A Comprehensive Tutorial on Science DMZ

Jorge Crichigno, Elias Bou-Harb, and Nasir Ghani

***Abstract*—Science and engineering applications are now generating data at an unprecedented rate. From large facilities such as the Large Hadron Collider to portable DNA sequencing devices, these instruments can produce hundreds of terabytes in short periods of time. Researchers and other professionals rely on networks to transfer data between sensing locations, instruments, data storage devices, and computing systems. While general-purpose networks, also referred to as enterprise networks, are capable of transporting basic data, such as e-mails and Web content, they face numerous challenges when transferring terabyte- and petabyte-scale data. At best, transfers of science data on these networks may last days or even weeks. In response to this challenge, the Science Demilitarized Zone (Science DMZ) has been proposed. The Science DMZ is a network or a portion of a network designed to facilitate the transfer of big science data. The main elements of the Science DMZ include: 1) specialized end devices, referred to as data transfer nodes (DTNs), built for sending/receiving data at a high speed over wide area networks; 2) high-throughput, friction-free paths connecting DTNs, instruments, storage devices, and computing systems; 3) performance measurement devices to monitor end-to-end paths over multiple domains; and 4) security policies and enforcement mechanisms tailored for high-performance environments. Despite the increasingly important role of Science DMZs, the literature is still missing a guideline to provide researchers and other professionals with the knowledge to broaden the understanding and development of Science DMZs. This paper addresses this gap by presenting a comprehensive tutorial on Science DMZs. The tutorial reviews fundamental network concepts that have a large impact on Science DMZs, such as router architecture, TCP attributes, and operational security. Then, the tutorial delves into protocols and devices at different layers, from the physical cyberinfrastructure to application-layer tools and security appliances, that must be carefully considered for the optimal operation of Science DMZs. This paper also contrasts Science DMZs with general-purpose networks, and presents empirical results and use cases applicable to current and future Science DMZs.**

***Index Terms*—Science DMZ, network flows, friction-free paths, data transfer node, bandwidth-delay product, perfSONAR.**

I. Introduction

WHEN the United States (U.S.) decided to build the interstate highway system in the 1950s, the country already had city streets and two-lane highways for daily-life transportation. While at first this system appeared to be redundant, the interstate highway system increased the ease of travel for Americans and the ability to transport goods from east to west, without stoplights [1].

Tracing similarities with the current cyberinfrastructure, today's general-purpose networks, also referred to as enterprise networks, are capable of efficiently transporting basic data. These networks support multiple missions, including organizations' operational services such as email, procurement systems, and Web browsing. However, when transferring terabyte- and petabyte-scale science data, enterprise networks face many unsolved challenges [2]. Key issues preventing high throughput include slow processing by CPU-intensive security appliances, inability of routers and switches to absorb traffic bursts generated by large flows, end devices that are incapable of sending and receiving data at high rates, lack of data transfer applications that can exploit the available network bandwidth, and the absence of end-to-end path monitoring to detect failures.

The need for a suitable cyberinfrastructure for large flows is illustrated in Fig. 1, which shows the monthly average traffic volume through the Energy Science network (ESnet) [3]. ESnet is a high-performance network that carries science traffic for the U.S. Department of Energy. As of 2018, this network is transporting tens of petabytes (PBs) per month, an increase of several orders of magnitude from some years ago.

In response to this challenge of transmitting big science data via a cyber-highway system without stoplights, ESnet developed the concept of Science Demilitarized Zone (Science DMZ or SDMZ) [4]. The Science DMZ is a network or a portion of a network designed to facilitate the transfer of

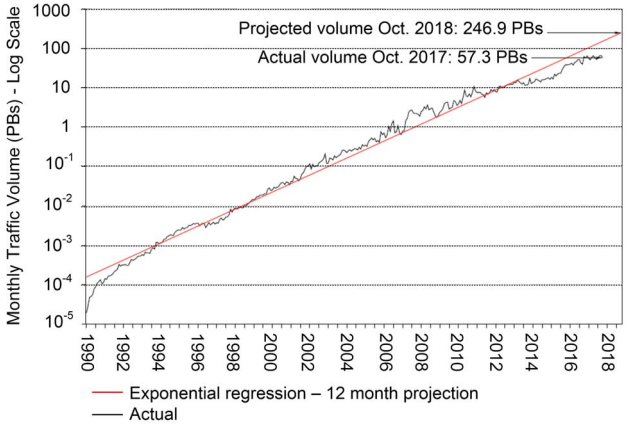

Fig. 1. Monthly average traffic volume through ESnet [3].

Manuscript received December 21, 2017; revised July 28, 2018; accepted September 11, 2018. Date of publication October 17, 2018; date of current version May 31, 2019. This work was supported by the Office of Advanced Cyberinfrastructure of the U.S. National Science Foundation under Award 1541352 and Award 1829698. *(Corresponding author: Jorge Crichigno.)*

J. Crichigno is with the Department of Integrated Information Technology, College of Engineering and Computing, University of South Carolina, Columbia, SC 29208 USA (e-mail: jcrichigno@cec.sc.edu).

E. Bou-Harb is with the Department of Computer Science, Florida Atlantic University, Boca Raton, FL 33431 USA (e-mail: ebouharb@fau.edu).

N. Ghani is with the Electrical Engineering Department, University of South Florida, Tampa, FL 33620 USA, and also with the Cyber Florida Center, University of South Florida, Tampa, FL 33620 USA (email: nghani@usf.edu).

Digital Object Identifier 10.1109/COMST.2018.2876086

1553-877X © 2018 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission.
See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

Fig. 2. Science DMZ data transfer applications. Top left: The Large Hadron Collider (LHC) produces approximately 30 PBs per year. Data is transmitted to multiple computing centers around the world. Photo courtesy of The European Organization for Nuclear Research [5]. Top center: The Very Large Array (VLA) is composed of 27 radio antennas of 25 meters in diameter each. Daily data collection comprises of several TBs, which are transmitted to research laboratories worldwide. Photo courtesy of the U.S. National Radio Astronomy Observatory [6]. Top right: Experimental Advanced Superconducting Tokamak. Data generated by the energy reactor is transmitted for analysis via a Science DMZ. Photo courtesy of ESnet [7]. Bottom left: magnetic resonance imaging scanner. Major brain imaging studies such as the Alzheimer's disease neuroimaging requires storage and transmission of multiple PBs of data [8]. Medical data can now be transported via medical Science DMZs [9], [10]. Photo courtesy of General Electric Healthcare [11]. Bottom center: Atomic, Molecular, and Optical (AMO) instrument. The instrument is used for a variety of experiments, such as illumination of single molecules. A single experiment can produce 150 to 200 TBs [12]. Photo courtesy of the U.S. SLAC National Accelerator Laboratory [13]. Bottom right: portable device for DNA and RNA sequencing which generates tens of GBs of data per experiment [14]. Photo courtesy of Nanopore Technologies [15].

big science data across wide area networks (WANs), typically at rates of 10 Gbps and above. In order to operate at such rates, this setup integrates the following key elements: i) end devices, referred to as data transfer nodes (DTNs), that are built for sending/receiving data at a high rate over WANs; ii) high-throughput paths connecting DTNs, instruments, storage devices, and computing systems. These paths are composed of highly-capable routers and switches and have no devices that may induce packet losses. They are referred to as friction-free paths; iii) performance measurement devices that monitor end-to-end paths over multiple domains; and iv) security policies and enforcement mechanisms tailored for high-performance science environments.

The Science DMZ architecture is similar to building the interstate highway system, whereas stoplights are removed to permit the high-speed movement of large flows. The interconnection of Science DMZs is also analogous to the development of the National Science Foundation network (NSFnet) in 1985, one of the predecessors of today's Internet. NSF, the main government agency in the U.S. supporting research and education in science and engineering, established the NSFnet to link together five supercomputer centers that were then deployed across the U.S. [16]. With Science DMZs, institutions are similarly linked together and have access to a virtual co-location of data that may rest anywhere in the world through a high-speed data-sharing architecture. Along these lines, Fig. 2 highlights applications that currently exploit the Science DMZ architecture to transmit large flows from instruments to laboratories for data analysis. From very large to portable devices, these instruments generate a large amount of data in short periods of time.

A. Contribution

At present, there is an increasing need to deploy Science DMZs in support of big science data transfers. However, efforts to prepare researchers and other professionals with the right knowledge are limited to dispersed work by the academia and the industry. Despite the importance of Science DMZs, currently there is no structured material in the forms of tutorials, surveys, or books.

This article addresses this gap in the literature by presenting a comprehensive tutorial on Science DMZs. Following a systematic approach through every layer of the protocol stack, the tutorial integrates information and tools for a better understanding of the issues, key challenges, best practices, and future research directions related to Science DMZs. The paper also presents empirical results and use cases obtained from state-of-the-art facilities and across a continental backbone. The results and use cases reinforce concepts and provide findings that are applicable to current and future Science DMZs. Since current researchers and practitioners are mostly trained to design and operate enterprise networks, this article will familiarize readers with Science DMZs, resulting in a broadening of the development and deployment of Science DMZs. The article reflects the wide interest of academia and industry in Science DMZs as an integrative system to build a high-speed cyber-highway. Examples include the

strong support of NSF and communities around the world endorsing the upgrade of network connectivity for science data transmissions [17], [18] and initiatives to improve WAN data transfers [19], [20]. Leading manufacturers of routers and switches, such as Cisco [21], Brocade [22], Ciena [23], and others, are now responding to the need for equipment suitable for Science DMZs. Window-based congestion control (used since the 1990s at the transport layer) is now being challenged by new paradigms such as rate-based congestion control [24], [25]. Application-layer tools targeted for Science DMZs are incorporating high-performance features to facilitate the sharing of big data [26]. Industry security leaders [27] and U.S. national laboratories [28] are now designing appliances amenable for large flows while protecting the Science DMZ and increasing rates beyond 100 Gbps [29].

B. Paper Structure

The article follows a bottom-up approach, from the physical cyberinfrastructure to the application layer and security aspects. Section II presents the motivation for and architecture of Science DMZs. This section also describes the WAN cyberinfrastructure supporting Science DMZs. Section III discusses attributes related to routers and switches, which are at the core of Science DMZs. Section IV describes key features that must be considered at the transport layer in Science DMZs. Section V presents application-layer tools used in Science DMZs and their features to support science data transfers. Section VI describes security challenges arising in Science DMZs and presents best practices. Section VII presents empirical results and use cases. Section VIII describes key challenges and open research issues, and Section IX concludes this article. Each section describes Science DMZ features at a particular layer in the protocol stack. As these features are described and analyzed, they are also compared with those features used in enterprise networks. Contrasting Science DMZs with enterprise networks provides essential information for a better understanding of the former. The abbreviations used in this article are summarized in Table XI, at the end of the article.

C. Definition of a Flow

Central to the discussion of the Science DMZ is the concept of a flow. This article follows the definition of a flow by the IP Flow Information Export (IPFIX) working group within the Internet Engineering Task Force (IETF) [30], [31]:

A flow is defined as a set of IP packets passing an observation point in the network during a certain time interval. All packets belonging to a particular flow have a set of common properties.

The common properties adopted in this article are the source and destination IP addresses, source and destination transport-layer ports, and transport-layer protocol. Additionally, there are two flow characteristics that are significant in this paper. The first characteristic is the duration of the flow, which is the time interval elapsed between the first and last packets with the same common properties. The second characteristic is the volume or size of the flow, which is the aggregate number of bytes contained in the packets with the same common properties.

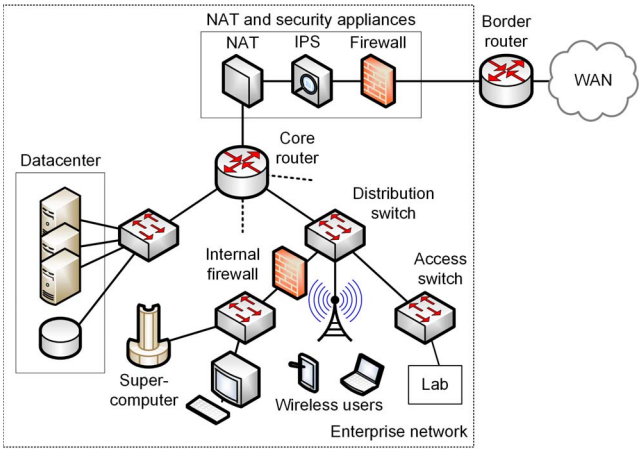

Fig. 3. A campus enterprise network.

II. Science DMZ Architecture and Physical Cyberinfrastructure

A. Limitations of Enterprise Networks and Motivation for Science DMZs

An enterprise network is composed of one or more interconnected local area networks (LANs). Common design goals are:

- To serve a large number of users and platforms: desktops, laptops, mobile devices, supercomputers, tablets, etc.
- To support a variety of applications: email, browsing, voice, video, procurement systems, and others.
- To provide security against the multiple threats that result from the large number of applications and platforms.
- To provide a level of Quality of Service (QoS) that satisfies user expectations.

To serve multiple applications and platforms, the network is designed for general purposes. To provide an adequate security level, the network may use multiple CPU-intensive appliances. Besides a centrally-located firewall, internal firewalls are often used to add stringent filtering capability to sensitive subnetworks. The network may only provide a minimum level of QoS, which is often sufficient. The level of QoS does not need to be strict, as applications can improve on the service provided by the network. Moderate bandwidth, latency, and loss rates are most of the time acceptable, as flows have a small size (from few KBs to MBs) and a short duration. Rates of few Kbps to tens of Mbps can satisfy bandwidth requirements. Furthermore, most applications are elastic and can adapt to the bandwidth provided by the network. Similarly, packet losses can be repaired with retransmissions and jitter can be smoothed by buffering packets at the receiver.

Fig. 3 shows a typical campus enterprise network. Packets coming from the WAN are inspected by multiple inline security appliances, including a firewall and an intrusion prevention system (IPS). Further processing is performed by a network address translator (NAT). Packets traverse through the network, from core-layer routers to access-layer switches. Important components of routers and switches, such as switching fabric, forwarding mechanism, size of memory buffers, etc. are adequate for small flows only. The devices

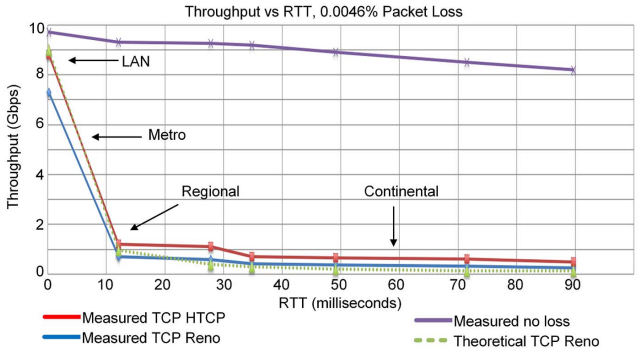

Fig. 4. Throughput vs round-trip time (RTT), for two devices connected via a 10 Gbps path. The performance of two TCP implementations are provided: Reno [32] (blue) and Hamilton TCP [33] (HTCP) (red). The theoretical performance with packet losses (green) and the measured throughput without packet losses (purple) are also shown [4].

also use processing techniques that yield poor performance when processing large flows, such as cut-through forwarding [4]. Additional security inspection by internal firewalls and distribution- and access-layer switches is common. These switches segregate LANs into virtual LANs (VLANs), requiring further frame processing and inter-VLAN routing. Further, end devices do not have the hardware nor software capabilities to send and receive data at high speeds. The bandwidth of the network interface card (NIC) and the input/output and storage systems are often below 10 Gbps. Similarly, software applications perform poorly on WAN data transfers because of limitations such as small buffer size, excessive processing overhead, and inadequate flow and congestion control algorithms.

Packet losses may occur at different locations in the enterprise network, including routers, switches, firewalls, IPS, etc. As a result of a packet loss, TCP reacts by drastically decreasing the rate at which packets are sent. The following example [4] illustrates the impact of a small packet loss rate. Fig. 4 shows the TCP throughput of a data transfer across a 10 Gbps path. The packet loss rate is 1/22,000, or 0.0046%. The purple curve is the throughput in a loss-free environment; the green curve is the theoretical throughput computed according to the following equation [34]:

$$\text{throughput} = \frac{MSS}{RTT \cdot \sqrt{L}}. \quad (1)$$

Eq. (1) indicates that the throughput of a TCP connection in steady state is directly proportional to the maximum segment size (MSS) and inversely proportional to the round-trip time (RTT) and the square root of the packet loss rate (L). The red and blue curves are real measured throughput of two popular implementations of TCP: Reno [32] and Hamilton TCP (HTCP) [33]. Because TCP interprets losses as network congestion, it reacts by decreasing the rate at which packets are sent. This problem is exacerbated as the latency increases between the communicating hosts. Beyond LAN transfers, the throughput decreases rapidly to less than 1 Gbps. This is often the case when research collaborators sharing data are geographically distributed.

B. Science DMZ Architecture

The Science DMZ is designed to address the limitations of enterprise networks and is typically deployed near the main enterprise network. It is important to highlight, however, that the two networks, the Science DMZ and the enterprise network, are separated either physically or logically. There are important reasons for this choice. First, the path from the Science DMZ to the WAN must involve as few network devices as possible, to minimize the possibility of packet losses at intermediate devices. Second, the Science DMZ can also be considered as a security architecture, because it limits the application types and corresponding flows supported by end devices. While flows in enterprise networks are numerous and diverse, those in Science DMZs are usually well-identified, enabling security policies to be tied to those flows.

A Science DMZ example is illustrated in Fig. 5(a). The main characteristics of a Science DMZ are the deployment of a friction-free path between end devices across the WAN, the use of DTNs, the active performance measurement and monitoring of the paths between the Science DMZ and the collaborator networks, and the use of access-control lists (ACLs) and offline security appliances. Specifically:

- Friction-free network path: DTNs are connected to remote systems, such as collaborators' networks, via the WAN. The high-latency path is composed of routers and switches which have large buffer sizes to absorb transitory packet bursts and prevent losses. The path has no devices that may add excessive delays or cause the packet to be delivered out of order; e.g., firewall, IPS, NAT. The rationale for this design choice is to prevent any packet loss or retransmission which can trigger a decrease in TCP throughput.
- Dedicated, high-performance DTNs: These devices are typically Linux devices built and configured for receiving WAN transfers at high speed. They use optimized data transfer tools such as Globus' gridFTP [26], [35], [36]. General-purpose applications (e.g., email clients, document editors, media players) are not installed. Having a narrow and specific set of applications simplifies the design and enforcement of security policies.
- Performance measurement and monitoring point: Typically, there is a primary high-capacity path connecting the Science DMZ with the WAN. An essential aspect is to maintain a healthy path. In particular, identifying and eliminating soft failures in the network is critical for large data transfers [4]. When soft failures occur, basic connectivity continues to exist but high throughput can no longer be achieved. Examples of soft failures include failing components and routers forwarding packets using the main CPU rather than the forwarding plane. Additionally, TCP was intentionally designed to hide transmission errors that may be caused by soft failures. As stated in RFC 793 [37], *As long as the TCPs continue to function properly and the Internet system does not become completely partitioned, no transmission errors will affect the users.* The performance measurement and monitoring point provides an automated

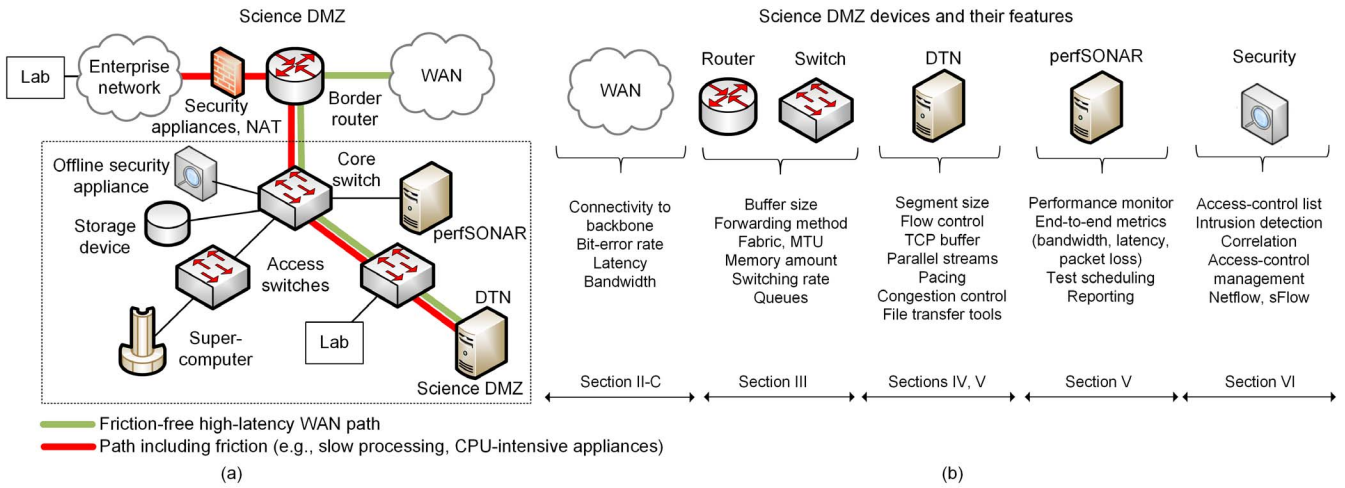

Fig. 5. Science DMZ location and device features. (a) A Science DMZ co-located with an enterprise network. Notice the absence of firewall or any stateful inline security appliance in the friction-free path. (b) Features of Science DMZ's devices.

mechanism to actively measure end-to-end metrics such as throughput, latency, and packet loss. The most used tool is perfSONAR [38], [39].

- ACLs and offline security appliances: The primary method to protect a Science DMZ is via router's ACLs. Since ACLs are implemented in the forwarding plane of a router, they do not compromise the end-to-end throughput. Additional offline appliances include payload-based and flow-based intrusion detection systems (IDSs).

In Fig. 5(a), when data sets are transferred to a DTN from the WAN, they may be stored locally at the DTN or written into a storage device. DTNs can be dual-homed, with a second interface connected to the storage device. This approach allows the DTN to simultaneously receive data from the WAN and transfer the data to the storage device, avoiding double-copying it. Users located in a laboratory inside the Science DMZ have friction-free access to the data in the storage device. On the other hand, users from a laboratory located in the enterprise network are behind the security appliances protecting that network. These users may achieve reasonable performance accessing the stored data/Science DMZ. The reason here is that, because of the very low latency between the Science DMZ and enterprise users, the retransmissions caused by the security appliances have much less performance impact. TCP recovers from packet losses quickly at low latencies (discussed in Section IV), contrasting with the slow recovery observed when packet losses are experienced in high-latency WANs. The key is to provide the long-distance TCP connections with a friction-free service.

1) Addressing the Enterprise Network Limitations: The Science DMZ addresses the limitations encountered in enterprise networks by using the coordinated set of resources shown in Fig. 5(b). At the physical layer/cyberinfrastructure, the WAN must be capable of handling large traffic volumes, with a predictable performance. Bit-error rates should be very low and congestion should not occur. The WAN path between end devices should include as few devices as possible. These requirements contrast with typical services delivered by commercial Internet Service Providers (ISPs), used in enterprise networks. ISPs often minimize operating costs at the expense of performance. For large data transfers and research purposes, many institutions are connected to regional or national backbones dedicated to supporting research and education, such as Internet2 [40].

At the data-link and network layers, the switches and routers must have a suitable architecture to forward frames/packets at a high speed (10 Gbps and above). Important attributes are the fabric, queueing, and forwarding techniques. These devices must also have large buffer sizes to absorb transient packet bursts generated by large flows. These requirements are opposite to those implemented by devices used in enterprise networks, which are driven by datacenter needs. The paths interconnecting devices inside a datacenter are characterized by a low latency. On the other hand, the paths interconnecting DTNs to remote networks are characterized by a high latency.

At the transport layer, the protocol must transfer a large amount of data between end devices without errors. TCP is the protocol used by most application-layer tools. A large amount of memory must be allocated to the TCP buffer, which permits the sender to continuously send segments to fill up the WAN capacity. Otherwise, the TCP flow control mechanism leads to a stop-and-wait behavior. The transport layer should also permit the enabling or disabling of TCP extensions, the use of large segment sizes, and the selection of the congestion control algorithm. The segment size depends on the maximum transmission unit (MTU), which is defined by the layer-2 protocol. The congestion control algorithm must be suitable for high-throughput high-latency networks, as data transfers are often conducted over WANs.

At the application layer, applications are limited to data transfer tools at the DTN and perfSONAR at the measurement and monitoring point. The prevalent data transfer tool is Globus' gridFTP [26], [35], [36]. Globus implements features such as parallel streams and re-startable data transfer. perfSONAR [38], [39] provides an automated mechanism to actively measure and report end-to-end performance metrics.

TABLE I
DIFFERENCES BETWEEN INTERNET AND INTERNET2/REN

Feature	Internet	Internet2/REN
Traffic flows	Commercial flows: millions of small flows.	Research flows: smaller number of large flows.
Bandwidth	Limited, subject to ISPs policies/throttling.	Paths of up to 100 Gbps.
Network devices	Heterogeneous environment, routers and switches are not optimized for large flows.	Routers and switches with large buffer sizes suitable for accommodating large data transfers.
Bottlenecks	Congestion and outages are common.	Clear expectations, predictable WAN performance in terms of bandwidth, latency, and packet loss.
End-to-end path monitoring	Difficult to detect and solve soft failure problems. ISPs do not typically collaborate in keeping the internetwork healthy.	Easier to detect and solve soft failure problems. Active tools, such as perfSONAR, are used in Internet2 and partner networks.
Routing	Routing is achieved independently by each ISP. Routing decisions are based on policies that minimize operating costs at the expense of performance.	Routing is optimized for performance, leading to high-throughput, shorter paths.
Frame size	The maximum frame size in routers located in an ISP is typically 1,500 bytes.	Routers within the Internet2 backbone support 9,000-byte frames. Large frame sizes increase the throughput and the recovery speed from losses.
IPv6	Support for IPv6 is not ubiquitous.	Full IPv6 support.

With respect to security, by avoiding general-purpose applications and by separating the Science DMZ from the enterprise network, specific policies can be applied to the science traffic. Also, data transfer tools are relatively simple to monitor and to secure. Security policies are implemented with ACLs and offline appliances, such as IDSs. Routers and switches also provide functionality for collecting flow information, such as Netflow [41] and sFlow [42]. Netflow is a protocol used for collecting and exporting flow information that is increasingly used for monitoring big data transfers [43]. Similarly, sFlow uses sampling to decrease the amount of collected information. At high rates, inline security appliances such as firewalls and IPSs lead to packet losses and thus are not used in Science DMZs.

