMA over Converged Ethernet (RoCE)</i>	12
Climbing Up the Networking Layers	12
<i>Internet Protocol</i>	12
<i>Transmission Control Protocol (TCP)</i>	13
<i>User Datagram Protocol (UDP)</i>	14
<i>iSCSI</i>	14
<i>NVMe/TCP</i>	14
<i>NFS</i>	14
<i>SMB</i>	15
<i>HTTP</i>	15

Crossing the Boundaries of Network Types—Again	15
<i>Fibre Channel over IP (FCIP)</i>	15
<i>RDMA-Capable Protocols</i>	15
<i>Storage Protocols That Use RDMA</i>	18
Storage Networks	21
Storage Network Designs	21
<i>Single-Switch Design</i>	21
<i>Edge-Core Design</i>	21
<i>Edge-Core-Edge Design</i>	23
<i>Mesh Design</i>	23
<i>Spine-Leaf Design</i>	23
Terminology	24
Fibre Channel and FCoE Terminology	24
Choice of Storage	25
Choice of Storage Network	25
Dedicated Versus Shared Networks for Storage Traffic	26
Common Questions on Storage Networks	27
Q: <i>What is the difference between a network and a fabric?</i>	27
Q: <i>What's the difference between a storage area network (SAN) and a storage network?</i>	27
Q: <i>Do storage networks have a role in the cloud?</i>	28
Q: <i>Do storage networks have a role in container storage?</i>	28
Congestion in Storage Networks: An Overview	28
Congestion Spreading	29
Causes of Congestion in Storage Networks	31
<i>Congestion Due to Slow End Devices</i>	31
<i>Congestion Due to Overutilization of a Link</i>	32
<i>Bit Errors on a Link</i>	38
<i>Lack of Buffers for the Distance, Frame Size, and Speed of a Link</i>	39
Source of Congestion in Storage Networks	40
<i>Congestion from End Devices</i>	40
<i>Congestion on ISLs</i>	40
<i>Congestion Within Switches</i>	40
Common Questions About Congestion in Storage Networks	41
Q: <i>What is backpressure?</i>	41

- Q: What are traffic burst and microburst? 41*
- Q: Isn't increasing network capacity the ultimate solution to network congestion? 41*
- Q: I was told that unlike Fibre Channel, RoCEv2 does not suffer from slow drain. Is this correct? 42*
- Q: Is slow drain the same as PFC storm? 42*
- Q: Would moving to the cloud eliminate congestion in storage networks? 43*
- Q: Would moving to HCI or SDS eliminate congestion in storage networks? 43*
- NVMe over Fabrics 43
- Common Questions on NVMe over Fabrics 44*
- Q: I have heard that NVMe supports 64K queues, each with 64K commands. How can I be ready for it? 44*
- Q: Doesn't NVMe have mechanisms to control network congestion? 44*
- Q: I built a new environment with NVMe over Fabrics, but the network throughput did not increase. Why? 44*
- Q: What effects does NVMe over Fabrics have on network congestion? 45*
- Q: Someone told me that congestion in their networks vanished after they upgraded to NVMe over Fabrics. Is that possible? 45*
- Q: Is building a dedicated network for NVMe over Fabrics best for congestion management? 45*
- Quality of Service (QoS) 46
- Sources of Delay in a Network 46
- Forwarding Delay 46*
- Propagation Delay 47*
- Serialization Delay 47*
- Queuing Delay 47*
- Common Questions on QoS in Storage Networks 48
- Q: Why do network devices need buffers? 48*
- Q: What is the difference between buffers and queues? 48*
- Q: What is the difference between buffers, pause buffers, and B2B credits? 48*
- Q: Why is queue a common term in IP/Ethernet networks but not in Fibre Channel fabrics? 49*
- Q: What are some common misconceptions about using QoS in storage networks? 49*

- Q: Why is QoS not commonly used in Fibre Channel fabrics?* 50
- Q: Which is better for storage traffic in Ethernet networks: policing or shaping?* 50
- Q: What is the difference between priority and bandwidth in the context of QoS?* 51

Summary 51

References 52

Chapter 2 Understanding Congestion in Fibre Channel Fabrics 55

Fibre Channel Flow Control 55

Initial Communication of B2B Credits 56

Return of B2B Credits During Frame Flow 58

B2B credit counters 60

Important Details About R_RDYs and B2B Credits 61

B2B Flow Control in a Multi-Hop Fabric 63

B2B Flow Control in a Multi-hop Fabric Without Congestion 63

B2B Flow Control in a Multi-hop Fabric with Congestion 64

Buffer Overrun Situation 67

Frame Rate Equalization Using B2B Flow Control 67

Congestion Spreading in Fibre Channel Fabrics 67

Congestion Due to Slow-Drain Devices 68

Congestion Due to Overutilization 70

Congestion Due to Overutilization on Host-Edge Links 70

The Culprit Host 73

Comparing Congestion Due to Slow Drain and Overutilization 73

Effect on the Culprit Host 74

Effect on the Culprit's Port and Its Connected Switchport 74

Effect on the Fabric 74

Congestion in Single-Switch Fabrics 75

Congestion in an ISL 76

Congestion Spreading Due to Edge Devices 77

Overutilization of an ISL 77

Lack of B2B Credits for the Distance, Speed, and Frame Size of an ISL 78

Buffering and the Ability to Absorb Congestion 83

Dependency on Traffic Patterns 84

<i>Effects on Latency</i>	84
<i>The Number of Buffers</i>	85
<i>User Action</i>	85
Frame Flow Within a Fibre Channel Switch	86
Frame Switching Within a Cisco MDS Switch	86
Frame Switching Architecture of a Fibre Channel Switch	89
<i>Location of Buffers: Ingress, Egress, or Both</i>	89
<i>Number of Buffers</i>	89
<i>Preventing Head-of-Line Blocking</i>	89
<i>Store-and-Forward Versus Cut-Through Switching</i>	90
<i>The Ability to Detect and Drop CRC-Corrupted Frames</i>	91
<i>Load-Balancing Schemes on ISLs</i>	92
<i>Congestion Management Features</i>	92
The Effects of Bit Errors on Congestion	92
Fibre Channel Frame Format	93
Fibre Channel Levels	95
Data Transmission on Fibre Channel Media	95
<i>Transforming an I/O Operation to FC Frames</i>	96
<i>Encoding the Frames and Special Functions</i>	97
<i>Special Functions: Delimiters, Primitive Signals, and Primitive Sequences</i>	98
<i>Transmitting Bits on the Media</i>	99
<i>Fibre Channel Baud Rate</i>	100
<i>Fibre Channel Bit Rate</i>	100
<i>Fibre Channel Data Rate</i>	100
<i>Difference Between Fibre Channel Speed and Bit Rate</i>	101
<i>The Effects of Primitive Signals on Data Rate</i>	101
Counters on Fibre Channel Ports	103
<i>Link Initialization Counters</i>	103
<i>Invalid Transmission Words</i>	104
CRC	104
<i>Forward Error Correction (FEC)</i>	105
Case Study: An Online Retailer	108
<i>Observations</i>	109
<i>Conclusions</i>	111
<i>Lessons Learned</i>	111

	Effect of Bit Errors on Congestion: Summary	112
	B2B Credit Loss and Recovery	112
	Loss of Tx B2B Credits Due to Bit Errors	113
	Zero Tx B2B Credits for an Extended Duration	115
	Credit Loss Recovery Using the B2B State Change Mechanism	116
	<i>Negotiation at Link Initialization</i>	117
	<i>Periodic Detection and Recovery of Credit Loss</i>	118
	<i>Important Details About the B2B State Change Mechanism</i>	119
	Credit Loss Recovery Using Link Reset Protocol	121
	Comparison of the B2B State Change Mechanism and Link Reset Protocol	122
	Fibre Channel Counters Summary	123
	Summary	127
	References	127
Chapter 3	Detecting Congestion in Fibre Channel Fabrics	129
	Congestion Detection Workflow	129
	Effects of Congestion (Congestion Severity)	130
	Cause of Congestion	131
	Source of Congestion (Culprits)	131
	Spread of Congestion (Victims)	132
	Time of Congestion Events	132
	How to Detect Congestion	132
	<i>Reactive Approaches</i>	132
	<i>Proactive Approaches</i>	132
	<i>Predictive Approaches</i>	132
	<i>Reactive, Proactive, Predictive, or All?</i>	133
	Where to Detect Congestion	133
	<i>Detecting Congestion on Network Devices</i>	133
	<i>Detecting Congestion on Remote Monitoring Platforms</i>	133
	Congestion Direction: Ingress or Egress	134
	<i>Egress Congestion</i>	135
	<i>Ingress Congestion</i>	135
	Congestion Detection Metrics	135
	Congestion Detection Metrics on Cisco MDS Switches	137
	Tx Credit Unavailability in Microseconds: TxWait	137
	<i>Raw TxWait</i>	139
	<i>Percentage TxWait</i>	139

