Maximizing Performance and Network Utility with a Science DMZ

Jason Zurawski
zurawski@es.net
ESnet / Lawrence Berkeley National Laboratory
Outline

• Motivation
• TCP Basics
• DMZ Design Patterns
• Integration w/ Science
• Conclusions
The Science DMZ in 1 Slide

Consists of four key components, all required:

- "Friction free" network path
  - Highly capable network devices (wire-speed, deep queues)
  - Virtual circuit connectivity option
  - Security policy and enforcement specific to science workflows
  - Located at or near site perimeter if possible

- Dedicated, high-performance Data Transfer Nodes (DTNs)
  - Hardware, operating system, libraries all optimized for transfer
  - Includes optimized data transfer tools such as Globus and GridFTP

- Performance measurement/test node
  - perfSONAR

- Once it’s up, users often need training – Engagement is key
  Details at http://fasterdata.es.net/science-dmz/
The Science DMZ has two main purposes:

1. A security architecture which allows for better segmentation of risks, and more granular application of controls to those segmented risks. In the case of the Science DMZ, the goal is to limit the segmented risk profile to that of a dedicated data transfer host.

2. An attempt to streamline tuning and troubleshooting by removing degrees-of-freedom (from both a reliability and performance perspective) from the active data transfer path.
The Science DMZ in 1 Picture

User experience

Design
Outline

• Motivation
• *TCP Basics*
• DMZ Design Patterns
• Integration w/ Science
• Conclusions
Science DMZ Background

• The data mobility performance requirements for data intensive science are beyond what can typically be achieved using traditional methods
  • Default host configurations (TCP, filesystems, NICs)
  • Converged network architectures designed for commodity traffic
  • Conventional security tools and policies
  • Legacy data transfer tools (e.g. SCP)
  • Wait-for-trouble-ticket operational models for network performance
Science DMZ Background

- The Science DMZ model describes a performance-based approach
  - Dedicated infrastructure for wide-area data transfer
    - Well-configured data transfer hosts with modern tools
    - Capable network devices
    - High-performance data path which does not traverse commodity LAN
  - Proactive operational models that enable performance
    - Well-deployed test and measurement tools (perfSONAR)
    - Periodic testing to locate issues instead of waiting for users to complain
  - Security posture well-matched to high-performance science applications
TCP – Ubiquitous and Fragile

• Networks provide connectivity between hosts – how do hosts see the network?
  • From an application’s perspective, the interface to “the other end” is a socket
  • Communication is between applications – mostly over TCP

• TCP – the fragile workhorse
  • TCP is (for very good reasons) timid – packet loss is interpreted as congestion
  • Packet loss in conjunction with latency is a performance killer
  • Like it or not, TCP is used for the vast majority of data transfer applications (more than 95% of ESnet traffic is TCP)
Packet/Data Loss?!

• “Wait a minute! I thought TCP was a reliable protocol? What do you mean ‘packet loss’, where is the data going!?”

• The data isn’t loss forever, it’s dropped somewhere on the path.
  • Usually by a device without enough buffer space to accept it, or by someone who thinks the data is corrupted and they won’t send it

• Once its dropped, we have a way of knowing its been dropped.
  • TCP is reliable, each end is keeping track of what was sent, and what was received.
  • If something goes missing, its resent. Resending is what takes the time, and causes the slowdown.

• TCP is able to reliably and transparently recover from packet loss by retransmitting any/all lost packets
  • This is how it provides a reliable data transfer services to the applications
  • The reliability mechanisms dramatically reduce performance when they are exercised

• We want to eliminate the causes of packet loss – so that we don’t need to test out the (slow) way that TCP can recover.

• What is the impact of that recovery?
A small amount of packet loss makes a huge difference in TCP performance.

With loss, high performance beyond metro distances is essentially impossible.
How Do We Accommodate TCP?