C. WAN Cyberinfrastructure

The Science DMZ can be treated as the portion of the cyberinfrastructure where the end devices are located. The second piece of the cyberinfrastructure is the WAN. In the U.S., there are multiple backbones and regional networks connecting institutions and corresponding Science DMZs. The primary backbone for science and engineering is Internet2 [40]. While most of this section focuses on the cyberinfrastructure needs for large flows using Internet2 as an example, the discussion is still applicable to other Research and Education Networks (RENs). A REN is a service provider network dedicated to supporting the needs of the research and education communities within a region. A particular REN which is deployed by a country is referred to as a National Research and Education Network (NREN). Examples of RENs include Internet2 in North America, GEANT [44] in Europe, UbuntuNet [45] in East and Southern Africa, APAN [46] in the Asia-Pacific region, and RedCLARA [47] in Latin America. Internet2 and RENs may contrast with commercial ISPs and Internet in several aspects, as summarized in Table I.

Internet2 has multiple point of presences (POPs) distributed across the U.S., where institutions can connect to the network. While institutions located in the proximity of a POP can readily access a REN, others remotely located may only connect to a REN indirectly. The connection of a Science DMZ to a REN can be accomplished in different ways, including a direct connection to the REN's POP, via a regional network, or via a commercial ISP.

1) Connecting a Science DMZ via an Internet2 POP: Many research institutions and universities connect directly to Internet2 via a direct link between the Science DMZ and an Internet2 POP. This connection type minimizes the number of devices or hops between the DTN and the WAN. Additionally, Internet2 is also optimized for throughput by avoiding the use of appliances that may reduce performance. Sometimes the POP is located in the institution campus, co-located with the border router. Alternatively, the institution campus may be located a few miles/kilometers away from the POP.

2) Connecting a Science DMZ via a Regional REN: A second option to access a major backbone/Internet2 is via a regional research network, which in turn is connected to Internet2. A representative example is the Western Regional Network (WRN) [48]. The WRN is a regional 100 Gbps REN in the western part of the U.S., as shown in Fig. 6. The interconnection with Internet2 is shown in blue. Connections to the Internet are achieved by peering with a tier-1 ISP, Level 3. The WRN is also connected to other research networks such as the Corporation for Education Network Initiatives in California (CENIC) network [49] and ESnet [3].

Fig. 6 highlights the case of the University of Hawaii (UH), which has a link to the WRN. The WRN has access to Internet2 at several POPs. Although this alternative requires that flows traverse across two hierarchical levels (i.e., the WRN and Internet2), these research networks are typically optimized for performance.

3) Connecting a Science DMZ via a Commercial ISP: Most ISPs may have policies/throttling mechanisms that do not favor performance. Bottlenecks and congestion are common and clear performance expectations cannot be established, because of the lack of collaborative monitoring between ISPs. Furthermore, policy criteria tend to dominate routing decisions rather than optimization criteria.

Fig. 7(a) shows a use case of a campus enterprise network connected to the WAN via an ISP service. The lower level of the Internet hierarchy is the access ISP, whereas a second level provides connectivity to access ISPs, namely the regional ISP.

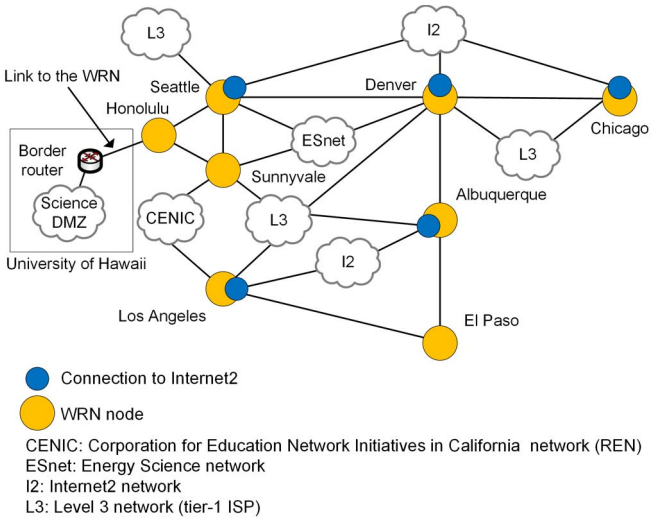

Fig. 6. A Science DMZ connected to a REN, the Western Regional Network (WRN) [48].

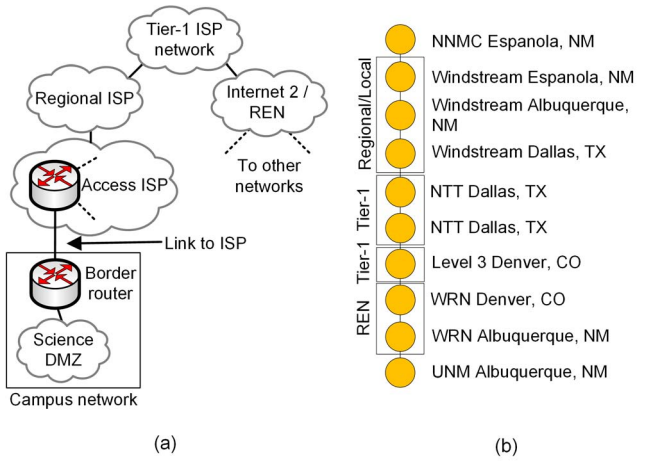

Fig. 7. Connecting a Science DMZ via an ISP. (a) A viewpoint of the connection in the Internet hierarchy. (b) The path between two Science DMZs, one attached to an ISP (NNMC) and another attached to a REN (UNM). NM, TX, and CO stand for New Mexico, Texas, and Colorado.

Sometimes, the regional ISP can also provide connectivity to the end customer, i.e., the campus network. Each regional ISP then connects to a tier-1 ISP.

Fig. 7(b) illustrates the communication between two Science DMZs in the state of New Mexico, U.S., located at Northern New Mexico College (NNMC) and at the University of New Mexico (UNM). The geographic distance between the two institutions is 90 miles (145 kilometers). NNMC is located in Espanola, where connectivity is provided by a commercial ISP. On the other hand, UNM is located in Albuquerque and has a direct connection to a REN, namely the WRN. Note the long path between the two locations, which crosses a local/regional ISP (Windstreams), two tier-1 ISPs (NTT and Level 3), and a REN (WRN). The resulting RTT is approximately 60 milliseconds.

The above example illustrates that existing routing policies at ISPs can cause excessive delays. If instead NNMC was directly connected to a REN or Internet2, or the traffic was routed more efficiently when it entered Albuquerque, the delay would only be a few milliseconds.

Fig. 8. Locations of institutions that have implemented cyberinfrastructure improvements and/or have deployed Science DMZs with the support of the NSF Campus Cyberinfrastructure program, as of 2016 [50].

4) Connecting a Science DMZ via a Commercial ISP Circuit: Science DMZs can be connected to Internet2 or a REN via layer-1 or layer-2 services provided by an ISP. A layer-1 service provides a dedicated wavelength on a fiber channel from the campus location to a POP of Internet2 or regional REN. A layer-2 service includes pseudowire emulation, Virtual Private LAN service (VPLS), and others. The advantage of this approach is that the terms of the service can be negotiated between the ISP and the institution, including a deterministic path to be followed by packets from the border router to the POP. Table II summarizes the four alternatives discussed in this section to connect Science DMZs to Internet2.

D. Current State: Science DMZ Deployment in the U.S.

The NSF recognizes the Science DMZ model as a proven operational best practice for university campuses supporting data-intensive science. This model has also been identified as eligible for funding through the NSF Campus Cyberinfrastructure program (CC*) [17]. Established in 2012, this program has funded more than 200 projects for network infrastructure deployment/Science DMZs. The locations of these institutions are shown in Fig. 8. Since a design goal of the Science DMZ is the establishment of a high-speed path across a WAN, the impact on improving the exchange of large data sets is significant. In essence, because of the data-sharing architecture of the Science DMZ, institutions implementing it have fast access to virtual co-location of large data that could reside anywhere in the world.

III. Data-Link and Network-Layer Devices

Two essential functions performed by routers are routing and forwarding. Routing refers to the determination of the route taken by packets. Forwarding refers to the switching of a packet from the input port to the appropriate output port. The term switching is also used interchangeably with forwarding.

Traditional routing approaches such as static and dynamic routing (e.g., Open Shortest Path First (OSPF) [51], BGP [52]) are used in the implementation of Science DMZs. Routing

TABLE II
ALTERNATIVE APPROACHES TO CONNECT A SCIENCE DMZ TO INTERNET2

Connection	Advantages	Disadvantages
SDMZ to Internet2 via a direct POP link	▷ Optimal technical approach; no additional hops from the Science DMZ to Internet2. ▷ Routing is optimized for performance. ▷ Internet2 has active performance monitoring. Thus, it is easier to detect soft failures.	▷ Based on location and providers, service may be more expensive than that of commercial service providers. ▷ Location; POPs to Internet2 may not be accessible to the client institution.
SDMZ to Internet2 via a regional research network	▷ If the regional research network is optimized for performance, there is a minimal performance degradation with respect to a direct link connection to Internet2 POP. ▷ Costs may be lower than that of establishing a direct link to an Internet2 POP.	▷ Additional hops are added to reach Internet2 backbone; packets must traverse at least two levels in the network hierarchy: to the research network and to Internet2.
SDMZ to Internet2 via commercial ISP circuit	▷ Resources are reserved in advanced (bandwidth), and a more predictable quality of service is guaranteed (compared to regular commercial service).	▷ Additional hops and latency are added to reach Internet2 backbone; packets traverse at least two levels in the network. ▷ Soft failures may not be easy to detect if they occur within the network of the service provider.
SDMZ to Internet2 via a regular commercial ISP	▷ Costs are typically lower than that of connecting the Science DMZ to a research network or to an Internet2 POP.	▷ Performance is unpredictable due to congestion, latency, inadequate equipment for large flows, and bandwidth policies. ▷ End-to-end path monitoring and detection of soft failures are difficult.

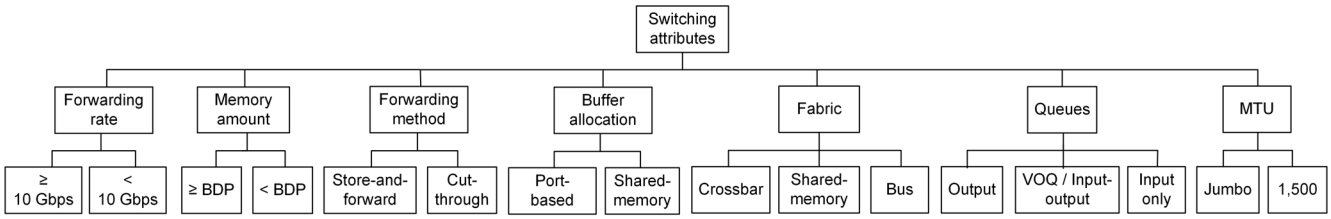

Fig. 9. Switching attributes requiring consideration in a Science DMZ.

events, such as routing table updates, occur at the millisecond, second, or minute timescale, and best practices used in regular enterprise networks are applicable to Science DMZs. On the other hand, with transmission rates of 10 Gbps and above, the forwarding operation occurs at the nanosecond timescale. Since forwarding functionality is common in both routers and switches, this section reviews the architecture and forwarding-related attributes of switches. These attributes are applicable to routers as well; thus, for the remainder of this section, the terms switch and router are used interchangeably. Switching attributes discussed in this section are illustrated in Fig. 9.

A. Switching Review

A generic router architecture is shown in Fig. 10. Modern routers may have a network processor (NP) and a table derived from the routing table in each port, which is referred to as the forwarding table (FT) or forwarding information base (FIB). The router in Fig. 10 has two input ports, iP1 and iP2, with their respective queues. iP1 has three packets in its queue, which will be forwarded to output ports oP1 (green packets) and oP2 (blue packet) by the fabric.

Router queues/buffers absorb traffic fluctuations. Even in the absence of congestion, fluctuations are present, resulting mostly from coincident traffic bursts [24]. Consider an input buffer implemented as a first-in first-out in the router of Fig. 10. As iP1 and iP2 both have one packet to be forwarded to oP1 at the front of the buffer, only one of them, say the packet at iP2, will be forwarded to oP1. The consequence of this is that not only the first packet at iP1 must wait, so too must the second packet that is queued at iP1 wait, even though there is no contention for oP2. This phenomenon is known as head-of-line (HOL) blocking [53]. To avoid HOL blocking, many switches use output buffering, a mixture of internal and output buffering, or techniques emulating output buffering such as Virtual Output Queueing (VOQ).

B. Switching Considerations for Science DMZs

There are critical switching attributes that must be considered for a well-designed Science DMZ. These attributes are related to the characteristics of the science traffic and the role of switches in mitigating packet losses. Key considerations are now presented.

1) Traffic Profile: At a switch, buffer size, forwarding or switching rate, and queues should be selected based on the traffic profile to be supported by the network. Enterprise networks and Science DMZs are subject to different traffic profiles, as listed in Table III. In a typical enterprise network, a very large number of flows consume a relatively small amount of bandwidth each. Fig. 11 shows an example of a traffic profile at a small campus enterprise network serving approximately 1,000 hosts. The number of flows observed in a week-long period is approximately 33 million, 81% TCP, 18% UDP, and 1% other protocols. According to the cumulative distribution function (CDF) of the flow duration, more than

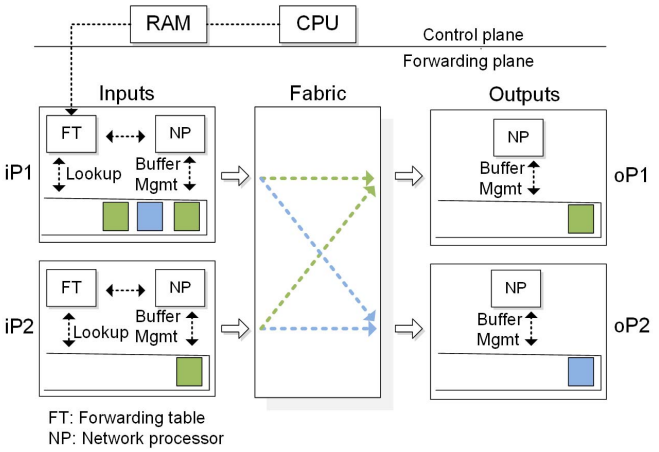

Fig. 10. A generic router architecture.

TABLE III
COMPARISON BETWEEN ENTERPRISE NETWORK AND SCIENCE DMZ FLOWS

Feature	Enterprise network flow	Science DMZ flow
Duration	Short	Long
Data size	KBs, MBs	TBs, PBs
Nature of the data	Large variety: web, email, media content, database-related, mobile applications, streaming	Files
Bursty	Yes	No
Packet loss	Less sensitive	Very sensitive
Latency	Sensitive	Less sensitive
Throughput	Less sensitive	Very sensitive
Concurrent flows	Thousands of flows per second	Few flows per second

90% of these flows have a duration of less than 200 seconds. Similarly, approximately 90% of the flows have a size of 10 KBs or less. This traffic profile is very different from that of a science flow, which may last several hours and consume the total available bandwidth. For example, transferring 100 TBs at 10 Gbps takes over 24 hours. In this type of flow, bursts occur occasionally but are not the norm. When both small and large flows are transported across the same network, smaller flows do not saturate ports. However, when bursts associated with a science flow occur, then these events can cause the starvation of the small flows [54].

Enterprise flows are less sensitive than Science DMZ flows to packet loss and throughput requirements. Typically, the size of files in enterprise applications is small. Even though packet losses reduce the TCP throughput, from a user perspective this reduction results in a modest increase of the data transfer time. On the other hand, Science DMZ applications typically transfer terabyte-scale files. Hence, even a very small packet loss rate can cause the TCP throughput to collapse below 1 Gbps, as illustrated in Fig. 4. As a result, a terabyte-scale data transfer requires many additional hours or days to complete.

A well-designed Science DMZ is minimally sensitive to latency. One of the goals of the Science DMZ is to prevent packet loss and thus to sustain high throughput over high-latency WANs. Hence, the Science DMZ uses dedicated DTNs and switches capable of absorbing transient bursts. It also

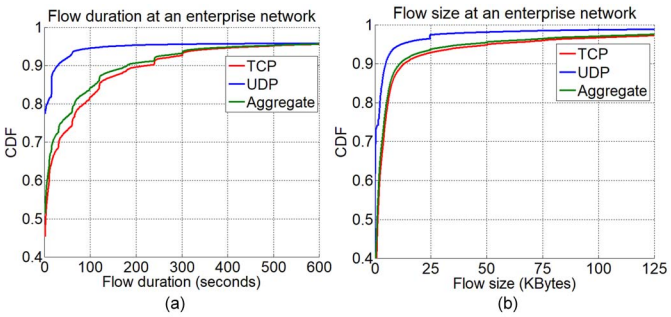

Fig. 11. A week-long (Apr. 16-22, 2018) measurement data for a small campus enterprise network. The total number of observed flows is approximately 33 million; 81% of flows are TCP, 18% UDP, and 1% other protocols. (a) Cumulative distribution function (CDF) of the flow duration and (b) the flow size. The flow duration is the time interval between the first and last packets of the flow observed in the network, whereas the flow size is the aggregate number of bytes contained in the packets of that flow.

avoids inline security appliances that may cause packets to be dropped or delivered out of order. By fulfilling these requirements, the achievable throughput can approach the full network capacity. For example, with no packet losses, the throughput illustrated in Fig. 4 approaches 10 Gbps (purple curve). Note that the throughput is only slightly sensitive to latency.

2) Maximum Transmission Unit: The MTU has a prominent impact on TCP throughput. As observed in Eq. (1), the throughput is directly proportional to the MSS. Congestion control algorithms perform multiple probes to see how much the network can handle. With high-speed networks, using half a dozen or so small probes to see how the network responds wastes a huge amount of bandwidth. Similarly, when a packet loss is detected, the rate is decreased by a factor of two. TCP can only recover slowly from this rate reduction. The speed at which the recovery occurs is proportional to the MTU (discussed in Section IV). Thus, for Science DMZs, it is recommended to use large frames.

3) Buffer Size of Output or Transmission Ports: The buffer size of a router's output port must be large enough, since packets from coincident arrivals from different input ports may be forwarded to the same output port. Additionally, buffers prevent packet losses when traffic bursts occur. A key question is how large should buffers be to absorb the fluctuations generated by large flows. The rule of thumb has been that the amount of buffering (in bits) in a router's port should equal the RTT (in seconds) multiplied by the capacity C (in bits per seconds) of the port [55], [56]:

$$\text{buffer size} = C \cdot RTT. \tag{2}$$

The above quantity is also known as the bandwidth-delay product (BDP). The rationale behind this quantity is explained in Fig. 12 [57]. In a TCP connection, a sender can have at most W_{max} in-flight or outstanding bits (or the equivalent in segments), where W_{max} is the TCP buffer size dictated by the receiver. Assume that the output port of the router is the bottleneck link of the end-to-end connection. Due to the additive increase behavior of TCP, the sender will keep increasing the rate. The number of queued packets at the router will also

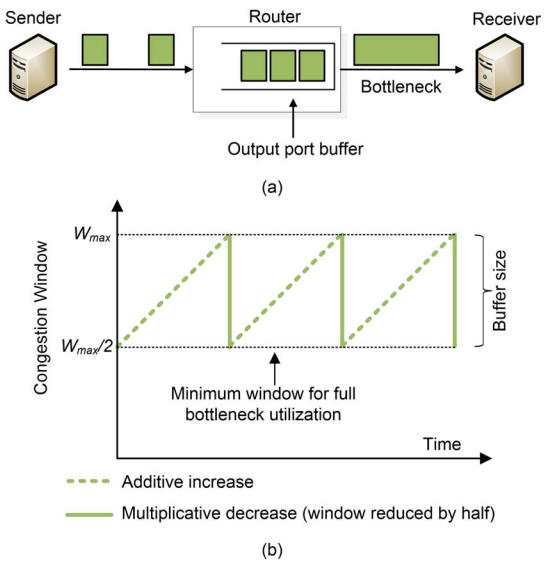

Fig. 12. TCP viewpoint of a connection and its behavior. (a) A simplified TCP interpretation of the connection. (b) The congestion control behavior characterized by the additive increase and multiplicative decrease.

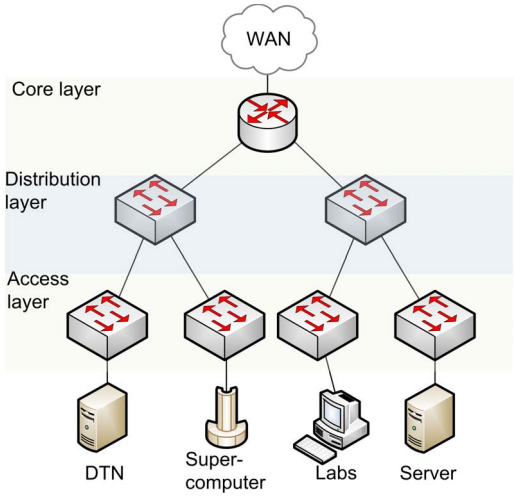

Fig. 13. Hierarchical network.

increase, until it becomes full and a packet is dropped. At that point, TCP decreases the congestion window to $\frac{W_{max}}{2}$. In order to maximize the throughput of the connection, the bottleneck link should always be utilized. With sufficient buffering, the window size is always above the critical threshold $\frac{W_{max}}{2}$. Since the buffer size is equal to the height of the TCP sawtooth [58], then the size needs to be equal to BDP as well. Notice that the buffer absorbs the changes observed in the TCP window size.

Appenzeller *et al.* [57] demonstrated that when there is a large number of TCP flows passing through a link, say *N*, the amount of buffering can be reduced to:

$$\text{buffer size} = \frac{C \cdot RTT}{\sqrt{N}}. \tag{3}$$

This result is observed when there is no dominant flow and the router aggregates thousands of flows.

Empirical results [59], [60] suggest that the buffer size of a router in a Science DMZ should equal the bandwidth-delay product. However, a formal proof remains an open research problem. The main challenge in finding an analytical solution is the mathematical complexity of queueing systems with complex packet inter-arrival times. Specifically, the network traffic exhibits high levels of burstiness and self-similarity. A critical characteristic of self-similar traffic is that there is no natural length of a burst; at every time scale ranging from a few milliseconds to minutes and hours, similar-looking traffic bursts are present. Thus, the results predicted by the *M*/*M*/1 model from queueing theory (which models packet arrivals as a Poisson process) deviate from the actual performance [61].

In this context, consider again Fig. 12(a). Assume that the router behaves like an *M*/*M*/1 queue, and *X* is the number of packets in the system. The utilization factor is defined as:

$$\rho = \frac{\text{packet arrival rate at the input port/s}}{\text{packet departure rate at the output port}}. \tag{4}$$

Note that ρ can be interpreted as the utilization of the bottleneck link. According to the *M*/*M*/1 model, the expected number of packets in the system is $\mathbb{E}(X) = \frac{\rho}{1-\rho}$, and the probability that at least *B* packets are in the system is given by ρ^B. For a link utilization of $\rho = 0.8$, the expected number of packets in the system is small, namely, $\mathbb{E}(X) = 4$. Thus, with a modest buffer size, say 60 packets, the packet drop rate would be less than $\rho^{60} = 0.0000015$. By contrast, the buffer size of a modern 10 Gbps router interface can be over 1,000,000 packets.

Modeling packet arrivals as a Poisson process severely underestimates the traffic burstiness. Traditional TCP congestion control algorithms typically send as many packets as possible at once. Hence, a potential approach to reduce the traffic burstiness (which would permit to reduce the buffer size of a router as predicted by the *M*/*M*/1 model) is to space out or *pace* packets at sender nodes. The pacing technique can be accomplished by requiring sender nodes to send packets at a fixed rate, so that they are spread over an RTT interval. Results by Beheshti *et al.* [62] indicate that high throughput can be achieved with small buffer sizes provided that short-term bursts are minimized. Notably, the first TCP congestion control algorithm based upon pacing has been recently proposed, namely, the Bottleneck Bandwidth and Round-Trip Time (BBR) algorithm [24]. Thus, studying the impact of BBR on routers' buffer size is a promising open research direction.

4) Bufferbloat: While allocating sufficient memory for buffering is desirable, it is also important to note that the term RTT in Eq. (2) depends upon the use case at hand. Hence, allocating additional unneeded buffer space may result in more latency. This undesirable latency phenomenon is known as bufferbloat [63], [64] and can be mitigated by avoiding the over-allocation of buffers. Controlling excess delay is an active research area. For example, new active-queue management techniques based on control theory have been recently proposed in [65].

5) Routers and Switches in a Hierarchical Network: Fig. 13 illustrates a typical hierarchical network. The access layer represents the network edge, where traffic enters or exits the

TABLE IV
COMPARISON BETWEEN ENTERPRISE NETWORK AND SCIENCE DMZ SWITCHES

Feature		Enterprise network switch	Science DMZ switch
Fabric	Crossbar	Recommended.	Recommended.
	Shared memory[1]	Suitable for low-latency, datacenters.	Not recommended; buffers usually cannot be allocated on a per-port basis.
	Bus	Suitable for small enterprise networks.	Not recommended; low switching capacity.
Queues	Input queue only	Not recommended; it suffers HOL blocking.	Not recommended; it suffers HOL blocking.
	Input and output queues	Adequate performance.	Adequate performance.
	VOQ	Adequate, attainable throughput approximates 100% of total capacity.	Adequate, attainable throughput approximates 100% of total capacity.
Forwarding	Cut-through	Preferred for low-latency enterprise networks.	Not recommended.
	Store-and-forward	Adequate performance.	Recommended.
Output buffer size	$\frac{RTT \cdot C}{\sqrt{N}}$	Adequate for enterprise flows.	Not sufficient to accommodate large flows.
	$RTT \cdot C$	Not needed.	Recommended; adequate to absorb large flows' bursts and changes in TCP window size.
Buffer allocation	Port-based	Adequate performance.	Recommended.
	Dynamic shared memory[1]	Adequate performance.	Not recommended.
Jumbo frame		Minimum impact for small, short duration flows.	Recommended.

[1] 1. A shared memory fabric often implies dynamic shared memory allocation for ports.

network. In Science DMZs, usually DTNs, supercomputer, and research labs have access to the network through access-layer switches. The distribution layer interfaces between the access layer and the core layer, aggregating traffic from the access layer. The core layer is the network backbone. Core routers forward traffic at very high speeds. In this simplified topology, the core is also the border router, connecting the network to the WAN.

Access-layer switches must support a range of traffic capacity needs, sometimes starting as low as 10 Mbps and reaching to as much as 100 Gbps. This wide mix can strain the choice of buffers required, particularly on output switch ports connecting to the distribution layer [66]. Specifically, buffer sizes must be large enough to absorb bursts from the end devices (DTNs, supercomputer, lab devices).

