Implementing Zero Trust Cloud Networks with Transport Access Control and First Packet Authentication

Casimer DeCusatis, Piradon Liengtiraphan
School of Computer Science and Mathematics
Marist College/NY State Cloud Computing Center
Poughkeepsie, NY USA
casimer.decusatis@marist.edu

Anthony Sager, Mark Pinelli
Research and Development Lab
BlackRidge Technologies
Reno, NV USA
tsager@blackridge.us

***Abstract*—Cyberinfrastructure is undergoing a radical transformation as traditional enterprise and telecommunication data centers are replaced by cloud computing environments hosting dynamic, mobile workloads. Traditional data center security best practices involving network segmentation are not well suited to these new environments. We discuss a novel network architecture which enables an explicit zero trust approach, based on a steganographic overlay which embeds authentication tokens in the TCP packet request, and first-packet authentication. Experimental demonstration of this approach is provided in both an enterprise-class server and cloud computing data center environment.**

Keywords-cybersecurity, token, cloud, transport, authentication

I. INTRODUCTION

In recent years, network-based cybersecurity attacks have increased in both frequency and severity, far outstripping traditional defense methods [1-3]. For example, a moderately-sized commercial data center network can experience over 100,000 security events per day [2]. These attacks may be launched by a variety of hostile actors ranging from individual hacktivists and cyber-gangs motivated by creating social disruption to large, well organized groups with political or financial motivations who are backed by nation-states. Increasingly, these attacks have multiple goals, including compromising critical network resources such as the domain name server (DNS) or a software defined network (SDN) controller.

In response to the growing number and sophistication of cybersecurity threats, a United States Presidential Executive Order on Cybersecurity was issued in February 2013 [4]. This order outlined a clear and present danger from cyberattacks and made Cyber Defense a national priority for organizations such as the Department of Homeland Security and the National Science Foundation. In particular, this Executive Order included a call to action which tasked the National Institute of Standards and Technology (NIST) with creating a set of voluntary policies and guidelines to help develop the U.S. cybersecurity framework. In response to this request, numerous federal agencies and industry representatives from the finance, utility, and telecommunication sectors began to develop a fundamentally different approach to cybersecurity, taking into account changing environmental trends such as pervasive mobility and big data analytics. The resulting report to NIST proposed the so-called "zero trust model" for information security [5]. While this remains a theoretical abstraction, and many of the elements described in this framework are not commercially available at this time, significant progress has been made in recent years towards the development of enabling technologies to support zero trust architectures. The importance of cyberinfrastructure has since been reinforced by additional Executive Orders in this area [6].

Zero trust is intended to provide a scalable security infrastructure which can be applied across many different types of organizations. A fundamental principle of zero trust involves guaranteeing secure access to all resources, regardless of location, and assuming all network traffic is a threat until it is authorized, inspected, and secured [5]. This is not merely an extension of security principles such as deny by default, least privileges, or role-based access control [7, 8]. Rather, it redefines the approach to resource segmentation, a fundamental principle in which resources to be protected are grouped together and securely isolated or partitioned to limit unauthorized access.

Traditional security models are based on a perimeter security model (also known as an implicit trust model or "trust but verify" approach), in which all communication is trusted between devices within a specified security group. This model is based on the assumption that the network is segmented and the data center architecture can create a boundary or demilitarized zone (DMZ) between trusted and untrusted portions of the network. This relatively static approach to security, based on physical or virtual perimeters, breaks down in modern cloud computing and mobile device environments, where dynamic features render the concept of a traditional DMZ obsolete. The cloud has become the new network edge, and it cannot be adequately defended using an implicit trust approach [9].

While network segmentation using VLANs and similar techniques remains a long standing security practice, it is

DOI 10.1109/SmartCloud.2016.22

generally recognized that such techniques alone do not provide sufficient network security [5,7,8]. Many organizations attempt to segment their networks in a coarse granularity fashion to reduce risk, subject to limitations imposed by legacy hardware, complex virtualization software, and a lack of programmable, portable OSS/BSS resources [9]. However, the widespread use of cloud computing and mobile platforms has caused the network edge to blur and dissolve. For example, when an organization implements hybrid cloud and stores critical data in a combination of on-site and cloud storage, the concept of a network edge loses its meaning).