<i>TxWait History Graphs</i>	139
<i>TxWait History in the OBFL Buffer</i>	142
Rx Credit Unavailability in Microseconds: RxWait	143
Continuous Tx Credit Unavailability in Milliseconds: Slowport-monitor	144
<i>Slowport-monitor Events in Real Time</i>	146
<i>Slowport-monitor History in OBFL</i>	147
Continuous Tx Credit Unavailability for 100 ms: Tx-credit-not- available	147
<i>Tx-credit-not-available in Real Time</i>	148
<i>Tx-credit-not-available History in the OBFL Buffer</i>	149
Differences Between TxWait, Slowport-monitor, and Tx-credit-not-available	150
When to Enable Slowport-monitor?	153
Continuous Rx Credit Unavailability for 100 ms: Rx-credit-not-available	155
Timeout Discards and Timeout-Drops	155
Tx Credit Loss Recovery	158
Link Failure: Link Reset Failed Nonempty Recv Queue (LR Rcvd B2B)	160
Credits and Remaining Credits	162
Credit Transition to Zero	163
Link Utilization	165
<i>Tx-datarate</i>	166
<i>Tx-datarate-burst</i>	167
<i>Rx-datarate</i>	167
<i>Rx-datarate-burst</i>	168
Bit Errors	168
Automatic Alerting	168
Port-Monitor on Cisco MDS Switches	168
<i>Port-Monitor Policy Types</i>	169
<i>Port-Monitor Policy Parameters</i>	169
<i>Port-Monitor Counters</i>	170
Detecting Congestion Using Remote Monitoring Platforms	177
NDFC Congestion/Slow-Drain Analysis	178
The MDS Traffic Monitoring (MTM) App	180
<i>MTM Architecture</i>	180
<i>MTM Use Cases</i>	181

	Metric Export Mechanisms	185
	<i>Parsing the Command-Line Output over SSH</i>	185
	<i>Simple Network Management Protocol (SNMP)</i>	185
	<i>Application Programming Interfaces (APIs)</i>	186
	<i>Streaming Telemetry</i>	187
	<i>Recommendations</i>	187
	The Pitfalls of Monitoring Network Traffic	189
	<i>Percentage Utilization of Fibre Channel Ports</i>	189
	<i>Average and Peak Utilization</i>	189
	Detecting Congestion Due to Slow Drain and Overutilization	192
	Slow Drain and Overutilization at the Same Time	194
	Detecting Congestion on long-distance links	195
	Summary	195
	References	196
Chapter 4	Troubleshooting Congestion in Fibre Channel Fabrics	199
	Troubleshooting Methodology and Workflow	199
	Congestion Severities and Levels	200
	<i>Mild Congestion (Level 1 and Level 1.5)</i>	200
	<i>Moderate Congestion (Level 2)</i>	201
	<i>Severe Congestion (Level 3)</i>	202
	Goals of Troubleshooting	202
	<i>Identifying the Source (Culprits) and Cause of Congestion</i>	202
	<i>Identifying the Affected Devices (Victims)</i>	203
	Methodology	205
	<i>Step 1: Troubleshooting Congestion in Decreasing Severity Levels</i>	205
	<i>Step 2: Chasing the Source of Congestion (Culprit)</i>	206
	Hints and Tips for Troubleshooting Congestion	214
	Investigating Higher Congestion Levels First	214
	<i>Finding Level 3 Congestion: Credit Loss</i>	214
	<i>Finding Level 2 Congestion: Frame Drops</i>	215
	<i>Finding Level 1/1.5 Congestion: TxWait and Overutilization</i>	216
	Using the show tech-support slowdrain Command	217
	Synchronizing Clocks and Considering Timing	217
	Timeout-Drop Anomaly	218
	Enabling and Using Automatic Alerting	219
	Using a Remote Monitoring Platform (NDFC/DCNM)	219

Cisco MDS NX-OS Commands for Troubleshooting Congestion	219
The show interface Command	220
The show interface counters [detailed] Command	222
The show interface txwait-history and rxwait-history Commands	225
The OBFL Commands: show logging onboard	226
<i>TxWait</i>	227
<i>RxWait</i>	227
<i>Error Statistics</i>	227
<i>Flow Congestion Drops</i>	234
Generic Troubleshooting Commands	234
<i>The show topology Command</i>	235
<i>The show flogi database Command</i>	235
<i>The show fcns database Command</i>	236
<i>The show zone member Command</i>	236
<i>The show zone name Command</i>	236
<i>The show zoneset active Command</i>	237
<i>The show fcs Ie Command</i>	237
<i>The show fcdomain Command</i>	237
<i>The show fspf database Command</i>	238
<i>The show rdp Command</i>	238
<i>The show fdmi database Command</i>	240
System Messages: show logging log	241
“ <i>Link failure Link Reset failed nonempty recv queue</i> ” System Message	241
“ <i>Link failure Link reset failed due to timeout</i> ” System Message	241
“ <i>TCP conn. closed - retransmit failure</i> ” System Message	242
Case Study 1: Finding Congestion Culprits and Victims in a Single-Switch Fabric	242
Fabric A Analysis	244
<i>Loss of Information Due to Clearing the OBFL Counters</i>	247
<i>TxWait Analysis</i>	248
<i>Traffic Utilization (Tx-datarate) Analysis</i>	249
<i>Graphical Correlation of Congestion Symptoms</i>	251
Fabric B Analysis	253
Culprit Analysis	254
Victim Analysis	255

<i>Direct Victims</i>	255
<i>Same-Path Victims</i>	267
<i>Indirect Victims</i>	267
Case Study 1 Summary	270
Case Study 2: Credit Loss Recovery Causing Frame Drops	271
Initial Investigation	272
Fabric A Analysis	273
<i>Edge Switch Fab_A_MDS_9396T_14</i>	275
<i>Core Switch Fab_A_MDS_9718_01</i>	276
<i>Core Switch Fab_A_MDS_9718_02</i>	278
<i>Fabric A Conclusion</i>	279
Fabric B Analysis	279
<i>Edge Switch Fab_B MDS_9396T_14</i>	279
<i>Core Switch Fab_B MDS_9718_01</i>	286
<i>Core Switch Fab_B MDS_9718_02</i>	287
<i>Fabric B Conclusion</i>	290
Culprit Analysis	290
Victim Analysis	292
<i>Direct Victims</i>	292
<i>Same-Path Victims</i>	294
<i>Indirect Victims</i>	294
Case Study 2 Summary	296
Case Study 3: Overutilization on a Single Device Causing Massive Congestion Problems	297
Level 3	298
Level 2	298
<i>MDS_9513_03</i>	299
<i>MDS_9710_03</i>	303
<i>MDS_9710_01</i>	308
<i>MDS_9513_01</i>	312
Culprit Analysis	318
Victim Analysis	318
<i>Direct Victims</i>	319
<i>Same-Path Victims</i>	321
<i>Indirect Victims</i>	321
Case Study 3 Summary	321

Case Study 4: Long-Distance ISLs Causing Congestion	323
Level 3	323
Level 2	324
Level 1.5	324
MDS_9148S_01	324
MDS_9148S_02	326
MDS_9148S_03	326
Culprit Analysis	334
Victim Analysis	334
Case Study 4 Summary	336
Summary	336
References	337
Chapter 5 Solving Congestion with Storage I/O Performance Monitoring	339
Why Monitor Storage I/O Performance?	339
How and Where to Monitor Storage I/O Performance	340
Storage I/O Performance Monitoring in the Host	340
Storage I/O Performance Monitoring in a Storage Array	341
Storage I/O Performance Monitoring in a Network	342
Cisco SAN Analytics Architecture	344
Traffic Inspection	344
Metric Calculation	345
Metric Export	345
Understanding I/O Flows in a Storage Network	347
I/O Flows in Fibre Channel Fabrics	347
I/O Flows Versus I/O Operations	350
I/O Flow Metrics	350
Latency Metrics	351
<i>Exchange Completion Time</i>	352
<i>Data Access Latency</i>	352
<i>Host Response Latency</i>	353
<i>Using Latency Metrics</i>	353
<i>The Location for Measuring Latency Metrics</i>	354
Performance Metrics	355
<i>I/O Operations per Second (IOPS)</i>	355
<i>I/O Size</i>	355

<i>Throughput</i>	357
<i>Outstanding I/O</i>	357
I/O Operations and Network Traffic Patterns	358
Read I/O Operation in a Fibre Channel Fabric	358
Write I/O Operation in a Fibre Channel Fabric	359
Network Traffic Direction	360
Network Traffic Throughput	362
Correlating I/O Operations, Traffic Patterns, and Network Congestion	363
Case Study 1: A Trading Company That Predicted Congestion Issues Using SAN Analytics	365
<i>Background</i>	365
<i>Initial Investigation: Finding the Cause and Source of Congestion</i>	366
<i>A Better Host Upgrade Plan</i>	366
<i>Case Study 1 Summary</i>	369
Case Study 2: A University That Avoided Congestion Issues by Correcting Multipathing Misconfiguration	369
<i>Background</i>	369
<i>Investigation</i>	369
<i>Case Study 2 Summary</i>	371
Case Study 3: An Energy Company That Eliminated Congestion Issues	371
<i>Background</i>	372
<i>Investigation</i>	372
<i>Case Study 3 Summary</i>	376
Case Study 4: A Bank That Eliminated Congestion Through Infrastructure Optimization	376
<i>Background</i>	376
<i>Investigation</i>	377
<i>Case Study 4 Summary</i>	379
Summary	379
References	379
Chapter 6 Preventing Congestion in Fibre Channel Fabrics	381
An Overview of Eliminating or Reducing Congestion	382
Defining the Outcome of an Approach	384
Manual Versus Automatic Approaches	385