Distribution- and core-layer switches must have as much buffer space as possible to handle the bursts coming from the access-layer switches and from the WAN. Hence, attention must be paid to bandwidth capacity changes (e.g., aggregation of multiple smaller input ports into a larger output port).

Switches manufactured for datacenters may not be a good choice for Science DMZs. They often use fabrics based upon shared memory designs. In these designs, the size of the output buffers may not be tunable, which may become a key performance limitation during the transfer of large flows.

C. Switches in Enterprise Networks and Science DMZs

Table IV compares switches for enterprise networks and Science DMZs. In general, the crossbar switch fabric is suggested for Science DMZs, because of its high bandwidth. A crossbar switch is also non-blocking; a packet being forwarded to an output port will not be blocked from reaching the output port as long as no other packet is currently being forwarded to that output port. The shared memory technology usually does not allow the allocation of per-port memory for buffering. In Science DMZs, ideally output ports will be statically allocated enough memory for buffering, as suggested by Eq. (2). Although the bus technology still provides sufficient bandwidth for enterprise networks, e.g., Cisco Catalysts 6500 switches [67], its underlying time-sharing operation is not appropriate for Science DMZs. Consider now buffering; HOL blocking limits the throughput of an input-buffered switch to 59% of the theoretical maximum (which is the sum of the link bandwidths for the switch) [68]. While this technology may be acceptable for small enterprise networks, it should not be used in high-throughput high-latency environments. Science DMZs should use switches that implement output buffering, a mixture of input and output buffering, or techniques emulating output buffering such as VOQ [68].

Forwarding techniques include cut-through and store-and-forward. Cut-through switches start forwarding a packet before the entire packet has been received, normally, as soon as the destination address is processed. They are designed to avoid buffering packets and to minimize latency. Store-and-forward switches buffer the entire packet before it is forwarded to the output port.

Store-and-forward switches provide flexibility to support any mix of speeds. Consider an incoming packet traveling at 10 Gbps that must be forwarded to a 100 Gbps output port. The bit time at the input port is 10 times longer than that at the output port. In a cut-through switch, as incoming bits are processed, they are transmitted to the output port. As soon as a bit is sent out, the 100 Gbps output port is idle waiting for the next bit, which is still being received by the 10 Gbps input port. Hence, much of the 100 Gbps bandwidth would be wasted. Thus, in order to optimize the use of the available bandwidth, the cut-through switch would have to change its operation mode to store-and-forward. However, a significant throughput degradation has been observed when a cut-through switch operates as a store-and-forward switch [4]. This degradation is partially attributed to the small buffer size of a typical cut-through switch. On the other hand, a store-and-forward switch provides automatic buffering of all incoming packets. The forwarding process from a slower interface to a faster

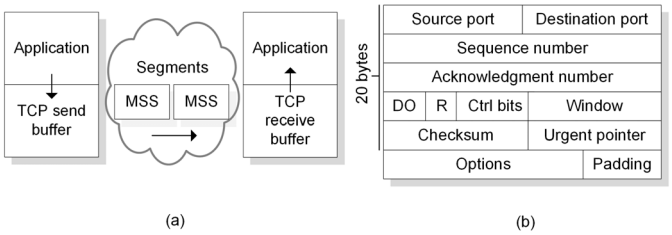

Fig. 14. TCP connection and header. (a) End points of the TCP connection. (b) TCP header. Ctrl, R, and DO fields stand for control, reserved, and data offset.

interface is made easier, as the reception process at the input port and transmission process at the output port are decoupled.

For Science DMZs, port-based buffer allocation is highly recommended. To absorb transient bursts formed by large flows, or when traffic streams are merged and multiplexed to the same output port, the amount of memory allocated to that port is recommended to be equal to the bandwidth-delay product. Many enterprise networks use switches based on dynamic shared memory. These switches deposit packets into a common memory that is shared by all ports. With dynamic shared memory, there is no guarantee that a port will be allocated an appropriate amount of memory, as this is dynamically allocated.

IV. Transport-Layer Issues

Applications transmit a large amount of data between end devices. Data must be correctly delivered from one device to another (e.g., from an instrument to a DTN). This is one of the services provided by TCP and a reason why TCP is the protocol used by data transfer tools. There are several TCP attributes that should be considered for their use in Science DMZs, including segment size, flow control and buffer size, selective acknowledgement, parallel connections, pacing, and congestion control. After a brief review of TCP, this section discusses these attributes.

A. TCP Review

TCP receives data from the application layer and places it in the TCP send buffer, as shown in Fig. 14(a). Data is typically broken into MSS units. The MSS is simply the MTU minus the combined lengths of the TCP and IP headers (typically 40 bytes). Ethernet's normal MTU is 1,500 bytes. Thus, the MSS's typical value is 1,460. The TCP header is shown in Fig. 14(b).

TCP implements flow control by requiring the receiver indicate how much spare room is available in the TCP receive buffer. For a full utilization of the path, the TCP send and receive buffers must be greater than or equal to the bandwidth-delay product. This buffer size value is the maximum number of bits that can be outstanding (in-flight) if the sender continuously sends segments.

For reliability, TCP uses two fields of the TCP header: sequence number and acknowledgement (ACK) number. The sequence number is the byte-stream number of the first byte in the segment. The acknowledgement number that the receiver

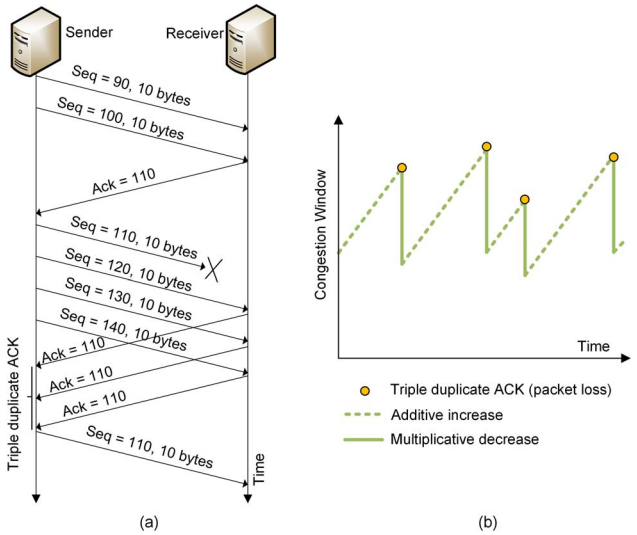

Fig. 15. TCP operation. (a) Exchange of segments between end devices. (b) Evolution of the congestion window.

puts in its segment is the sequence number of the next byte the receiver is expecting from the sender. Fig. 15(a) shows an example of the use of these two fields. If an acknowledgement for an outstanding segment is not received, TCP retransmits that segment. Alternatively, the sender can also detect a packet loss by detecting a triple duplicate ACK.

TCP maintains a congestion window whose size is the number of bytes the sender may have in the network at any time. The connection throughput is the minimum between the flow control and the congestion window, divided by the RTT. Assuming a large TCP receive buffer, the congestion window is used to adjust the rate at which the sender sends data.

B. TCP Considerations for Science DMZs

Features such as TCP buffer size and parallel streams are usually overlooked in enterprise networks, where a slight throughput degradation is often acceptable for small flows. However, inadequate transport-layer settings may have a high negative impact for large flows. These features are discussed next.

1) Maximum Segment Size: One obvious advantage of using large segments is efficiency in processing, because a 20-byte header overhead can be amortized over more data. Moreover, the recovery after a packet loss is proportional to the MSS. During the additive increase phase of the congestion control algorithm, TCP increases the congestion window by approximately one MSS every RTT. This means that by using a 9,000-byte MSS instead of a 1,500-byte MSS, the throughput increases six times faster. Even when losses are occasional, the performance improvement can be significant.

2) Flow Control and TCP Receive Buffer: TCP flow control imposes a limit in the utilization of the channel from the source to the destination. In order to maximize the utilization of the channel and increase throughput, the TCP buffer must be at least as large as the BDP, and preferably larger. By having a large TCP buffer, the sender can keep transmitting

at full speed until the first acknowledgement comes back. Increasing the TCP buffer above BDP, for example to a value that equals 2BDP, also adds robustness. Thus, if a sporadic loss occurs, TCP would decrease the window size to BDP. Therefore, after the sporadic loss, the sender would still fully utilize the channel.

For applications that use parallel collaborating TCP connections or streams in the transmission of a dataset, the TCP buffer can be reduced. This requires an application-layer software, such as gridFTP [26], [35], [69] (discussed in Section V) to orchestrate the transmission over multiple connections. Since the full bandwidth is shared by the parallel connections, the TCP buffer needs not to be equal to the BDP. Instead, it can be reduced in proportion to the number of parallel connections.

3) Selective Acknowledgment: Much of the complexity of TCP is related to inferring which packets have arrived and which packets have been lost. The cumulative acknowledgement number does not provide this information. A selective acknowledgement (SACK) lists up to three ranges of bytes that have been received. With this information, the sender can more directly decide what segments to retransmit.

The impact of using SACK on large data transfers at 10 Gbps is not conclusive. In paths with small to medium RTT, the use of SACK is encouraged in [70]. However, in paths with large RTT and bandwidth, using SACK may reduce performance. For very large BDP paths where the TCP buffer size is in the order of tens of MBs, there is a large number of in-flight segments. For example, for a TCP receive buffer of 64 MBs and a MSS of 1,500 bytes, there could be almost 45,000 outstanding segments. When a SACK event occurs, the TCP performance may be degraded by the process of locating and resending the packets listed in the SACK lists. This in turn causes TCP to trigger a timeout and to reduce the congestion window. If such issues are observed, a solution is to disable SACK.

4) Parallel TCP Connections: The advent of Science DMZs and the need to combat random packet losses have recently initiated new research in the use of parallel TCP connections for large flows [19], [71], [72]. Assuming that losses, RTT, and MSS are the same in each connection, the total throughput is essentially the aggregation of the *K* single TCP connection throughputs [73]. Since the throughput of a single TCP connection is given by Eq. (1), the aggregate throughput of *K* connections is given by the following equation:

$$\text{aggregate throughput} = \sum_{i=1}^{K} \frac{MSS}{RTT\sqrt{L}} = K\frac{MSS}{RTT\sqrt{L}}. \tag{5}$$

Thus, an application opening *K* parallel TCP connections essentially creates a large virtual MSS on the aggregate connection that is *K* times the MSS of a single connection. A larger MSS increases the rate of recovery from a loss event from one MSS per successful segment transmission to *K* MSSs per successful segment transmission. When the aggregate TCP connection begins to create congestion, any router or switch along the path begins dropping packets and Eq. (5) is no longer valid. Parallel TCP connections must be implemented and managed by the application layer. Its use is further discussed in Section V.

5) TCP Fair Queue Pacing: Data transmissions can be bursty, resulting in packets being buffered at routers and switches and dropped at times. End devices can contribute to the problem by sending a large number of packets in a short period of time. If those packets were transmitted at a steady pace, the formation of queues could be reduced.

TCP pacing is a technique by which a transmitter evenly spaces or paces packets at a pre-configured rate. TCP pacing has been applied for years in enterprise networks [74], with mixed results. However, its recent application to data transfers in Science DMZs suggests that its use has several advantages [75]. TCP pacing has also been applied to datacenter environments [76].

The existing TCP congestion control algorithms, with the exception of BBR [24], indicate how much data is allowed for transmission. Those algorithms do not provide a time period over which that data should be transmitted and how the data should be spread to mitigate potential bursts. The rate, however, can be enforced by a packet scheduler such as a fair queue (FQ). The packet scheduler organizes the flow of packets of each TCP connection through the network stack to meet policy objectives. Some Linux distributions such as CentOS [77] implement FQ scheduling in conjunction with TCP pacing [24], [78].

FQ is intended for locally generated traffic (e.g., a sender DTN). Fig. 16 illustrates the operation of FQ pacing. Application 1 generates green packets, and application 2 generates blue packets. Each application opens a TCP connection. FQ paces each connection according to the desired rate, evenly spacing out packets within an application based on the desired rate. The periods T_1 and T_2 represent the time-space used for connections 1 and 2 respectively.

TCP pacing reduces the typical TCP sawtooth behavior [60] and is effective when there are rate mismatches along the path between the sender and the receiver. This is the case, for example, when the ingress port of a router has a capacity of 100 Gbps, and the egress port has a capacity of 10 Gbps. Because of the TCP congestion control mechanism, the sawtooth behavior always emerges. As TCP continues to increase the size of the congestion window, eventually the bottleneck link becomes full while the rest of the links become underutilized. These mismatches produce a continuous circle of additive increases and multiplicative decreases [60].

6) TCP Congestion Control Algorithms: A loss-based signal is still the main mechanism used to adjust the congestion window and thus the throughput. The key difference among loss-based congestion control algorithms is the strategy after a packet loss is detected. The rate at which the congestion window grows after the loss may follow different mathematical functions. Examples include Reno [32], Cubic [79], and HTCP [33]. Reno uses a linear rate increase while Cubic and HTCP use cubic and quadratic functions.

Essentially, the main issue observed in high-speed networks and Science DMZs is that, after a packet loss, the additive increase is too slow to reach full speed. Consider Fig. 17(a), which shows a TCP's viewpoint of a connection. At any time,

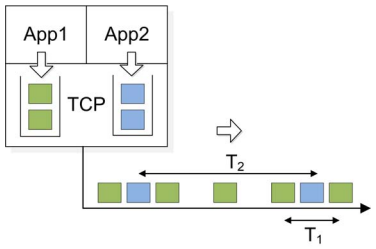

Fig. 16. TCP pacing. Packets of application 1 and application 2 are evenly spaced by T_1 and T_2 time units respectively.

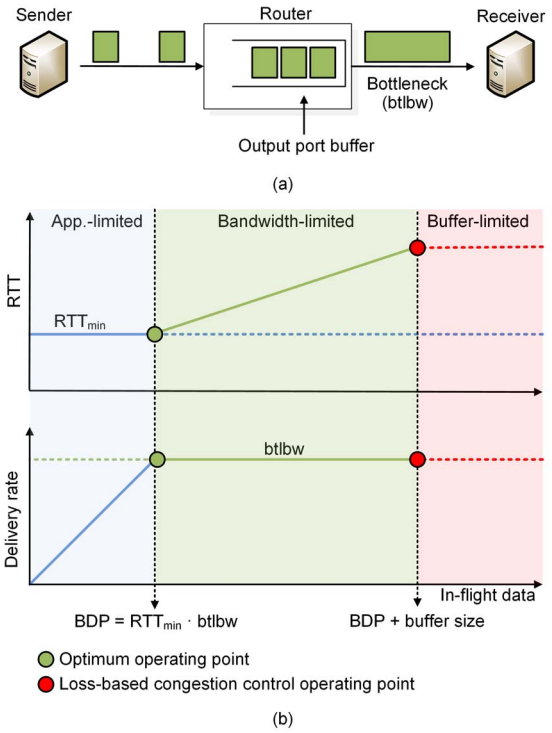

Fig. 17. TCP viewpoint of a connection and relation between throughput and RTT. (a) Simplified TCP interpretation of the connection. (b) Throughput and RTT, as a function of in-flight data [24].

the connection has exactly one slowest link or bottleneck bandwidth (btlbw) that determines the location where queues are formed. When the router's buffer is large, the loss-based congestion control keeps it full. When the router's buffer is small, the loss-based congestion control misinterprets a packet loss as a signal of congestion, leading to low throughput. The output port queue increases when the input link arrival rate exceeds btlbw. The throughput of loss-based congestion control algorithms is less than btlbw, because of the frequent packet losses [24].

Fig. 17(b) illustrates the RTT and delivery rate as functions of the amount of data in-flight [24]. RTTmin is the minimum RTT, when no congestion exists. In the application-limited region, the delivery rate/throughput increases as the amount of data generated by the application layer increases, while the RTT remains constant. The pipeline between the sender and the receiver becomes full when the in-flight number of bits is equal to BDP, at the edge of the bandwidth-limited region. The queue size starts increasing, resulting in an increase of the RTT. The delivery rate/throughput remains constant, as the bottleneck link is fully utilized. Finally, when no buffer is available at the router to store arriving packets (the amount of in-flight bits is equal to BDP plus the buffer size of the router), then packets are dropped.

BBR, the recently proposed congestion control algorithm [24], is a disruption of previous algorithms in that the control is based on the rate rather than on the window. At any one time, BBR sends at a given calculated rate, instead of sending new data in response to each received acknowledgement. BBR attempts to find the optimal operating point, shown as a green dot in Fig. 17(b), by estimating RTTmin and btlbw.

A natural question is how well does BBR, a rate-based congestion control algorithm, perform with respect to a Science-DMZ recommended traditional loss-based congestion control? Preliminary results indicate that BBR shows better performance than traditional loss-based congestion control algorithms when packet losses occur. Of particular interest to Science DMZs is the range of corruption before the throughput completely collapses. The results, which are presented in Section VII, show that BBR can achieve a better performance than loss-based congestion control algorithms. Specifically, BBR can tolerate a larger rate of packet losses before the throughput collapses.

C. Transport Layer Issues in Enterprise Networks and Science DMZs

Table V shows a comparison between enterprise networks and Science DMZs, regarding transport-layer features.

Reliability is required for file and dataset transfers and therefore, Science DMZ applications use TCP. While TLS [80] and SSL [81] also offer reliable service and security on top of TCP, they introduce additional overhead and a redundant service. Globus, a well-known application-layer tool for transferring large files, offers confidentiality, integrity and authentication services.

The flow control rate is managed by the TCP buffer size. For Science DMZ applications, the buffer size must be greater than or equal to the bandwidth-delay product. With this buffer size, TCP behaves as a pipelined protocol. On the other hand, general-purpose applications often use a small buffer size, which produces a stop-and-wait behavior.

The study of congestion control algorithms is an active research area. Although the traditional window-based loss-based congestion control may not be appropriate for modern enterprise networks, there were no alternatives until recently, and thus its use can be labeled as indifferent. However, recent preliminary results, including those presented in Section VII, indicate that BBR performs better than window-based loss-based algorithms.

If the TCP buffer size at DTNs is smaller than the bandwidth-delay product, the utilization of the channel is lower than 100%. The sender must constantly wait for acknowledgement segments before transmitting additional data segments. On the other hand, if the buffer size is greater than

TABLE V
COMPARISON OF TRANSPORT-LAYER FEATURES IN ENTERPRISE NETWORKS AND SCIENCE DMZS

Feature		Enterprise network	Science DMZ
Protocol	TCP	Used in applications requiring reliability; e.g., email, http.	Used for main applications: data transfers.
	UDP	Used in applications that do not require reliability; e.g., voice and video	Not used.
	TLS/SSL	Used in applications requiring security; e.g., online banking.	Not recommended; it adds an additional flow control layer.
Flow control	Pipelined	Not necessary; BDP is typically small.	Recommended; data transfers occur across high-throughput high-latency networks.
	Stop-and-wait behavior	Not recommended, but performance is not dramatically impacted when RTT is small.	Not recommended; throughput is severely reduced.
Use of SACK	With large MSS	Indifferent.	Results suggest the use of SACK may reduce throughput, especially when RTT is large.
	With small MSS	Throughput is slightly improved, in particular when RTT is small.	Indifferent.
Congestion control	Window-based	Only alternative until recently.	Only alternative until recently.
	Rate-based	Performance evaluations of BBR indicate an increase in throughput in small flows. Under severe-loss scenarios, throughput can be much larger than that of window-based algorithms.	Under very low-loss scenarios, BBR's performance is between 2-3% lower than that of window-based. In lossy scenarios, performance is superior than that of window-based.
TCP buffer size $\geq$ bandwidth-delay product		Not essential for small RTT.	Required for a full utilization of the end-to-end path.
Large MSS		Not essential for small RTT.	Recommended; it speeds up the recovery of the congestion window.
TCP pacing		Encouraging results when bottleneck bandwidth is known or can be estimated.	Encouraging results when bottleneck bandwidth is known or can be estimated.
Parallel streams		Impact in small flows is not substantial.	Recommended; it minimizes the impact of packet losses.

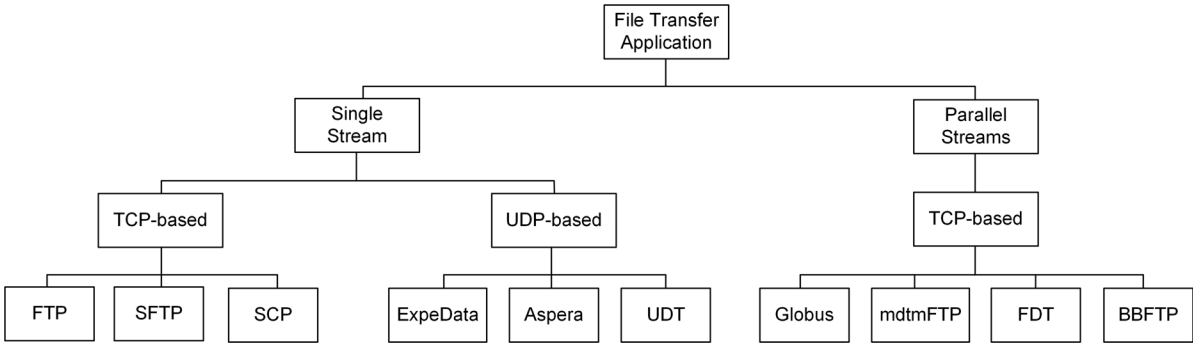

Fig. 18. Taxonomy for data transfer applications.

or equal to the bandwidth-delay product, the path utilization approaches the maximum capacity and many data segments are allowed to be in transit while acknowledgement segments are simultaneously received. For small and short-duration flows, this may not be essential. However, for large science flows, to achieve full performance, the buffer size must be at least equal to the bandwidth-delay product. The MSS is perhaps one of the most important features in high-throughput high-latency networks with packet losses. TCP pacing is a promising feature. The challenge for its wide adoption is the complexity of developing a mechanism to discover the bottleneck link and its capacity.

V. APPLICATION-LAYER TOOLS

The essential end devices inside a Science DMZ are the DTNs and the performance monitoring stations. DTNs run a data transfer tool while monitoring stations run a performance monitoring application, typically perfSONAR. Other useful tools at deployment and evaluation times are WAN emulation and throughput measurement applications. These tools are convenient because they facilitate early performance evaluation without a need of connecting the Science DMZ to a real WAN. Additionally, in contrast to enterprise networks, virtualization technologies have not been adopted in Science DMZs, because of performance limitations.

This section provides an overview of application-layer tools used in Science DMZs. The section also discusses the performance limitations of virtualization technologies preventing their adoption in Science DMZs.

A. File Transfer Applications

File transfer applications are used by researchers and practitioners to share data. Historically, applications were built around the File Transport Protocol (FTP) [82]. While FTP-based applications work well in enterprise networks, their performance in high-throughput, high-latency environments is often poor. Fig. 18 presents a taxonomy for file transfer applications.

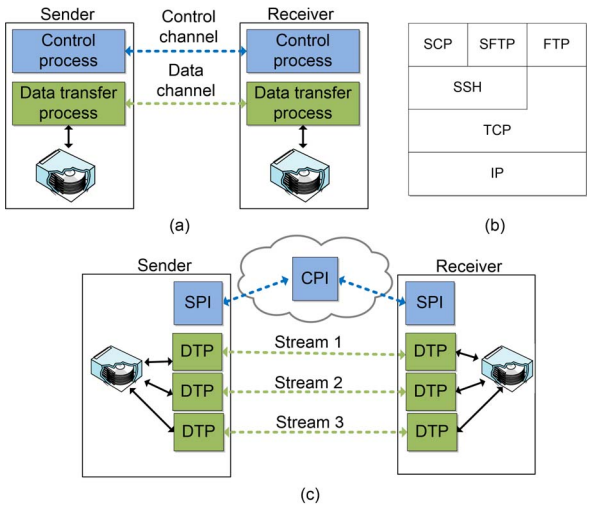

Fig. 19. Data transfer models. (a) FTP model. (b) Location of file transfer protocols in the protocol stack. (c) Globus model. Control information is exchanged between Client Protocol Interpreter (CPI) and Server Protocol Interpreter (SPI). Data transfer processes (DTPs) exchange actual data.

1) Traditional File Transfer Applications: Fig. 19(a) shows the basic FTP model. It uses two TCP connections: a data channel and a control channel. FTP has several limitations when used in Science DMZs, including limited negotiation capability of the TCP buffer size, poor throughput performance in long fat networks, lack of uniform interfaces between data transfer processes and sources and sinks (local hard disks, parallel file systems, distributed data sources, etc.), and lack of support for partial file transfer and restartable transfer.

Other file transfer protocols used in enterprise networks include Secure Copy (SCP) and Secure FTP (SFTP). These protocols are implemented above the Secure Shell protocol (SSH) [83], which in turn is implemented above TCP. Fig. 19(b) shows their respective location in the protocol stack. When a file transfer is performed by SCP or SFTP, an SSH channel is open between the end points. This channel uses a window-based flow control. Even though this feature works well for enterprise file transfers, it constitutes another rate limitation for large science flows.

2) File Transfer Applications for Science DMZs: The prevalent tool for science data transfers is Globus gridFTP. As of 2017, there are over 40,000 Globus end points deployed [26]. While the following description corresponds to Globus, many of its features apply to other applications recommended for Science DMZs.

GridFTP is an extension of FTP for high-speed networks. Globus is an implementation of gridFTP [26], [35] and its architecture is shown in Fig. 19(c). Globus has the following features.

- The control channel is established between the client protocol interpreter (CPI), a third party located in the cloud, and the server protocol interpreters (SPIs).
- Multiple parallel TCP connections are supported. These connections are referred to as streams and constitute the data channels. Typical values are between 2-8 streams.
- Globus includes support for partial and restartable file transfers [19]. Science DMZs are used for transferring large data files, which may take hours. If a disruption occurs momentarily, it is beneficial to transfer just the remaining portion of a file.
- The maximum size of the TCP buffer can be explicitly adjusted.

Other file transfer applications for big data are Multicore-Aware Data Transfer Middleware FTP (mdtmFTP) [84] and Fast Data Transport (FDT) [85]. mdtmFTP is designed to efficiently use the multiple computing cores (multicore CPUs) on a single chip that are common in modern computer systems. mdtmFTP also improves the throughput in DTNs that use a non-uniform memory access (NUMA) model. In the traditional Uniform Memory Access (UMA) model, the access to the RAM from any CPU or core takes the same amount of time. However, with NUMA, accessing some parts of memory by a core may take longer than other parts, creating a performance penalty. FDT is an application optimized for the transfer of a large number of files [85]. Hence, thousands of files can be sent continuously, without restarting the network transfer between files. However, FDT and mdtmFTP have not been widely adopted despite encouraging performance results [84].