• High-performance wide area TCP flows must get loss-free service
  • Sufficient bandwidth to avoid congestion
  • Deep enough buffers in routers and switches to handle bursts
    • Especially true for long-distance flows due to packet behavior
    • No, this isn’t buffer bloat
• Equally important – the infrastructure must be verifiable so that clean service can be provided
  – Stuff breaks
    • Hardware, software, optics, bugs, ...
    • How do we deal with it in a production environment?
  – Must be able to prove a network device or path is functioning correctly
    • Regular active tests should be run – perfSONAR
  – Small footprint is a huge win
    • Fewer the number of devices = easier to locate the source of packet loss
Outline

• Motivation
• TCP Basics
• **DMZ Design Patterns**
• Integration w/ Science
• Conclusions
Science DMZ Takes Many Forms

• There are a lot of ways to combine these things – it all depends on what you need to do
  • Small installation for a project or two
  • Facility inside a larger institution
  • Institutional capability serving multiple departments/divisions
  • Science capability that consumes a majority of the infrastructure

• Some of these are straightforward, others are less obvious

• Key point of concentration: eliminate sources of packet loss / packet friction
Ad Hoc DTN Deployment

Site Border Router

WAN

10G

Test and measurement not aligned with data resource placement

perfSONAR

Global security policy mixes rules for science and business traffic

Conflicting requirements result in performance compromises

Site / Campus LAN

10GE

Perimeter Firewall

Building or Wiring Closet Switch/Router

High performance Data Transfer Node with high-speed storage

DTN traffic subject to limitations of general-purpose networking equipment/config

Note: Site border router and perimeter firewall are often the same device

DTN traffic subject to firewall limitations
Legacy Method: Ad Hoc DTN Deployment

• This is often what gets tried first
• Data transfer node deployed where the owner has space
  • This is often the easiest thing to do at the time
  • Straightforward to turn on, hard to achieve performance
• If lucky, perfSONAR is at the border
  – This is a good start
  – Need a second one next to the DTN
• Entire LAN path has to be sized for data flows
• Entire LAN path is part of any troubleshooting exercise
• This usually fails to provide the necessary performance.
A better approach: simple Science DMZ

- **Border Router**
  - **WAN**
  - **Science DMZ Switch/Router**
  - **Enterprise Border Router/Firewall**
  - **Site / Campus LAN**
  - **Site / Campus access to Science DMZ resources**
  - **Clean, High-bandwidth WAN path**
  - **High performance Data Transfer Node with high-speed storage**
  - **Per-service security policy control points**
  - **perfSONAR**

- **ESnet Science Engagement** (engage@es.net) - 5/3/19
Small-scale Science DMZ Deployment

• Add-on to existing network infrastructure
  • All that is required is a port on the border router
  • Small footprint, pre-production commitment
• Easy to experiment with components and technologies
  • DTN prototyping
  • perfSONAR testing
• Limited scope makes security policy exceptions easy
  • Only allow traffic from partners
  • Add-on to production infrastructure – lower risk
Prototype Science DMZ Data Path

- **WAN**
  - **perfSONAR**
- **Border Router**
  - Clean, High-bandwidth WAN path
  - Site / Campus access to Science DMZ resources
- **Science DMZ Switch/Router**
  - Per-service security policy control points
- **High performance Data Transfer Node**
  - with high-speed storage
- **Enterprise Border Router/Firewall**
  - High Latency WAN Path
  - Low Latency LAN Path

**EPOC**

Engagement and Performance Operations Center
Support For Multiple Projects

• Science DMZ architecture allows multiple projects to put DTNs in place
  • Modular architecture
  • Centralized location for data servers

• This may or may not work well depending on institutional politics
  • Issues such as physical security can make this a non-starter
  • On the other hand, some shops already have service models in place

• On balance, this can provide a cost savings – it depends
  • Central support for data servers vs. carrying data flows
  • How far do the data flows have to go?
Multiple Projects

- Border Router
- Enterprise Border Router/Firewall
- Clean, High-bandwidth WAN path
- Site / Campus access to Science DMZ resources
- Science DMZ Switch/Router
- Per-project security policy control points
- Project A DTN
- Project B DTN
- Project C DTN

perfSONAR
Supercomputer Center Deployment

• High-performance networking is assumed in this environment
  • Data flows between systems, between systems and storage, wide area, etc.
  • Global filesystem often ties resources together
    • Portions of this may not run over Ethernet (e.g. IB)
    • Implications for Data Transfer Nodes
• “Science DMZ” may not look like a discrete entity here
  • By the time you get through interconnecting all the resources, you end up with most of the network in the Science DMZ
  • This is as it should be – the point is appropriate deployment of tools, configuration, policy control, etc.
• Office networks can look like an afterthought, but they aren’t
  • Deployed with appropriate security controls
  • Office infrastructure need not be sized for science traffic
Supercomputer Center Data Path

