White Paper

Event-Based Software-Defined Networking: Build a Secure Science DMZ

What You Will Learn

As the need to efficiently move large data sets around the world increases, the Science DMZ - built at the network edge and designed to be secure without the performance limitations imposed by traditional security devices such as firewalls - is becoming vital. This document explores an event-based software-defined networking (SDN) solution that improves both the security and efficiency of the traditional science DMZ. The document discusses:

- Current science DMZ implementations and weaknesses
- Network functions needed for a secure science DMZ
- Concepts of event-based SDN
- Details of the reference implementation

This document is intended for individuals responsible for designing and engineering solutions for networks that involve the movement of large amounts of data.

Background

Scientific research increasingly relies on very large data flows, with large collaborative partnerships of researchers and the transfer of data from experiments and simulations around the world. The unique characteristics of this huge data transfer pose new networking challenges:

- Traditional campus networks are designed for enterprise business operations. They typically are designed for a very large number of small flows and are not well suited to the bulk transfer of scientific data, which is characterized by a small number of very large flows.
- Sharing scientific data characterized by large flows with traditional campus networks has significant drawbacks. For typical data traffic, packet loss is often tolerated in the campus LAN. However, even very small amounts of packet loss can reduce TCP performance by an order of magnitude when WAN latency is introduced, and hence such a solution does not meet the stringent requirements for the movement of scientific data.
- The hardware limitations of firewalls are generally exposed when the heavy network-traffic loads of big data flows are managed under complex firewall rule-set constraints. Other limitations, such as old fiber optics also pose performance constraints on these large flows.
- Traditional campus networks are optimized for security and partially sacrifice performance for this purpose. The security optimization in traditional networks leads to campus firewall policies that block ports or limit flows needed for various data-intensive experiments.
- The traffic engineering methods in traditional campus networks cannot perform the detailed classification of flows needed to enforce big data policies for bandwidth provisioning.

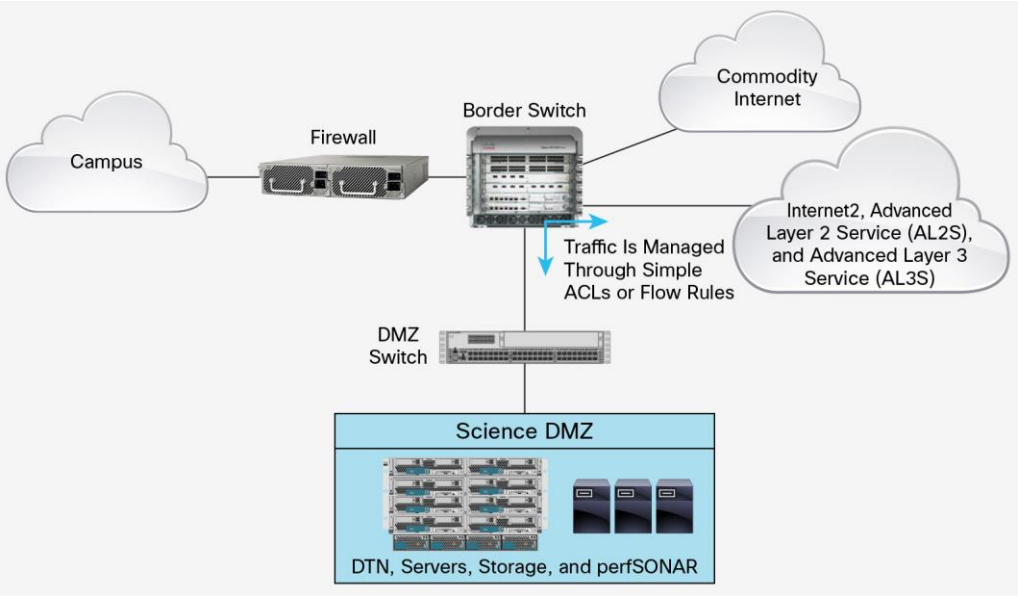
Science DMZ

The inability of the traditional campus infrastructure to cater to on-demand science and big data applications requires a new approach. The term “Science DMZ” refers to a class of networks built outside an institutional firewall and that are structured to be secure without the performance impacts that are common with traditional security devices (for example, inline firewalls and intrusion prevention systems [IPSs]).