By contrast, a zero trust network security architecture incorporates a dynamic, automated security policy which extends across conventional security boundaries but still provides fine granularity segmentation and isolation of critical resources. Such an approach is preferably based on an explicit trust model (also known as a "trust nothing, verify everything" approach). In other words, all traffic needs to be validated, even between virtual machines (VMs) sharing a common physical host. Explicit security is part of a layered, defense-in-depth approach, which avoids kill chains and thus prevents single points of failure from compromising the entire security defense system. Fine grain segmentation improves management visibility and makes it feasible to disrupt network attacks as early as possible in the attack process, preferably to prevent data reconnaissance techniques from even identifying the resources which are being protected.

We note that this approach is compatible with recently proposed "micro-segmentation" and "micro-service" approaches, and that traditional network segmentation approaches break down as we introduce these approaches combined with automated network service chaining [10]. Ideally, micro-segmentation of a zero trust network would include authentication of not just users or applications, but would extend down to the level of authenticating individual packets. As noted in the NIST report [5], conventional networks assert the identity of a user or application based on a series of attributes such as network addresses, which may be forged. Such networks may decide to trust a user or application based on some criteria, but the concept of trust does not apply to conventional network packets, which are the fundamental building blocks of any network. It is desirable to implement a form of authentication with packet level granularity, which should offer several advantages. A finely grained zero trust approach improves network analysis and visibility, especially when combined with exhaustive logging and analysis of management plane data. Other potential benefits include simpler, vendor agnostic architectures, better scalability, and improved application portability.

Further, network segmentation can only be realized if we can avoid unauthorized awareness (a request for access to the network should not only be denied, it should avoid providing the requestor with any information about the nature of resources which are connected to the network). For example, modern data center networks are subjected to a constant stream of access requests at the network edge, since even a denied TCP connection request will return some information about the nature of the network, assisting attackers in fingerprinting the target system [8]. This weakness of the TCP/IP protocol stack means that potential attackers can gather information about a potential target by repeatedly trying to complete a connection request, even if the request attempt fails. The information collected in this manner can be used to plan future attacks or identify weaknesses in the perimeter defenses. It is desirable to prevent error message reconnaissance information from reaching a potential attacker without compromising performance of the remaining system.

There are several disruptive technologies which align with zero trust network architectures; we will focus on network centric approaches for the remainder of this paper. For example, zero trust networks can benefit from the centralized management plane and dynamic configurability offered by software defined networks (SDN). While the basic principles of SDN networks are well defined [11], we highlight several useful features. Programmable SDN controllers are able to implement dynamic network segmentation based on data collected from sources outside the network itself, such as honeypots, security analytic engines, and other sources. The application of security analytics to monitoring or management data sets enables the creation of actionable threat intelligence, allowing an SDN network to proactively discourage security threats and respond in near real time when new threats become apparent. This approach is particularly effective when combined with virtualized network functions (VNFs) such as virtual routers, firewalls, or other appliances. Recently SDN has begun to disrupt MAN/WAN networks, as noted in recent enterprise-class service deployments [12] and proposals from major telecom carriers including AT&T's Domain 2.0 network [13]. Since a global deployment of SDN at the scales proposed in this work implies large numbers of local and regional SDN controllers, each of which resides in its own VM, it would be beneficial to support zero trust on a computer platform capable of scaling to hundreds or thousands of VMs.

In this paper, we describe an approach which enables zero trust networks by providing first-packet based authentication, and demonstrate the use of this approach in defending an SDN controller from cyberattacks. We describe a steganographic overlay approach which embeds network authentication tokens in a TCP connection request, and blocks unauthorized traffic from completing a request. Resources protected in this manner are effectively concealed from reconnaissance attempts by attackers. We then demonstrate a commercially viable approach to transport layer identity management and authentication. Further, we show that this approach prevents fingerprinting of key resources such as the SDN controller by blocking any

response to unauthorized packets at the transport layer and below. We experimentally demonstrate the application of this approach in both large, enterprise-class servers and a cloud computing test bed.