Link Capacity	386
Congestion Recovery by Disconnecting the Culprit Device	387
Considerations for Disconnecting a Culprit	387
How to Disconnect?	388
Congestion Recovery by Dropping Frames	388
Dropping Frames Based on Their Age in the Switch	389
<i>Configuring Congestion-Drop Timeout on Cisco MDS Switches</i>	389
<i>Details on Congestion-Drop Timeout</i>	389
Dropping Frames Based on Slow Drain on an Edge Port	391
<i>Enabling No-Credit-Drop Timeout on Cisco MDS Switches</i>	393
<i>Details on No-Credit-Drop Timeout</i>	393
<i>No-Credit-Drop Timeout in Action</i>	394
<i>Finding the Optimum No-Credit-Drop Timeout Value</i>	397
Traffic Segregation	398
Categorizing Traffic for Segregation	400
Traffic Segregation to Dedicated ISLs	400
<i>Using VSANs for Traffic Segregation on Dedicated ISLs</i>	401
<i>Considerations for Traffic Segregation to Dedicated ISLs Using Multiple VSANs</i>	405
Case Study 1: A Bank That Avoided Congestion with Traffic Segregation	406
<i>Background and Investigation</i>	407
<i>Solution: Traffic Segregation to Dedicated ISLs</i>	408
<i>Case Study 1 Summary</i>	410
Traffic Segregation Using Virtual Links	410
<i>Understanding Virtual Links</i>	410
<i>Flow Control in a Virtual Link</i>	411
<i>Congestion Segregation Using Virtual Links</i>	412
<i>Scope of Congestion Segregation Using Virtual Links</i>	414
<i>Extending Virtual Links to the End Devices</i>	416
<i>Enabling Virtual Links on ISLs on Cisco MDS Switches</i>	416
<i>Traffic Assignment to Virtual Links</i>	417
<i>Automatic Assignment of Traffic to Virtual Links: Congestion Isolation</i>	418
<i>Manual Assignment of Traffic to Virtual Links</i>	423
<i>Comparing No-Credit-Drop Timeout with Congestion Isolation</i>	424

<i>No-Credit-Drop Timeout and Congestion Isolation in Action</i>	425
<i>Too Many VLs: The Hidden Side Effects</i>	431
Traffic Segregation Considerations	432
<i>Comparing Traffic Segregation Using VSANs and Virtual Links</i>	432
<i>Congestion Segregation Using Virtual Links: Caution</i>	432
Congestion Prevention Using Rate Limiters on Storage Arrays	433
Congestion Prevention Using Dynamic Ingress Rate Limiting on Switches	436
How DIRL Prevents Congestion	436
<i>How DIRL Prevents Congestion Due to Overutilization</i>	436
<i>How DIRL Prevents Congestion Due to Slow Drain</i>	437
<i>Details of DIRL</i>	437
Benefits of DIRL	439
Enabling and Using DIRL on Cisco MDS Switches	439
Enable FPM	440
Configure Port-Monitor	440
DIRL in Action	441
Test Setup	441
<i>Scenario 1: Congestion Due to Slow Drain Without Spreading</i>	443
<i>Scenario 2: Congestion Due to Slow Drain with Spreading</i>	444
<i>Scenario 3: Preventing Congestion Due to Slow Drain Using DIRL</i>	444
<i>Scenario 4: Preventing Congestion Due to Overutilization Using DIRL</i>	450
Comparing DIRL with Other Approaches	455
<i>DIRL Versus No-Credit-Drop Timeout</i>	455
<i>DIRL Versus Traffic Segregation Using Virtual Links</i>	456
Preventing Congestion by Notifying the End Devices	457
Readiness of Notifications and Signals in Fibre Channel	458
Notifications and Signals in Fibre Channel Fabrics	459
Register Diagnostic Functions	459
Exchange Diagnostic Capabilities	460
Fabric Performance Impact Notification (FPIN)	460
Congestion Signals	462
Examples of RDF, EDC, FPIN, and Congestion Signals	463
Comparing FPIN Frames and Congestion Signals	466
The Possible Results of FPIN Frames and Signals	466

	<i>Configuring Sending of FPIN Frames and Congestion Signals on Cisco MDS Switches</i>	467
	Using DIRM Versus Notifying the End Devices for Congestion Prevention	468
	Network Design Considerations	469
	Lowering the Link Speed of Storage Ports	470
	Edge-Core-Edge or Edge-Core or Collapsed-Core Design	471
	Increased Traffic Localization to a Single Switch	473
	Splitting Large Fabrics into Smaller Islands	474
	Summary	475
	References	476
Chapter 7	Congestion Management in Ethernet Storage Networks	479
	Ethernet Flow Control	479
	How Ethernet Flow Control Works	480
	<i>Pause Time</i>	480
	<i>When Are Pause Frames Sent?</i>	481
	<i>Ingress and Egress Queues</i>	483
	<i>Location of Ingress No-Drop Queues</i>	484
	<i>Number of Ingress No-Drop Queues per Port</i>	484
	<i>Implementation Differences and the Scope of This Book</i>	484
	<i>Pause Threshold and Resume Threshold</i>	485
	Ethernet Pause Frames Compared with Fibre Channel B2B Credits	495
	Priority Flow Control	496
	<i>Mapping Traffic Classes to the Pause Frame Class Enable Vector Field</i>	497
	<i>Layer 2 Priority Flow Control</i>	498
	<i>Layer 3 Priority Flow Control</i>	499
	Converged Ethernet Networks	503
	Configuring Lossless Ethernet	503
	Dedicated and Converged Ethernet Network	505
	Understanding Congestion in Lossless Ethernet Networks	506
	Slow Drain in Lossless Ethernet Networks	506
	Overutilization of a Link in Lossless Ethernet Networks	506
	Bit Errors	506
	Congestion Spreading in a Single-Switch Lossless Ethernet Network	507

Congestion Spreading in an Edge–Core Lossless Ethernet Network	508
Congestion Spreading in a Lossless Spine–Leaf Network	508
<i>Slow Drain in a Lossless Ethernet Spine–Leaf Network</i>	510
<i>Overutilization of a Host–Edge Link in a Lossless Ethernet Spine–Leaf Network</i>	510
<i>Comparing Congestion Due to Slow Drain and Overutilization in a Lossless Ethernet Spine–Leaf Network</i>	510
Detecting Congestion in Lossless Ethernet Networks	511
Congestion Direction: Ingress or Egress	511
Congestion Detection Metrics	512
<i>Duration of Traffic Pause: TxWait and RxWait</i>	513
<i>The Number of Pause Frames</i>	516
<i>Frame Drops or Discards</i>	519
<i>Bit Errors</i>	520
<i>Link Utilization</i>	522
<i>PFC Storms</i>	524
Storage I/O Performance Monitoring	527
<i>UDP Flow Monitoring Versus I/O Flow Monitoring</i>	528
<i>Unavailability of I/O Flow Monitoring in Lossless Ethernet Networks</i>	528
<i>Alternative Approaches</i>	528
<i>FCoE I/O Operations</i>	529
<i>RoCE I/O Operations</i>	529
<i>Correlating I/O Operations, Traffic Patterns, and Network Congestion</i>	531
Detecting Congestion on a Remote Monitoring Platform	531
<i>Congestion Detection Using Cisco Nexus Dashboard Insights</i>	531
<i>Metric Export Mechanisms</i>	532
Troubleshooting Congestion in Lossless Ethernet Networks	534
Goals	535
Congestion Severities and Levels	535
Methodology	536
Troubleshooting Congestion in Spine–Leaf Topology	536
Reality Check	537
Troubleshooting Congestion by Using a Remote Monitoring Platform	538
<i>Comparative Analysis</i>	538

<i>Trends and Seasonality</i>	539
<i>Monitoring a Slow-Drain Suspect</i>	539
<i>Monitoring an Overutilization Suspect</i>	540
FC and FCoE in the Same Network	540
<i>Congestion Spreading Due to Slow Drain</i>	541
<i>Congestion Spreading Due to Overutilization</i>	541
<i>Bit Rate Differences Between FC and FCoE</i>	543
Multiple No-Drop Classes on the Same Link	543
Bandwidth Allocation Between Lossless and Lossy Traffic	544
<i>The Effect of Lossy Traffic on the No-Drop Class</i>	545
<i>Case Study 1: An Online Gaming Company</i>	545
<i>Case Study 2: Converged Versus Dedicated Storage Network</i>	547
Preventing Congestion in Lossless Ethernet Networks	547
Eliminating or Reducing Congestion: An Overview	547
Congestion Recovery by Dropping Frames	549
<i>Dropping Frames Based on Their Age in the Switch</i>	549
<i>Dropping Frames Based on Slow Drain on an Edge Port</i>	549
Congestion Notification in Routed Lossless Ethernet Networks	556
<i>Solution Components</i>	556
<i>RoCEv2 Transport Overview</i>	557
<i>RoCEv2 Congestion Management</i>	557
<i>RoCEv2 Congestion Management Considerations</i>	559
<i>PFC and ECN</i>	561
Lossless Traffic with VXLAN	565
VXLAN Overview	565
VXLAN Transport	565
Physical Topology	566
MAC Address Learning	566
Lossless Traffic over VXLAN	566
VXLAN Encapsulation	567
VXLAN Decapsulation	567
Congestion Notification over VXLAN	567
Flow Control and Congestion Notification with VXLAN	568
Congestion Management in VXLAN	569
Summary	569
References	570

Chapter 8 Congestion Management in TCP Storage Networks 573

Understanding Congestion in TCP Storage Networks	574
Comparison with Lossless Networks	574
How iSCSI and NVMe/TCP Exchange Data	575
<i>Bit Errors in Lossy Ethernet Networks with TCP Transport</i>	578
<i>How TCP Provides Reliable Data Transfer</i>	579
<i>TCP Flow Control</i>	581
<i>TCP Congestion Control</i>	582
Congestion in TCP Storage Networks	585
<i>Congestion Due to Overutilization of the Host Link</i>	585
<i>Congestion Within the Host</i>	586
Storage I/O Performance Monitoring	587
TCP Flow Monitoring Versus I/O Flow Monitoring	588
<i>Unavailability of I/O Flow Monitoring in TCP Storage Networks</i>	588
<i>Alternative Approaches</i>	589
iSCSI I/O Operations	589
NVMe/TCP I/O Operations	591
Correlating I/O Operations, Traffic Patterns, and Network Congestion	594
Comparison with Lossless Networks	594
Estimating I/O Flow Performance from TCP Flow Performance	594
IP MTU and TCP MSS Considerations	595
<i>The Number of Packets for an I/O Operation</i>	596
<i>Packet Fragmentation</i>	596
<i>Comparison with Lossless Networks</i>	596
Preventing Congestion in TCP Storage Networks	597
Eliminating or Reducing Congestion: An Overview	597
Congestion Notification in TCP Storage Networks	599
<i>Solution Components</i>	599
<i>Explicit Congestion Notification in TCP/IP Networks</i>	600
<i>Comparison with RoCEv2 Networks</i>	601
<i>Comparison with Fibre Channel Fabrics</i>	602
<i>ECN Considerations for Block-Storage Traffic</i>	602
Switch Buffer Management	604
<i>Queue Utilization</i>	604
<i>Queue Utilization Considerations</i>	606