- They are based on a network architecture explicitly designed for high-performance applications and in which the science network is distinct from the general-purpose network.
- They typically are deployed at or near the edge of an organization and are optimized for a moderate number of high-speed flows rather than a larger number of general-purpose business-system or enterprise-computing flows.
- They provide infrastructure and security policies and enforcement mechanisms that are tailored for high-performance science environments.
- They can be customized according to application needs to start the movement of data-intensive science flows to fast WAN backbones (for example, Internet2 in the United States).
- They allow dynamic identification and orchestration of big data application traffic to bypass the campus enterprise firewall.
- They use dedicated systems for data transfer (data transmission networks [DTNs]) that foster flow acceleration.
- They include performance measurement and network testing systems that are regularly used to characterize the network and are available for troubleshooting.

Figure 1 illustrates transit selection within a campus access network in which the intelligence at the campus border routes research data flows to specific research network paths, bypassing the campus firewall. However, enterprise traffic, such as web browsing and email, is routed through the same campus access network to the Internet through the firewall-policed paths.

Figure 1. Science DMZ Overview



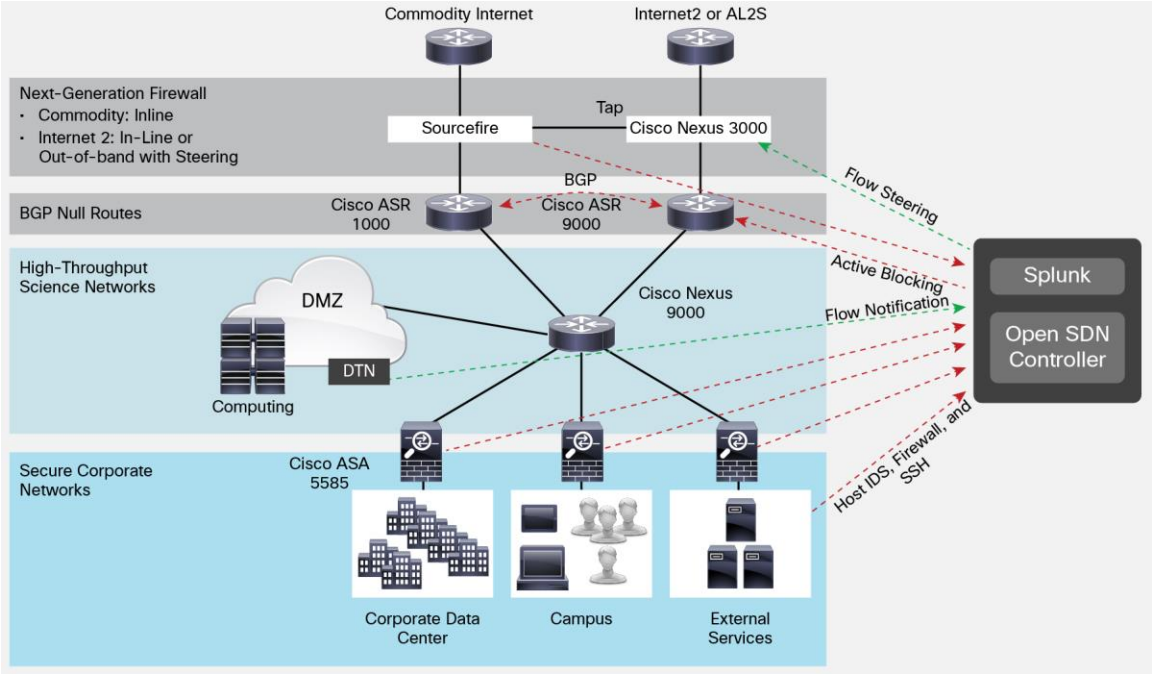
Build a Solution to Secure the Science DMZ

This document explores a solution to secure a Science DMZ in a production environment that meets the policy requirements of most institutions.

SDN controllers can rapidly respond to changing conditions in the network. In the case of a Science DMZ, the controller can actively block attacks at the enterprise border and steer huge data flows around firewalls and intrusion detection systems (IDSs) that would otherwise be overwhelmed by them or limit their performance.