This paper is organized as follows. Following the introduction, we describe the operation of a transport layer identity management scheme in section 2. We then present experimental results for the enterprise server and cloud test bed use cases in section 3. Finally, section 4 presents our conclusions and recommendations for future work.

II. TRANSPORT ACCESS CONTROL ARCHITECTURE

Our approach is based on a combination of two technologies, namely transport access control (TAC) and first packet authentication. To our knowledge, these approaches have not previously been combined into a single unified defense. In the proposed explicit trust model, each network session is independently authenticated at the transport layer before any access to the network or protected servers is granted. Unauthorized traffic is simply rejected from the network, and there is no feedback to a potential attacker attempting to fingerprint the system. Explicit trust is established by generating a network identity token during session setup. The network token is a 32 bit, cryptographically secure, single use object which expires after four seconds. Tokens are associated with identities from existing Identity Access Management (IAM) systems and credentials, such as Microsoft Active Directory or the IAM system used by Amazon Web Services [14]. Explicit trust is established by authenticating these identity tokens on the first packet of a TCP connection and applying security policy, before sessions with cloud or network resources are established (see Figure 1).

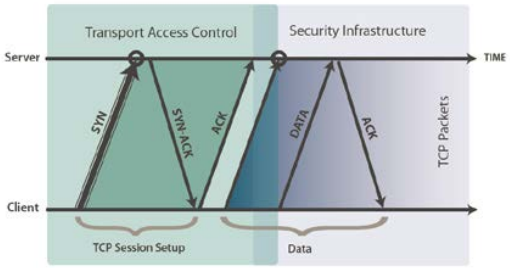

Figure 1 – Transport Layer TAC approach

Tokens are generated for each unique entity requesting access to a network resource; these entities are generally a user or device. An in-line virtual security gateway is then implemented between the equipment being protected and the rest of the network. The approach is illustrated in Figure 2, which shows two protected resources within a Corporate Network, namely the Accounting an HR servers. A third party user at IP address 192.168.7.10 is authorized to access only the Accounting server. A TAC gateway appliance is connected in the path between this user and the remaining network, and a second gateway is positioned before the protected resources. The first gateway inserts an identity token in the first packet of the TCP connection request. The second gateway enforces the network access policy by extracting the token, resolving the token to an identity, and determining the identity's authorizations. If the user is attempting a connection request to the Accounting server, the gateway grants access and allows the connection request to complete normally. However, if the user is attempting a connection request to the HR server, the request is denied and discarded. The user receives no feedback from the system when a connection request fails, blocking both network discovery and fingerprinting attempts. The attempted access is logged in an external Syslog server, which allocates enough memory to avoid wrapping and over-writing log entries. Existing tools such as SIAM can be used to analyze the logs or generate alerts of suspicious activity. A sample alert message is shown at the bottom of Figure 2. We note that continuous logging of all access attempts is consistent with the approach of a zero trust network (i.e. not allowing any access attempts to go unmonitored).

When the second gateway receives a connection request, it extracts and authenticates the inserted identity token and then applies a security policy (such as forward, redirect, or discard) to the connection request based on the received identity. This gateway acts as a policy enforcement point transparent to the rest of the system architecture and backwards compatible with existing network technologies. If the network access token for a TCP request fails to resolve to an identity or resolves to an identity that lacks the authority to access the requested resource, then the connection request is rejected without providing any further response to the requestor. In this way, the requestor receives no information about what sort of devices might be attached behind the gateway, effectively cloaking the presence of a protected scientific instrument or data repository. Both the identity insertion gateway and identity authentication gateway appliances can be implemented as VNFs hosted on a virtual server or router, intelligent optical transport device, or computerized research equipment controller.