<i>User Actions</i>	608
Comparison with Lossless Ethernet	609
Comparison with Fibre Channel Fabrics	610
Active Queue Management	610
<i>Tail Drop</i>	610
<i>Random Early Detect (RED)</i>	611
<i>Weighted Random Early Detection (WRED)</i>	611
<i>Approximate Fair Dropping (AFD)</i>	612
<i>Dynamic Packet Prioritization (DPP)</i>	614
Detecting Congestion in TCP Storage Networks	615
Source of Congestion Within the End Devices	616
<i>Congestion Detection Notes</i>	616
<i>Comparison with Lossless Networks</i>	616
The Source of Congestion Within the Network	617
<i>Packet Drops or Discards</i>	617
<i>ECN Counters</i>	617
<i>Link Utilization</i>	619
<i>Queue Depth Monitoring and Microburst Detection</i>	620
<i>Bit Errors</i>	623
Detecting Congestion Using a Remote Monitoring Platform	623
<i>Comparative Analysis</i>	623
<i>Trends and Seasonality</i>	624
Congestion Detection Using Cisco Nexus Dashboard Insights	624
Metric Export Mechanisms	625
Troubleshooting Congestion in TCP Storage Networks	625
Goals	625
Congestion Severities and Levels	626
Methodology	626
Load Balancing in TCP Storage Networks	627
QoS Considerations for Dedicated and Shared Storage Networks	628
<i>The Effect of Other Traffic Classes on Storage Traffic Class</i>	628
<i>Configuring Versus Operating a Shared Storage Network</i>	629
<i>QoS Expertise</i>	629
FCoE, RoCE, iSCSI, and NVMe/TCP in the Same Network	629
iSCSI and NVMe/TCP in a Lossless Network	630

iSCSI and NVMe/TCP with VXLAN	631
Fibre Channel over TCP/IP (FCIP)	631
TCP Optimizations for Storage Traffic on Cisco FCIP Switches	631
Detecting Congestion on FCIP Links	633
Modified TCP Implementations	637
Summary	638
References	639

Chapter 9 Congestion Management in Cisco UCS Servers 641

Cisco UCS Architecture	641
UCS Domain	642
Traffic Flow in a UCS Domain	642
Flow Control in a UCS Domain	644
Understanding Congestion in a UCS Domain	644
Detecting Congestion in a UCS Domain	645
Ingress Congestion	645
Egress Congestion	646
Congestion Between FI Server Ports and IOM/FEX Fabric Ports	646
UCS Congestion Detection Notes	646
The UCS Traffic Monitoring (UTM) App	648
The Journey of UTM	649
Getting Started with UTM	650
UTM Architecture	650
An Overview of Using UTM	650
Troubleshooting Congestion Using UTM	651
Congestion Troubleshooting Workflow in UTM	651
<i>Proactively Detecting Congestion Due to Slow Drain</i>	653
<i>Proactively Detecting Congestion Due to Overutilization</i>	655
Case Study 1: Finding the Cause and Source of Congestion in a UCS Domain	657
<i>Background</i>	657
<i>Investigation</i>	658
<i>Conclusion</i>	661
<i>Solution</i>	661
<i>Case Study 1 Summary</i>	662
Case Study 2: Congestion Due to Slow Drain on the Backplane Port	662
<i>Investigation</i>	662

Conclusions 663
Case Study 2 Summary 664
Case Study 3: Non-Uniform Utilization of FI Uplink Ports 665
Investigation 665
Conclusion 666
Solution 666
Case Study 3 Summary 667
Case Study 4: Congestion Due to Multipathing I/O Imbalance 667
Investigation 667
Conclusion 668
Solution 668
Case Study 4 Summary 668
Summary 668
References 669
Index 671

Sample pages

Solving Congestion with Storage I/O Performance Monitoring

This chapter explains the use of storage I/O performance monitoring for handling network congestion problems.

This chapter covers the following topics:

- Why Monitor Storage I/O Performance?
- How and Where to Monitor Storage I/O Performance.
- Cisco SAN Analytics Architecture
- Understanding I/O Flows in a Storage Network
- I/O Flow Metrics
- I/O Operations and Network Traffic Patterns
- Case studies

Why Monitor Storage I/O Performance?

Storage I/O performance monitoring provides advanced insights into network traffic, which can then be used to accurately address network congestion. This information is in addition to what the network ports already provide by counting the number of packets sent and received, the number of bytes sent and received, and link errors. In addition, storage I/O performance monitoring brings visibility to the upper layers of the stack and can explain why a network has or lacks traffic by providing the following information:

- The upper-layer protocol—SCSI or NVMe—that generated the network traffic
- Upper-layer protocol errors such as SCSI queue full, reservation conflict, NVMe namespace not ready, and so on

- IOPS, throughput, I/O size, and so on
- How long I/O operations take to complete, the delay caused by storage arrays, and the delay caused by hosts

This performance can also be monitored for every flow, giving granular insights into the traffic on a network port. This flow-level performance monitoring is extremely useful because most production environments are virtualized. When a host causes congestion due to overutilization of its link, the network can detect this condition, as explained in earlier chapters. In addition, storage I/O performance monitoring can detect the cause of the high amount of traffic and which virtual machine (VM) is asking for it.

Likewise, when a host causes congestion due to slow drain, investigating the SCSI- and NVMe-level performance and error metrics can explain why the host has become slower in processing the traffic. It is also possible to determine whether a particular VM has caused the entire host to slow down. In addition, storage I/O performance monitoring can also predict the likeliness of network congestion. These and many more benefits of storage I/O performance monitoring are explained in this chapter, and case studies are provided.

Storage I/O performance monitoring is a detailed subject. Its use cases involve application and storage performance insights, storage provisioning recommendations, infrastructure optimization, change management, audits, reporting, and so on. The scope of this book, however, is limited only to congestion use cases. We recommend continuing your education on this topic beyond this book. Refer to the References section later in this chapter.

This chapter focuses on the SCSI and NVMe protocols in the block-storage stack for performance monitoring. But these protocols initiate I/O operations only when an application wants them to read or write data. Therefore, monitoring higher layers in the stack, up to the application layer, can provide even more insights into why the network has traffic. Application-level monitoring, however—such as that provided by the Cisco AppDynamics observability platform—is beyond the scope of this book. This is another area that we recommend to continue your education outside this book.

How and Where to Monitor Storage I/O Performance

At a high level, storage I/O performance can be monitored within a host, in storage arrays, or in a network. These are three viable options because an I/O operation passes through many layers within the initiator (host), the target (storage array), and multiple switches in the network. This section explains these approaches briefly, but the primary focus of this chapter is on monitoring storage I/O performance in the network.

Storage I/O Performance Monitoring in the Host

Most operating systems, such as Linux, Windows, and ESXi, monitor storage I/O performance. Example 5-1 shows an example of monitoring storage I/O performance in Linux by using the `iostat` command.

Example 5-1 *Storage I/O Performance Monitoring in Linux*

```
[root@stg-tme-lnx-b200-7 ~]# iotop
```

TID	PRIO	USER	DISK READ	DISK WRITE	SWAPIN	IO>	COMMAND
Total DISK READ :		36.30 M/s		Total DISK WRITE :		36.85 M/s	
Actual DISK READ:		36.31 M/s		Actual DISK WRITE:		36.80 M/s	
941	be/3	root	0.00 B/s	0.00 B/s	0.00 %	3.31 %	[jbd2/dm-101-8]
46303	be/4	root	6.42 M/s	6.37 M/s	0.00 %	1.93 %	fiio config_fio_1
542	be/3	root	0.00 B/s	0.00 B/s	0.00 %	1.89 %	[jbd2/dm-22-8]
26496	rt/4	root	0.00 B/s	0.00 B/s	0.00 %	1.26 %	multipathd
46383	be/4	root	7.13 M/s	7.11 M/s	0.00 %	0.42 %	fiio config_fio_1
46284	be/4	root	11.96 M/s	12.34 M/s	0.00 %	0.00 %	fiio config_fio_1
46384	be/4	root	5.19 M/s	5.40 M/s	0.00 %	0.00 %	fiio config_fio_1
46402	be/4	root	5.61 M/s	5.63 M/s	0.00 %	0.00 %	fiio config_fio_1

For the purpose of dealing with network congestion, monitoring storage I/O performance within hosts involves the following considerations:

- Per-path storage I/O performance should be monitored because although multiple paths that perform at different levels exist between the host and the storage array, the host may, by default, report only cumulative performance.
- Metrics from thousands of hosts should be collected and presented in a single dashboard for early detection of congestion.
- Collecting the metrics from hosts may require dedicated agents, and there is overhead involved in maintaining them.
- Different implementations on different operating systems, such as Linux, Windows, and ESXi, may take non-uniform approaches to collecting the same metrics.
- Be aware that measuring the performance within hosts makes the measurements prone to issues on a particular host. Is the “monitored” end device “monitoring” itself? What happens when it gets congested or becomes a slow-drain device?
- Because of organizational silos, hosts and storage arrays may be managed by different teams.

Storage I/O Performance Monitoring in a Storage Array

Most arrays monitor storage I/O performance. For example, Figure 5-1 shows I/O performance on a Dell EMC PowerMax storage array.