Border Router

Firewall

Core Switch/Router

WAN

Virtual Circuit

Front end switch

Data Transfer Nodes

High Latency WAN Path

Low Latency LAN Path

High Latency VC Path

Supercomputer

Parallel Filesystem
Distributed Science DMZ

• Fiber-rich environment enables distributed Science DMZ
  • No need to accommodate all equipment in one location
  • Allows the deployment of institutional science service

• WAN services arrive at the site in the normal way

• Dark fiber distributes connectivity to Science DMZ services throughout the site
  • Departments with their own networking groups can manage their own local Science DMZ infrastructure
  • Facilities or buildings can be served without building up the business network to support those flows

• Security is potentially more complex
  • Remote infrastructure must be monitored
  • Solutions depend on relationships with security groups
Distributed Science DMZ – Dark Fiber

Border Router

WAN

Clean, High-bandwidth WAN path

10G

Site / Campus access to Science DMZ resources

Enterprise Border Router/Firewall

10GE

10GE

Science DMZ Switch/Router

Dark Fiber

Per-project security policy control points

Project A DTN (remote)

Project B DTN (remote)

PerfSONAR

Dark Fiber

Dark Fiber

Project C DTN (remote)

Site / Campus LAN
Common Threads

• Two common threads exist in all these examples
• Accommodation of TCP
  • Wide area portion of data transfers traverses purpose-built path
  • High performance devices that don’t drop packets
• Ability to test and verify
  • When problems arise (and they always will), they can be solved if the infrastructure is built correctly
  • Small device count makes it easier to find issues
  • Multiple test and measurement hosts provide multiple views of the data path
    • perfSONAR nodes at the site and in the WAN
    • perfSONAR nodes at the remote site
Equipment – Routers and Switches

• Requirements for Science DMZ gear are different
  • No need to go for the kitchen sink list of services
  • A Science DMZ box only needs to do a few things, but do them well
  • Support for the latest LAN integration magic with your Windows Active Directory environment is probably not super-important
  • A clean architecture is important
    • How fast can a single flow go?
    • Are there any components that go slower than interface wire speed?

• There is a temptation to go cheap
  • Hey, it only needs to do a few things, right?
  • You typically don’t get what you don’t pay for
    • (You sometimes don’t get what you pay for either)
Common Circumstance: Multiple Ingress Data Flows, Common Egress

Hosts will typically send packets at the speed of their interface (1G, 10G, etc.)

• Instantaneous rate, not average rate
• If TCP has window available and data to send, host sends until there is either no data or no window

Hosts moving big data (e.g. DTNs) can send large bursts of back-to-back packets

• This is true even if the average rate as measured over seconds is slower (e.g. 4Gbps)
• On microsecond time scales, there is often congestion
• Router or switch must queue packets or drop them
Rant Ahead

N.B. You are entering into rant territory on the matter of switch buffering. If you are going to take away anything from the next section:

1. Under buffered network devices are the **single greatest threat** to data intensive use of the network. You can make hosts, operating systems, and application choices perform better for ‘free’, it will cost $$$ to fix a crappy switch or router

2. You will be steered toward non-optimal choices when you talk with the vendor community because they don’t understand simple math (but by the end of this, you will).

3. A 1U/2U data center/racklan network device should never be in the path of your data intensive network use case.

4. Non-passive (e.g. stateful) security devices are the same for buffering, and are actually worse due to the processing overhead.

5. Anytime you jump around the OSI stack – add friction (e.g. routing when you don’t need to, application layer inspection, etc.)
All About That Buffer (No Cut Through)

Figure 1: Basic Router Architecture
All About That Buffer (No Cut Through)

- Data arrives from multiple sources

Buffers have a finite amount of memory
- Some have this per interface
- Others may have access to a shared memory region with other interfaces

The processing engine will:
- Extract each packet/frame from the queues
- Pull off header information to see where the destination should be
- Move the packet/frame to the correct output queue

Additional delay is possible as the queues physically write the packet to the transport medium (e.g. optical interface, copper interface)

Figure 1: Basic Router Architecture
All About That Buffer (No Cut Through)