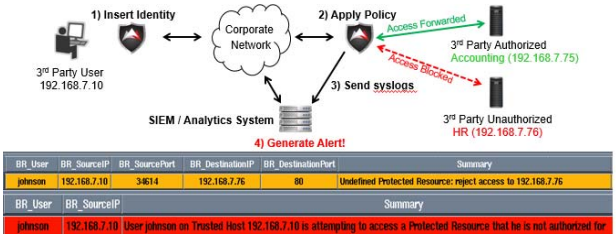

Figure 2 – Block diagram of TAC authentication scheme

This approach has several advantages, including separation of security policy from the network design (addresses and topologies) [7]. This approach works for any network topology or addressing scheme, including IPv4, IPv6, and networks which use the Network Address

Translation (NAT) protocol and is compatible with dynamic addressing often used with mobile devices. This approach extracts, authenticates, and applies policy to the connection requests, not only protecting against unauthorized external reconnaissance of the network devices but also stopping any malware within the protected devices from calling home(exfiltration). Security policies can be easily applied at the earliest possible time to conceal network attached devices from unauthorized awareness. By preventing unauthorized awareness and access, transport access control blocks both known and unknown attack vectors. This approach is low latency and high bandwidth since packet content is not inspected. Since the network tokens are embedded in the TCP session request, they do not consume otherwise useful data bandwidth. The combination of transport access control and a segmented, multi-tenant network implements a layered defense against cybersecurity threats, and contributes to non-repudiation of archival data. These techniques are also well suited to protecting public and hybrid cloud resources, or valuable, high performance cloud resources such as enterprise-class mainframe computers. Further, this approach can be applied to protecting the centralized SDN network controller from unauthorized access, and enable only authorized SDN controllers to manage and configure the underlying network. TAC uses an innovative identity token cache to provide high scalability and low, deterministic latency. The token cache is tolerant of packet loss and enables TAC deployments in low bandwidth and high packet loss environments.

III. EXPERIMENTAL RESULTS

A. Cloud Test Bed Use Case

The cybersecurity cloud test bed is illustrated in Figure 3. A protected resource (in this case, the SDN controller) is intended to be accessible only from a trusted client (in this case, one of the two Trusted SDN Admins). A BlackRidge hardware appliance gateway which implements TAC with first packet authentication [15] is placed in-line with the trusted clients, where it inserts tokens in the transport frame headers. Tokenized packets flow through the Corporate Network, which eventually routes them to the SDN controller. A virtual appliance is placed between the Corporate Network and the SDN controller, which will authenticate the tokenized packets and only allow authenticated and authorized packets to pass through to the SDN controller. Any packets without Tokens or identified traffic without the authority to access the SDN controller will be dropped. Our test configuration uses a hardware appliance to insert Tokens and a Virtual Appliance running on VMWare Esxi to authenticate Tokens. The hardware and software appliances are only addressable through their management ports and use the management ports to access the required network time protocol (NTP) Servers. A list of trusted devices to be allowed access is provisioned in the TAC gateways, and the list of trusted devices can be edited using the gateway management ports.

The gateway has three modes of operation, known as Bridge, Enforce, and Monitor. In Bridge mode the gateway does not perform authentication or insert tokens into the data packets; rather, it simply functions as a two port, Layer 2 bridge device. Enforce mode will perform authentication and insert tokens into the 32 bit sequence and acknowledgement number fields of a TCP frame, according to the established address list policy. Monitor mode has the same functionality as Enforce mode, with the exception that it does not enforce the security policy. Monitor mode is useful to validate configurations during installation and setup for a new gateway. By toggling a configured gateway between Bridge and Enforce modes, it's possible to observe the effects of turning token-based authentication off and on. The gateway architecture is a "bump in the wire" approach, and the gateway device is only addressable through its management port.