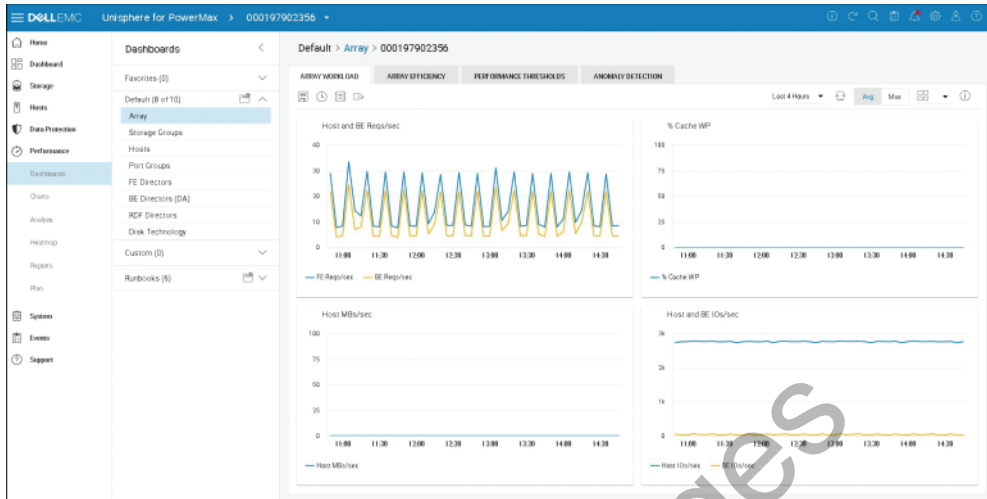


Figure 5-1 Storage I/O Performance Monitoring on a Dell EMC PowerMax Storage Array

The metrics collected by the BE storage arrays can be used for monitoring I/O performance, but this approach involves similar challenges to the host-centric approach, as explained in the previous section.

Storage I/O Performance Monitoring in a Network

I/O operations are encapsulated within frames for transporting the frames via a storage network. The network switches only need to look up the headers to send the frames toward their destination. In other words, a network, for its typical function of frame forwarding, need not know what's inside the frame. However, monitoring storage I/O performance in the network requires advanced capability on the switches for inspecting the transport (such as Fibre Channel) header, and upper-layer protocol (such as SCSI and NVMe) headers.

Cisco SAN Analytics monitors storage I/O performance natively within a network because it is integrated by design with Cisco MDS switches. As Fibre Channel frames are switched between the ports of an MDS switch, the ASICs (application-specific integrated circuits) inspect the FC and NVMe/SCSI headers and analyze them to collect I/O performance metrics such as the number of I/O operations per second, how long the I/O operations are taking to complete, how long the I/O operations are spending in the storage array, how long the I/O operations are spending in the hosts, and so on. Cisco SAN Analytics does not inspect the frame payload because there is no need for it, as the metrics can be calculated by inspecting only the headers.

Cisco SAN Analytics, because of its network-centric approach and unique architecture, has the following merits for monitoring storage I/O performance:

- **Vendor neutral:** Cisco SAN Analytics is not dependent on server vendor (HPE, Cisco, Dell, and so on), host OS vendor (Red Hat, Microsoft, VMware, and so on), or storage array vendor (Dell EMC, HPE, IBM, Hitachi, Pure, NetApp, and so on).
- **Not dependent on end-device type:** Cisco SAN Analytics is not dependent on any of the following:
 - **Server architecture:** Rack-mount, blade, and so on
 - **OS type:** Linux, Windows, or ESXi
 - **Storage architecture:** All-flash, hybrid, non-flash, and so on

Legacy end devices can also benefit because no changes are needed on them, such as installation of an agent or firmware updates.

- **No dependency on the monitoring architecture of end devices:** Different products use different logic for collecting similar metrics. For example, some storage arrays collect I/O completion time on the front-end ports, whereas other storage arrays collect it on the back-end ports. Different host operating systems may collect I/O completion time at different layers in the host stack. Cisco SAN Analytics doesn't have this dependency.
- **Flow-level monitoring:** Cisco SAN Analytics monitors performance for every flow separately. When a culprit switchport is detected, flow-level metrics help in pinpointing the issue to an exact initiator, target, virtual machine, or LUN/namespace ID.
- **Flexibility of location of monitoring:** Cisco SAN Analytics can monitor storage I/O performance at any of the following locations:
 - **Host-connected switchports:** Close to apps and servers
 - **Storage-connected switchports:** Close to storage arrays
 - **ISL ports:** Flow-level granularity in the core of the network
- **Granular:** Cisco SAN Analytics monitors storage I/O performance at a low granularity—microseconds for on-switch monitoring and seconds for exporting metrics from the switch.

This chapter focuses on using Cisco SAN Analytics for addressing congestion in storage networks, although the education and case studies can be used with host-centric and storage array-centric approaches as well.

Cisco SAN Analytics Architecture

Cisco SAN Analytics architecture can be divided into three components (see Figure 5-2):

- Traffic inspection by ASICs on Cisco MDS switches
- Metric calculation by an onboard network processing unit (NPU) or by the ASIC
- Streaming of flow metrics to an external analytics and visualization engine for end-to-end visibility

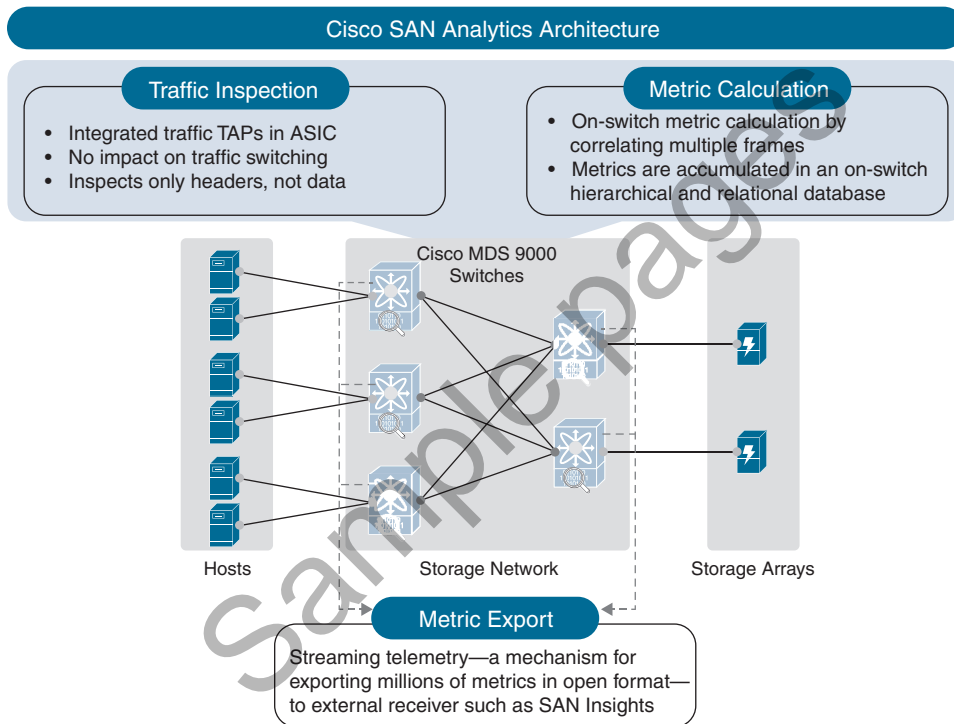


Figure 5-2 *Cisco SAN Analytics Architecture*

Traffic Inspection

Traffic inspection is integrated by design into Fibre Channel ASICs. In addition to switching the frames between the switchports, these ASICs can inspect the traffic in ingress and egress directions without any performance or feature penalty. In other words, traffic access points (TAPs) are built into the ASICs.

This approach is secure because the ASICs inspect only the Fibre Channel and SCSI/NVMe headers of the relevant frames. The frame payload (application data) is not inspected.

These ASICs are custom designed by Cisco, and they are exclusively used in MDS switches. Cisco Nexus switches and UCS fabric interconnects, despite supporting FC ports on selective models, use a different ASIC and thus don't offer SAN Analytics.

Metric Calculation

After inspecting the frame headers, Cisco MDS switches calculate the metrics by correlating multiple frames with common attributes, such as frames belonging to the same I/O operation and frames belonging to the same flow.

The metric calculation logic in the 32 Gbps MDS switches resides in an onboard network processing unit (NPU), which is a powerful packet processor. In 64 Gbps MDS switches, the metric calculation logic resides within the ASIC itself, although the NPU continues to exist on the switches. Regardless of this architectural detail, the overall metric calculation logic remains the same.

Cisco MDS switches accumulate the metrics in a hierarchical and relational database for on-switch visibility or export to a remote receiver.

Note At the time of this writing, Cisco SAN Analytics does not collect I/O flow metrics in FICON environments.

Metric Export

Cisco SAN Analytics is designed to inspect every flow that passes through a storage network in an always-on fashion. As a result, it collects millions of metrics per second. A traditional approach (such as SNMP) for exporting a large number of metrics may not work at this scale, and thus, Cisco introduced streaming telemetry for this purpose. In addition to being efficient, streaming telemetry exports metrics in open format, which simplifies third-party integrations.

The receiver of streaming telemetry can use I/O flow metrics from multiple switches to provide fabric-wide and end-to-end visibility into a single pane of glass for long-term metric retention, trending, correlation, predictions, and so on. SAN Insights is an example of such a receiver and is a feature in Cisco Nexus Dashboard Fabric Controller (NDFC), formerly known as Cisco Data Center Network Manager (DCNM). Figure 5-3 shows the SAN Insights dashboard, which provides many ready-made use cases, such as automatic learning, baselining, and deviation calculations for up to 1 million I/O flows per NDFC server as of release 12.1.2. This high scale gives visibility into issues anywhere in the fabric.