Figures 2 presents an overview of this solution.

Figure 2. Secure Science DMZ Overview



In this solution, the Cisco® Open SDN Controller directs changes in the network. It is a central point that accepts representational state transfer (REST) commands, updates any applicable state within the controller, and then constructs the appropriate device-specific action and sends it using the appropriate protocol for that device (OpenFlow, NETCONF, etc.).

As a powerful event aggregation database with the capability to process arbitrary Python code, Splunk is an excellent platform to act as a clearinghouse for all security and application events in the science DMZ. In this solution, Splunk logically sits on top of the Open SDN Controller as an application and ingests information from three main sources: the Cisco FireSIGHT® Management Center, Cisco Adaptive Security Appliance (ASA) firewall, and Globus. However, it can collect events from many other devices on the network, such as host-based IDSs, web-security appliances, and identity services, providing additional situational awareness. Splunk can then correlate events to identify patterns and thresholds and act on them by sending REST commands to the Open SDN Controller.

In this design, Splunk initiates two types of actions: it blocks security threats and it steers huge data flows around IDSs and firewalls:

- Blocking of security threats is accomplished with information from the Cisco FirePOWER™ IPS appliance, which sits out-of-band of the data flows, and ASA firewalls elsewhere in the network.
- Steering is accomplished by detecting huge data flows with active flow notification from Globus. Splunk monitors the event logs from the Globus GridFTP control mechanism for specific data-transfer flow information.

In either case, Splunk then uses available northbound REST APIs in the Open SDN Controller to block or steer traffic as described in more detail in the following sections.

Data Transfer Tools in the Science DMZ

GridFTP is the underlying technology for Globus file transfers. The GridFTP protocol specification extends FTP to provide secure, reliable, and efficient transfer of data across wide-area distributed networks. GridFTP extensions provide, among other things, parallelism (that is, the use of multiple socket connections between pairs of data movers), restart markers, and data channel security. GridFTP is an open-source implementation developed primarily at Argonne National Laboratory and the University of Chicago.

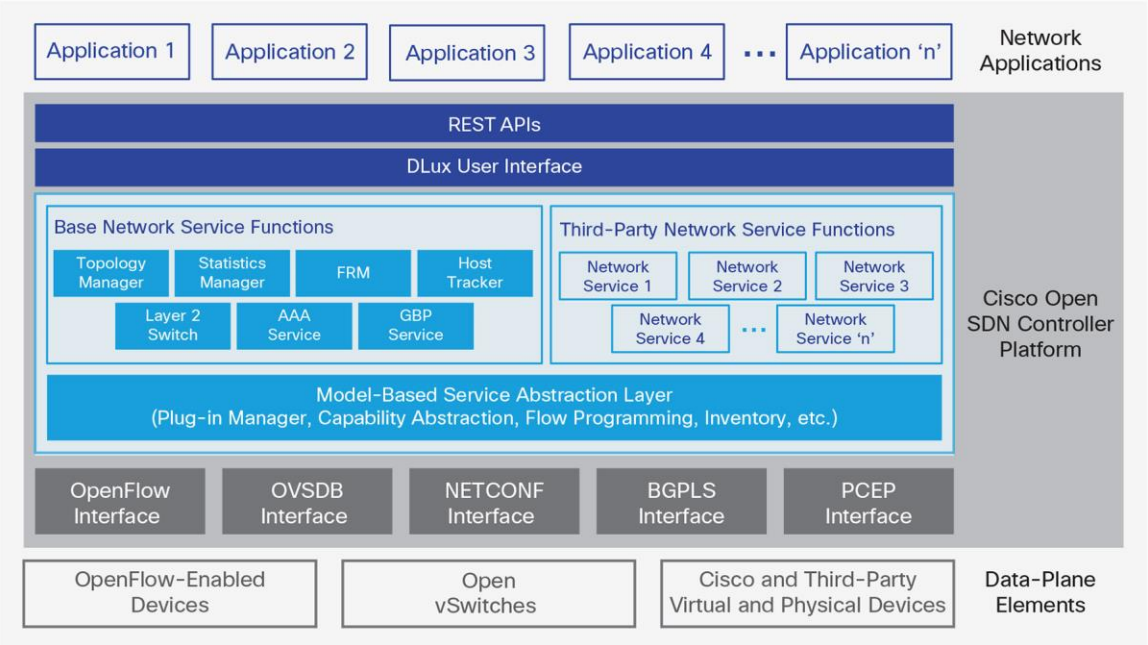
Wide-area file transfer involves optimal use of a variety of complex resources and protocols. This task is inherently complicated; GridFTP and similar tools have historically been complicated for the end user to control and to fully understand. In addition, large-size data transfers may take an extended time period to complete. This transfer time introduces reliability and recovery problems that the end user should not have to manage directly. To address these problems, Argonne National Laboratory and the University of Chicago have developed a software-as-a-service (SaaS) solution to manage transfers, called Globus transfer service, to which users can direct requests to transfer or synchronize files and directories between two locations. Globus transparently orchestrates GridFTP transfers and handles security, monitors transfers, and restarts transfers upon failure. With more than 10,000 active endpoints as of April 2015, Globus is an important element of the research-networking ecosystem.