The gateway can also be used in a Layer 3 operating mode, which performs NAT for selected ports on the gateway. This is useful in cloud computing environments, allowing the gateway to present a public IP address on its client facing, untrusted port and a private IP address on its trusted port. In this case, the insertion and authentication of tokens is performed before NAT. In a public cloud deployment (such as BlackRidge Technology recently demonstrated within Amazon Web Services), the cloud service provider infrastructure resides on the right hand side of figure 3, with the public Internet acting as the untrusted network and the cloud user on the left side. Public IP addresses are used on ingress ports facing the untrusted network, and the cloud service provider's protected resource connect to a trusted egress port.

We configured the test bed as shown in Figure 3, and toggled the gateway between Bridge and Enforce modes. This allowed us to verify that Enforce mode would only permit tokenized packets from one of the trusted clients to reach the SDN controller. We then attempted a reconnaissance scan of the Corporate Network from an untrusted client. These scans were conducted using several industry standard tools, including Metasploit, HTTPrint, Firewalk, and PuTTY [7, 8]. When the TAC gateways were in Bridge mode, we were able to successfully fingerprint the SDN controller, as shown in Figure 4. We can easily determine that the controller is running OpenFlow protocols, as well as the specific version of OpenFlow. We then repeated the scans with the gateway in Enforce mode. As shown in Figure 5, we are now unable to identify the presence of an SDN controller on the management network. The TAC gateway blocks all potential responses at the transport layer and below. We also cannot determine if there is a TAC gateway present on the Corporate Network as TAC was also used to protect the management port of the gateway. This implements both packet level authentication

and unauthorized awareness, both desirable properties in a zero trust architecture.

To evaluate the effectiveness of the TAC gateway in defending against denial of service (DoS) attacks, we launched a DoS attack at the network protocol layer from the untrusted client in Figure 3. Common DoS simulation tools require knowledge of the target IP address, but as previously demonstrated the TAC gateway effectively cloaks the IP address for our SDN controller. For test purposes, we assume that an attacker has somehow obtained the SDN controller IP address through outside channels (perhaps a spear phishing attack on the network administrator) and we proceed to launch a DoS attack against the controller. Using a standard tool such as Low Orbit Ion Cannon (LOIC), we launched an attack against the IP address of the gateway data port, management port, and SDN controller. All packets were blocked by the TAC gateway without providing any additional intelligence to the attacker.

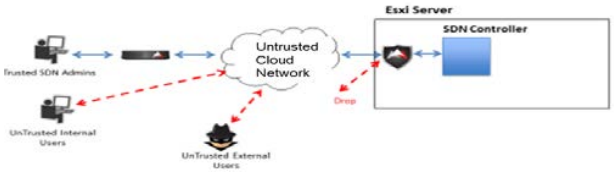

Figure 3 – Cloud test bed use case

```
8080/tcp open  http    Apache Tomcat/Coyote JSP engine 1.1
| http-methods: GET HEAD POST PUT DELETE TRACE OPTIONS
| Potentially risky methods: PUT DELETE TRACE
|_See http://nmap.org/nsedoc/scripts/http-methods.html
|_http-server-header: Apache-Coyote/1.1
|_http-title: OpenDaylight - Login
8181/tcp open  http    Jetty 8.1.14.v20131031
|_http-methods: No Allow or Public header in OPTIONS response (status code 404)
|_http-server-header: Jetty(8.1.14.v20131031)
|_http-title: Error 404 Not Found
Device type: general purpose
Running: Linux 3.X
OS CPE: cpe:/o:linux:linux_kernel:3
OS details: Linux 3.11 - 3.14
```

Figure 4 – Fingerprinting the SDN controller

631/udp open|filtered ipp
1013/udp open|filtered unknown

Figure 5 – Cloaking the SDN controller

B. *Enterprise-class cloud server use case*

The virtual gateways were tested in a highly virtualized server environment using IBM Z Systems enterprise servers (mainframes) as shown in Figure 6. These servers are commonly used in public, private, and hybrid cloud environments for Fortune 500 applications (particularly in the financial markets) as well as within cloud service providers such as SoftLayer. An IBM model z13 enterprise server was provisioned into 2 partitions running the z/OS operating system, and 2 partitions running zLinux. For each operating system, one partition served as the protected resource while the other served as the trusted host. All four partitions share common physical network interfaces, provided by an Open System Adapter (OSA) card. The virtual appliances were hosted in two additional logical partitions (LPARs), interconnected with the protected resources, trusted hosts, and OSA's as shown in Figure 4. Additional OSA cards were provisioned to serve as interfaces for the network management ports on the virtual appliances.