Table VI lists additional data transfer applications and implemented features. Besides TCP-based applications, three UDP-based applications are listed. Here, a common feature in UDP-based applications is the use of a single UDP stream, rather than multiple streams. A key limitation observed in Science DMZs when using a single UDP stream is CPU-related. Namely, in multicore processor architectures, a UDP stream is adhered to a single core, which may become saturated. As a result, the UDP transmission rate can be lower than the available bandwidth. At the same time, other cores may be idle and underutilized [91]. UDP-based applications do not use congestion windows for congestion control. Instead, they use rate-based congestion control, similar to BBR [24]. Here, congestion is signaled by an increase in the RTT, which triggers a decrease in the transmission rate. ExpeData [88] and Aspera Fast [86] are two proprietary implementations and some details are not available. Some specialized applications are used in enterprise environments, but their performance for big science data transfers is not available.

B. Virtual Machines and Science DMZs

The idea behind a virtual machine is to abstract the hardware of a computer into several execution environments. As a physical resource, access to a NIC is also shared. In this context, Fig. 20 shows a sample topology with three hosts. Host 1 contains three virtual machines connected by a virtual switch. One virtual machine is a DTN. Hosts 2 and 3 are native (non-virtual) DTNs. A virtual switch is implemented inside the hypervisor. Similar to a physical switch, the virtual switch constructs its own forwarding table and forwards frames at the data-link layer. The virtual switch also connects to the external network through a physical NIC.

A virtual machine is connected to the internal network through a virtual NIC. There are different types of virtual NICs, including the following:

TABLE VI
FEATURES OF VARIOUS DATA TRANSFER APPLICATIONS. U INDICATES UNKNOWN

Application	Transport protocol	Adjustable buffer size	Parallel streams	Partial file transfer	Re-startable file transfer	Security	Sharing and publishing	Adoption	SDMZ recommended
FTP, SCP, SFTP [83]	TCP	No	No	No	No	Yes; SCP, SFTP	No	High; enterprise networks	No
Globus [26], [35]	TCP	Yes	Yes	Yes	Yes	Yes, algorithm is determined via openSSL, based on DTNs capability	Yes	High; universities and research centers	Recommended, high adoption and available support
mdtmFTP [84]	TCP	Yes	Yes	No	No	Yes, U	No	Low	Acceptable; limited support
FDT [85]	TCP	Yes	Yes	No	Yes	No[1]	No	Low	Acceptable; limited support
Aspera Fast [86]	UDP	No	Yes	No	No	Yes; Advanced Encryption Standard (AES) 128	No	Medium; enterprise networks	Unknown performance
BBFTP [87]	TCP	Yes	Yes	No	No	No	No	Low	Unknown performance
ExpeData [88]	UDP	U	U	U	U	Yes; AES 128	No	Medium; enterprise networks	Unknown performance
UDT [89], [90]	UDP	No	Yes	No	No	No	No	Low	No; lack of parallel streams

[1]Security can be incorporated via third party software package.
U: Unknown.

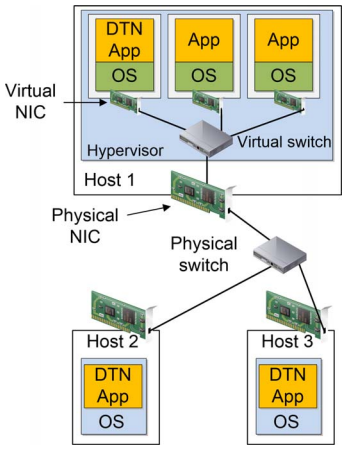

Fig. 20. Network topology including a virtual DTN contained in host 1 and two native (non-virtual) DTNs, host 2 and host 3.

- E1000: an emulated version of the Intel 82545EM Gigabit Ethernet NIC. Older (Linux and Windows) guest operating systems use this virtual NIC.
- E1000e: this virtual NIC emulates newer models of Intel Gigabit NICs. It is the default virtual NIC for newer (Windows) guest operating systems.
- VMXNET: this virtual NIC has no physical counterpart. There are two enhanced versions, VMXNET2 and VMXNET3. The latter is recommended for high-speed data transfers [92].

While virtual technologies have been widely adopted in enterprise networks, their use in Science DMZs has been discouraged for several reasons. First, the hypervisor represents a software layer that adds processing overhead. Second, the physical NIC is potentially shared among multiple virtual machines. Third, even if the virtual DTN is the only virtual machine running on a physical server, the CPU must be shared with the hypervisor and the virtual switch. Moreover, commercial vendors may not disclose important attributes of the virtual switch, such as buffer size and switching architecture.

Based on the above limitations, virtualization is not recommended for Science DMZs operating at speeds above 10 Gbps. For Science DMZs operating at 10 Gbps, preliminary results in Section VII suggest that virtual DTNs may achieve an acceptable performance, provided the physical server they run on has a high CPU capacity and the workload is controlled.

C. Monitoring and Performance Applications for Science DMZs

One of the essential elements of a Science DMZ is the performance measurement and monitoring point. The monitoring process in Science DMZs focuses on multi-domain end-to-end performance metrics. On the other hand, the monitoring process in enterprise networks focuses on single-domain performance metrics. Accordingly, Fig. 21 presents monitoring applications: perfSONAR [38], [39], Simple Network Management Protocol (SNMP) [93], Syslog [94], and Netflow [41]. The latter is also used for security purposes and is discussed in Section VI.

1) PerfSONAR: perfSONAR [38], [39] is an application that helps locate network failures and maintain optimal end-to-end usage expectations. Each organization deciding to use this

TABLE VII
COMPARISON BETWEEN SNMP AND PERFSONAR

Feature	SNMP	perfSONAR
Main uses	Enterprise networks: offices, campuses, commercial ISPs.	Science DMZ, RENs.
Scope	Single-domain.	Multi-domain.
Network monitoring under controlled load	Difficult; SNMP agents can collect statistics or report events.	Easy; perfSONAR is composed of several active testing tools.
Performance instrumentation	Difficult; SNMP uses polling to track individual network elements rather than end-to-end performance.	Easy, perfSONAR's probing tests measure end-to-end performance.
Soft failure detection	Difficult; failures could be inferred locally only through polling byte counters.	Easier; multi-domain visibility and active monitoring from the local network to any deployed perfSONAR node.
End-to-end failure detection	Difficult; limited multi-domain visibility.	Easier; variety of end-to-end tools for performance and troubleshooting; e.g., One-Way Active Measurement Protocol (OWAMP), Bandwidth Test Controller (BWCTL), Network Path and Application Diagnostics (NPAD).
Sub-path testing	No.	Yes; perfSONAR's NPAD tool allows the testing of portions of paths.
Hard failure	Easier; SNMP can report on asynchronous events via trap messages.	Difficult; perfSONAR does not report asynchronous events.
Measurable variables	CPU usage, packet counters, dropped packets, number of flows.	Bandwidth, latency, packet loss, jitter.
Schedulable tests	No.	Yes; pScheduler.
Programmability in configuration and task specification	Commercial products are available, but custom coding to automatically configure/test devices may be required.	Easier; it supports jq [95], a command-line JSON [96] processor for parsing and processing commands.
Confidentiality, integrity and authentication	Yes; SNMPv3.	No.
Reporting	Yes; multiple tools for automatic generation of reports. Usually, reports are for single-domain only.	Yes; automatic generation of reports and dashboards for end-to-end multi-domain paths; esmond stores and reports time-series measurements.

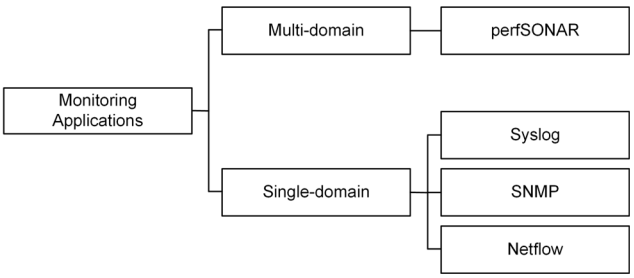

Fig. 21. Monitoring applications.

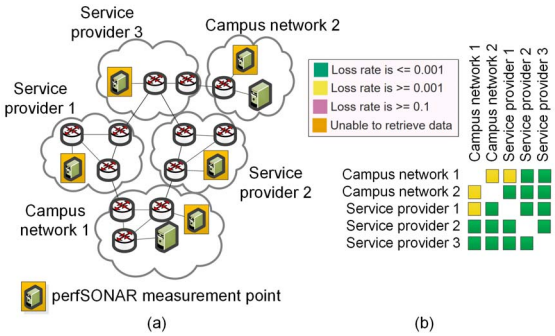

Fig. 22. perfSONAR application. (a) Multi-domain topology. Each network has a perfSONAR node. (b) Corresponding perfSONAR dashboard.

tool is required to install a measurement point in its network, as shown in Fig. 22(a). The service providers 1, 2, and 3 provide connectivity to campus networks 1 and 2. A measurement point is a Linux machine running the perfSONAR application. perfSONAR offers several services, including automated bandwidth tests and diagnostic tools.

One of the main features of perfSONAR is its cooperative nature by which an institution can measure several metrics (e.g., throughput, latency, packet loss) to different intermediary domains and to a destination network. Using the example of Fig. 22(a), campus network 1 can measure metrics from itself to campus network 2. Campus network 1 can also measure metrics to the service providers. Fig. 22(b) shows a sample dashboard view for packet loss rate for the perfSONAR node at campus network 1.

Given the increasing number of Science DMZs, perfSONAR has seen a steady increase in deployments. Currently, there are more than 2,000 perfSONAR measurement points deployed around the world. In the U.S., most Science DMZs include at least one perfSONAR node. Fig. 23 shows the location of perfSONAR nodes as of June 2017.

2) Comparison of Monitoring Applications in Enterprise Networks and Science DMZs: The ubiquitous SNMP protocol is also widely used for monitoring purposes. Accordingly, Table VII compares SNMP and perfSONAR. Overall, their functionalities are complementary and a well-monitored Science DMZ may include both. SNMP is used to monitor a single administrative domain; thus it lacks ability to detect failures beyond the local domain. Also SNMP can only infer, to some extent, a performance metric based on polling of individual network elements. Meanwhile, perfSONAR includes a set of active testing tools to measure performance via probing. End-to-end and soft failures can be detected with perfSONAR because of its multi-domain characteristic, sub-path testing, and end-to-end tools. On the other hand, some hard failures

TABLE VIII
COMPARISON OF DATA TRANSFER AND MONITORING APPLICATIONS, AND VIRTUALIZATION USE IN ENTERPRISE NETWORKS AND SCIENCE DMZ

Data transfer application		
Feature	Enterprise network	Science DMZ
Rates	Tens of Mbps to few Gbps.	10 Gbps and above.
Transport protocol	UDP, TCP, TSL/SSL.	TCP.
Partial and re-startable transfer	Usually not required.	Highly desirable.
Management of parallel streams	Not required.	Required.
Parallel file system	Typically not used nor required.	Highly desirable; provides parallelism opportunities and higher rates.
Sharing and publishing	Mature tools, high adoption; e.g., Google drive, Dropbox.	Maturing feature, in developing phase; e.g., Globus.
Security	Mature feature, supported with HTTPS and TLS/SSL.	Maturing feature, not fully in compliance with rules and regulations yet.
Data synchronization between repositories	Supported (e.g., Google drive, Dropbox).	Minimal supported; manual procedure required.
Monitoring application		
Feature	Enterprise network	Science DMZ
Monitoring scope	Single-domain.	Multi-domain.
Soft failure detection	Desirable but not essential.	Highly desirable.
Sub-path testing	Not required; path in typical switched LAN environments often are single hop.	Highly desirable; paths are typically composed of many hops in multiple domains.
Hard failure detection	Easy, highly granular; e.g., more than 6,000 Syslog events and 90 SNMP trap notifications in enterprise devices [103].	Few available features.
Monitored network type	Focus on LANs and/or interconnected LANs.	Focus on inter-networks composed of LANs and WANs.
Virtualization technology		
Feature	Enterprise network	Science DMZ
Virtual host	High adoption: server consolidation, multiple execution environments, mobility.	Low adoption, limited need for consolidation (often data transfer and perfSONAR applications only); performance penalty.
Virtual switch	High adoption; virtual switch used with VLANs to isolate VMs.	Low adoption, unavailability of buffer capability and configuration; performance penalty.
Virtual router	Medium adoption; new technology (e.g., NSX [99]), suitable for east-west traffic routing in datacenters.	Low adoption, not required; performance penalty.
virtual NIC	High adoption; E1000e, VMXNET, VMXNET2, VMXNET3 [92].	Low adoption; e.g., VMXNET3 [92] supports 10 Gbps rates.
Protocols for network virtualization	Used for LAN management, e.g., 802.1Q, overlay VXLAN [100].	Used for resource reservation in WANs, e.g., OSCARS [101], MPLS [102].

are easily detected by SNMP while they may not be detected quickly by perfSONAR. While reporting applications are available for both, perfSONAR's reports include multi-domain results. Regarding security, SNMPv3 includes confidentiality, integrity and authentication.

D. WAN Emulation and Other Performance Applications

When deploying a Science DMZ, routers, switches, and DTNs should be tested. Problems associated with routers and switches may not be observed in a testing environment unless WAN conditions, such as delay and jitter, are introduced. Thus, inadequate buffer sizes can easily be overlooked. Hence for testing purposes, in the absence of a WAN, a useful alternative is a network emulator. With such a tool, applications and devices can be tested over a virtual network. Now, two applications widely used to emulate a WAN are netem [97] and iPerf [98]. netem is a Linux application that emulates the properties of a WAN and permits to vary parameters such as delay, jitter, packet loss, and duplication and re-ordering of packets. Meanwhile, iPerf measures memory-to-memory throughput from a client (sender) to a server (receiver). The client generates dummy application-layer data in main memory, which is then moved down through the protocol stack and over the network media. The server receives the data and moves it up through the protocol stack. The two applications, netem and iPerf, can be used together to emulate data transfers between DTNs and test TCP parameters (congestion control algorithms, buffer size, TCP extensions), routers, and switches.

Fig. 23. perfSONAR nodes deployed as of June 2017.

E. Applications in Enterprise Networks and Science DMZs

Table VIII compares data transfer and monitoring applications used in enterprise networks and Science DMZs. The use

of virtualization in both environments is also compared. Owing to the nature and duration of data transfers, Science DMZ applications should incorporate features such as partial and re-startable transfers. In addition, features to combat packet losses are important, including the use and orchestration of parallel streams. On the other hand, data synchronization is a mature feature already implemented by applications used in enterprise networks.

Virtualization has not been adopted for Science DMZ deployments. The main concern is the performance penalty associated with virtual devices. Additionally, although products such as NSX [99] perform well in enterprise networks, the capacity and architecture of virtual routers and switches (switching rate, buffer size, fabric) are often not available.

In enterprise networks, protocols for network virtualization are mostly used for LAN management. Examples include 802.1Q and Virtual Extensible LAN (VXLAN) [100]. In Science DMZs, protocols and platforms such as On-demand Secure Circuits and Reservation System (OSCARS) [101] and Multi-Protocol Label Switching (MPLS) [102] are used for resource reservation and creation of virtual circuits across WANs.

VI. Security

Security is also a growing concern in Science DMZs. Hence, associated security problems can be divided in operational security, confidentiality, integrity, and authentication:

- Operational security: attackers can attempt unauthorized access, introduce malware into devices, and conduct denial of service (DoS) attacks. ACLs, firewalls, IPSs, and IDSs are commonly used to counter attacks.
- Confidentiality: only the sender and the intended receiver should understand the contents of the transmitted message. This requires that the message be encrypted.
- Integrity: the content of the communication between the sender and the intended receiver must not be altered, maliciously or by accident. Hash functions are used for integrity control.
- Authentication: the sender and the receiver should confirm the identity of the other party. Authentication methods typically rely on pre-shared key and digital signatures.

Of the above four areas, operational security is the most relevant at the time of designing and deploying a Science DMZ. The remaining three areas (confidentiality, integrity, and authentication) can be implemented at different layers, including relying on the application layer for these services.

Table IX lists security-related differences between enterprise networks and Science DMZs. The volume distribution differs substantially, as typically there are few simultaneous flows in a Science DMZ, whereas there can be thousands or millions of small flows in an enterprise network. There are a variety of applications in enterprise networks while there are only a couple in Science DMZs, namely, data transfer and performance monitoring tools. As a result, delivery attack options are abundant in enterprise networks (e.g., cross-site scripting, SQL injection, XML injection, etc.). By contrast, Science DMZs only see specific data transfer and performance monitoring tools, such as Globus and perfSONAR. Hence, the number of open ports is not an indicator of risk, as a large number of exploits in enterprise networks are delivered via ports 80 and 443. On the other hand, Globus often requires hundreds of open ports for data channels. Also, the number of operations executed on data in a Science DMZ is small. In addition to having hundreds or thousands of servers and desktops, most enterprise networks adopt a bring-your-on-device (BYOD) policy [104], which allows users to use their personal mobile device. BYOD represents additional risks, because mobile devices include a large variety of applications and operating systems with their respective vulnerabilities.

TABLE IX
Security-Related Differences Between Enterprise Networks and Science DMZs

Feature	Enterprise network	Science DMZ
Volume	Thousands of concurrent small flows.	Typically few concurrent large flows.
Application type	Web, emails, HTML, XML, mobile applications, media content, SQL, etc.	Data transfers, performance monitoring.
Most used ports	80 (HTTP), 443 (HTTPS).	2811 (Globus control channel), 50,000 to 51,000 (Globus data channels).
Operations over data on used ports	Multimedia, image processing, games, mobile code execution, HTML, XML, SQL operations.	File operations: open, read, write, close.
Number of devices	Typically hundreds to thousands.	Few, it could be even a single DTN.
Bring-your-own-device policy	Yes.	No.
Operating systems and platforms	A variety of OSs and platforms, including Windows, Linux, MAC, RIM Blackberry, Android, Windows Mobile, Oracle, Kindle.	Typically only Linux (e.g., CentOS) for all DTNs and perfSONAR nodes.
Application changes and updates	Continuous changes in applications and operating systems updates.	Changes are not frequent.

Inline devices are discouraged in Science DMZs, since they check each packet in real time. On the other hand, offline devices operate with copies of packets and do not interfere with traffic flows. Fig. 24 illustrates the typical placement of security appliances. A taxonomy of security appliances discussed next is shown in Fig. 25.

A. Operational Security for Science DMZs

1) Network Segregation: Since traffic characteristics and security policies differ between a Science DMZ and an enterprise network, their segregation is natural. Hence, when implemented contiguously, the two networks must either be physically or logically separated. Fig. 5(a) shows an example of physical separation, where the two networks are attached to different interfaces at the border router. Note that traffic flowing into the campus enterprise network is subject to inspection

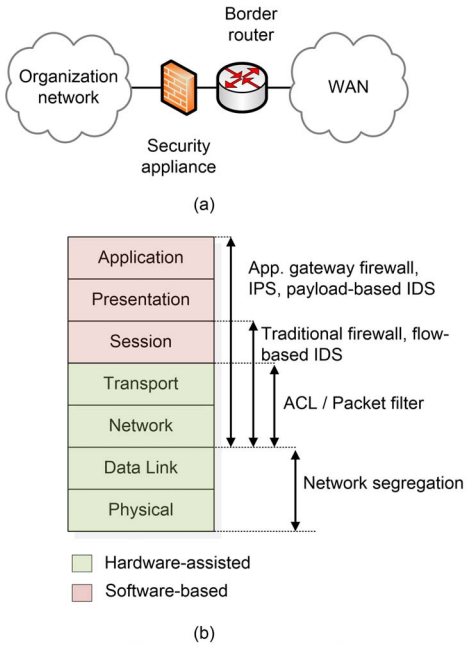

Fig. 24. Physical and logical locations of security appliances. (a) Security appliance (ACL, IPS, IDS, and/or firewall) co-located with the border router. (b) Security appliances in the OSI model.

by a firewall and other security appliances, while traffic flowing into the Science DMZ is subject to a minimal inspection by the border router only.

An alternative to physical segregation is logical segregation by using VLAN technology. A VLAN is a logical subgroup within the LAN that is created at the switch via software rather than physical hardware. The Science DMZ can be isolated from the campus enterprise network by establishing one or more VLANs assigned to the Science DMZ. Since a VLAN can have its own IP address scheme, different access and security policies can be implemented based on IP. However, a disadvantage of a VLAN-based segregation is that the bandwidth of the interface to which both the Science DMZ and enterprise network are attached must be shared. Hence, if there is no mechanism to control the bandwidth allocation to each network, the enterprise network may starve when DTNs are actively transferring data. Additionally, switches must be dimensioned based on Science DMZ requirements.

2) Access-Control List: ACLs are used to control the access to a Science DMZ. Since ACLs are implemented in the forwarding plane of routers and switches, they do not compromise performance. Additionally, as collaborators' IP addresses may be known in advance, a targeted security policy can be used. Fig. 26 shows an example of an ACL implementation [105]. Fig. 26(a) shows the input pipeline, where a packet arrives at the input port and the termination line performs physical-layer functions. The parser engine parses the incoming packets and extracts the fields required for lookup. The lookup process results in an output port the packet will be forwarded to. At this moment, the inbound ACL is applied to the packet. The packet is then switched through the fabric and buffered. The output pipeline, shown in Fig. 26(b), follows a similar scheme.

Fig. 26(c) shows an ACL used to protect a Science DMZ. The DTN with IP address 37.96.87.13 is located in the protected Science DMZ. The IP addresses of other DTNs from collaborators' networks are 143.10.21.2 and 98.103.6.12. The ACL is applied in the inbound direction at the interface facing the WAN. The ACL has three rules; the first two rules permit any TCP segment coming from the collaborators' addresses and going to the local DTN. The last rule denies any other packets from entering the Science DMZ. Note that stricter rules can also be applied, even incorporating port information (e.g., an ACL may only permit TCP segments from collaborators at the ports used by Globus).

3) Firewalls: These devices are capable of processing a large number of small flows characterized by short durations and low transfer rates. Additionally, firewalls typically have small buffers [4]. Clearly, Science DMZ flows do not match this traffic profile. As a result, when a large flow crosses a firewall, the throughput of the flow deteriorates rapidly.

Consider Fig. 27, which shows the throughput for data transfers between a DTN located at Brown University in Providence, Road Island, and a DTN located at the University of Colorado in Boulder, Colorado [106]. These two DTNs are connected by a 1 Gbps path. Fig. 27(a) shows the throughput achieved when there is one firewall located at Brown University. The blue curve is the throughput from the DTN at Brown University to the DTN at the University of Colorado. This traffic is referred to as outbound, and the firewall is not intended to inspect this flow. The green curve is the throughput from the DTN at the University of Colorado to the DTN at Brown University. This traffic is referred to as inbound, and the firewall inspects each packet of this flow.

While both curves show that throughput is affected, the inspection impact on the inbound traffic is critical. For example, the inbound throughput does not even reach 50 Mbps, or 5% of the 1 Gbps path capacity. Fig 27(b) shows the performance between the same two DTNs, but for the case where the firewall is removed from the path. In this instance, the throughput is approximately 900 Mbps, or 90% of the capacity. The reason of this performance difference is related to TCP retransmissions. Namely, every time that TCP receives a triple duplicate ACK for a packet that is lost, a fast retransmission is triggered and the congestion window is reduced by half, thus reducing throughput.

Fig. 27(c) illustrates a generic architecture of a 10 Gbps enterprise firewall. Internally, load balancing among 20 firewall processors is achieved on a per-flow basis (each firewall processor has a capacity of 0.5 Gbps). Note that the maximum throughput is not determined by a single large flow; instead, the maximum throughput is the aggregate throughput of thousands of small flows. When a flow with a rate above 0.5 Gbps arrives at the input interface, all packets of the flow are processed by the same firewall processor. Eventually, incoming packets are dropped as a consequence of the low capacity of the individual firewall processor.

Note that data transfers in LANs may still achieve reasonable performance in the presence of firewalls. Namely, since the latency is small, the TCP throughput can increase quickly after a packet loss. Specifically, after reducing the congestion

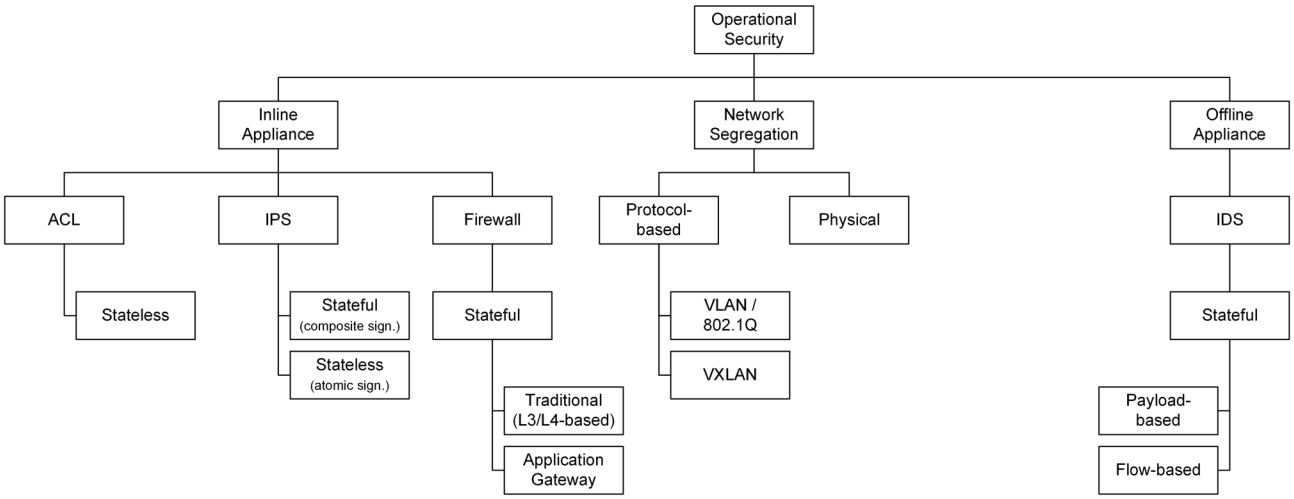

Fig. 25. Operational security appliances and techniques.