Globus is one of the most widely adopted platforms for moving data between science sites globally and therefore makes an attractive target for science DMZ optimization. However, the general-purpose nature of the solution allows support for other data transfer applications with little effort.

Cisco Open SDN Controller

The Open SDN Controller (Figure 3) is a commercial distribution of OpenDaylight that delivers business agility through automation of standards-based network infrastructure. It abstracts away the complexity involved in managing heterogeneous network environments to improve service delivery and reduce operating costs. As open-source-based software, the Open SDN Controller continuously advances through the ongoing innovation and support of the OpenDaylight community.

Figure 3. Cisco Open SDN Controller Overview



Event Detection and Correlation with Splunk

In general, the completeness of cybersecurity increases with the quantity and quality of events. However, each event in isolation presents an incomplete picture. Event correlation takes all the events into account, combining them with site-specific heuristics to determine when to take a particular action. This approach dramatically reduces the number of false positives and missed events, which can otherwise limit the effectiveness of the system and compromise security. The solution discussed here uses Splunk for both event correlation and event storage.

The solution uses Splunk to detect three types of events: events from the FirePOWER IDS sensor, deny notifications from the ASA firewall, and flow notifications from Globus (see Figure 2). These events are acted on by a combination of real-time and scheduled searches within Splunk.

Real-Time Searches

Real-time searches are used for events that require immediate action or that require no correlation because of their severity or clarity. One example of such an event is a high-priority event from the FirePOWER IDS:

```
index=estreamer sourcetype=estreamer rec_type_simple="IPS EVENT" priority="high"
src_ip | eval action="block" | eval reason="IDS-HIGH"
```

This search looks for incoming security events from the FireSIGHT Event Streamer (eStreamer) for Splunk application that have a high priority. For events of this type, you typically automatically set the action to block and call Python code that sends the REST calls needed to block the source IP address.

Another example is a flow event generated by Globus during a file transfer request:

```
sourcetype=globus-netmgr event=flow action | eval params="event=flow"
```

This search looks for events coming from the globus-netmgr log with the event parameter set to **flow**. Splunk then picks up the detailed five-tuple supplied in the event notification and calls Python code that sends the REST calls needed to steer these known-good flows around security devices.

Scheduled Searches

Scheduled searches are used for events that need to be evaluated on a regular basis and are not necessarily handled in real time. The science DMZ solution uses the following scheduled search to automatically clear blocked source IP addresses if they have been quiet for more than 24 hours:

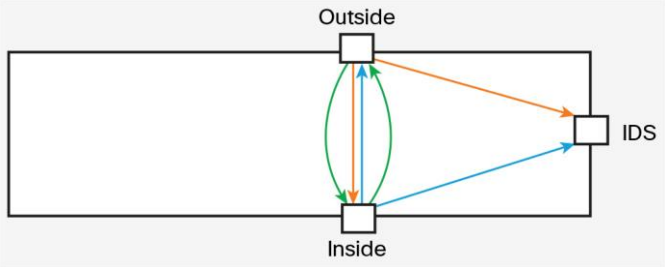
```
sourcetype=splunk_odl_action action=block earliest=-48h | stats count by src_ip |
table src_ip | search NOT [ search sourcetype=splunk_odl_action action=block
earliest=-24h | stats count by src_ip | table src_ip ] | eval
params="action=unblock,event=block_timeout".
```