- **The Bandwidth Delay Product**
  - The amount of “in flight” data for a TCP connection (BDP = bandwidth * round trip time)
  - Example: 10Gb/s cross country, ~100ms
    - \(10,000,000,000 \text{ b/s} \times .1 \text{ s} = 1,000,000,000 \text{ bits}\)
    - \(1,000,000,000 / 8 = 125,000,000 \text{ bytes}\)
    - \(125,000,000 \text{ bytes} / (1024*1024) \approx 125\text{MB}\)
  - Ignore the math aspect: it’s making sure there is memory to catch and send packets
    - As the speed increases, there are more packets.
    - If there is not memory, we drop them, and that makes TCP sad.
All About That Buffer (No Cut Through)

• Buffering isn’t as important on the LAN (this is why you are normally pressured to buy ‘cut through’ devices)
  • Change the math to make the Latency 1ms = **1.25MB**
  • ‘Cut through’ and low latency switches are designed for the data center, and can handle typical data center loads that don’t require buffering (e.g. same to same speeds, destinations within the broadcast domain)

• Buffering *MATTERS* for WAN Transfers
  • Placing something with inadequate buffering in the path reduces the buffer for the entire path. E.g. if you have an expectation of 10Gbps over 100ms – don’t place a 12MB buffer anywhere in there – your reality is now ~10x less than it was before (e.g. 10Gbps @ 10ms, or 1Gbps @ 100ms)

• Ignore the math aspect, its really just about making sure there is memory to catch packets. As the speed increases, there are more packets. If there is not memory, we drop them, and that makes TCP sad.
  • Memory on hosts, and network gear
All About That Buffer (No Cut Through)

- What does this “look” like to a data transfer? Consider the test of iPerf below
  - See TCP ‘ramp up’ and slowly increase the window
  - When something in the path has no more space for packets – a drop occurs. TCP will eventually react to the lost packet, and ‘back off’
  - In the example, this first occurs when we reach a buffer of around 6-8MB. Then after backoff the window is halved a couple of times
  - This happens again later – at a slightly higher buffer limit. This could be because there was cross traffic the first time, etc.