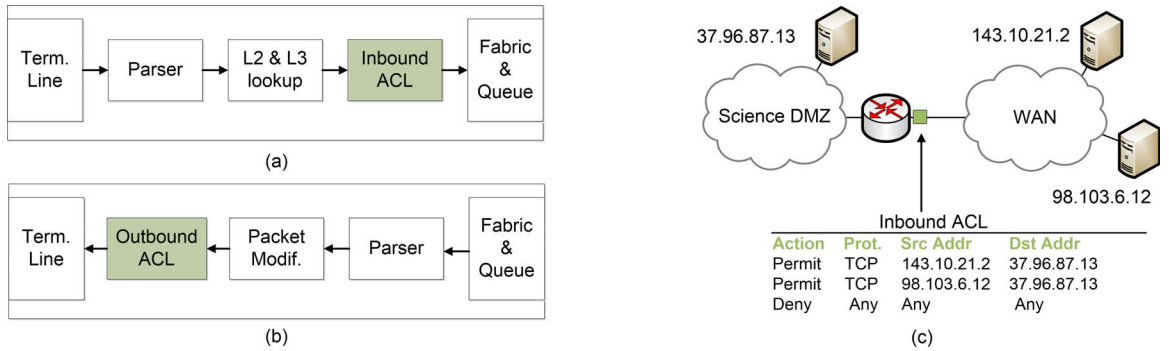

Fig. 26. Implementation and use of ACLs. (a) Diagram of the input and (b) output ports of a router, and the placement of inbound and outbound ACLs. (c) A Science DMZ protected by an inbound ACL. Notice the targeted security by which only specific collaborators' DTNs at 143.10.21.2 and 98.103.6.12 are permitted to connect to the DTN 37.96.87.13.

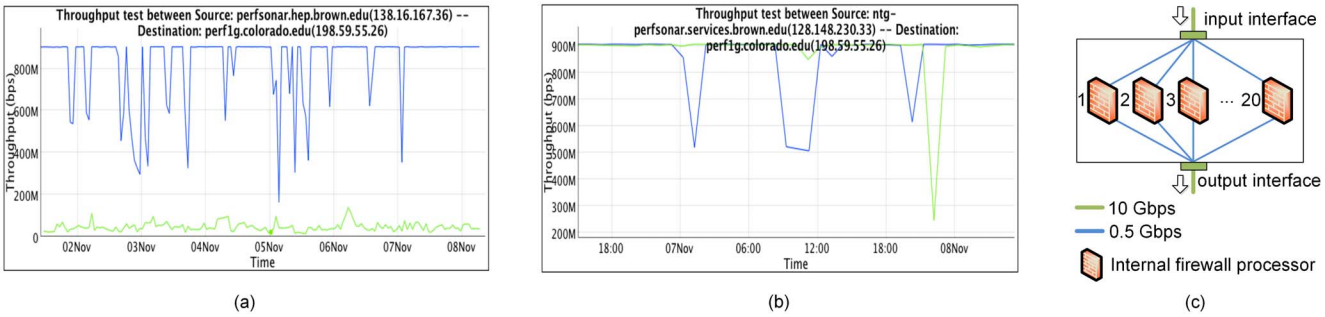

Fig. 27. Impact of a firewall on a data transfer. Throughput performance between a DTN at the University of Colorado in Boulder, Colorado, and a DTN at Brown University in Providence, Rhode Island. The blue curve is the throughput from the DTN at Brown University to the DTN at University of Colorado. The green curve is the throughput from the DTN at the University of Colorado to the DTN at Brown University. (a) Data transfer when a firewall is located in the path of the two hosts. (b) Data transfer when the firewall is removed from the path. The results of (a) and (b) are reproduced from [106]. (c) Conceptual 10 Gbps firewall architecture.

window by half, TCP increases the congestion window again in a time that is proportional to the RTT.

4) Intrusion Prevention System: One of the main features of an IPS is the database containing attack signatures. However, the process of matching a signature with the content of the packet in real time is time consuming. Even new IPS devices such as the Next Generation IPS (NGIPS) [107], which advertise throughputs of tens to hundreds of Gbps, are not suitable for processing large flows. Akin to firewalls, they are designed for processing thousands of small flows simultaneously. For example, the underlying technology of the NGIPS is Snort [108], an open source IPS engine. For an NGIPS appliance rated at 10 Gbps with 20 internal processors, its maximum throughput is only achieved by aggregating the individual throughput of these 20 internal Snort instances. Since each instance has a capacity of 0.5 Gbps and packets belonging to the same flow are processed by the same Snort instance, inspecting a 10 Gbps science flow is not feasible here [27].

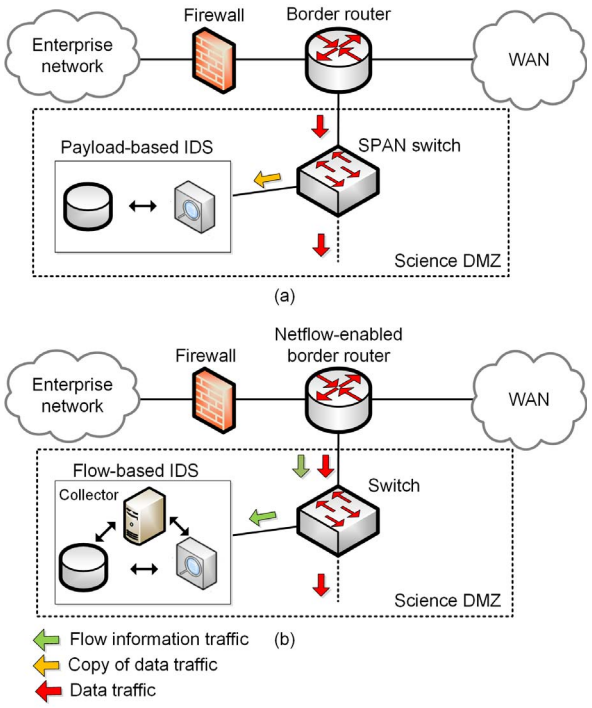

Fig. 28. IDS implementations. (a) Payload-based IDS model. (b) Flow-based IDS model.

5) Intrusion Detection System: For Science DMZs, IDSs represent a better option than IPSs. These systems can be classified based on the information used to detect attacks: payload-based IDS and flow-based IDS [109], [110]. Payload-based IDSs inspect the content of every packet. For high-speed networks, the main challenge of this approach is the processing capability. Meanwhile, flow-based IDSs analyze the communication patterns within the network rather than the contents of individual packets. These devices are attractive for Science DMZs, because of the substantial processing reduction.

Fig. 28(a) illustrates the deployment of a payload-based IDS. The border router forwards traffic to a switch. Packets addressed to the protected network are copied and sent to the IDS. The copy is typically done by a switch with a feature called Switched Port Analyzer (SPAN). SPAN copies network traffic from a selected source port in the switch to a selected destination port. The latter is connected to the IDS.

A popular payload-based IDS is Bro [111]. Bro is well-suited for use in a Science DMZ for several reasons: 1) flexibility in defining security policies, which can be granularly customized by using a domain-specific scripting language interpreted by a policy script interpreter layer; 2) incorporation of hundreds of protocol analyzers in the event engine, allowing the IDS to detect anomalies carried in practically all existent protocols; and 3) scalability. As science flows are characterized at rates of tens of Gbps or more, the potential traffic volumes surpass the capacity of a single-instance IDS. Hence, Bro nodes can be organized in clusters, where clusters of nodes cooperate seamlessly [28]. However, at very high rates, the amount of processing may become excessive, even for a clustered IDS [18].

Flow-based IDSs track the lifetime of a flow and characterize its behavior [109], [110]. This characterization may incorporate several attributes such as the time the exchange of data started, the time it ended, the number of transferred bytes, etc. Fig. 28(b) shows a Netflow-based IDS protecting a Science DMZ. The Netflow-enabled router collects statistical information of all incoming flows passing through the interface facing the WAN. These statistics are collected in hardware by an interface' network processor. The router then extracts the packet header from each packet seen on the monitored interface, and marks the header with a timestamp. It then proceeds to update a flow entry in the flow cache of the router. Once a flow record expires (typically seconds or few minutes), it is sent to a flow collector. Note that the volume of information sent to and stored by the flow collector is several orders of magnitude lower than the actual traffic.

For campuses operating at 100 Gbps, the sampling flow (sFlow) technique [42] is a more scalable solution than Netflow. Here, for a given flow, instead of processing each packet, sFlow can process 1 out of a packets, where a is a configurable parameter. It should be noted, however, that sampling not only lowers the demands put on the flow exporter, but also could make the detection of intrusions harder [109].

6) Response Plan: In general, at least two actions that can be taken once an anomaly is detected are black hole routing and ACL blocking. The black hole routing approach drops packets coming from a suspicious source IP address (e.g., an attacker identified by an IDS) by installing a particular entry in the routing table. The mechanism used is called Unicast Reverse Path Forwarding (uRPF) [112]. This information can be disseminated to other routers via BGP [52]. The ACL blocking technique creates and installs an ACL in the border router when an offender is identified.

Black hole routing is more effective if the information is disseminated to other routers, thus the attack is prevented before packets reach the Science DMZ. On the other hand, ACL blocking is simpler and effective, but the router must still process each offender packet.

B. Confidentiality, Integrity, and Authentication

Confidentiality, integrity, and authentication services are typically provided by the application layer, specifically, by the data transfer tool. These security aspects are required for certain applications. For example, medical Science DMZs [10], [113] transport medical information that must adhere to security and privacy laws and regulations.

Globus [26], [35] provides authentication on the control channel by default. Confidentiality and integrity are both supported on the data channel but are not enabled by default. Vardoyan *et al.* [69] showed that by using Globus with multiple threads, the encryption of the data channel has a minimal performance impact. Globus also includes a feature called striped configuration, which is illustrated in Fig. 29(a). In this configuration, multiple cooperating DTNs can exchange data with remote DTNs [35]. The DTNs are coordinated by a server protocol interpreter (SPI), which implements the control channel. Transfers are then divided over all available DTNs, thus

TABLE X
SECURITY CONSIDERATIONS IN SCIENCE DMZS

Device / Technique	Advantage	Disadvantage	SDMZ recommended
Physical segregation	Easy bandwidth allocation for each network; equipment specifically dimensioned for each network: enterprise network and Science DMZ; easy to apply different security policies for each network.	More expensive; having two physical infrastructures may require higher maintenance and operation costs.	Yes
VLAN segregation	Only one physical infrastructure is required; cheaper.	Shared infrastructure between enterprise network and Science DMZ; more complex allocation of resources (bandwidth, buffer, etc.); potential bandwidth starvation in enterprise network, if resources are not adequately allocated.	Acceptable, if shared resources are appropriately allocated
ACL	Very scalable; it is implemented in router's forwarding plane; minimal performance impact; easy to implement.	Decisions are not based on context; addresses of collaborators must be identified in advance; fragmented packets are unreliably filtered; susceptible to IP spoofing.	Yes
Firewalls	Session tracking adds context to decisions; they are robust against IP spoofing; rich data log.	Inspection capacity is below required rates; for science flows, throughput is severely impacted; it represents a bottleneck for Science DMZs and leads to packet losses and/or out-of-order delivery.	No
Payload-based IDS	Payload inspection provides full application-layer information; state information is collected without interference with flows; no performance impact on switches or routers.	Additional resources for scalability may be needed (e.g., cluster of servers); peak of traffic inspection may be very large at times; attacks might only be stopped from reaching target after they occur; without large clusters, monitoring 100 Gbps links is very difficult.	Yes
Flow-based IDS	The most scalable IDS solution; ability to inspect hundreds of Gbps with a single CPU (sFlow); state information is collected without interference; minimal performance impact on routers and switches.	Application-layer payload is not inspected, but only flow information; flows represent aggregate information only; when sampling is used (sFlow), flow information may be lost; attacks might only be stopped from reaching target after they occur.	Yes
IPS	Inspection of application-layer payload provides full information; attacks can be detected and stopped immediately.	Inspection capacity under single large flow is well below required rates; for science flows, throughput is severely impacted.	No
Confidentiality, integrity, authentication at application layer	Encryption at modern DTNs can now be achieved at high rates; scalable alternatives are available, if needed (e.g., Globus' striped configuration).	There is a throughput degradation when file integrity check is performed; additional resources (e.g., CPU) may be needed for scalability.	Yes
Confidentiality, integrity, authentication with IPsec	Integrity is check on a per-packet basis at the router, avoiding a resource-expensive file integrity check at the end of the transfer; well-known, proven technology (IPsec).	If router is overloaded, the additional processing overhead may lead to packet losses.	Not recommended, but acceptable if the router has sufficient CPU capability

allowing the combined bandwidth of all DTNs to be used. An advantage of this configuration when implementing full encryption is the distribution of processing load.

Modern symmetric-key algorithms can also efficiently encrypt and decrypt data. Some manufacturers, such as Intel and AMD, now offer hardware-based instructions to improve the encryption and decryption throughput of some algorithms, such as the Advanced Encryption Standard (AES). Also, block ciphers provide abundant parallelism's opportunities. For example, when operating in counter mode, AES can encrypt and decrypt blocks in parallel, and the throughput can be increased according to the amount of parallelism provided by multiple cores. Current encryption technology is suitable for 10 Gbps rates.

On the other side of encryption is the file integrity check (FIC). To verify the integrity of a file, the entire file must be received first. Only then can the destination DTN run a cryptographic hash function on the received file. Thus, FIC may represent a larger performance penalty than encryption and decryption.

For authentication purposes, an industry standard that is increasingly being adopted is OAuth 2.0 [114]. Consider a client DTN attempting to download a large file from a server DTN. Here the client DTN is provided with a delegated access to the file resting at the server DTN without sharing credentials.

Confidentiality, integrity, and authentication can also be implemented via a site-to-site virtual private network (VPN). Here, the sender router encrypts the traffic before it enters the WAN. The receiver router then decrypts the traffic upon arrival to the destination Science DMZ. Fig. 29(b) illustrates this alternative. Now, most routers implement VPNs based on the IP security (IPsec) architecture [115]. While IPsec is a well-proven technology, its main disadvantage is the additional processing overhead at the router.

C. Security Summary

Table X summarizes the various security techniques. Clearly, securing a Science DMZ cannot be done with a single device or technique. ACLs are strongly recommended for

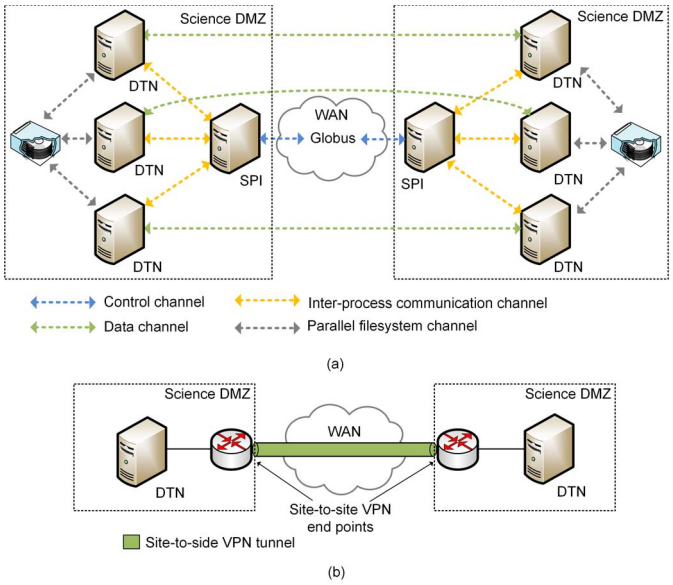

Fig. 29. Configuration options for confidentiality, integrity, and authentication. (a) Cluster-to-cluster configuration of the Globus striped configuration. The server protocol interpreter (SPI) implements the control channel and coordinates the load distribution among DTNs. The communication between SPI and DTNs is via an inter-process communication channel. Data is read/write from/to a storage device using a parallel filesystem channel. (b) Site-to-site virtual private network (VPN) tunnel. Confidentiality, integrity, and authentication are implemented by the routers located at the points of the tunnel.

protecting Science DMZ; however, other offline devices, such as IDSs, should also be implemented to supplement an ACL's lack of context. Inline devices must be avoided.

VII. Empirical Results and Use Cases

This section presents empirical results and use cases. These results were conducted in a laboratory testbed and in ESnet. The two associated topologies are shown in Fig. 30. The laboratory testbed consists of DTNs connected by a 10 Gbps path. The sender and the receiver are purposed-built DTNs. The operating system on these devices is Linux CentOS [77]. Many of the experiments conducted in the laboratory testbed introduce delay, jitter, and packet corruption. These parameters are varied using the netem application [97]. Specifically, when jitter is introduced, it is expressed as a percentage of the RTT. For example, for a path with 10 milliseconds RTT, introducing 10% jitter means that the measured RTT of a packet is between 9-11 milliseconds, with a uniform distribution probability. Most of the empirical results present memory-to-memory throughput, which is computed as the number of bits transferred from the main memory at the sender device to the main memory at the receiver device within a certain amount of time. Unless explicitly indicated, the reported throughput is the average of ten experiments, where each experiment lasts 70 seconds. In order to disregard initial transient behaviors, the first 10 seconds of an experiment are not considered at the time of measuring the throughput.

The tests over ESnet were conducted by the ESnet team and are reproduced from [59] and [60]. Note that ESnet provides services to more than 40 research sites of the U.S. Department of Energy, including the entire U.S. national laboratory system and supercomputing facilities. Accordingly, Fig. 30(b) shows ESnet nodes in yellow and U.S. national laboratories in green. The capacity of links varies from 10 Gbps to 100 Gbps.

The following Section VII-A discusses evaluation results related to buffer sizes in routers and switches. Next, Section VII-B presents evaluation results for transport-layer attributes. Section VII-C then describes performance evaluation of application-layer tools, and the impact of virtualization on DTNs. Finally, Section VII-D discusses some security-related use cases.

A. Buffer Size of Routers and Switches

1) Varying Buffer Size: The results presented in this section are reproduced from [59] for experiments done over ESnet. The source of the flows is Lawrence Berkeley National Laboratory (LBNL) and the destination is Brookhaven National Laboratory (BNL). LBNL is located in Sunnyvale and BNL is located in New York City. The switch under test is a Juniper MX80 [116]. The buffer size of the output port of this switch, located in LBNL, is set to a variety of values, from 4 MBs to 190 MBs. This switch aggregates two science flows and two background flows. All other switches/routers along the path connecting LBNL and BNL do not constitute bottlenecks. The RTT is 70 milliseconds and the path capacity is 10 Gbps. Experiments with the laboratory testbed shown in Fig. 30(a) were also conducted. The same Juniper MX80 switch is tested and the 70-millisecond RTT is emulated with netem. Note that the bandwidth-delay product is 83.4 MBs.

Fig. 31 shows the aggregate throughput of the two science flows. The blue bars indicate the throughput between Sunnyvale and New York over ESnet, whereas the orange bars indicate the throughput using the laboratory topology with emulated RTT. Both results are consistent with each other. Overall an increase in buffer size from 4 MBs to 76 MBs results in a corresponding increase in throughput. Once the buffer size approaches the bandwidth-delay product, the throughput levels off at approximately 7 Gbps. The remaining capacity of the 10 Gbps is consumed by the background UDP traffic. Additional tests using other router models were consistent with that of Fig. 31. With buffer sizes below the bandwidth-delay product, the throughput is not maximized.

2) Small Buffer Size - Datacenter Switches: The switch market is largely driven by datacenter needs. Typically, the paths interconnecting devices inside a datacenter are characterized by low latency. Thus, datacenter switches often have a small amount of buffer space. Hence, a key concern is how these switches perform as the distance between DTNs increases.

The small buffer size scenario consists of a single science flow from the sender DTN to the receiver DTN of Fig. 30(a). Switches 1 and 2 are Cisco Nexus 3172 devices [117]. This switch model has a switching capacity of 1.4 Terabits per seconds (Tbps) and implements dynamic shared memory for port buffer allocation. A port can also be allocated up to 8 MBs. For this buffer size and 10 Gbps bandwidth capacity, the corresponding RTT, according to Eq. (2), is 6.7 milliseconds. This value is referred to as the critical RTT.

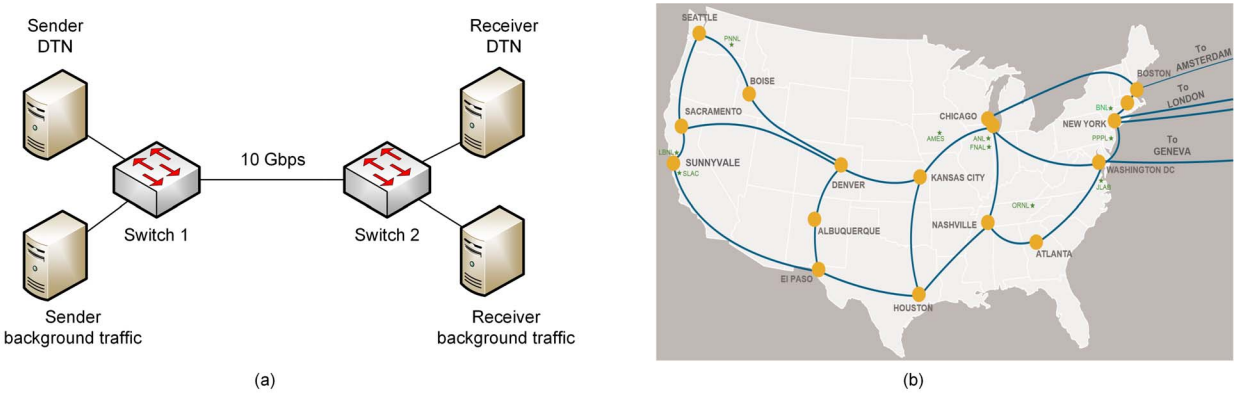

Fig. 30. Network topologies used for testing network performance. (a) Laboratory testbed. All links operate at 10 Gbps. (b) ESnet topology [3]. Links operate at 10 and 100 Gbps.

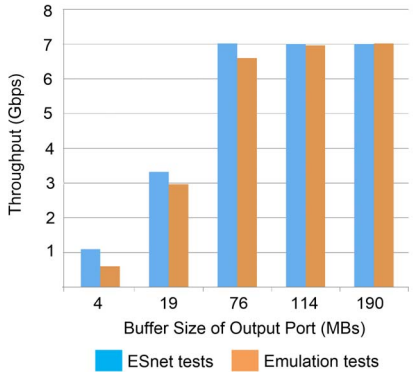

Fig. 31. Aggregate throughput of the two TCP flows described in Section VII-A.1. The switch used for testing performance is a Juniper MX80 [116]. RTT: 70 milliseconds; link bandwidth: 10 Gbps for all links; bandwidth-delay product: 83.4 MBs. The results are reproduced from [59].

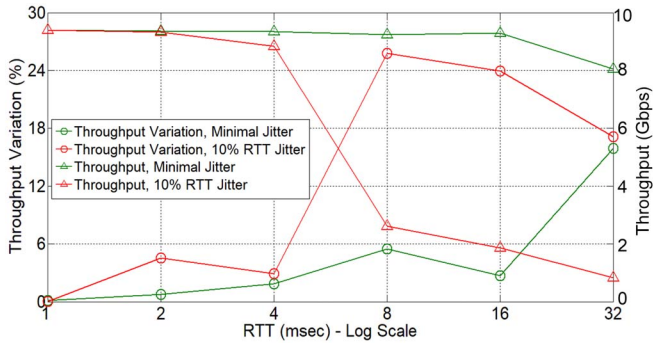

Fig. 32. Throughput and throughput variation as a function of the RTT. For tests presented in this section, switches 1 and 2 in Fig. 30(a) have 8 MBs of memory to allocate per port. The critical RTT is 6.7 milliseconds.

Fig. 32 shows the throughput as a function of the RTT. The green curves represent the throughput and the throughput variation obtained when no artificial jitter is added into the system. Note that the throughput variation is simply the coefficient of variation of the throughput, which is the ratio of the standard deviation to the average throughput. Meanwhile, the red curves represent the throughput and the throughput variation obtained when additional jitter is added. By introducing variance into the RTT of a packet, the emulated WAN produces bursts and packet reordering. The MTU is 1,500 bytes.

Consider the throughput curves. For RTT values of 1, 2, and 4 milliseconds, the green and red curves are similar and the throughput is above 9 Gbps. However, when the RTT increases to 8 milliseconds, the difference between the two curves becomes substantial. At that point, the throughput with minimal jitter is still above 9 Gbps whereas the throughput with 10% RTT jitter is drastically reduced to approximately 2.6 Gbps. This reduction is produced by the buffer limitation and the reception of out-of-order packets. According to Eq. (2), in order to sustain the throughput when the RTT is 8 milliseconds, the amount of buffer space must be almost 10 MBs. Instead, switches 1 and 2 can only allocate 8 MBs per port. However, notice that when the jitter is minimal, the throughput is still 8 Gbps, even when the RTT is 32 milliseconds. This result suggests that throughput can be improved by minimizing jitter and increasing inter-packet spacing. While jitter may not be removed completely, the inter-packet spacing can be increased with TCP pacing.

Consider now the throughput variation results. When the RTT is less than 4 milliseconds, the variation is less than 5%. For the red curve, note that the maximum value occurs when the RTT is 8 milliseconds (i.e., the RTT is approximately equal to the critical RTT). At that point, the variation is above 25% and throughput degradation is accentuated. As a result, predictability, an important feature for Science DMZs, is no longer attained.

B. Transport-Layer Attributes

1) Maximum Segment Size: The MSS value has an important impact on throughput, in particular when packet losses and corruption occur. Hence, Fig. 33 illustrates the results of a data transfer between the two DTNs in Fig. 30(a), with 20 milliseconds RTT. The bandwidth-delay product is 11.92 MBs. Using netem, the system introduces on average 10 single-bit errors during a 60-second period, or its equivalent of 0.1667 errors per second. These errors are uniformly distributed over the test period and force TCP to retransmit segments and decrease throughput when a triple duplicate ACK is received. Now, consider the red curve of Fig. 33. When the MSS is 1,000 bytes,

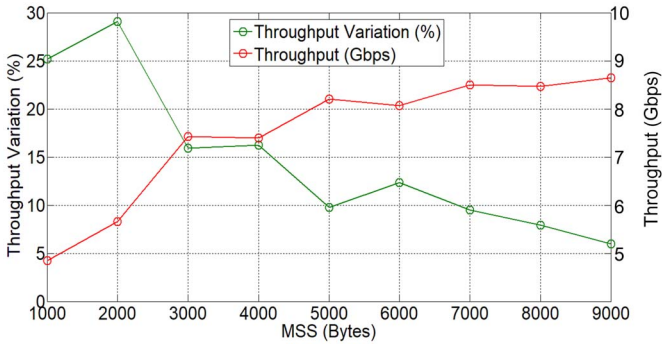

Fig. 33. Red curve: throughput as a function of the MSS, for data transfers between two DTNs connected by a 10 Gbps, 20 milliseconds path. The path introduces 0.1667 single-bit errors per second. Green curve: corresponding throughput variation.