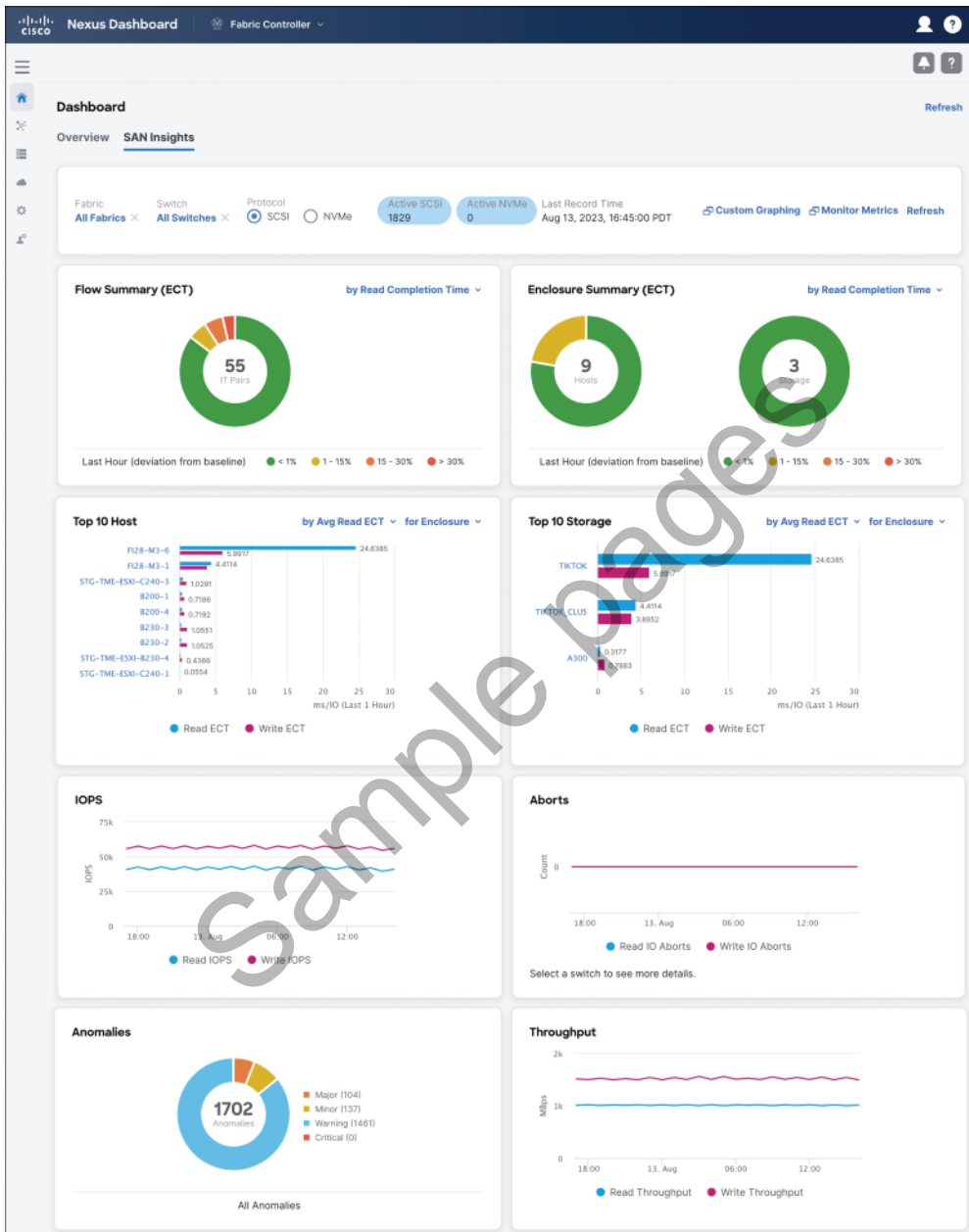


Figure 5-3 SAN Insights Dashboard in Cisco NDFC

Understanding I/O Flows in a Storage Network

Without considering I/O flows, a network is only aware of the frames in ingress and egress directions. Categorizing network traffic into I/O flows helps in correlating it with initiators, targets, and the logical unit number (LUN) for SCSI I/O operations and namespace ID (NSID) for NVMe I/O operations. In addition, storage performance can be monitored for every I/O flow individually to get detailed insights into the traffic. For example, when a switchport is 90% utilized, throughput per I/O flow can tell which initiator, target, and LUN/namespace are the top consumers.

I/O Flows in Fibre Channel Fabrics

The following can be the I/O flow types in a Fibre Channel fabric:

- **Port flow:** Traffic belonging to all the I/O operations that pass through a network port makes a port flow. It can be a SCSI port flow for SCSI traffic or an NVMe port flow for NVMe traffic.
- **VSAN flow:** A port of a Cisco Fibre Channel switch may carry traffic in one or more VSANs. Hence, a port flow can be further categorized into one or more VSAN flows.
- **Initiator flow:** Traffic belonging to all the I/O operations that are initiated by an initiator makes an initiator flow.
- **Target flow:** Traffic belonging to all the I/O operations that are destined for a target makes a target flow.
- **Initiator-target (IT) flow:** Traffic belonging to all the I/O operations between a pair of initiator and target makes an IT flow.
- **Initiator-target-LUN (ITL) flow:** Traffic belonging to all the I/O operations between an initiator, a target, and a logical unit makes an ITL flow. An ITL flow is applicable only for SCSI I/O operations.
- **Initiator-target-namespace (ITN) flow:** Traffic belonging to all the I/O operations between an initiator, a target, and a namespace makes an ITN flow. An ITN flow is applicable only for NVMe I/O operations.
- **Target-LUN (TL) flow:** Traffic belonging to all the I/O operations that are destined for a target port and a specific logical unit makes a TL flow. A TL flow is applicable only for SCSI I/O operations.
- **Target-namespace (TN) flow:** Traffic belonging to all the I/O operations that are destined to a target port and a specific namespace makes a TN flow. A TN flow is applicable only for NVMe I/O operations.

The definition of an I/O flow can also be extended to a virtual entity (VE), such as a virtual machine (VM) on the host. When combined with an ITL or ITN flow, the end-to-end flow becomes a VM-ITL flow or a VM-ITN flow. There are at least two approaches for achieving this visibility into the VMs.

The first approach needs support from hosts, and in some cases even from storage arrays, for tagging the VM identifier in the frame header. Although Cisco SAN Analytics on MDS switches supports VM-ITL and VM-ITN flows, because of the dependency on the end devices, most production deployments are not ready for it at the time of this writing.

The second approach uses the APIs from VMware vCenter to provide the correlation between the VM and the initiator and LUN (or namespace) from the ITL (or ITN) flow. The benefit of this approach, unlike the first approach, is that upgrading the end devices is not mandatory. Cisco SAN Insights uses this approach in NDFC 12.1.2 onward.

In environments where even the read-only access to VMware vCenter cannot be added to NDFC, this approach can still be used for manually correlating ITL or ITN flows with the VMs. The use of this approach is demonstrated further in the section “Case Study 3: An Energy Company That Eliminated Congestion Issues,” later in this chapter.

This chapter focuses only on ITL flows that are natively available on the Cisco MDS switches without any dependency on the end devices and NDFC. The environments with VM-ITL flows made available using either of the two approaches mentioned earlier can benefit by expanding ITL flows in the same way that port flows are expanded to IT flows and ITL flows.

To understand the I/O flows and how they help in gaining granular details about a network, consider the example in Figure 5-4. Two initiators, I-1 and I-2, connect to two targets, T-1, and T-2, via a fabric of Switch-1 and Switch-2. The ISL port on Switch-1 (Port-3) reports an ingress throughput of 800 MBps. After enabling SAN Analytics, Port-3 can categorize network traffic into multiple types of I/O flows and monitor the performance of every flow.

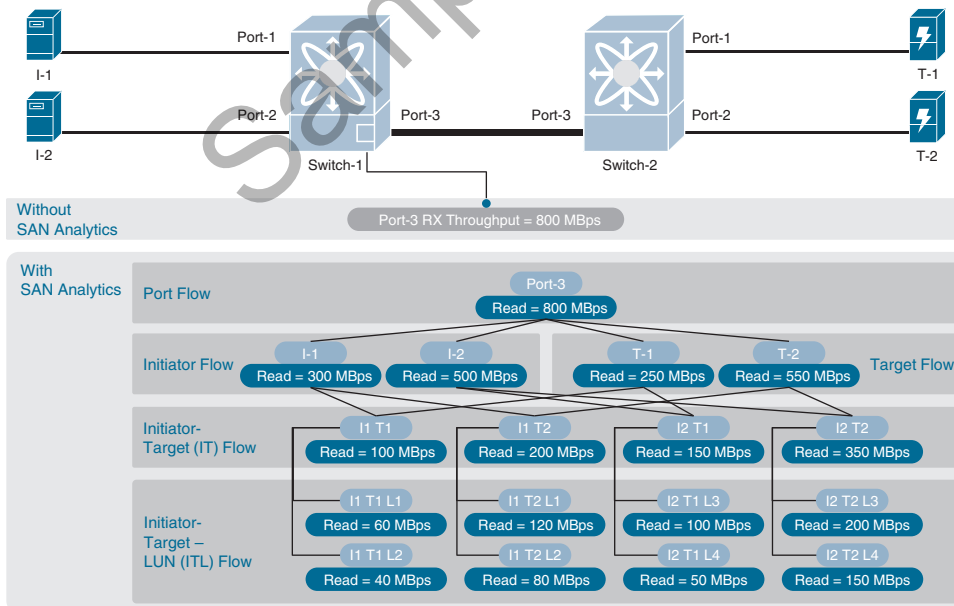


Figure 5-4 I/O Flows and Flow-Level Metrics Using Cisco SAN Analytics

SAN Analytics can find the following details:

- The 800 MBps throughput on Port-3 on Switch-1 is because of SCSI read I/O operations.
- Port-3 may have two VSANs: VSAN 100 and VSAN 200 (not shown in Figure 5-4). The VSAN flows provide a further breakdown of the port flow throughput, such as a read throughput of 600 MBps for VSAN 100 and a read throughput of 200 MBps for VSAN 200.
- I-1's read throughput via Port-3 is 300 MBps, whereas I-2's read throughput via Port-3 is 500 MBps.
- T-1's read throughput via Port-3 is 250 MBps, whereas T-2's read throughput via Port-3 is 550 MBps.
- Port-3 has four IT flows: I1-T1, I1-T2, I2-T1, and I2-T2. The read throughput for each is as follows:
 - I1-T1: 100 MBps
 - I1-T2: 200 MBps
 - I2-T1: 150 MBps
 - I2-T2: 350 MBps
- Port-3 has eight ITL flows. I-1 uses LUN-1 and LUN-2, whereas I-2 uses LUN-3 and LUN-4. The read throughput for each is as follows:
 - I1-T1-L1: 60 MBps
 - I1-T1-L2: 40 MBps
 - I1-T2-L1: 120 MBps
 - I1-T2-L2: 80 MBps
 - I2-T1-L3: 100 MBps
 - I2-T1-L4: 50 MBps
 - I2-T2-L3: 200 MBps
 - I2-T2-L4: 150 MBps

As is evident from this example, the hierarchical and relational definitions of I/O flows help create a precise breakdown of traffic on a switchport. During congestion, the per-flow metrics, such as throughput, help in pinpointing the root cause of the exact entity, such as initiator, target, LUN, or namespace. Without per-flow storage I/O performance monitoring, as provided by Cisco SAN Analytics, such detailed insights are not possible.

I/O Flows Versus I/O Operations

I/O flows shouldn't be confused with I/O operations. An I/O flow is identified by end-to-end tuples such as initiator, target, LUN, or namespace (ITL or ITN flows). In contrast, I/O operations transfer data within an I/O flow. For example, when Initiator-1 initiates 100 read I/O operations per second to LUN-1 on Target-1, the ITL flow is identified as Initiator-1–Target-1–LUN-1, whereas there were 100 I/O operations per second.

An I/O flow is created only after an initial exchange of I/O operations between the identifying tuples. Later, if the initiator doesn't read or write data, the I/O flows may still exist, but no I/O operations flow through it, which results in zero IOPS for these I/O flows.

I/O Flow Metrics

The I/O flow metrics collected by Cisco SAN Analytics can be classified into the following categories:

- **Flow identity metrics:** These metrics identify a flow, such as switchport, initiator, target, LUN, or namespace.
- **Metadata metrics:** The metadata metrics provide additional insights into the traffic. For example:
 - **VSAN count:** Number of VSANs carrying traffic on a switchport.
 - **Initiator count:** Number of initiators exchanging I/O operations behind a switchport.
 - **Target count:** Number of targets exchanging I/O operations behind a switchport.
 - **IT flow count:** Number of pairs of initiators and targets exchanging I/O operations via a switchport.
 - **TL and TN flow count:** Number of pairs of targets and LUNs/namespaces behind a switchport exchanging I/O operations.
 - **ITL and ITN flow count:** Number of pairs of initiators, targets, and LUNs/namespaces exchanging I/O operations via a switchport.
- **Metric collection time:** Start time and the end time for I/O flow metrics during a specific export. This metric helps in knowing the precise duration when a metric was calculated at the link.
- **Latency metrics:** Latency metrics identify the total time taken to complete an I/O operation and the time taken to complete various steps of an I/O operation. For example:
 - **Exchange Completion Time (ECT):** Total time taken to complete an I/O operation.
 - **Data Access Latency (DAL):** Time taken by a target to send the first response to an I/O operation. DAL is one component of ECT that's caused by the target.

- **Host Response Latency (HRL):** Time taken by an initiator to send the response after learning that the target is ready to receive data for a write I/O operation. HRL is one component of ECT that's caused by the initiator.
- **Performance metrics:** These metrics measure the performance of I/O operations. For example:
 - **IOPS:** Number of read and write I/O operations completed per second.
 - **Throughput:** Amount of data transferred by read and write operations, in bytes per second.
 - **Outstanding I/O:** The number of read and write I/O operations that were initiated but are yet to be completed.
 - **I/O size:** The amount of data requested by a read or write I/O operation.
- **Error metrics:** The error metrics indicate errors in read and write I/O operations (for example, Aborts, Failures, Check condition, Busy condition, Reservation Conflict, Queue Full, LBA out of range, Not ready, and Capacity exceeded).

An exhaustive explanation of all these metrics is beyond the scope of this chapter. This chapter is just a starting point for using end-to-end I/O flow metrics in solving congestion and other storage performance issues.

Latency Metrics

Latency is a generic term to convey storage performance. But as Figure 5-5 and Figure 5-6 show, there are multiple latency metrics, each conveying a specific meaning. Latency metrics are measured in time (microseconds, milliseconds, and so on).

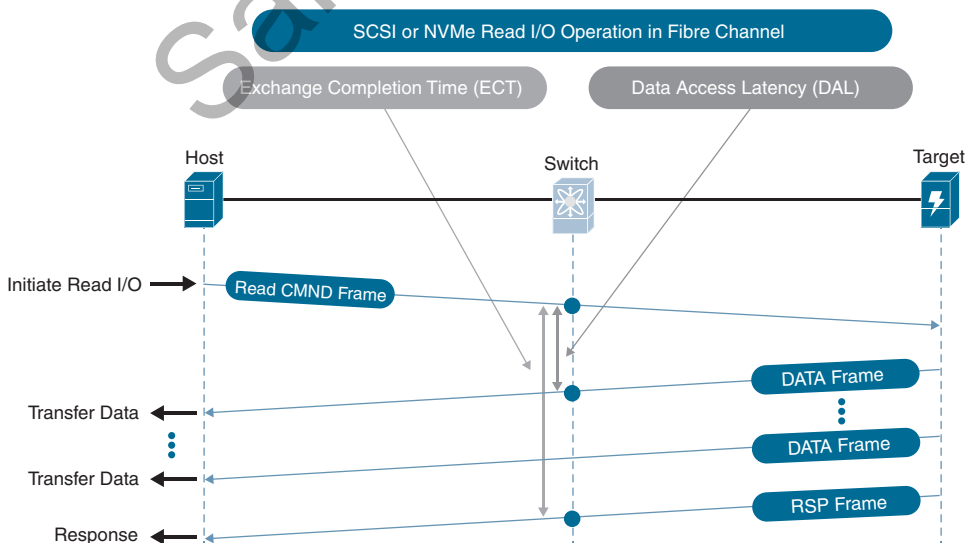


Figure 5-5 Latency Metrics for a Read I/O Operation

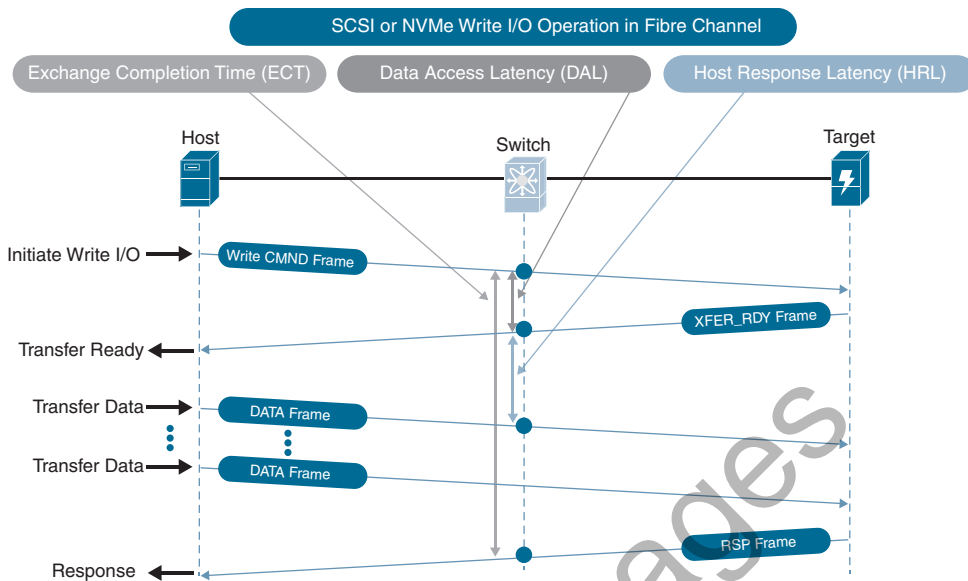


Figure 5-6 Latency Metrics for a Write I/O Operation

Exchange Completion Time

Exchange Completion Time (ECT) is the time taken to complete an I/O operation. It is a measure of the time difference between the command (CMND) frame and the response (RSP) frame. In Fibre Channel, an I/O operation is carried out by an exchange, and hence it's called Exchange Completion Time, but ECT can also be known as I/O completion time.

ECT is an overall measure of storage performance. In general, the lower the ECT, the better. This is because lower ECTs result in improved application performance.

At the same time, a direct correlation between ECT and application performance is not straightforward because it's dependent on the application I/O profile. In general, when application performance degrades and if ECT increases (degrades) at the same time, the reason for the performance degradation is the slower I/O performance.

Data Access Latency

Data Access Latency (DAL) is the time taken by a storage array in sending the first response after receiving a command (CMND) frame. For a read I/O operation, DAL is calculated as the time difference between the command (CMND) frame and the first-data (DATA) frame. For a write I/O operation, DAL is calculated as the time difference between the command (CMND) frame and the transfer-ready (XFER_RDY) frame.