<table>
<thead>
<tr>
<th>ID</th>
<th>Interval</th>
<th>Transfer</th>
<th>Bandwidth</th>
<th>Retr</th>
<th>Cwnd</th>
</tr>
</thead>
<tbody>
<tr>
<td>14</td>
<td>0.00-1.00 sec</td>
<td>524 KBytes</td>
<td>4.29 Mbits/sec</td>
<td>0</td>
<td>157 KBytes</td>
</tr>
<tr>
<td>14</td>
<td>1.00-2.00 sec</td>
<td>3.31 MBytes</td>
<td>27.8 Mbits/sec</td>
<td>0</td>
<td>979 KBytes</td>
</tr>
<tr>
<td>14</td>
<td>2.00-3.00 sec</td>
<td>17.7 MBytes</td>
<td>148 Mbits/sec</td>
<td>0</td>
<td>5.36 MBytes</td>
</tr>
<tr>
<td>14</td>
<td>3.00-4.00 sec</td>
<td>18.8 MBytes</td>
<td>157 Mbits/sec</td>
<td>214</td>
<td>1.77 MBytes</td>
</tr>
<tr>
<td>14</td>
<td>4.00-5.00 sec</td>
<td>11.2 MBytes</td>
<td>94.4 Mbits/sec</td>
<td>0</td>
<td>1.88 MBytes</td>
</tr>
<tr>
<td>14</td>
<td>5.00-6.00 sec</td>
<td>10.0 MBytes</td>
<td>83.9 Mbits/sec</td>
<td>0</td>
<td>2.39 MBytes</td>
</tr>
<tr>
<td>14</td>
<td>6.00-7.00 sec</td>
<td>16.2 MBytes</td>
<td>136 Mbits/sec</td>
<td>0</td>
<td>3.63 MBytes</td>
</tr>
<tr>
<td>14</td>
<td>7.00-8.00 sec</td>
<td>23.8 MBytes</td>
<td>199 Mbits/sec</td>
<td>0</td>
<td>5.50 MBytes</td>
</tr>
<tr>
<td>14</td>
<td>8.00-9.00 sec</td>
<td>38.0 MBytes</td>
<td>325 Mbits/sec</td>
<td>0</td>
<td>8.23 MBytes</td>
</tr>
<tr>
<td>14</td>
<td>9.00-10.00 sec</td>
<td>57.5 MBytes</td>
<td>482 Mbits/sec</td>
<td>0</td>
<td>11.8 MBytes</td>
</tr>
<tr>
<td>14</td>
<td>10.00-11.00 sec</td>
<td>81.2 MBytes</td>
<td>682 Mbits/sec</td>
<td>0</td>
<td>16.2 MBytes</td>
</tr>
<tr>
<td>14</td>
<td>11.00-12.00 sec</td>
<td>50.0 MBytes</td>
<td>419 Mbits/sec</td>
<td>35</td>
<td>3.93 MBytes</td>
</tr>
<tr>
<td>14</td>
<td>12.00-13.00 sec</td>
<td>15.0 MBytes</td>
<td>126 Mbits/sec</td>
<td>0</td>
<td>2.20 MBytes</td>
</tr>
<tr>
<td>14</td>
<td>13.00-14.00 sec</td>
<td>11.2 MBytes</td>
<td>94.4 Mbits/sec</td>
<td>0</td>
<td>2.53 MBytes</td>
</tr>
<tr>
<td>14</td>
<td>14.00-15.00 sec</td>
<td>13.8 MBytes</td>
<td>115 Mbits/sec</td>
<td>1</td>
<td>1.50 MBytes</td>
</tr>
<tr>
<td>14</td>
<td>15.00-16.00 sec</td>
<td>6.25 MBytes</td>
<td>52.4 Mbits/sec</td>
<td>5</td>
<td>813 KBytes</td>
</tr>
<tr>
<td>14</td>
<td>16.00-17.00 sec</td>
<td>5.00 MBytes</td>
<td>41.9 Mbits/sec</td>
<td>0</td>
<td>909 KBytes</td>
</tr>
<tr>
<td>14</td>
<td>17.00-18.00 sec</td>
<td>5.00 MBytes</td>
<td>41.9 Mbits/sec</td>
<td>0</td>
<td>1.37 MBytes</td>
</tr>
<tr>
<td>14</td>
<td>18.00-19.00 sec</td>
<td>10.0 MBytes</td>
<td>83.9 Mbits/sec</td>
<td>0</td>
<td>2.43 MBytes</td>
</tr>
<tr>
<td>14</td>
<td>19.00-20.00 sec</td>
<td>17.5 MBytes</td>
<td>147 Mbits/sec</td>
<td>0</td>
<td>4.22 MBytes</td>
</tr>
</tbody>
</table>
Decoding Specifications

• “The buffering behaviors of the switches and their operating system, such as behavior under memory stress, are typically proprietary information and not well documented”
  
http://www.measurementlab.net/blog/traffic-microbursts-and-their-effect-on-internet-measurement/

• “Even if you know how much packet buffer is in the switch, assumptions on how it is deployed that are not backed up by testing can lead to unhappiness. What we like to say is that is the job of the network engineers to move bottlenecks around.”
  
  • Jim Warner

• http://people.ucsc.edu/~warner/buffer.html
Decoding Specifications

• So let's say the spec sheet says this:

• What does ‘module’ mean?
  • Typically this means the amount of memory for the entire switch (if it’s a single unit) or a blade (if the chassis supports more than one).
  • BUT … this memory can be allocated in a number of different ways:
    • Shared between all ports
    • Dedicated (smaller) amounts per-port
    • Shared between ASICS, which control a bank of ports

| VOQ buffer | 72 MB per module |
Decoding Specifications

• Consider this architecture
  • 48 Ports
    • 12 ASICS
    • 4 Ports per ASIC
  • **72MB** total
    • **6MB per ASIC**
    • If all ports are in use – expect that each port has access to **1.5MB**. If only one is in use, it can use 6MB
  • Additional memory is often available in a ‘burst buffer’ in the fabric

ASIC = application-specific integrated circuit, think ‘small routing engine’
Decoding Specifications

• Recall: [https://www.switch.ch/network/tools/tcp_throughput/](https://www.switch.ch/network/tools/tcp_throughput/)

• What does 6MB get you?
  • 1Gbps @ <= 48ms (e.g. ½ needed for coast-to-coast)
  • 10Gbps @ <= 4.8ms (e.g. metro area)