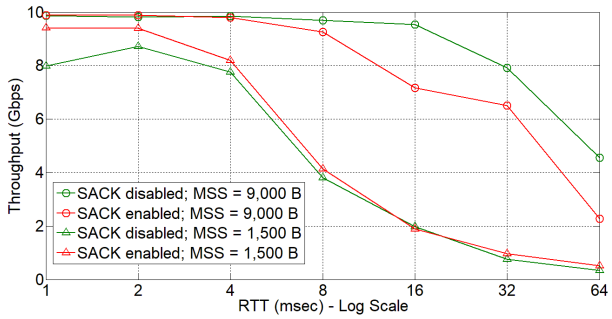

Fig. 34. Experimental results of a data transfer between to DTNs connected by a 10 Gbps path with approximately 0.1667 single-bit errors per second.

even this modest corruption rate reduces the throughput to 4.85 Gbps. As the MSS increases to 9,000 bytes, the throughput also increases to 8.65 Gbps. Clearly, a large MSS value makes TCP more tolerant to corruption.

A large MSS value also enhances the performance predictability of TCP. The green curve of Fig. 33 shows the throughput variation for the same scenario. For small 1,000- and 2,000-byte MSSs, the throughput variability is above 25%. This result indicates a high level of performance uncertainty, which goes against the operational principles of Science DMZs. As the MSS increases, however, the throughput variation decreases significantly. For example, when the MSS is 9,000 bytes, the throughput variation is slightly above 5%.

2) Selective Acknowledgment: This section studies the impact of TCP SACK on large flows. Namely, Fig. 34 shows the performance of TCP with and without SACK, for data transfers between the two DTNs in Fig. 30(a). Switches 1 and 2 have 8 MBs of buffer memory. Thus, the critical RTT based on Eq. (2) is 6.7 milliseconds. The system introduces single-bit errors (which trigger selective acknowledgements) at a rate of 0.1667 errors per second.

When the RTT is small, enabling SACK improves performance, in particular for 1,500-byte MSS. However, when the RTT is 8 milliseconds or above, the throughput is highly degraded for 1,500-byte MSS, independent of the use of SACK. On the other hand, when MSS is 9,000 bytes and RTT is above 4 milliseconds, the throughput is increased by not enabling SACK. For example, when the RTT is 16 milliseconds, disabling SACK allows DTNs to obtain a throughput of approximately 9.5 Gbps, while enabling it results in a throughput of approximately 6.5 Gbps.

The impact of SACK on large data transfers at 10 Gbps is not conclusive. In paths with small to medium RTT, the use of SACK is encouraged in [70] and confirmed in the above experiments. However, in high-bandwidth high-latency paths with a small MSS, using SACK may end up degrading performance.

3) TCP Pacing: The results of this section are reproduced from [60]. TCP pacing has been described as a potential scheme to mitigate traffic bursts generated by small flows [62]. With the increase in science traffic across networks, researchers have recently explored the impact of pacing on large flows [60].

Fig. 35(a) shows the results of data transfers in ESnet. The path capacity and RTT between DTNs are 100 Gbps and 92 milliseconds respectively. Transfers use TCP Cubic [79] without FQ pacing and a MSS of 1,500 bytes. Four concurrent TCP connections are generated from a single source DTN to a single destination DTN. These four connections exhibit the typical sawtooth behavior [58], which in part is attributed to the inability of switches to absorb traffic bursts. Notice that according to Eq. (2), a large amount of buffer space is needed to absorb bursts for this setup, namely 1,097 MBs. Fig. 35(b) shows the behavior of TCP Cubic with FQ pacing. The pacing rate for the four TCP connections is approximately 20 Gbps (curves are overlapped at nearly 20 Gbps). The throughput is slightly lower than 20 Gbps per connection. However, notice how the sawtooth behavior is reduced and stable rates are obtained.

In general, TCP FQ pacing is also effective when there are rate mismatches along the path between the sender and the receiver. This is the case, for example, when the ingress port of a router has a capacity of 100 Gbps and the egress port has a capacity of 10 Gbps. As TCP increases the congestion window during the additive increase phase, eventually the bottleneck link becomes full while the rest of the links become underutilized. The mismatches produce a continuous circle of additive increases and multiplicative decreases, thus generating the sawtooth behavior.

Fig. 35(c) shows the data transfer between two DTNs over ESnet. One DTN is located in Amarillo, Texas, and the other DTN is located in New York City. Although the WAN connecting the two sites has 100 Gbps capacity, one of the DTNs is attached to the network via a 1 Gbps network interface card. Thus, the entirety of the path includes multiple 100 Gbps links and one bottleneck link of 1 Gbps. The figure shows three curves: the throughput when both DTNs are based on Linux CentOS [77] Version 6 (violet), the throughput when DTNs are based on Linux CentOS Version 7 (green), and the throughput when DTNs are based on Linux CentOS Version 7 and packets are paced at 800 Mbps (blue). Note that pacing also leads to much more stable behaviors, almost removing the TCP sawtooth behavior.

4) Congestion Control: TCP BBR is an example of a congestion control algorithm that uses pacing to adjust the rate near the estimated bottleneck bandwidth [24]. Hence, this

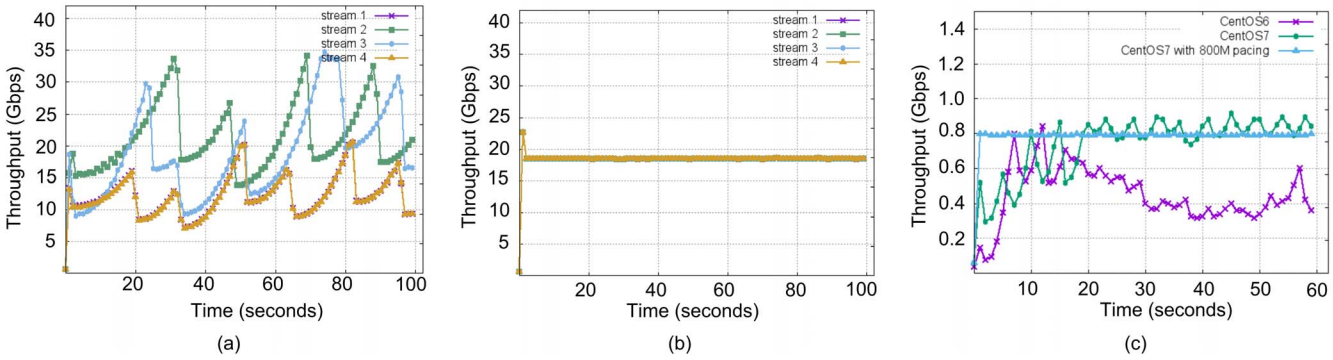

Fig. 35. Impact of TCP pacing on throughput. (a) Data transfers of four parallel TCP connections across a 100 Gbps, 92 milliseconds RTT path. (b) The same data transfer as in (a), but using TCP pacing. (c) Data transfers between two DTNs connected by a path with a bottleneck link of 1 Gbps. The curves show the performance when the DTNs use different Linux operating systems (violet: CentOS 6; green: CentOS 7, and blue: CentOS7 with pacing). The results are reproduced from [60].

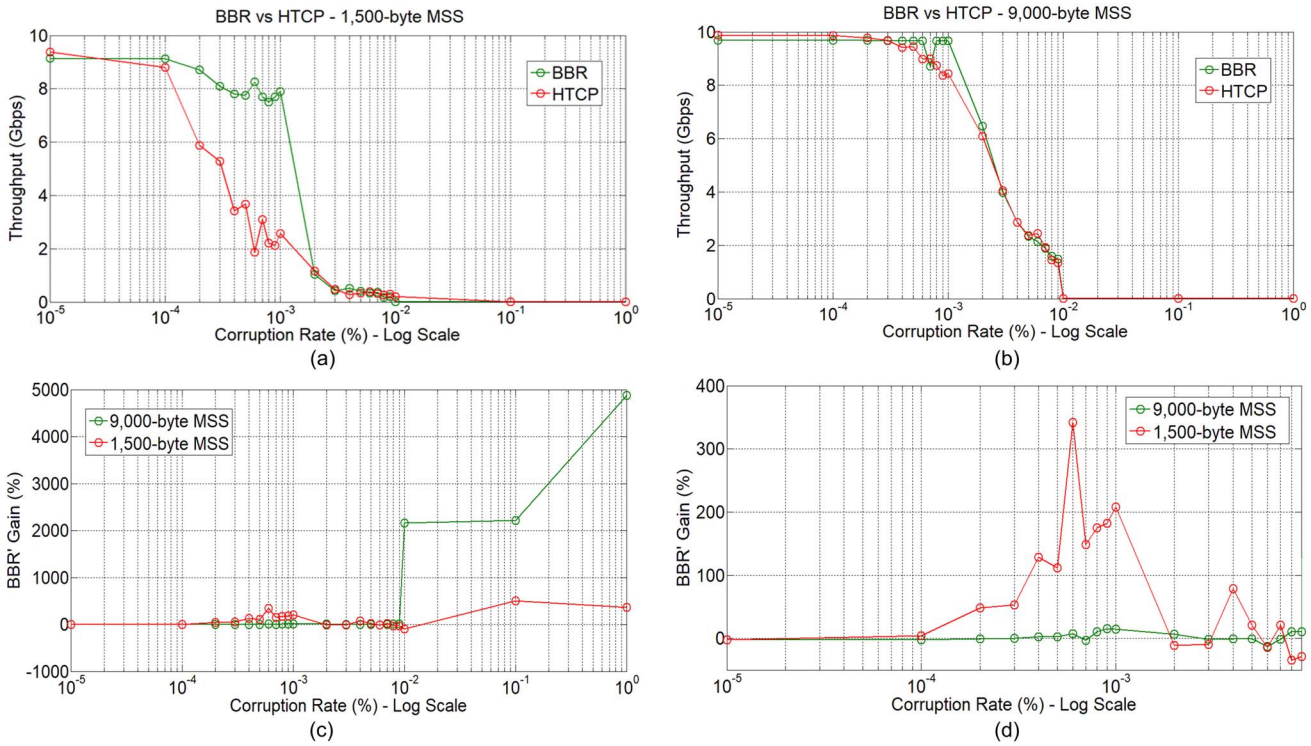

Fig. 36. Throughput as a function of the packet corruption rate, for data transfers between two DTNs connected by a 10 Gbps, 10 milliseconds path. The congestion control algorithms are BBR and HTCP. (a) Results for 1,500-byte MSS. (b) Result for 9,000-byte MSS. (c) BBR's gain given by Eq. (6). (d) BBR's gain focusing on low corruption rates.

section compares the performance of HTCP and BBR. HTCP is a representative loss-based congestion control algorithm used in high-speed networks [33], [118].

Fig. 36 shows the performance of BBR and HTCP for data transfers between the two DTNs in Fig. 30(a). Switches 1 and 2 have 8 MBs of memory for output-port buffers. Packet corruption is introduced at rates ranging from 10^{-5} to 1%. Figs. 36(a) and 36(b) show the throughput obtained with 1,500-byte and 9,000-byte MSSs respectively. Fig. 36(c) shows the BBR's gain with respect to HTCP, computed as

$$\text{BBR's gain} = 100 \cdot \frac{BBR_T - HTCP_T}{HTCP_T}, \qquad (6)$$

where BBR_T and $HTCP_T$ are the throughput of BBR and HTCP. Fig. 36(d) shows the same results as Fig. 36(c), but focusing on the corruption range from 10^{-5} to 10^{-2}. This interval is relevant as HTCP's throughput starts collapsing at a corruption rate of 10^{-4}, and BBR's throughput at a corruption rate of 10^{-3}.

Consider the case for 1,500-byte MSS, Fig. 36(a). At very low corruption rates, BBR's throughput is 2 to 3% lower than that of HTCP. As the corruption increases from 10^{-4} to 10^{-3}, BBR's gain increases up to 342% (see Fig. 36(d), red curve). Between corruption rates of 10^{-4} to 10^{-3}, the throughput decays rapidly for HTCP while BBR also experiences a less-severe decrease. Between corruption rates of $2 \cdot 10^{-3}$ to 10^{-2},

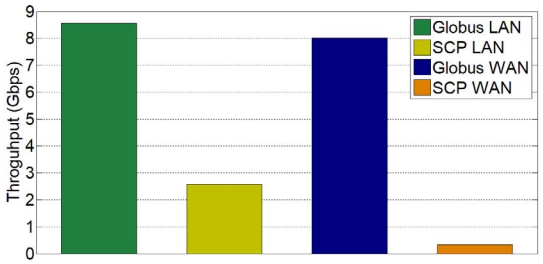

Fig. 37. Throughput comparison between Globus and SCP. The RTT is 0.1 milliseconds in the LAN environment and 53 milliseconds in the WAN environment.

the throughput is similar for both algorithms. At very high corruption rates, the throughput collapses, independently of the congestion control algorithm. When the corruption rate is above 10^{-3}, rates are clearly not suitable for Science DMZs.

Consider now the case for 9,000-byte MSS, Fig. 36(b). Between corruption rates of 10^{-5} to 10^{-2}, BBR and HTCP have similar performance. However, when the corruption rate is 10^{-2}, BBR's gain suddenly increases by more than 2,000% (see Fig. 36(c)). Nevertheless, despite the better performance of BBR, the throughput is low and inadequate for big data transfers.

The above preliminary results indicate that BBR shows better performance than HTCP when packet losses occur. In particular, BBR is able to extend the range of corruption rate that can be tolerated before the throughput collapses, as seen in Fig. 36(a).

C. Application-Layer Tools

1) Data-Transfer Tools: This section compares the disk-to-disk throughput of two data transfer tools: Globus and SCP. Globus is used in Science DMZ environments, while SCP is used in enterprise environments. Globus uses the services of TCP, whereas SCP uses the services of SSH, which implements its own flow control process. Data transfers are performed between the two DTNs shown in Fig. 30(a). The MSS is 1,500 bytes.

Fig. 37 shows the evaluation results. The first scenario is labeled as LAN and the RTT is 0.1 milliseconds. The second scenario is labeled as WAN and the RTT is 53 milliseconds. For SCP, the buffer size is 1 MB. Notice that this buffer size is greater than the bandwidth-delay product for the LAN scenario. Thus, the buffer size is not a limitation. However, the throughput of Globus here is more than three times that of SCP: 8.515 Gbps versus 2.532 Gbps. When the RTT increases to 53 milliseconds, the throughput gap is further increased. Specifically, the throughput of Globus is now 8 Gbps (6% reduction) while that of SCP is 0.33 Gbps (87% reduction). These results reflect the negative impact on SCP's throughput of adding an addition layer (SSH) with its corresponding overhead and flow control, and the positive impact on Globus' throughput of incorporating features for big data transfers, such as parallel streams and large TCP buffers.

2) Parallel Streams: This section illustrates the impact of parallel streams on data transfers conducted in the testbed of Fig. 30(a). The RTT is 20 milliseconds and switches 1 and 2 have 8 MBs of memory for output-port buffers.

Fig. 38 shows the throughput as a function of the corruption rate. Consider Fig. 38(a) which shows the results when the MSS is 1,500 bytes. For small corruption rates, between 10^{-6} to 10^{-4}, single-stream rates are between 6.4 and 4.8 Gbps. Adding parallel streams increases throughput substantially; with 13 streams, the throughput increases by almost 100% with respect to single-stream transfers, and attainable rates are up to 9.5 Gbps. The throughput obtained with 5 and 9 streams is similar to that obtained with 13 streams. However, the throughput degradation is accentuated as the corruption rate increases above 10^{-4}, independently of the number of streams. Still, notice that when parallel streams are used, the rates are significantly higher, until the corruption rate reaches 10^{-2}.

Consider now Fig. 38(b) which shows the results for a MSS value of 9,000 bytes. Here, at low corruption rates, the performance of single-stream and parallel-stream transfers are similar. Using jumbo frames helps attain faster recovery after a packet loss, which adds robustness to single-stream transfers. However, as the corruption rate increases above 10^{-4}, parallel-stream transfers outperform single-stream transfers.

3) Performance of Virtual DTNs: The use of virtualization in Science DMZs has been low. However, as virtual components (e.g., virtual NICs, virtual switches) become more capable of processing frames at 10 Gbps rates, the deployment of virtual DTNs may be a viable alternative. Hence, this section evaluates the performance of virtual DTNs.

Using the topology of Fig. 20, two scenarios are considered: a) from the host 2 DTN to the virtual DTN located in host 1 (virtual environment using VMware ESXi Hypervisor 6.0 [119]), and b) from the host 2 DTN to the host 3 DTN (native environment). The physical switch has 8 MBs of memory for output-port buffering. The throughput between the two native DTNs is labeled as native-native, while that between a native DTN and the virtual DTN is labeled as native-virtual. The path capacity is 10 Gbps and the virtual DTN uses a VMXNET3 virtual NIC [92].

The performance evaluation results capture the impact of jitter, RTT, and MSS. Consider Fig. 39(a), which shows the results when the MSS is 1,500 bytes. When jitter is added into the system, the penalty of using a virtual DTN is more pronounced. At 4 milliseconds RTT, the use of virtualization leads to a decrease in throughput of 26%, i.e., from 9.25 Gbps (1 millisecond RTT) to 6.82 Gbps (4 milliseconds RTT). Meanwhile, the throughput using native DTNs is only reduced by 2.87% for the same RTT interval, from 9.41 Gbps to 9.14 Gbps. The performance degradation is even larger at 16 and 32 milliseconds RTT, since the processing overhead at the virtual DTN adds inefficiencies and consumes virtual CPU resources. For 32 milliseconds RTT, the number of in-flight packets is more than 26,000. Note that in the above scenario, when jitter is present, some packets arrive out of order, further decreasing throughput. When the sender receives a triple duplicate ACK, a fast retransmission is triggered and the congestion window is reduced by half. The estimated RTT must also be recomputed and the TCP timers must be updated accordingly. A timer is associated with each unacknowledged

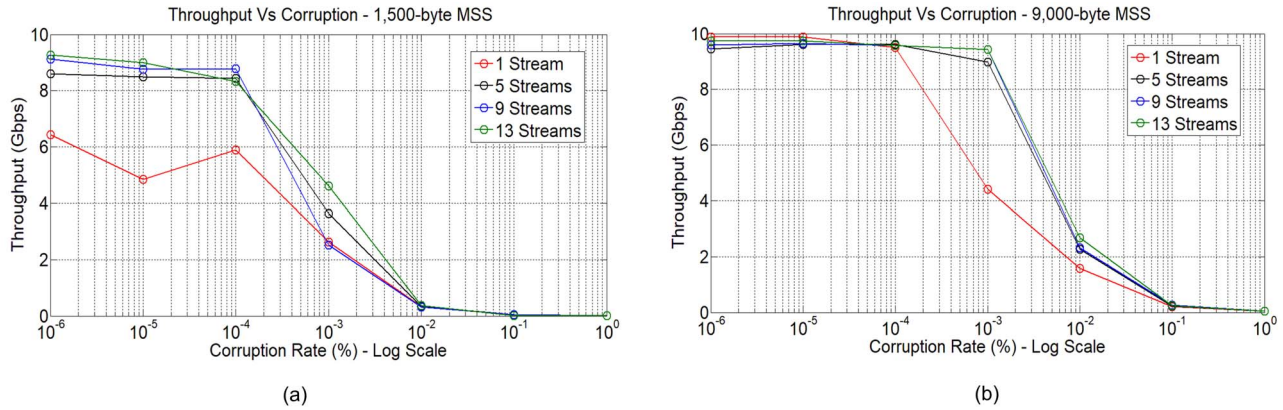

Fig. 38. Throughput of data transfers between two DTNs connected by a 20 milliseconds RTT, 10 Gbps path. (a) Results for 1,500-byte MSS. (b) Results for 9,000-byte MSS.

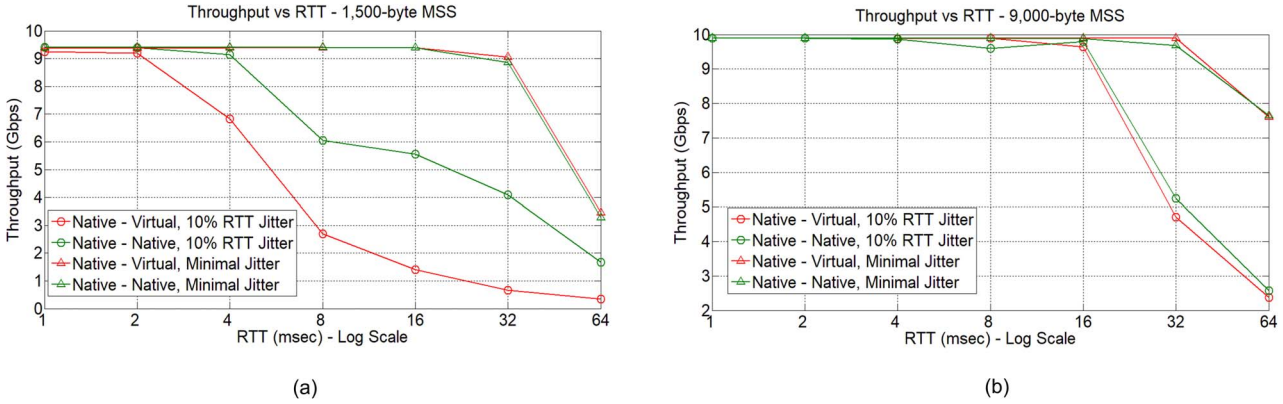

Fig. 39. Impact of latency, jitter, and MSS on virtualization. (a) Results for 1,500-byte MSS. (b) Results for 9,000-byte MSS.

segment. Moreover, at the receiver side, packets are buffered not only by the TCP receive buffer but also by the virtual NIC and by the virtual switch. Further processing occurs at the virtual switch, which must read each Ethernet frame and forward it to the virtual DTN. Thus, any additional processing overhead is amplified at host 1. Note that when jitter is minimal or when the MSS is 9,000 bytes, the performance difference between the virtual and native environments is reduced. With jumbo frames and 32 milliseconds RTT, the number of in-flight packets is reduced to approximately 4,400 packets. Using 1,500-byte segments means six times as much per-segment overhead as using 9,000-byte segments. If virtual DTNs are used, the processing overhead should be minimized.

D. Security Use Cases

As noted earlier, ACLs are the primary protection mechanism used in Science DMZs. Offline protection mechanisms include payload-based and flow-based IDSs. Hence, this section presents three use cases to protect Science DMZs.

1) Payload-Based IDS: Fig. 40 illustrates the architecture of a Bro-based system implemented by LBNL to protect a Science DMZ attached to ESnet [28]. The 100 Gbps input traffic is forwarded to a high-speed switch which splits and forwards the traffic to five Bro nodes, each containing ten Bro workers (each worker is a server machine). Ultimately, inspection is carried out by Bro workers. The switch distributes the input traffic across the five nodes by applying a hash function to a combination of source and destination IP addresses, transport-layer protocol, and source and destination ports. All packets belonging to the same TCP flow are forwarded to the same Bro node. Thus, the IDS can correlate the packet's information with the state information collected by the system. At most, each Bro node receives approximately 20 Gbps or 20% of the total input traffic. In practice, the amount of traffic forwarded to each Bro node is considerably less than 20 Gbps, because after an initial inspection, most packets from a trusted source are not inspected. This technique is now used more frequently to alleviate the load on inspection engines [27], [120].

Each Bro node is attached to the network via a network interface card with the capability of evenly redistributing the traffic to Bro workers. Therefore, each worker must process at most 2 Gbps of traffic. Now, Bro classifies flows into two categories: interesting and uninteresting. Only interesting flows are fully inspected. Uninteresting flows simply bypass the IDS and are not inspected. Some key features of this system include:

- When a new flow is identified, it is subject to full inspection. Specifically, the first 128 MBs of all flows are always inspected. This initial inspection attempts to identify anomalies in the flow and thus to classify it

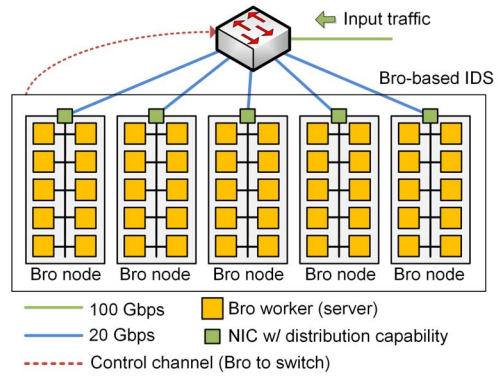

Fig. 40. A Bro-based IDS [28].

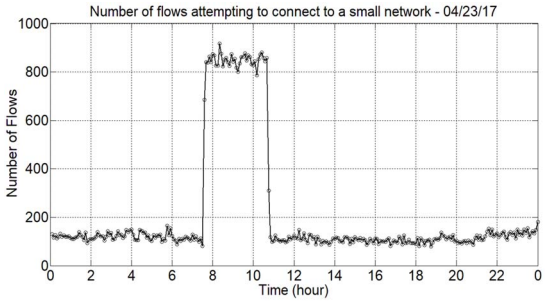

Fig. 41. Data collected by a flow-based IDS at a small campus network. (a) Number of flows per five-minute unit. (b) Data collected per flow. (c) Remote devices connecting to the enterprise network sorted by the number of flows.

as interesting. If no anomalies are detected, the flow is classified as uninteresting.

- All TCP segments with their flag bits enabled (control segments) are forwarded to the Bro cluster. These segments may indicate a change in the state of the flow.
- A control channel is used to dynamically update the status of the flows. Once a flow is classified as uninteresting, Bro installs an ACL in the switch so that packets belonging to that flow are no longer forwarded to the Bro cluster (with the exception of the control segments).

According to [28], typically only 2-4 Gbps of traffic is inspected from the 100 Gbps input traffic, with occasional peaks of up to 25 Gbps.