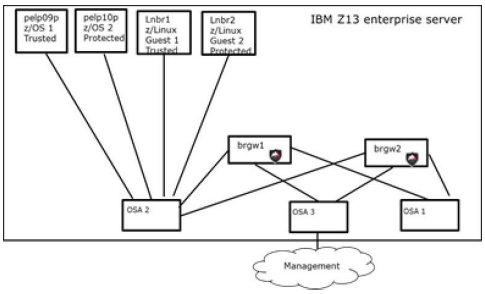

Figure 6 – Enterprise server use case

Three use cases were tested using this configuration. First, the gateways were configured to allow connectivity only between Linux partitions 1 and 2. We confirmed normal operation of resource connectivity including SSH, SCP, iperf, sftp, and wget functions, and verified that untrusted hosts such as zOS-1 could not access the Linux partitions. Second, the gateways were configured to allow connectivity between Linux-1 and zOS-2 partitions (note that the gateway authentication is independent of the operating system running in either the supplicant or the trusted resource). As before, we verified basic functionality (including multiple sftp file transfers between the trusted host and protected resource) and confirmed that other partitions, such as zOS-1, could not access the protected \resource in this configuration. Third, the gateway was configured to allow connectivity between the two zOS partitions. As in the previous use cases, we verified basic functionality (including multiple sftp file transfers between the protected resource and trusted hosts) and confirmed that the Linux-1 partition was unable to access protected resources in this configuration. These three test cases established that the gateways could be configured to enable or disable applications running between any two partitions on the same physical server, even if the gateway itself is hosted in a partition on the same physical server. This approach directly supports the zero trust architecture we intended to implement.

The gateway functionality was further demonstrated in a pre-production test environment at Marist College, part of the New York State Cloud Computing and Analytics Center (CCAC), as shown in Figure 7. In this case, the network spans multiple buildings on the campus MAN (about 1 km apart). In the first building, ten sysadmin terminals were

interconnected to a Layer 2 switch, and one switch port was connected to the insertion gateway. In this configuration, all terminals connected to the trusted switch port receive authentication tokens and will be allowed to access the protected resource. In a second building, the second gateway was configured in Layer 3 mode, which was then attached to the protected resource (a sysadmin application(Syswiki) running in a SUSE Linux guest VM on an IBM z144 enterprise server (mainframe).

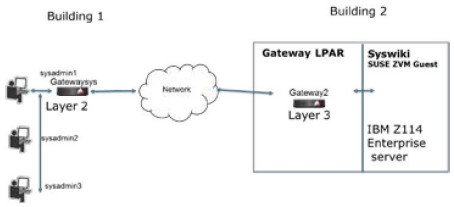

Figure 7 – Proof of concept campus test bed

Using this configuration, we verified that the gateway allowed the Sysadmin trusted host terminals to perform operations including SSH, scp, and http post/get functions to the protected resource. We also verified that other terminals (untrusted hosts) on the same network were unable to access the protected resource when the gateways were set to Enforce mode, but could access the protected resource when the gateway was put into Bridge mode. The Sysadmins showed no measurable degradation in response time or performance of the protected resource application with and without the gateway operating in enforce mode.

Further, there was no measurable performance impact when accessing other resources on the campus network or accessing Internet resources when the gateway was in Enforce or Bridge mode. We also conducted a port scan of the gateway using Nmap/Zenmap tools from an untrusted terminal, and were unable to identify any open ports or fingerprint the Syswiki when the gateways were in Enforce mode.