Architectural Options

This solution can be deployed in two ways depending on the goals and priorities of the organization. If the main priority is the fastest possible data transfer, the IDS can be placed out-of-band and block attackers by injecting null routes into Border Gateway Protocol (BGP). If the rate of the ports being secured is greater than the capability of the IDS, huge data flows can be steered using events logged by Globus or other data-transfer utilities. If initiation of the huge flow is not logged, then the IDS may be overwhelmed, but the data transfer will not be disturbed. The main difference in the deployment of these two scenarios is the initial setup of the flows in the Cisco Nexus® switch.

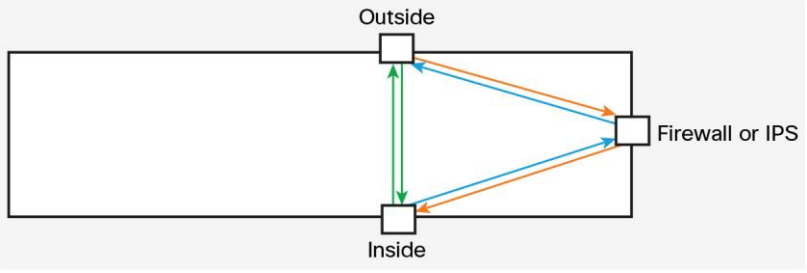
Figure 4 shows two initial flows: one flow with an input port of “Outside” and output ports of “Inside” and “IDS” (orange line), and a second flow with an input port of “Inside” and output ports of “Outside” and “IDS” (blue line). The actual implementation includes a pair or flows for both Ethernet and Address Resolution Protocol (ARP) packets.

Figure 4. Out-Of-Band Intrusion Detection System



If security is the utmost concern, however, the opposite approach can be taken by placing a firewall in-band and steering the huge data flows around it. In this case, not detecting a huge data flow may degrade the performance of the flow, but it will not compromise the security posture of the organization. Figure 5 shows the initial flows for the in-band setup. In this case, flows are created from the “Outside” port to the “Firewall or IPS” port; then they are created from the “Firewall or IPS” port to the “Inside” port. An opposite set of flows is created. The actual implementation includes a pair or flows for both Ethernet and ARP packets.

Figure 5. In-Band Firewall or Intrusion Prevention System



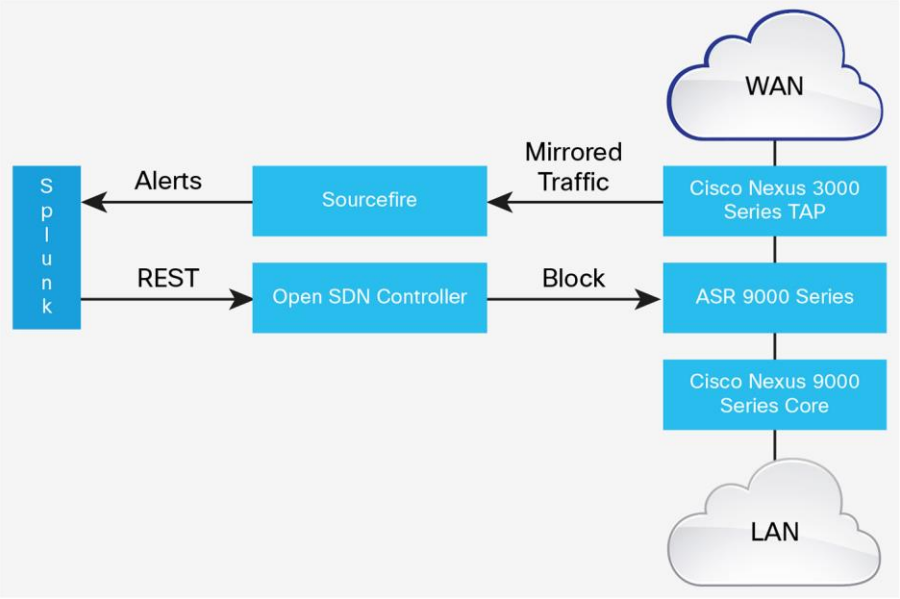
In both the in-band and out-of-band use cases, the initial setup is different, but the bypass remains the same. As depicted by the green lines in Figure 4 and Figure 5, bypass flows are created at a higher priority for the particular data transfer flows from the “Outside” port directly to the “Inside” port.