2) Flow-Based IDS: A flow-based IDS is a more scalable option than a payload-based IDS. A generic flow-based IDS is shown in Fig. 28(b). Fig. 41 shows typical information

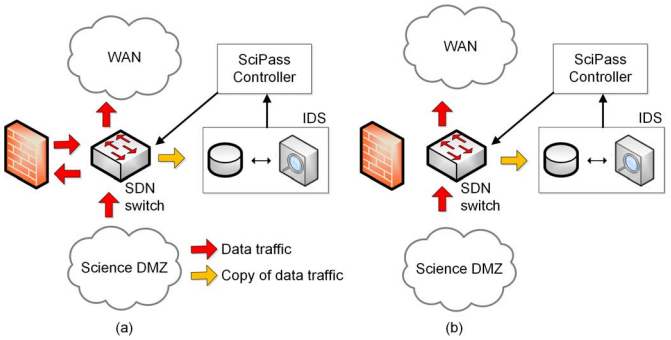

Fig. 42. SciPass architecture. (a) Default behavior; packets are forwarded through the firewall. (b) Science flow being re-routed by the SDN controller.

collected by a flow-based IDS. Specifically, this example corresponds to an IDS deployed to protect both the enterprise network and Science DMZ at a small institution. Each point in Fig. 41(a) is computed by aggregating five minutes of data. At any time except between 7:45-10:30, the number of flows is less than 200. Fig. 41(b) shows the information collected per flow, for the five-minute period between 8:30-8:35. The columns *pps* and *Bpp* indicate the packets per second and the bytes per packet. The *Src IP Addr* and *Src Pt* are the IP address and the port number of the device attempting to connect to the network respectively. The column *Flags* indicates whether at least one segment of the flow has already included a TCP segment with the bit ACK (A), PUSH (P), RESET (R), SYN (S), and FIN (F) set. This column is important because it provides information about the state of the TCP connection. Finally, Fig. 40(c) shows external IP addresses generating flows, sorted by the number of flows (i.e., flows are aggregated based upon the source IP address). The first entry is an example of a dictionary attack from IP address 75.161.37.158 (3,385 flows coming from this IP during a five-minute period). Aggregating flows can certainly help identify cyberattacks. For example, during a data transfer operation, a collaborator's DTN would only have a few connections open to a DTN located inside the Science DMZ. Anomalies such as dictionary attacks, on the hand, open tens or hundreds of parallel connections, as seen in Fig. 41(a). As a highly scalable technique, it is expected that flow-based IDS will become a prevalent method for protecting Science DMZs.

3) SDN-Based Security: Software Defined Networking (SDN) [121] is increasingly being used to protect Science DMZs. An example of an SDN-based architecture is SciPass [122], which is illustrated in Fig. 42. SciPass consists of an SDN switch, an IDS, a controller, and a firewall. By default, all traffic is forwarded to the firewall, as shown in Fig. 42(a). At the same time, a copy of the data traffic is sent to the IDS. Based on the security policy, the IDS determines whether a flow is trusted to bypass the firewall. The IDS then signals the SciPass controller, which generates new forwarding rules and downloads the new rules into the switch, as illustrated in Fig 42(b). The new forwarding rules contain an idle timeout so that once the flow completes, the rules are purged from the switch. By avoiding the firewall, the throughput of the science flows are dramatically improved.

The technique of allowing trusted flows to bypass security appliances is also increasingly being used in enterprise networks [120], [123]. This technique may use authentication information. For example, a client DTN is typically required to authenticate against a server DTN prior to the data transfer. Flows from an authenticated party could therefore be considered as low-risk flows and bypass the security appliances.

VIII. Challenges and Open Research Issues

Owing to its proven efficiency to move large data sets, the number of deployed Science DMZs has been rapidly increasing in the last few years. However, there are still many challenges and open research issues that must be addressed.

A. Connectivity to the WAN

1) Cyberinfrastructure: The deployment cost of high-speed connections is still an unresolved problem in developing countries and many areas of developed countries. In the U.S., this is observed in areas such as remote Native lands, where there is a lack of cyberinfrastructure for WAN connectivity at Gbps rates. The deployment of fiber connections and access to POPs from such remote locations have prohibitive costs. As an example, in 2010, the U.S. Federal Communications Commission (FCC) released the National Broadband Plan, an effort to narrow the digital divide between urban and rural areas. Some key problems that must be addressed include:

- Terrain. Historically, service providers have dismissed the prospect of installing cables in areas located in remote, mountainous regions with extreme variations in elevation. The process of digging and laying underground fiber in these terrains is arduous, time-consuming, and expensive. Typically these areas are also far away from regional networks, exchange points, RENs and Internet2.
- Regulations pose a unique set of challenges. Many developing countries have only recently opened up to market forces, from a totally centralized scheme. Similarly, sovereign tribal nations in the U.S. require telecommunications providers to meet certain criteria to protect the land and culture, and most carriers are not interested in complying with additional rules on top of the regular bureaucracy. In developing countries and tribal nations, if a service provider is interested in laying fiber, it could see significant hurdles and consultations with local government. The provider would also have to perform environmental protection and historic preservation studies [124], [125].
- Cost. In rural areas with a limited potential subscriber base, RENs and service providers in particular see no possibility for any return on investment.

2) Connection-Oriented Networks: Since the early days of the Internet, there have been two camps regarding the service provided by the network layer: connectionless and connection-oriented. The first one has adopted the end-to-end argument [126] that shaped the Internet. However, many parts of the Internet and RENs are evolving to connection-oriented services as QoS becomes more important. For large data transfers and as a means to connect Science DMZs, a connection-oriented service provides bandwidth guarantee, a key advantage. An example of a connection-oriented service is the On-demand Secure Circuits and Reservation System (OSCARS) [127], [128], which connects WAN layer-2 circuits directly to the DTNs and provides bandwidth reservation and traffic engineering capabilities. Similarly, MPLS is used by large service providers, RENs, and Internet2 to provide QoS and establish long-term connections. Other connection-oriented schemes for bandwidth and delay guarantees have been also proposed [101], [129]. The adoption of connection-oriented technology to connect Science DMZs is expected to continue. Moreover, the wider adoption of this paradigm may encourage further development of upper-layer protocols. For example, given a guaranteed bandwidth, congestion control schemes based on pacing would be simpler to implement than current TCP congestion control schemes.

B. Data-Link and Network-Layer Devices

1) Features for Large Flows: Typically, datacenter devices are designed for low-latency networks. These devices have a small amount of memory for buffering and use cut-through and fabric designs that are only suitable for small flows. Additionally, even when a device has sufficient memory to accommodate large flows, default configurations result in buffer underutilization. Since the technical expertise of cyberinfrastructure engineers mostly focuses on enterprise networks, suboptimal configurations are not uncommon. Fortunately, the market has recently noticed the need for Science DMZ-capable devices. Hence, many manufacturers such as Cisco [21], Brocade [22], and Ciena [23] are now providing features amenable for large flows, such as adequate buffer allocation and application-programming interfaces to automate processes and enforce preset policies (e.g., bypassing a firewall according to traffic type or trust level).

2) Maximum Transmission Unit: The maximum segment size has notable performance impact in high-throughput, high-latency networks, in particular under random-loss regimes. Unfortunately, supporting end-to-end jumbo frames is still an open challenge. Foremost, all hosts in a single broadcast domain must use the same MTU, and this can be difficult and error-prone. Additionally, Ethernet has no mechanism of detecting an MTU mismatch. A device that receives a frame larger than its MTU simply drops it silently. Secondly, since different administrative domains (ISPs, RENs) are independently operated, packets are routed through devices that either do not support jumbo frames or at best have different MTUs. Hence, there is a need to establish a standard for jumbo frames, so there is a reasonable guarantee that if vendors comply with the specifications, then there would be no interoperability problems.

C. TCP Optimization

1) Congestion Control: Most TCP algorithms for congestion control use packet loss as a signal of congestion. According to Eq. (2), in order to achieve a throughput of 10 Gbps, TCP can only tolerate one segment loss for every

6,944,000,000 segments, which is incredibly small. The use of alternative congestion control mechanisms where packet loss is not a signal of congestion is a promising direction. The recently proposed BBR algorithm [24] has shown preliminary throughput improvement in medium- and high-loss packet regimes. Note that using TCP pacing to adjust the bit rate at an estimated bottleneck bandwidth is a departure from the traditional window-based congestion control mechanism. Additionally, since rate-based congestion control does not require constant congestion window updates, this approach avoids the long delays inherent in the receiver sending the congestion window. Moreover, the promising performance results of BBR may lead to the development of other congestion control algorithms. The use of parameters for detecting congestion and random losses that have stronger correlation to congestion than packet losses also needs to be explored.

2) Pacing: TCP FQ pacing has shown promising results in long fat networks. However, the main concern with this technique is finding the bottleneck link along the path between the end devices. Once the bottleneck link is identified, pacing packets at the bottleneck link's capacity mitigates the TCP sawtooth behavior and produces stable throughput. Pacing can also be easier in connection-oriented networks, as packets can be paced at the guaranteed bandwidth allocated to the connection.

3) TCP Extensions: Many TCP extensions have been proposed over the years, including selective acknowledgement, timestamp, window scale, and RTT measurement [70]. As most of these extensions were targeted to mitigate issues observed in the Internet's best-effort service model, they may not be suitable for large data transfers over well-conditioned networks such as Internet2 and other RENs. Hence, investigating the use of TCP extensions in Science DMZ environments is required.

D. Optimization in the Protocol Stack

As routers and switches are optimized for Science DMZs [21]–[23], the protocol stack at DTNs may become the bottleneck for many implementations. Reducing DTN processing overheads is desirable to increase throughput.

Software techniques can help optimize the TCP performance on 10 Gbps WANs and above. However, optimizing a DTN to operate at 100 Gbps is currently a persistent challenge. Most TCP implementations have a considerable overhead and produce a very high CPU utilization, which raises questions about the viability of TCP as the network bandwidth continues to grow [20]. UDP-based tools such as Aspera FAST [86] and UDT [89] may suffer a performance penalty due to context switching and the process of copying data to user-space buffers. Kissel *et al.* [20] have recently proposed a new protocol called wide-area Remote Direct Memory Access (RDMA). RDMA decreases TCP processing overheads by using optimization techniques such as zero-copy and splice. Zero-copy is a procedure that relieves the CPU of copying data from one memory area to another (e.g., from lower-level layers to the TCP buffer). This technique saves CPU cycles and memory bandwidth when transmitting a file over a WAN. Similarly, splice is a system call used to move data between two file descriptors. Splice minimizes the movement of data between kernel space and user space.

Overall, zero-copy and splice are two techniques that can minimize the movement of data within a DTN. Similar cross-layer optimization techniques can further reduce processing overheads. For example, TCP and IP are usually implemented together, so that there is no need to copy the layer-3 payload when moving it from the network process to the transport process. This idea can be extended to the upper layers, i.e., from transport to application.

E. Applications

1) Data Transfer Tools: As the main applications used in Science DMZs, data transfer tools must be designed for high-throughput, high-latency networks. Namely, these tools should implement features such as parallel streams, large buffer sizes, and partial and restartable file transfers. At present, engineers rely on rule of thumbs to configure many of these features. For example, there is no formal solution to the problem of selecting the number of parallel TCP streams that should be open for a data transfer. Globus suggests that the number of streams should be between 2-8. Moreover, the optimal value may depend on the RTT, bandwidth, congestion control algorithm, etc.

Data transfer tools should minimize the time spent in input/output operations (which are expensive) and exploit the multi-core capability of modern DTNs. For example, FDT [85] uses independent threads to read and write on physical devices in parallel. Data transfer tools should also avoid copying data multiple times within the DTN. Improvements may involve several layers, including transport and application.

The adoption of UDP-based data transfer applications has been minimal. Tests conducted in 10 Gbps networks indicate that the throughput is limited by the high CPU-utilization [91]. Also, current UDP-based applications do not use parallel streams. Instead, they only open one stream per data transfer. Typically, the stream's process is tied to one core while other cores are idle. With this approach, UDP-based applications may only achieve higher rates by increasing the CPU's clock rate. However, increasing the CPU rate is a challenge. Instead, during the last decade, the throughput has been increased by using multicore CPUs. Thus, an open research issue includes the use of UDP-based applications using multiple streams, in particular when parallelism opportunities exist [101], [127].

2) Monitoring Applications: The effectiveness of perfSONAR in measuring end-to-end metrics and in detecting soft failures relies on its deployment across multiple domains [38]. While perfSONAR has been extensively deployed on RENs (e.g., ESnet [3], Internet2 [40], GEANT [44], CESNET [130], etc.), its deployment by ISPs is still lacking. A contributing factor here is the lack of familiarity of engineers who are more familiar with single-domain tools used in enterprise networks, such as SNMP, Syslog, and Netflow. Thus, there is a need to outreach to the networking community to widen the adoption of collaborative multi-domain tools without compromising the privacy and commercial interests of ISPs.

Integrating and correlating data collected from different applications is an immediate research direction. For example, SNMP and perfSONAR complement each other. The former can detect intra-domain hard failures while the latter can detect inter-domain soft failures. In this context, Gonzalez *et al.* [18] describe a monitoring application integrating perfSONAR, SNMP, and other tools. The proposed platform also integrates data visualization and analytics modules. With the advent of SDN, this type of integration and the addition of network programmability are expected to continue.

3) Virtualization: The research community has been reluctant to adopt virtual components into Science DMZs, mainly because of the performance degradation of virtual DTNs. However, in small institutions where resources are often limited, using virtual DTNs is a cost-efficient alternative. Preliminary results suggest that virtual DTNs may be adequate for 10 Gbps Science DMZs, provided the physical server they run on has sufficient CPU capacity and the workload is minimal. However, when packet losses occur and DTNs require more processing capability for handling retransmissions, the performance degradation can be significant (see Fig. 39(a)). Additionally, virtual components are unable to perform at 40/100 Gbps. Thus, research on minimizing processing overheads on virtual devices (virtual switch, virtual NIC, hypervisor) is still required.

F. Security

In general, not having Web, email, and other general-purpose applications running on DTNs mitigates the delivery of malicious payloads via XML, SQL, cross-site injection, and other methods. However, since transfer rates are high, the data inspection in Science DMZs may be minimal. For example, the typical inspection rate of a payload-based IDS protecting a 100 Gbps Science DMZ connected to ESnet is between 2-4 Gbps [28], which is less than 5% of the total network input. While the reported number of malware attacks in current Science DMZs has been minimal, there is a trade-off between performance and security that should be carefully analyzed when deploying this type of IDS, in particular for 40/100 Gbps Science DMZs. A specific approach that can be explored for high rates may combine both flow-based and payload-based IDSs. A first layer of detection may preselect suspicious flows using a flow-based IDS, while a second layer may scan packets of the preselected flows using a payload-based IDS.

Confidentiality, integrity, and authentication are usually implemented at the application layer. Although current encryption algorithms are capable of performing at or near 10 Gbps, Globus' file integrity checks may introduce a penalty of up to 10%. Encryption rates of 40 and 100 Gbps are still uncommon in DTN deployments. However, recent development of specialized hardware indicates that a rate of 100 Gbps is achievable for in-transit encryption [131]. The use of medical Science DMZs [9], [10] and the need to comply with regulations [132], [133] are expected to accelerate these developments. Finally, preventing DoS and scanning attacks is also an ongoing research direction, as these attack types are continuously evolving.

IX. Conclusion

This article presents a comprehensive tutorial on Science DMZ. Motivated by the ever increasing science data production and by the need to transfer and manipulate it across WANs, the tutorial delves into every layer in the protocol stack, reviewing protocols and devices of a well-designed Science DMZ. At the cyberinfrastructure level, the need for high-speed connectivity and the alternatives for connecting to RENs and Internet2 are discussed. Transferring large flows requires routers and switches with appropriate buffer space. Important features of these devices are discussed, including fabrics, rates, queues, and forwarding methods, which must be considered for successful deployments. Performance evaluations conducted in ESnet and in a laboratory testbed reinforce the importance of having well-conditioned equipment when transferring large flows.

At the transport layer, as data must be correctly delivered from one device to another, the article examines TCP and its attributes including segment size, pacing, TCP extensions, flow control, and congestion control. These attributes play an important role in Science DMZs, as the end-to-end path utilization and the throughput depend on them. At the application layer, Science DMZs are often limited to data transfer tools and performance and security monitoring applications. Data transfer tools differ substantially from those used in enterprise networks. Important features such as partial transfer capability, parallel streams, and application buffer size can have significant performance impact, in particular in scenarios with random losses. Hence, these features should be incorporated. Similarly, a performance measurement application such as perfSONAR is necessary to monitor end-to-end paths over multiple domains and to detect soft failures.

The article also describes advantages and disadvantages of security appliances and techniques, such as ACL, IPS, IDS, and related best practices to secure Science DMZs while avoiding performance degradation. Finally, this article concludes by discussing open research issues that need further investigation.

Acknowledgment

The authors would like to acknowledge the ESnet team members for their contribution to this work. Namely, Jason Zurawski and Eli Dart suggested key changes and corrections and provided material that is included in some sections of this manuscript. Brian Tierney and Michael Smitasin are the authors of the results shown in Figs. 31 and 35 and gave valuable information related to these results. Aashish Sharma is a member of the team that developed the Bro-based IDS shown in Fig. 40. He provided information included in Section VII-D1. The authors also would like to thank John Gerdes from the University of South Carolina and John Hicks from Internet2 for their helpful advice on various technical issues examined in this paper.

The authors are very grateful for the comprehensive review conducted by the anonymous reviewers. Their suggestions and corrections helped improve the quality of this manuscript.

TABLE XI
ABBREVIATIONS USED IN THIS ARTICLE

Abbreviation	Term
ACK	Acknowledgement
ACL	Access-Control List
AES	Advanced Encryption Standard
AMO	Atomic, Molecular, and Optical
BBR	Bottleneck Bandwidth and Round-Trip Time
BDP	Bandwidth-Delay Product
BGP	Border Gateway Protocol
BNL	Brookhaven National Laboratory (United States)
btlbw	Bottleneck Bandwidth
BWCTL	Bandwidth Test Controller
BYOD	Bring-Your-On-Device
CC*	Campus Cyberinfrastructure Program
CDF	Cumulative Distribution Function
CENIC	Corporation for Education Network Initiatives in California
CPI	Client Protocol Interpreter
CPU	Central Processing Unit
DMZ	Demilitarized Zone
DoS	Denial of Service
DTN	Data Transfer Node
DTP	Data Transfer Process
ESnet	Energy Science Network
FCC	Federal Communications Commission (United States)
FDT	Fast Data Transfer
FIB	Forwarding Information Base
FIC	File Integrity Check
FQ	Fair Queue
FT	Forwarding Table
FTP	File Transfer Protocol
GB	Gigabyte
Gbps	Gigabits Per Second
HOL	Head-Of-Line
HTCP	Hamilton Transmission Control Protocol
HTML	Hypertext Markup Language
HTTP	Hypertext Transfer Protocol
HTTPS	Hypertext Transfer Protocol Secure
IDS	Intrusion Detection System
IETF	Internet Engineering Task Force
IP	Internet Protocol
IPFIX	IP Flow Information Export
IPS	Intrusion Prevention System
IPsec	Internet Protocol Security
ISP	Internet Service Provider
KB	Kilobyte
Kbps	Kilobits Per Second
LAN	Local Area Network
LBNL	Lawrence Berkeley National Laboratory (United States)
LHC	Large Hadron Collider
MB	Megabyte
Mbps	Megabits Per Second
mdtmFTP	Multicore-Aware Data Transfer Middleware File Transfer Protocol
MPLS	Multi-Protocol Label Switching
MSS	Maximum Segment Size
MTU	Maximum Transmission Unit
NAT	Network Address Translator
netem	Network Emulator
NGIPS	Next Generation Intrusion Prevention System
NIC	Network Interface Card
NNMC	Northern New Mexico College
NP	Network Processor
NPAD	Network Path and Application Diagnostics
NREN	National Research and Education Network
NSF	National Science Foundation
NSFnet	National Science Foundation Network
NUMA	Non-Uniform Memory Access
OS	Operating System
OSCARS	On-Demand Secure Circuits and Reservation System
OSPF	Open Shortest Path First
OWAMP	One-Way Active Measurement Protocol
PB	Petabyte
POP	Point of Presence
RAM	Random Access Memory
RDMA	Remote Direct Memory Access
REN	Research and Education Network
RTT	Round-Trip Time
RTT_{min}	Minimum Round-Trip Time
SACK	Selective Acknowledgement
SCP	Secure Copy Protocol
SDMZ	Science Demilitarized Zone
SDN	Software Defined Networking
sFlow	Sampled Flow
SFTP	Secure File Transfer Protocol
SNMP	Simple Network Management Protocol
SPAN	Switched Port Analyzer
SPI	Server Protocol Interpreter
SQL	Structured Query Language
SSH	Secure Shell
TB	Terabyte
Tbps	Terabits Per Second
TCP	Transmission Control Protocol
UDP	User Datagram Protocol
UDT	UDP-based Data Transfer Protocol
UMA	Uniform Memory Access
UNM	University of New Mexico
uRPF	Unicast Reverse Path Forwarding
U.S.	United States
VLA	Very Large Array
VLAN	Virtual Local Area Network
VM	Virtual Machine
VOQ	Virtual Output Queueing
VPLS	Virtual Private LAN Service
VPN	Virtual Private Network
VXLAN	Virtual Extensible Local Area Network
WAN	Wide Area Network
WRN	Western Regional Network
XML	Extensible Markup Language

REFERENCES

[1] D. McNichol, *The Roads That Built America: The Incredible Story of the U.S. Interstate System*. New York, NY, USA: Sterling, 2003.

[2] L. Farrell, "Science DMZ: The fast path for science data," *Sci. Node*, May 2016. [Online]. Available: https://sciencenode.org/feature/science-dmz-a-data-highway-system.php

[3] *The Energy Science Network*. Accessed: Oct. 30, 2018. [Online]. Available: https://www.es.net

[4] E. Dart, L. Rotman, B. Tierney, M. Hester, and J. Zurawski, "The science DMZ: A network design pattern for data-intensive science," in *Proc. Int. Conf. High Perform. Comput. Netw. Stor. Anal.*, Denver, CO, USA, Nov. 2013, pp. 1–10.

[5] *European Organization for Nuclear Research*. Accessed: Oct. 30, 2018. [Online]. Available: https://home.cern/about/computing

[6] *National Radio Astronomy Observatory*. Accessed: Oct. 30, 2018. [Online]. Available: http://www.vla.nrao.edu/

[7] J. Bashor, *General Atomics Remote Controls Fusion Experiments, Bridges Collaborators Using ESnet-Championed Technology*, Lawrence Berkeley Nat. Lab., Energy Sci. Netw., Berkeley, CA, USA, Sep. 2015. [Online]. Available: https://es.net/news-and-publications/esnet-news/2015/science-dmz-fuels-fusion-research/

[8] J. Van Horn and A. W. Toga, "Human neuroimaging as a 'big data' science," *J. Brain Imag. Behav.*, vol. 8, no. 2, pp. 323–331, Jun. 2014.

[9] S. Peisert *et al.*, "The medical science DMZ: A network design pattern for data-intensive medical science," *J. Amer. Med. Inf. Assoc.*, vol. 25, no. 3, pp. 267–274, Mar. 2018. [Online]. Available: https://academic.oup.com/jamia/article/doi/10.1093/jamia/ocx104/4367749/The-medical-science-DMZ-a-network-design-pattern

[10] S. Peisert *et al.*, "The medical science DMZ," *J. Amer. Med. Inf. Assoc.*, vol. 23, no. 6, pp. 1199–1201, May 2016.

[11] *General Electric Health Care*. Accessed: Oct. 30, 2018. [Online]. Available: http://www3.gehealthcare.com/en/global_gateway

[12] G. Roberts, *Big Data and the X-Ray Laser*, SLAC Nat. Accelerator Lab., Symmetry Mag., Menlo Park, CA, USA, Jun. 2013. [Online]. Available: https://www.symmetrymagazine.org/article/june-2013/big-data-and-the-x-ray-laser

[13] *SLAC National Accelerator Laboratory*. Accessed: Oct. 30, 2018. [Online]. Available: https://www6.slac.stanford.edu/

[14] E. Waltz, "Portable DNA sequencer minion helps build the Internet of Living Things," *IEEE Spectr. Mag.*, Mar. 2016. [Online]. Available: https://spectrum.ieee.org/the-human-os/biomedical/devices/portable-dna-sequencer-minion-help-build-the-internet-of-living-things

[15] *Nanopore Technologies*. [Online]. Available: https://nanoporetech.com/

[16] "A brief history of NSF and the Internet," Nat. Sci. Found., Alexandria, VA, USA, Fact Sheet, Aug. 2003. Accessed: Oct. 30, 2018. [Online]. Available: https://www.nsf.gov/news/news_summ.jsp?cntn_id=103050

[17] *National Science Foundation (NSF) Campus Cyberinfrastructure Program*. Accessed: Oct. 30, 2018. [Online]. Available: https://www.nsf.gov/funding/pgm_summ.jsp?pims_id=504748

[18] A. Gonzalez *et al.*, "Monitoring big data transfers over international research network connections," in *Proc. IEEE Int. Congr. Big Data*, Jun. 2017, pp. 334–351.

[19] Z. Liu, P. Balaprakash, R. Kettimuthu, and I. Foster, "Explaining wide area data transfer performance," in *Proc. IEEE/ACM Int. Symp. High Perform. Distrib. Comput. (HPDC)*, Jun. 2017, pp. 167–178.

[20] E. Kissel, M. Swany, B. Tierney, and E. Pouyoul, "Efficient wide area data transfer protocols for 100 Gbps networks and beyond," in *Proc. 3rd Int. Workshop Netw. Aware Data Manag.*, Nov. 2013, Art. no. 3.

[21] "Event-based software-defined networking: Build a secure science DMZ," Cisco Syst., San Jose, CA, USA, White Paper, 2015. [Online]. Available: https://www.cisco.com/c/en/us/products/collateral/cloud-systems-management/open-sdn-controller/white-paper-c11-735868.html

[22] "Software-driven science DMZ networks," Brocade, San Jose, CA, USA, White Paper, 2016. [Online]. Available: https://www.brocade.com/content/dam/common/documents/content-types/solution-brief/brocade-software-driven-science-dmz-networks-sb.pdf

[23] "Transform large-scale science collaboration," Ciena, Hanover, MD, USA, White Paper. Accessed: Oct. 30, 2018. [Online]. Available: http://media.ciena.com/documents/Science+DMZ+AN.pdf

[24] N. Cardwell, Y. Cheng, C. S. Gunn, S. H. Yeganeh, and V. Jacobson, "BBR: Congestion-based congestion control," *Commun. ACM*, vol. 60, no. 2, pp. 58–66, Feb. 2017.

[25] J. Crichigno, Z. Csibi, E. Bou-Harb, and N. Ghani, "Impact of segment size and parallel streams on TCP BBR," in *Proc. IEEE Int. Conf. Telecommun. Signal Process.*, Athens, Greece, Jul. 2018, pp. 1–5.

[26] K. Chard, S. Tuecke, and I. Foster, "Globus: Recent enhancements and future plans," in *Proc. XSEDE16 Conf. Diversity Big Data Sci. Scale*, Miami, FL, USA, Jul. 2016, Art. no. 27.