• What does 1.5MB get you?
  • 1Gbps @ <= 12ms (e.g. regional area)
  • 10Gbps @ <= 1.2ms (e.g. data center [or more accurately, rack or row])

• In either case – remember this assumes you are the only thing using that memory ... congestion is a more likely reality
Outline

• Motivation
• TCP Basics
• DMZ Design Patterns
• Integration w/ Science
• Conclusions
NCAR RDA Data Portal

• Let’s say I have a nice compute allocation at NERSC – climate science

• Let’s say I need some data from NCAR for my project
  • https://rda.ucar.edu/

• Data sets (there are many more, but these are two):
  • https://rda.ucar.edu/datasets/ds199.1/ (1.5TB)
  • https://rda.ucar.edu/datasets/ds313.0/ (430GB)

• Download to NERSC (could also do ALCF or NCSA or OLCF)
NCAR RDA Performance to DOE HPC Facilities

- 1.5TB data set
- 1121 files
Next Steps – Building On The Science DMZ

• Enhanced cyberinfrastructure substrate now exists
  • Wide area networks (ESnet, GEANT, Internet2, Regionals)
  • Science DMZs connected to those networks
  • DTNs in the Science DMZs

• What does the scientist see?
  • Scientist sees a science application
    • Data transfer
    • Data portal
    • Data analysis
  • Science applications are the user interface to networks and DMZs

• Large-scale data-intensive science requires that we build larger structures on top of those components
HPC Facilities: The Petascale DTN Project

• Built on top of the Science DMZ model
• Effort to improve data transfer performance between the DOE ASCR HPC facilities at ANL, LBNL, and ORNL, and also NCSA.
  • Multiple current and future science projects need to transfer data between HPC facilities
  • Performance goal is 15 gigabits per second (equivalent to 1PB/week)
  • Realize performance goal for routine Globus transfers without special tuning
• Reference data set is 4.4TB of cosmology simulation data

Petascale DTN Project
March 2016
L380 Data Set

ALCF DTN cluster
Globus endpoint: alcf#dtm_mira

NERSC DTN cluster
Globus endpoint: nersc#dtm

OLCF DTN cluster
Globus endpoint: olcf#dtm_atlas

NCSA DTN cluster
Globus endpoint: ncsa#BlueWaters

Data set: L380
Files: 19260
Directories: 211
Other files: 0
Total bytes: 4442781786682 (4.4T bytes)
Smallest file: 0 bytes (0 bytes)
Largest file: 11313986248 bytes (11G bytes)
Size distribution:
1 - 10 bytes: 7 files
10 - 100 bytes: 1 files
100 - 1K bytes: 59 files
1K - 10K bytes: 3170 files
10K - 100K bytes: 1560 files
100K - 1M bytes: 2817 files
1M - 10M bytes: 3901 files
10M - 100M bytes: 3800 files
100M - 1G bytes: 2295 files
1G - 10G bytes: 1647 files
10G - 100G bytes: 3 files

engage@es.net

Petascale DTN Project
November 2017
L380 Data Set

Gigabits per second (min/avg/max), three transfers

NERSC DTN cluster
Globus endpoint: nersc#dtnc
Filesystem: /project

OLCF DTN cluster
Globus endpoint: oflc#dtnc_atlas
Filesystem: atlas2

ALCF DTN cluster
Globus endpoint: alcf#dtnc_mira
Filesystem: /projects

NCSA DTN cluster
Globus endpoint: ncsa#BlueWaters
Filesystem: /scratch

Data set: L380
Files: 19260
Directories: 211
Other files: 0
Total bytes: 4442781786482 (4.47 bytes)
Smallest file: 0 bytes (0 bytes)
Largest file: 11331896248 bytes (11G bytes)
Size distribution:
1 - 10 bytes: 7 files
10 - 100 bytes: 1 files
100 - 1K bytes: 59 files
1K - 10K bytes: 3170 files
10K - 100K bytes: 1560 files
100K - 1M bytes: 2817 files
1M - 10M bytes: 3901 files
10M - 100M bytes: 3800 files
100M - 1G bytes: 2295 files
1G - 10G bytes: 1647 files
10G - 100G bytes: 3 files