IV. CONCLUSIONS

The growing cybersecurity threat requires an architectural redesign of the data center network, based on the principles of an explicit zero trust network. We have demonstrated several principles of zero trust using a transport access control system, based on a steganographic overlay which embeds authentication tokens in the TCP packet request and first-packet authentication. This system can provide enhanced security in both enterprise computing and cloud environments as part of a defense-in-depth strategy and prevents unwanted fingerprinting of protected resources. Future research will continue penetration testing to identify and mitigate any additional vulnerabilities of this scheme.

ACKNOWLEDGMENT

The authors gratefully acknowledge the support of the National Science Foundation (NSF) grant 1541384, Campus Cyberinfrastructure – Data, Networking and Innovation Program (CC-DNI), per NSF solicitation 15-534, for the project entitled CC-DNI (Integration (Area 4): Application Aware Software-Defined Networks for Secure Cloud Services (SecureCloud). The authors also gratefully acknowledge the support of Marist College and the New York State Cloud Computing and Analytic Center (CCAC).

REFERENCES

[1] "Cisco 2014 annual security report", published by Cisco System Inc., https://www.cisco.com/web/offer/gist_ty2_asset/Cisco_2014_ASR.pdf . (Last accessed February 9, 2015)

[2] "Cisco 2015 annual security report", published by Cisco System Inc., https://www.cisco.com/web/offer/gist_ty2_asset/Cisco_2014_ASR.pdf . (last accessed February 9, 2015)

[3] "IBM X-Force trend and risk report", published by IBM Corporation, October 2013 http://www-03.ibm.com/security/xforce/downloads.html (last accessed December 18, 2015)

[4] U.S. Presidential Executive Order, "Improving critical infrastructure cybersecurity", (February 12, 2013), http://www.whitehouse.gov/the-press-office/2013/02/12/executive-order-improving-critical-infrastructure-cybersecurity (last accessed February 25, 2015)

[5] NIST report, "Developing a framework to improve critical infrastructure cybersecurity", submitted by Forrester Group, 18 p. (April 2013) http://csrc.nist.gov/cyberframework/rfi_comments/040813_forrester_research.pdf (last accessed January 5, 2015).

[6] U.S. Presidential Executive Order, "Promoting private sector cybersecurity information sharing", (February 13, 2015), http://www.whitehouse.gov/the-press-office/2015/02/13/executive-order-promoting-private-sector-cybersecurity-information-shari (last accessed February 25, 2015)

[7] R. Smith, Elementary Information Security, 2nd edition, Jones and Bartlett, Burlington, MA (2016)

[8] S. Oriyano, Hacker Techniques, Tools, and Incident Handling, 2nd edition, Jones and Bartlett, Burlington, MA (2014)

[9] J. Stewart, Cisco blog, October 2015 http://blogs.cisco.com/security/cybersecurity-what-needs-to-change-now (last accessed May 19, 2016)

[10] S. Cherukuri, "NFV architecture and orchestration for cloud based virtual managed services", Cisco Live 2015 paper BRKSDN-2065 https://www.ciscolive.com/online/connect/sessionDetail.ww?SESSION_ID=83663&tclass=popup (last accessed May 20, 2016)

[11] M. Casado, N. Foster, A. Guha. "Abstractions for Software-Defined Networks". Vol. 57, pp 86-95. Communications of the ACM. 2014

[12] J. Kindervag and R. Harrison, "Orchestrate a zero trust network", https://www.brighttalk.com/webcast/9591/186577?utm_campaign=communication_missed_you&utm_medium=email&utm_source=brighttalk-transact&utm_content=webcast (last accessed April 28, 2016)

[13] AT&T Domain 2.0 Vision White Paper, November 2013 https://www.att.com/Common/about_us/pdf/AT&T%20Domain%202.0%20Vision%20White%20Paper.pdf (last accessed May 20, 2016)

[14] Amazon Web Services Identity and Access Management, April 2016 https://aws.amazon.com/iam/ (last accessed May 20, 2016)

BlackRidge white paper, "Dynamic network segmentation", August 2012 http://www.blackridge.us/images/site/page-content/BlackRidge_Dynamic_Network_Segmentation.pdf (last accessed April 27, 2016)