Active Blocking with Out-of-Band Intrusion Detection

Traditionally, IDSs simply monitor networks or systems for malicious activities or policy violations and generate alerts to a management station. They are generally connected through test access points (TAPs) or Cisco Switched Port Analyzer (SPAN) ports and, because they are not in the path, do not actively respond to those threats. However, being out-of-band of the traffic reduces any impact on performance. When augmented with event-based SDN, IDSs can achieve an improved mix of performance and security.

Figure 6 shows a system for collecting traffic through either TAPs or mirroring from the Cisco ASR 9000 Series Aggregation Services Router at the border. The mirroring can be performed either statically or dynamically using event-based SDN, discussed later in this document. The mirrored traffic is sent to a Cisco Sourcefire® sensor, which then logs security events through the FireSIGHT Management Center to Splunk. Splunk can then generate an action determined by such factors as rate of alerts and reputation of the attacking host learned from other events or data sources.

Figure 6. Event-Based SDN for Out-of-Band IDS with Active Blocking



This action is a Python script that makes a REST call to the Open SDN Controller to add a null route for the source address of the attacker. This null route is sent to the ASR 9000 Series border router through the controller’s southbound NETCONF interface. Because it has several southbound protocol options, the controller could also block the address using other southbound APIs such as OpenFlow or BGP flow specification if more detailed blocking were required (Layer 3 and Layer 4 information).

Steering a Huge Data Flow around Out-of-Band IDS Using Globus Input

Dynamic detection of huge data flows is difficult to achieve because the data is not in a recognizable form and communicates over any number of ports. Therefore, a wide range of variables must be used, for ports, time, or source addresses, to classify the flow, making the border needlessly porous and insecure. Furthermore, any delay

in detection of the large data flow can either overwhelm the IDS, blinding it to other threats, or adversely affect the beginning of the data flow, causing it to achieve significantly lower performance due to the characteristics of TCP over WANs.

Globus provides a unique mechanism for detecting and steering data flows as they are started. To transfer a file using GridFTP, a user must first authenticate with Globus. Globus can then set up the transfer, which occurs as a separate flow. Globus can log the information about this flow to Splunk, which can then call an action to send a command through REST to the Open SDN Controller to bypass an IDS or firewall. Because this flow is essentially authenticated, it provides an extra level of security while creating an undisturbed path for the data flow. Figure 7 illustrates the process of traffic steering when using Globus.

Figure 7. Steering Huge Data Flows around IDS and Firewall Using Globus Data Flow Notification and Input