41.0/42.2/43.9 Gbps
21.2/22.6/24.5 Gbps
55.4/58.7/57.4 Gbps
23.1/33.7/39.7 Gbps
35.9/39.0/40.7 Gbps
43.0/50.0/56.3 Gbps
33.0/35.0/37.8 Gbps
44.1/46.8/48.4 Gbps
26.7/34.7/39.9 Gbps
33.2/43.4/50.3 Gbps
29.9/33.1/35.5 Gbps
34.6/47.5/56.8 Gbps

EPOC
Engagement and Performance Operations Center

engage@es.net - 5/3/19
Science Data Portals

• Large repositories of scientific data
  • Climate data
  • Sky surveys (astronomy, cosmology)
  • Many others
  • Data search, browsing, access

• Many scientific data portals were designed 15+ years ago
  • Single-web-server design
  • Data browse/search, data access, user awareness all in a single system
  • All the data goes through the portal server
    • In many cases by design
    • E.g. embargo before publication (enforce access control)
Legacy Portal Design

- Very difficult to improve performance without architectural change
  - Software components all tangled together
  - Difficult to put the whole portal in a Science DMZ because of security
  - Even if you could put it in a DMZ, many components aren’t scalable
- What does architectural change mean?
Architectural Examination of Data Portals

• Common data portal functions (most portals have these)
  • Search/query/discovery
  • Data download method for data access
  • GUI for browsing by humans
  • API for machine access – ideally incorporates search/query + download

• Performance pain is primarily in the data handling piece
  • Rapid increase in data scale eclipsed legacy software stack capabilities
  • Portal servers often stuck in enterprise network

• Can we “disassemble” the portal and put the pieces back together better?
  • Use Science DMZ as a platform for the data piece
  • Avoid placing complex software in the Science DMZ
Legacy Portal Design

Portal server applications:
- web server
- search
- database
- authentication
- data service

Browsing path
Query path
Data path

ESnet Science Engagement (engage@es.net) - 5/3/19
Next-Generation Portal Leverages Science DMZ
Put The Data On Dedicated Infrastructure

• We have separated the data handling from the portal logic
• Portal is still its normal self, but enhanced
  • Portal GUI, database, search, etc. all function as they did before
  • Query returns pointers to data objects in the Science DMZ
  • Portal is now freed from ties to the data servers (run it on Amazon if you want!)
• Data handling is separate, and scalable
  • High-performance DTNs in the Science DMZ
  • Scale as much as you need without modifying the portal software
• Outsource data handling to computing centers
  • Computing centers are set up for large-scale data
  • Let them handle the large-scale data, and let the portal do the orchestration of data placement

ESnet Science Engagement (engage@es.net) - 5/3/19
JGI Data Portal

**Searching for Projects**
- Explore what you can do here.
- Search projects/proposals using “Advanced Search” filters.

**Downloading Files**
- Download over the web
- Download large number of files with Globus service.
- Download via API using scripting or programming
- Download with “Cart” by collecting projects/portals of your interest.

**Looking for Access**
- Looking for data and do not have access to the private portal? Please contact PI
- How to grant access to your proposal/project/genome? Get Instructions.

**What’s New**
New feature: “Download with Cart”
A convenient way to collect projects/genomes/metagenomes of your interest and download all files associated with them in bulk.
Read more and provide your comments and suggestions for this feature to our team.

**My Favorites**
My Favorites: New Feature - Based on Your Feedback
This feature allows to save your filtered search results to "My Favorites" and access it later.

**The “Tree of Life”**
Please use our powerful search or go to the “Tree of Life” if it is the most convenient way for you to reach your genomes/projects.

---

EPOC
Engagement and Performance Operations Center

JGI Genome Portal

New Feature: “Bulk Downloads”

Collect your favorite projects and download them in bulk with our new feature “CART”. Ability to download files with Portal or via Globus.

Find out more details
ESnet Science Engagement (engage@es.net) - 5/3/19
Reasons To Scale Data Portals

• Some reasons are obvious
  • Increase in size of data objects (MB → GB → 100s of GB)
  • Number of data objects (many thousands per data set)

• Other reasons are paradigm shifts
  • Modern data analysis on HPC can use a *lot* of data
  • Today’s HPC facilities are far more capable than in the past

• Retrofit / rebuilt data portals and data repositories
  • Significant wins from increased data analysis
Science at Scale: Genomics

ORNL RESEARCHERS LEVERAGE GPU TENSOR CORES TO DELIVER UNPRECEDENTED PERFORMANCE

Researchers at the US Department of Energy’s Oak Ridge National Laboratory broke the exascale barrier, achieving a peak throughput of 1.88 exaflops—faster than any previously reported science application—while analyzing genomic data on the recently launched Summit supercomputer.