When a target receives a read I/O operation, if the data requested is not in cache, the target must first read the data from the storage media, which takes time. The amount of time it takes to retrieve the data from the media depends on several factors, such as overall system utilization and the type of storage media being used. Likewise, when a

target receives a write I/O operation, it must process all the other operations ahead of this operation, which takes time. An increase in these time values leads to a large DAL.

In most cases, it's best to investigate DAL while troubleshooting higher ECT because DAL may tell why ECT increased. An increase in ECT and also in DAL indicates a slowdown within the storage array.

Host Response Latency

Host Response Latency (HRL), for a write I/O operation, is the time taken by a host in sending the data after receiving the transfer ready. It is calculated as the time difference between the transfer-ready frame and the first data frame.

Because read I/O operations do not have transfer ready, HRL is not calculated for them.

In most cases, it's best to investigate HRL while troubleshooting higher-write ECTs because HRL may tell why ECT increased. An increase in write ECT and also in HRL indicates a slowdown within the host.

Using Latency Metrics

The following are important details to remember about latency metrics, such as ECT, DAL, and HRL, when addressing congestion in a storage network:

- A good way of using ECT is to monitor it for a long duration and find any deviations from the baseline. For example, consider two applications with an average ECT of 200 μ s and 400 μ s over a week. The I/O flow path of the first application gets congested, resulting in an increased ECT of 400 μ s. At this moment, although both applications have the same ECT, only the first application may be degraded, while the second application remains unaffected, even though their ECT values are the same.
- ECT measures the overall storage performance, but it doesn't convey the source of the delay, which can be the host, network, or storage array. The delay caused by the host is measured by HRL, whereas the delay caused by the storage array is measured by DAL.
- The delay caused by the network may be the direct result of congestion. For example, when a host-connected switchport has high TxWait, the frames can't be delivered to it in a timely fashion. As a result, the time taken to complete the I/O operations (ECT) increases.
- Although an increase in TxWait (or a similar network congestion metric) increases ECT, the reverse may not be correct. ECT may increase even when the network isn't congested. ECT is an end-to-end metric. It may increase due to delays caused by hosts, network, or storage. The block I/O stack within a host involves multiple layers. Similarly, an I/O operation undergoes many steps within a storage array. The delay caused by any of these layers increases ECT.

- Network congestion is one of the reasons for higher ECT. However, it's not the only reason. Other network issues may increase ECT even without congestion (for example, network traffic flowing through suboptimal paths, long-distance links, or poorly designed networks).
- All latency metrics increase under network congestion. This increase is seen in all the I/O flows whose paths are affected by congestion.
- While considering dual fabrics with active/active multipath, if only one fabric is congested, only the I/Os using the congested fabric report increases in ECT. The average increase in the ECT as reported by the host may or may not show this difference, depending on how much ECT degrades. For example, consider an application that measures I/O completion time (ECT) as 200 μ s. The application accesses storage via Fabric-A and Fabric-B. ECT over Fabric-A is 180 μ s, whereas ECT over Fabric-B is 220 μ s. If Fabric-A becomes congested, resulting in an increase in ECT from 180 to 270 μ s (50% deviation), the average ECT as measured by the application increases to 245 μ s, which is only a 22% increase.

How can you verify if an increase in ECT for an application is because of congestion or not? Here are some suggestions:

- Check the metrics for the ports (such as TxWait) in the end-to-end data path.
- Check the ECT of the I/O flows that use the same network path as the switchport. If ECT increases just for one I/O flow but the rest of the I/O flows don't show an increase, it is not a network congestion issue because the network doesn't do any preferential treatment for I/O flows. A fabric just understands the frames, and all frames are equal for it.
- Investigate other metrics, like I/O size, IOPS, and so on. A common example is an increase in I/O size because larger I/O size operations take longer to complete. Also, find any SCSI and NVMe errors and link-level errors.

The Location for Measuring Latency Metrics

Cisco SAN Analytics calculates latency metrics by taking the time difference between relevant frames on the analytics-enabled switchports on MDS switches. As a result, the absolute value of these metrics may differ by a few microseconds, depending on the exact location of the measurement. For example, the ECT reported by a storage-connected switchport may be a few microseconds lower than the ECT reported by a host-connected switchport. This is because the storage-connected switchport sees the command frame a few microseconds after the host-connected switchport does, and it sees the response frames a few microseconds earlier than the host-connected switchport. When the time difference between the command frame and the response frame on the storage port is considered, it comes out to be less than the time difference between the command frame and the response frame on the host-connected switchport.

This difference in the value of latency metrics based on the location of measurement is marginal. It may be a matter of discussion in an academic exercise, but for any real-world production environment, the difference is very small, increases complexity, makes it hard for various teams to understand the low-level details, and doesn't change the end result.

What is more important is to understand that in lossless networks, congestion spreads from end to end quickly. If this congestion increases ECT by 50% on the storage-connected switchport, the same percentage increase will be seen on the host-connected port also, although the absolute values may differ.

What happens if the congestion is only severe enough that the effect is limited to storage ports or host ports? In production environments, the spread of congestion can't be predicted. More importantly, if the congestion has not spread from end to end, it's not severe enough to act on. In such cases, it is best to monitor and use the metrics for future planning, but without an end-to-end spread, the effect of congestion is limited to a small subset of the fabric.

Performance Metrics

Performance metrics convey the rate of I/O operations, their pattern, and the amount of data transferred.

I/O Operations per Second (IOPS)

IOPS, as its name suggests, is the number of read or write I/O operations per second. Typically, IOPS is a function of the application I/O profile and the type of storage. For example, transactional applications have higher IOPS requirements than do backup applications. Also, SSDs provide higher IOPS than do HDDs.

It is not possible to infer the network traffic directly from IOPS. An I/O operation may result in a few or many frames, depending on the data transferred by that I/O operation. Likewise, the throughput caused by I/O operations depends on the amount of data transferred by those I/O operations. Hence, it's difficult to predict the effect of higher IOPS on network congestion without accounting for I/O size, explained next.

On the other hand, network congestion typically results in reduced IOPS because the network is unable to deliver the frames to their destinations in a timely fashion or can transfer fewer frames.

I/O Size

The amount of data transferred by an I/O operation is known as its I/O size. I/O size is a function of the application's I/O profile. For example, a transactional application may have an I/O size of 4 KB, whereas a backup job may use an I/O size of 1 MB.

This I/O size metric in the context of storage I/O performance monitoring or SAN Analytics is different from the amount of data that an application wants to transfer as

part of an application-level transaction or operation. For example, an application may want to transfer 1 MB of data, but the host may decide to request this data using four I/O operations, each of size 256 KB. This difference is worth understanding, especially while investigating various layers within a host.

I/O size is encoded in the command frame of I/O operations. It has no dependency on network health. As a result, I/O size doesn't change with or without congestion.

Large I/O size results in a higher number of frames, which in turn leads to higher network throughput. For example, a 2 KB read I/O operation results in just one Fibre Channel data frame of size 2 KB, whereas a 64 KB read I/O operation results in 32 Fibre Channel frames of size 2 KB. Because of this, I/O size directly affects the network link utilization and thus provides insights into why a host port or a host-connected switchport may be highly utilized. For example, a host link may not be highly utilized with an I/O size of 16 KB. But the same link may get highly utilized and thus become the source of congestion when the I/O size spikes to 1 MB.

To understand the effect of I/O size on link utilization, consider the example in Figure 5-7. Two hosts, Host-1, and Host-2, connect to the switchports at 8 GFC to access storage from multiple arrays. Both servers are doing 10,000 read I/O operations per second (IOPS). However, the I/O sizes used by the two servers are different. Host-1 uses an I/O size of 4 KB, whereas Host-2 uses an I/O size of 128 KB.

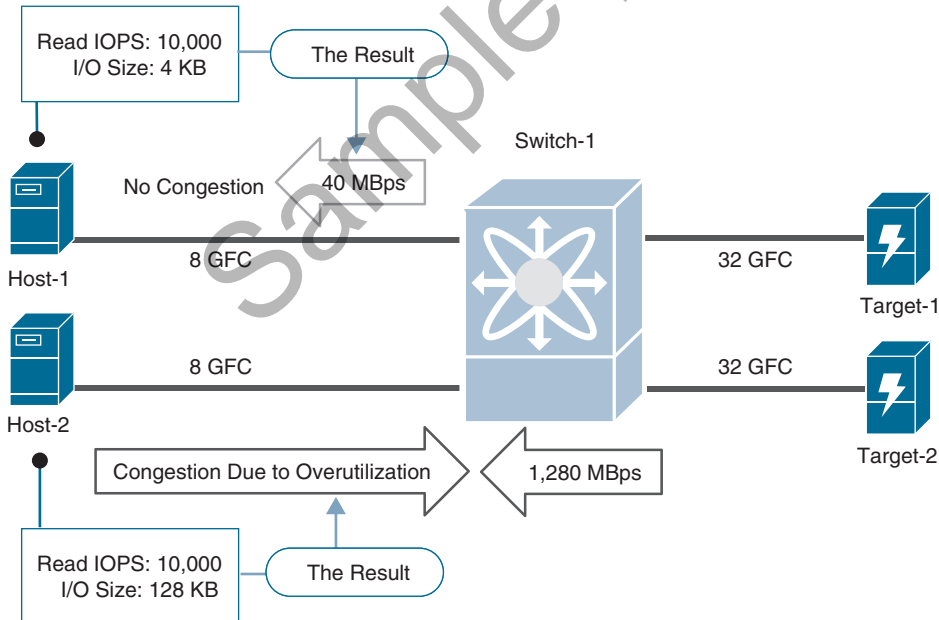


Figure 5-7 Detecting and Predicting the Cause of Congestion Using I/O Size