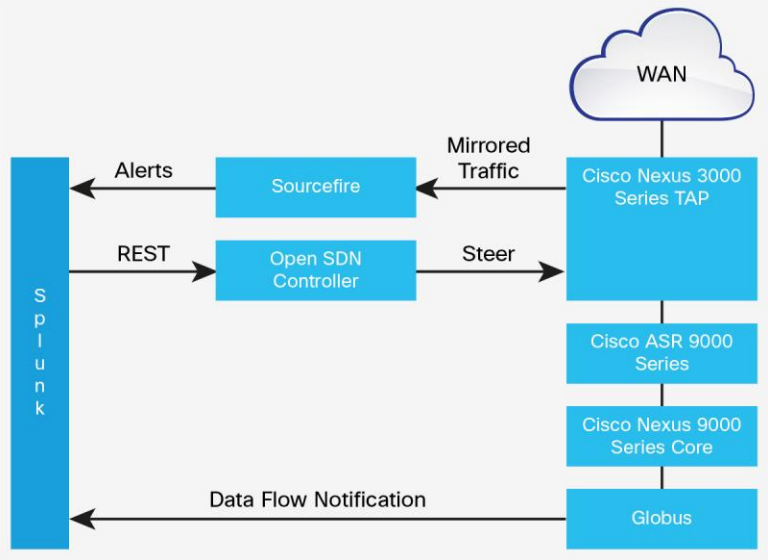
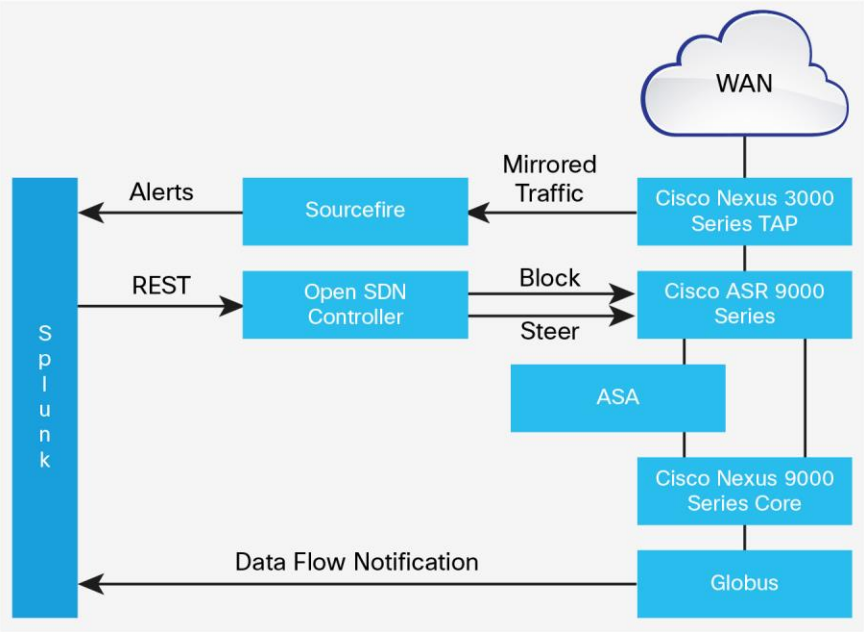


Figure 8 shows a security-first scenario in which all traffic is sent through a firewall by default. When huge flows are initiated by Globus, the flows are sent around the firewall, bypassing any performance degradation that it might impart. If the system fails to detect a huge data flow, the flow will not bypass the firewall, potentially affecting the performance of the transfer, but not affecting the security posture of the organization.

Figure 8. Steering Huge Data Flow around Firewall Using Data Flow Notification from Globus



Conclusion

The Science DMZ is a necessary architecture for moving large flows of scientific data around the world. However, it has been difficult to implement securely with traditional network protocols and hardware. SDN is often considered to offer a way to improve the security and performance of the Science DMZ. Although this solution uses SDN in the form of the Cisco Open SDN Controller, REST APIs, and protocols such as OpenFlow and NETCONF, SDN is employed only in those areas for which it provides value. For everything else, the Science DMZ solution uses proven products and technologies such as Splunk, BGP, traditional border routing designs, and platforms such as Cisco Nexus switches, ASR routers, and ASA firewalls.

Using just a small amount of Python code, the solution provides the two main functions needed to improve speed and security: blocking and steering. Splunk provides the means for collecting event data from nearly any source, correlating those events, and using a well-defined interface (REST API) with the Open SDN Controller. The controller then initiates the appropriate action on the physical network devices. The model is flexible and scalable, with well-defined interfaces between components.

Although traffic blocking and steering actions are required for the Science DMZ, they also have broad application across a number of other challenges facing modern networks. The use of an event-based approach with scalable, well-defined components should allow the solution to be adapted to many other needs.

For More Information

Send email to ask-opensdn@cisco.com or go to:

- http://www.cisco.com/go/opensdn
- developer.cisco.com/site/openSDN

Americas Headquarters
Cisco Systems, Inc.
San Jose, CA

Asia Pacific Headquarters
Cisco Systems (USA) Pte. Ltd.
Singapore

Europe Headquarters
Cisco Systems International BV Amsterdam,
The Netherlands

Cisco has more than 200 offices worldwide. Addresses, phone numbers, and fax numbers are listed on the Cisco Website at www.cisco.com/go/offices.

Printed in USA C11-735868-00 10/15