The ORNL team achieved the feat, the equivalent to carrying out nearly 2 trillion billion calculations per second, by using a mixture of numerical precisions. Traditionally, scientific computing has relied on double-precision floating-point operations, however, interest in reduced numerical precision has grown in recent years due to breakthroughs in artificial intelligence and machine learning. In this case, researchers were able to implement high-speed single- and half-precision operations to gain additional performance.

The record-setting run was carried out using a representative dataset on 4,000 of Summit’s GPU-accelerated nodes.
Science at Scale: Climate

Figure 3: Sample images of atmospheric rivers correctly classified (true positive) by our deep CNN model. Figure shows total column water vapor (color map) and land sea boundary (solid line).
They Can Use All The Data

• Groups like these need large data sets
• Much of the data in their field is behind legacy portals
  • Significant human effort to retrieve what they need
  • Legacy systems perform poorly, especially at scale
• Legacy data portals are a product of their time
  • We now live in the future from the perspective of those designs
  • Current systems far exceed the capabilities available 15 years ago
  • From the perspective of today’s systems, legacy portals are products of a bygone past
• It is now perfectly reasonable for a scientist to want all the data
  • Machine learning + HPC
  • But this only works if the scientists can get to the data at scale
IT/Engineering (not the scientists) Have To Do This

• The scientific community cannot do this for themselves
• Individual researchers do not control the resources
  • Computing centers
  • Data repositories
  • Science networks
  • Our community owns these – we have to do the work
• Integration, performance engineering, interoperability
• Science Engagement to teach scientists how to use the better platforms
• This is the path forward

¯\_(ツ)_/¯
Networks Cannot Do This Alone

• We need a whole-community effort
  • Networks
  • HPC facilities
  • Data repositories / Data portals
  • Experimental facilities
  • Science collaborations
  • Science programs

• Networks can help, and must be part of the conversation
  • Heavy lifting is now at the network edge, in collaboration with the network core
  • Need to get the architecture right – we know how to do this

• Engagement is critical
  • We have to teach people to use the new thing
  • They aren’t going to magically find out about new services
Long-Term Vision
It’s All A Bunch Of Science DMZs
It’s All A Bunch Of Science DMZs
Outline

• Motivation
• TCP Basics
• DMZ Design Patterns
• Integration w/ Science
• Conclusions
ESnet’s Science DMZ – Enabling Big Data Science

The Science DMZ design pattern supports secure, high-performance data movement between and among facilities, labs, universities, data portals, and clouds.

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Requirements analysis of DOE Office of Science facilities and projects from network and data perspective</td>
<td>Science Engagement model for science support</td>
<td>Science DMZs deployed at DOE labs and facilities</td>
<td>Development Support</td>
<td>Development Support</td>
</tr>
<tr>
<td>Performance engineering, troubleshooting, experience</td>
<td>OIN workshop series trains CI engineers throughout the community</td>
<td>NSF CC-NIE and successor programs fund over 100 Science DMZs at universities</td>
<td>Science DMZ supports development, prototyping, and incremental deployment of advanced network and data services</td>
<td></td>
</tr>
<tr>
<td>Creation of Science DMZ design pattern</td>
<td>Petascale DTN Project</td>
<td>Ongoing production operation across science complex</td>
<td>Production Data Movement</td>
<td>Science DMZs support large scale data movement and data placement across the science complex (DOE and collaborators)</td>
</tr>
<tr>
<td></td>
<td>Modern Research Data Portal design pattern</td>
<td></td>
<td></td>
<td>Performant security</td>
</tr>
<tr>
<td></td>
<td>Pacific Research Platform</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Scalable Data Platform</td>
<td></td>
<td>Science DMZs provide foundation for data-driven science globally</td>
</tr>
</tbody>
</table>
Maximizing Performance and Network Utility with a Science DMZ

Jason Zurawski
zurawski@es.net
ESnet / Lawrence Berkeley National Laboratory