[27] "Processing of single stream large session (elephant flow) by the firepower services," Cisco Syst., San Jose, CA, USA, White Paper, Jan. 2017. [Online]. Available: https://www.cisco.com/c/en/us/support/docs/security/firepower-management-center/200420-Processing-of-Single-Stream-Large-Sessio.pdf

[28] V. Stoffer, A. Sharma, and J. Krous, "100G intrusion detection," Lawrence Berkeley Nat. Lab., Berkeley, CA, USA, Rep., Aug. 2015. Accessed: Oct. 30, 2018. [Online]. Available: https://www.cspi.com/wp-content/uploads/2016/09/Berkeley-100GIntrusionDetection.pdf

[29] K. Roberts, Q. Zhuge, I. Monga, S. Gareau, and C. Laperle, "Beyond 100 Gb/s: Capacity, flexibility, and network optimization," *J. Opt. Commun. Netw.*, vol. 9, no. 4, pp. C12–C23, Apr. 2017.

[30] J. Quittek, T. Zseby, B. Claise, and S. Zander, "Requirements for IP flow information export (IPFIX)," Internet Eng. Task Force, Fremont, CA, USA, RFC 3917, Jul. 2008. [Online]. Available: http://www.ietf.org/rfc/rfc3917.txt

[31] B. Claise, "Specification of the IP flow information export (IPFIX) protocol for the exchange of IP traffic flow information," Internet Eng. Task Force, Fremont, CA, USA, RFC 5101, Jul. 2008. [Online]. Available: http://www.ietf.org/rfc/rfc5101.txt

[32] K. Fall and S. Floyd, "Simulation-based comparisons of Tahoe, Reno, and Sack TCP," *ACM SIGCOMM Comput. Commun. Rev.*, vol. 26, no. 3, pp. 5–21, Jul. 1996.

[33] D. J. Leith, R. N. Shorten, and Y. Lee, "H-TCP: A framework for congestion control in high-speed and long-distance networks," Hamilton Inst., Maynooth, Ireland, Rep., Aug. 2005. [Online]. Available: http://www.hamilton.ie/net/htcp2005.pdf

[34] M. Mathis, J. Semke, J. Mahdavi, and T. Ott, "The macroscopic behavior of the TCP congestion avoidance algorithm," *ACM SIGCOMM Comput. Commun. Rev.*, vol. 27, no. 3, pp. 67–82, Jul. 1997.

[35] W. Allcock, J. Bresnahan, R. Kettimuthu, and M. Link, "The globus striped gridFTP framework and server," in *Proc. ACM/IEEE Conf. Supercomput.*, Seattle, WA, USA, Nov. 2005, p. 54.

[36] B. Radic, V. Kajic, and E. Imamagic, "Optimization of data transfer for grid using gridFTP," in *Proc. Int. Conf. Inf. Technol. Interfaces*, Cavtat, Croatia, Jun. 2008, pp. 709–715.

[37] J. Postel, "Transmission control protocol (TCP)," Internet Eng. Task Force, Fremont, CA, USA, RFC 793, Sep. 1981. [Online]. Available: https://tools.ietf.org/html/rfc793

[38] J. Zurawski *et al.*, "perfSONAR: On-board diagnostics for big data," in *Proc. Workshop Big Data Sci. Infrastruct. Services*, Oct. 2013, pp. 1–6.

[39] A. Hanemann *et al.*, "perfSONAR: A service oriented architecture for multi-domain network monitoring," in *Proc. 3rd Int. Conf. Service Orient. Comput.*, Dec. 2005, pp. 241–254.

[40] *Internet2*. Accessed: Oct. 30, 2018. [Online]. Available: https://www.internet2.edu/

[41] B. Claise, "Cisco systems netflow services export version 9," Internet Eng. Task Force, Fremont, CA, USA, RFC 3954, Oct. 2004. [Online]. Available: https://www.ietf.org/rfc/rfc3954.txt

[42] K. Miller. *DDOS Mitigation With sFlow*. Accessed: Oct. 30, 2018. [Online]. Available: http://www.rn.psu.edu/2014/07/25/ddos-mitigation-with-sflow/

[43] R. Hofstede, A. Pras, A. Sperotto, and G. D. Rodosek, "Flow-based compromise detection: Lessons learned," *IEEE Security Privacy*, vol. 16, no. 1, pp. 82–89, Jan./Feb. 2018.

[44] F. Farina, P. Szegedi, and J. Sobieski, "GÉANT world testbed facility: Federated and distributed testbeds as a service facility of GÉANT," in *Proc. Int. Tele Traffic Congr.*, Karlskrona, Sweden, Sep. 2014, pp. 1–6.

[45] *Ubuntunet*. Accessed: Oct. 30, 2018. [Online]. Available: https://ubuntunet.net/

[46] *Asia Pacific Advanced Network*. Accessed: Oct. 30, 2018. [Online]. Available: https://apan.net/

[47] *Red Clara Network*. Accessed: Oct. 30, 2018. [Online]. Available: https://www.redclara.net/index.php/en/

[48] *The Western Regional Network*. Accessed: Oct. 30, 2018. [Online]. Available: http://nets.ucar.edu/nets/ongoing-activities/wrn/wrnroot/

[49] *The Corporation for Education Network Initiatives in California*. Accessed: Oct. 30, 2018. [Online]. Available: http://cenic.org

[50] K. Thompson, "Campus cyberinfrastructure," in *Proc. Principal Investigators Workshop NSF Campus Cyberinfrastructure Program*, Oct. 2016. [Online]. Available: https://www.thequilt.net/wp-content/uploads/CC_PIMeeting2016_KLT.pdf

[51] J. Moy, "Open shortest path first (OSPF) version 2," Internet Eng. Task Force, Fremont, CA, USA, RFC 2328, Apr. 1998. [Online]. Available: https://www.ietf.org/rfc/rfc2328.txt

[52] Y. Rekhter, T. Li, and S. Hares, "Border gateway protocol 4," Internet Eng. Task Force, Fremont, CA, USA, RFC 4271, Jan. 2006. [Online]. Available: https://tools.ietf.org/html/rfc4271

[53] J. F. Kurose and K. W. Ross, *Computer Networking: A Top-Down Approach*, 7th ed. Boston, MA, USA: Pearson, 2017.

[54] *Router/Switch Buffer Size Issues*. Accessed: Oct. 30, 2018. [Online]. Available: https://fasterdata.es.net/network-tuning/router-switch-buffer-size-issues/

[55] C. Villamizar and C. Song, "High performance TCP in ANSNET," *ACM SIGCOMM Comput. Commun. Rev.*, vol. 24, no. 5, pp. 45–60, Oct. 1994.

[56] R. Bush and D. Meyer, "Some Internet architectural guidelines and philosophy," Internet Eng. Task Force, Fremont, CA, USA, RFC 3439, Dec. 2003. [Online]. Available: https://www.ietf.org/rfc/rfc3439.txt

[57] G. Appenzeller, I. Keslassy, and N. McKeown, "Sizing router buffers," in *Proc. Conf. Appl. Technol. Archit. Protocols Comput. Commun.*, Portland, OR, USA, Oct. 2004, pp. 281–292.

[58] J. Padhye, V. Firoiu, D. Towsley, and J. Kurose, "Modeling TCP throughput: A simple model and its empirical validation," in *Proc. ACM SIGCOMM Conf. Appl. Technol. Archit. Protocols Comput. Commun.*, Vancouver, BC, Canada, Sep. 1998, pp. 303–314.

[59] M. Smitasin and B. Tierney, "Evaluating network buffer size requirements," in *Proc. Technol. Exchange Workshop*, Oct. 2015. [Online]. Available: https://meetings.internet2.edu/media/medialibrary/2015/10/05/20151005-smitasin-buffersize.pdf

[60] B. Tierney, "Improving performance of 40G/100G data transfer nodes," in *Proc. Technol. Exchange Workshop*, Sep. 2016. [Online]. Available: https://meetings.internet2.edu/2016-technology-exchange/detail/10004333/

[61] V. Paxson and S. Floyd, "Wide area traffic: The failure of Poisson modeling," *IEEE/ACM Trans. Netw.*, vol. 3, no. 3, pp. 226–244, Jun. 1995.

[62] N. Beheshti *et al.*, "Optical packet buffers for backbone Internet routers," *IEEE/ACM Trans. Netw.*, vol. 18, no. 5, pp. 1599–1609, Oct. 2010.

[63] V. G. Cerf, "Bufferbloat and other Internet challenges," *IEEE Internet Comput.*, vol. 18, no. 5, p. 80, Sep./Oct. 2014.

[64] H. Im, C. Joo, T. Lee, and S. Bahk, "Receiver-side TCP countermeasure to bufferbloat in wireless access networks," *IEEE Trans. Mobile Comput.*, vol. 15, no. 8, pp. 2080–2093, Aug. 2016.

[65] K. Nichols, V. Jacobson, A. McGregor, and J. Iyengar, "Controlled delay active queue management," Internet Eng. Task Force, Fremont, CA, USA, Internet Draft draft-ietf-aqm-codel-10, Oct. 2017. [Online]. Available: https://tools.ietf.org/html/draft-ietf-aqm-codel-10

[66] *Linux Tuning*. Accessed: Oct. 30, 2018. [Online]. Available: https://fasterdata.es.net/host-tuning/linux/

[67] "Cisco catalyst 6500 supervisor 2T architecture white paper," Cisco Syst., San Jose, CA, USA, White Paper, Jan. 2017. [Online]. Available: https://www.cisco.com/c/en/us/products/collateral/switches/catalyst-6500-series-switches/white_paper_c11-676346.html#_Toc390815326

[68] N. McKeown, A. Mekkittikul, V. Anantharam, and J. Walrand, "Achieving 100% throughput in an input-queued switch," *IEEE Trans. Commun.*, vol. 47, no. 8, pp. 1260–1267, Aug. 1999.

[69] G. Vardoyan, R. Kettimuthu, M. Link, and S. Tuecke, "Characterizing throughput bottlenecks for secure gridFTP transfers," in *Proc. IEEE Int. Conf. Comput. Netw. Commun. (ICNC)*, San Diego, CA, USA, Jan. 2013, pp. 861–866.

[70] D. Borman, B. Braden, V. Jacobson, and R. Scheffenegger, "TCP extensions for high performance," Internet Eng. Task Force, Fremont, CA, USA, RFC 7323, Sep. 2014. [Online]. Available: https://tools.ietf.org/html/rfc7323#section-4.2

[71] N. Mills, F. A. Feltus, and W. B. Ligon, III, "Maximizing the performance of scientific data transfer by optimizing the interface between parallel file systems and advanced research networks," *J. Future Gener. Comput. Syst.*, vol. 79, pp. 190–198, Feb. 2018. [Online]. Available: https://doi.org/10.1016/j.future.2017.04.030

[72] F. A. Feltus *et al.*, "The widening gulf between genomics data generation and consumption: A practical guide to big data transfer technology," *J. Bioinformat. Biol. Insights*, vol. 9, no. 1, pp. 9–19, Sep. 2015.

[73] T. J. Hacker, B. D. Athey, and B. Noble, "The end-to-end performance effects of parallel TCP sockets on a lossy wide-area network," in *Proc. Parallel Distrib. Process. Symp.*, Apr. 2001, p. 10.

[74] A. Aggarwal, S. Savage, and T. Anderson, "Understanding the performance of TCP pacing," in *Proc. Int. Conf. Comput. Commun. (INFOCOM)*, Mar. 2000, pp. 1157–1165.

[75] N. Hanford, B. Tierney, and D. Ghosal, "Optimizing data transfer nodes using packet pacing," in *Proc. Workshop Innov. Netw. Data-Intensive Sci.*, Nov. 2015, pp. 1–8.

[76] M. Ghobadi and Y. Ganjali, "TCP pacing in data center networks," in *Proc. IEEE Ann. Symp. High Perform. Interconnects (HOTI)*, Aug. 2013, pp. 25–32.

[77] *The Centos Project*. Accessed: Oct. 30, 2018. [Online]. Available: https://www.centos.org/

[78] *TSO Sizing and the FQ Scheduler*. Accessed: Oct. 30, 2018. [Online]. Available: https://lwn.net/Articles/564978/

[79] I. Rhee and L. Xu, "CUBIC: A new TCP-friendly high-speed TCP variant," *ACM SIGOPS Oper. Syst. Rev.*, vol. 42, no. 5, pp. 64–74, Jul. 2008.

[80] T. Dierks and E. Rescorla, "The transport layer security (TLS) protocol version 1.2," Internet Eng. Task Force, Fremont, CA, USA, RFC 5246, Aug. 2008. [Online]. Available: https://tools.ietf.org/html/rfc5246

[81] A. Freier, P. Karlton, and P. Kocher, "The secure sockets layer protocol version 3.0," Internet Eng. Task Force, Fremont, CA, USA, RFC 6101, Aug. 2011. [Online]. Available: https://tools.ietf.org/html/rfc6101

[82] J. Postel and J. Reynolds, "File transfer protocol," Internet Eng. Task Force, Fremont, CA, USA, RFC 959, Oct. 1985. [Online]. Available: https://tools.ietf.org/html/rfc959

[83] T. Ylonen and C. Lonvick, "The secure shell (SSH) connection protocol," Internet Eng. Task Force, Fremont, CA, USA, RFC 4254, Jan. 2006. [Online]. Available: https://tools.ietf.org/html/rfc4254

[84] L. Zhang, W. Wu, P. DeMar, and E. Pouyoul, "mdtmFTP and its evaluation on ESNET SDN testbed," *Future Gener. Comput. Syst.*, vol. 79, pp. 199–204, Feb. 2018. [Online]. Available: https://doi.org/10.1016/j.future.2017.04.024

[85] *Fast Data Transfer (FDT)*. Accessed: Oct. 30, 2018. [Online]. Available: http://monalisa.cern.ch/FDT

[86] "Ultra high-speed transport technology," Emeryville, CA, USA, Aspera, White Paper. Accessed: Oct. 30, 2018. [Online]. Available: http://asperasoft.com/resources/white-papers/ultra-high-speed-transport-technology/

[87] *High-End Computing Capability Using BBFTP for Remote File Transfers*. Accessed: Oct. 30, 2018. [Online]. Available: https://www.nas.nasa.gov/hecc/support/kb/using-bbftp-for-remote-file-transfers_147.html

[88] "Expedata, a multipurpose transaction protocol," Norman, OK, USA, Expedat, White Paper, Jan. 2017. [Online]. Available: http://www.dataexpedition.com/expedat/Docs/

[89] Y. Gu and R. L. Grossman, "UDT: UDP-based data transfer for high-speed wide area networks," *Comput. Netw.*, vol. 51, no. 7, pp. 1777–1799, May 2007.

[90] D. V. Bernardo and D. B. Hoang, "Empirical survey: Experimentation and implementations of high speed protocol data transfer for grid," in *Proc. IEEE Int. Conf. Adv. Inf. Netw. Appl. Workshops*, Mar. 2011, pp. 335–340.

[91] *UDP Tuning in Science DMZs*. Accessed: Oct. 30, 2018. [Online]. Available: https://fasterdata.es.net/network-tuning/udp-tuning/#toc-anchor-1

[92] "Performance evaluation of VMXNET3 virtual network device," VMware, Palo Alto, CA, USA, Rep. Accessed: Oct. 30, 2018. [Online]. Available: https://www.vmware.com/pdf/vsp_4_vmxnet3_perf.pdf

[93] D. Levi, P. Meyer, and B. Stewart, "Simple network management protocol (SNMP) applications," Internet Eng. Task Force, Fremont, CA, USA, RFC 3413, Dec. 2002. [Online]. Available: https://tools.ietf.org/html/rfc3413

[94] C. Lonvick, "The BSD Syslog protocol," Internet Eng. Task Force, Fremont, CA, USA, RFC 3164, Aug. 2001. [Online]. Available: https://www.ietf.org/rfc/rfc3164.txt

[95] *jq Command-Line JSON Processor*. Accessed: Oct. 30, 2018. [Online]. Available https://stedolan.github.io/jq/

[96] T. Bray, "The JavaScript object notation (JSON) data interchange format," Internet Eng. Task Force, Fremont, CA, USA, RFC 7159, Mar. 2014. [Online]. Available: https://tools.ietf.org/html/rfc7159

[97] S. Hemminger, "Network emulation with NetEm," in *Proc. Australia's Nat. Linux Conf.*, Apr. 2005, pp. 18–93.

[98] *iPerf3*. Accessed: Oct. 30, 2018. [Online]. Available: http://software.es.net/iperf/

[99] R. Mijumbi *et al.*, "Network function virtualization: State-of-the-art and research challenges," *IEEE Commun. Surveys Tuts.*, vol. 18, no. 1, pp. 236–262, 1st Quart., 2016.

[100] M. Mahalingam *et al.*, "Virtual extensible local area network (VXLAN): A framework for overlaying virtualized layer 2 networks over layer 3 networks," Internet Eng. Task Force, Fremont, CA, USA, RFC 7348, Aug. 2014.

[101] T. Orawiwattanakul, H. Otsuki, E. Kawai, and S. Shimojo, "Multiple classes of service provisioning with bandwidth and delay guarantees in dynamic circuit network," in *Proc. IEEE Int. Symp. Integr. Netw. Manag.*, May 2015, pp. 475–483.

[102] E. Rosen, A. Viswanathan, and R. Callon, "Multiprotocol label switching architecture," Internet Eng. Task Force, Fremont, CA, USA, RFC 3031, Jan. 2001. [Online]. Available: https://tools.ietf.org/html/rfc3031.txt

[103] "Building scalable Syslog management solutions," San Jose, CA, USA, Cisco, White Paper, 2009. [Online]. Available: https://www.cisco.com/c/en/us/products/collateral/services/high-availability/white_paper_c11-557812.html#wp9000392

[104] K. Johnson and T. DeLaGrange, "SANS survey on mobility/BYOD security policies and practices," North Bethesda, MD, USA, SANS, White Paper, Oct. 2012. [Online]. Available: https://www.sans.org/reading-room/whitepapers/analyst/survey-mobility-byod-security-policies-practices-35175

[105] "Cisco Nexus 3100 platform switch architecture," San Jose, CA, USA, Cisco Systems, White Paper, Oct. 2013. [Online]. Available: https://people.ucsc.edu/∼warner/Bufs/cisco-3100-arch.pdf

[106] *Brown University (Firewall) Example*. Accessed: Oct. 30, 2018. [Online]. Available: https://fasterdata.es.net/performance-testing/perfsonar/perfsonar-success-stories/brown-university-example/

[107] "Cisco firepower NGIPS," Data Sheet, Cisco Syst., San Jose, CA, USA, 2017. [Online]. Available: https://www.cisco.com/c/en/us/products/collateral/security/ngips/datasheet-c78-738196.html

[108] *Snort Open Source Intrusion Prevention System*. Accessed: Oct. 30, 2018. [Online]. Available: https://www.snort.org/

[109] A. Sperotto *et al.*, "An overview of IP flow-based intrusion detection," *IEEE Commun. Surveys Tuts.*, vol. 12, no. 3, pp. 343–356, 3rd Quart., 2010.
[110] R. Hofstede *et al.*, "Flow monitoring explained: From packet capture to data analysis with NetFlow and IPFIX," *IEEE Commun. Surveys Tuts.*, vol. 16, no. 4, pp. 2037–2064, 4th Quart., 2014.
[111] *The Bro Network Security Monitor*. Accessed: Oct. 30, 2018. [Online]. Available: http://www.broids.org
[112] W. Kumari and D. McPherson, "Remote triggered black hole filtering with unicast reverse path forwarding (uRPF)," Internet Eng. Task Force, Fremont, CA, USA, RFC 5635, Aug. 2009. [Online]. Available: https://tools.ietf.org/html/rfc5635
[113] N. Pho *et al.*, "Data transfer in a science DMZ using SDN with applications for precision medicine in cloud and high-performance computing," in *Proc. Int. Conf. High Perform. Comput. Netw. Storage Anal. (SC15)*, Nov. 2015, pp. 1–4.
[114] D. Hardt, "The OAuth 2.0 authorization framework," Internet Eng. Task Force, Fremont, CA, USA, RFC 6749, Oct. 2012. [Online]. Available: https://tools.ietf.org/html/rfc6749
[115] S. Kent and S. Seo, "Security architecture for the Internet protocol," Internet Eng. Task Force, Fremont, CA, USA, RFC 4301, Dec. 2005. [Online]. Available: https://tools.ietf.org/html/rfc4301
[116] "MX960, MX480, MX240, MX104 and MX80 3D universal edge routers," Data Sheet, Juniper Netw., Sunnyvale, CA, USA, 2018. [Online]. Available: http://www.juniper.net/assets/us/en/local/pdf/datasheets/1000597-en.pdf
[117] "Cisco Nexus 3172PQ, 3172TQ, 3172TQ-32T, 3172PQ-XL, and 3172TQ-XL switches," Data Sheets, Cisco Syst., San Jose, CA, USA, Jan. 2017. [Online]. Available: https://www.cisco.com/c/en/us/products/collateral/switches/nexus-3000-series-switches/data_sheet_c78-729483.html
[118] Y.-T. Li, D. Leith, and R. Shorten, "Experimental evaluation of TCP protocols for high-speed networks," *IEEE/ACM Trans. Netw.*, vol. 15, no. 5, pp. 1109–1122, Oct. 2007.
[119] *VMware ESXI*. Accessed: Oct. 30, 2018. [Online]. Available: https://www.vmware.com/products/esxi-and-esx.html
[120] J. Crichigno *et al.*, "Optimal traffic scheduling for intrusion prevention systems," *Int. J. Adv. Telecommun. Electrotechn. Signals Syst.*, vol. 6, no. 2, pp. 1–4, 2017.
[121] D. Kreutz *et al.*, "Software-defined networking: A comprehensive survey," *Proc. IEEE*, vol. 103, no. 1, pp. 14–76, Jan. 2015.
[122] E. Balas and A. Ragusa, "SciPass: A 100Gbps capable secure science DMZ using OpenFlow and Bro," in *Proc. Technol. Exchange Conf.*, Oct. 2014, pp. 73–79.
[123] J. Crichigno and N. Ghani, "A linear programming scheme for IPS traffic scheduling," in *Proc. IEEE Int. Conf. Telecommun. Signal Process. (TSP)*, Jul. 2015, pp. 16–20.
[124] J. Tveten, *On American Indian Reservations, Challenges Perpetuate the Digital Divide*, ARS Technica, Jan. 2016. [Online]. Available: https://arstechnica.com
[125] N. Sambuli, "Challenges and opportunities for advancing Internet access in developing countries while upholding net neutrality," *J. Cyber Policy*, vol. 1, no. 1, pp. 61–74, May 2016.
[126] J. Saltzer, D. Reed, and D. Clark, "End-to-end argument in system design," *ACM Trans. Comput. Syst.*, vol. 2, no. 4, pp. 277–288, Nov. 1984.
[127] J. M. Plante, D. A. P. Davis, and V. M. Vokkarane, "Parallel circuit provisioning in ESnet's OSCARS," in *Proc. IEEE Int. Conf. Adv. Netw. Telecommun. Syst. (ANTS)*, Dec. 2014, pp. 1–6.
[128] I. Monga, C. Guok, W. E. Johnston, and B. Tierney, "Hybrid networks: Lessons learned and future challenges based on ESnet4 experience," *IEEE Commun. Mag.*, vol. 49, no. 5, pp. 114–121, May 2011.
[129] A. Gumaste, T. Das, K. Khandwala, and I. Monga, "Network hardware virtualization for application provisioning in core networks," *IEEE Commun. Mag.*, vol. 55, no. 2, pp. 152–159, Feb. 2017.
[130] K. Slavicek, V. Novak, and J. Ledvinka, "CESNET fiber optics transport network," in *Proc. IEEE Int. Conf. Netw.*, Mar. 2009, pp. 403–408.
[131] *Transpacific Encryption Success at 100Gbps*, Ericsson, Stockholm, Sweden, Sep. 2017. [Online]. Available: https://www.ericsson.com/en/press-releases/2017/9/transpacific-encryption-success-at-100gbps
[132] W.-B. Lee and C.-D. Lee, "A cryptographic key management solution for HIPAA privacy/security regulations," *IEEE Trans. Inf. Technol. Biomed.*, vol. 12, no. 1, pp. 34–41, Jan. 2008
[133] M. A. Alyami and Y.-T. Song, "Removing barriers in using personal health record systems," in *Proc. IEEE Int. Conf. Comput. Inf. Sci. (ICIS)*, Jun. 2016, pp. 1–8.

Jorge Crichigno received the Ph.D. degree in computer engineering from the University of New Mexico, Albuquerque, USA. He is an Associate Professor with the Integrated Information Technology Department, College of Engineering and Computing, University of South Carolina. He has also been a Research Associate with the Electrical Engineering Department, University of South Florida since 2016. His current research interests are in the areas of network and protocol optimization for high-throughput high-latency systems, and Internet measurements for cyber security. He has served as a Reviewer and a TPC Member of journals and conferences, such as the IEEE TRANSACTIONS ON MOBILE COMPUTING and IEEE Globecom, and as a panelist for the National Science Foundation. He is an ABET Evaluator representing the IEEE.

Elias Bou-Harb received the Ph.D. degree in computer science from Concordia University, Montreal, Canada. He was a Visiting Research Scientist with Carnegie Mellon University, Pittsburgh, PA, USA, in 2015 and 2016. He is currently an Assistant Professor with the Computer Science Department, Florida Atlantic University. He is also a Research Scientist with National Cyber Forensic and Training Alliance, Canada. His current research interests are in the areas of cyber security, big data analytics, data-driven digital forensics, Internet measurements for cyber security, and cyber security for critical infrastructure.

Nasir Ghani received the Ph.D. degree in electrical engineering from the University of Waterloo, Canada. He was the Associate Chair of the Electrical and Computer Engineering Department, University of New Mexico from 2007 to 2013, and a Faculty Member with Tennessee Tech University from 2003 to 2007. He is a Professor with the Electrical Engineering Department, University of South Florida and a Research Liaison for Cyber Florida, a state-funded center focusing on cybersecurity research, education, and outreach. He has also spent several years working in industry at large blue chip organizations (IBM, Motorola, and Nokia) and several hi-tech startups. His research interests include cyberinfrastructure networks, cybersecurity, cloud computing, disaster recovery, and IoT/cyberphysical systems. He has published over 230 peer-reviewed articles and has several highly cited U.S. patents. He has also served as an Associate Editor for various journals, including the IEEE/OSA JOURNAL OF OPTICAL AND COMMUNICATIONS AND NETWORKING, IEEE SYSTEMS, and IEEE COMMUNICATIONS LETTERS. He has also guest-edited special issues of IEEE NETWORK and *IEEE Communications Magazine* and has organized and chaired symposia and workshops for numerous flagship IEEE conferences, such as IEEE Globecom, IEEE ICC, IEEE Infocom, and IEEE ICCCN. He was also the Chair of the IEEE Technical Committee on High Speed Networking from 2007 to 2010.