Building Security Perimeters to Protect Network Systems Against Cyberthreats

By Deepak Puthal, Saraju P. Mohanty, Priyadarsi Nanda, and Uma Choppali

Due to the wide variety of devices used in computer network systems, cybersecurity plays a major role in securing and improving the performance of the network or system. Although cybersecurity has received a large amount of global interest in recent years, it remains an open research space. Current security solutions in network-based cyberspace provide an open door to attackers by communicating first before authentication, thereby leaving a black hole for an attacker to enter the system before authentication. This article provides an overview of cyberthreats, traditional security solutions, and the advanced security model to overcome current security drawbacks.

BACKGROUND

Information and communication technology is the key in realizing smart cities [1]. The underlying technologies, including wireless sensor networks, the Internet of Things (IoT), and cyber-physical systems, facilitate the design and operations of such smart cities [1]–[3]. The bottom line for all of these technologies is the connectivity of wired or wireless networks to the Internet. In the IoT and smart cities, hierarchical information technology (IT) infrastructure connects sensors, the cloud, and the command and control center.

Transmission Control Protocol/Internet Protocol (TCP/IP) is still being used as one of the basic means for communication involving private and public networks (i.e., the Internet). There are several security methods implemented over TCP/IP protocol, such as IP security (IPsec). IPsec is a secured network protocol across IP-based networks whose main purpose is to authenticate and encrypt the data packets on an end-to-end basis, and IPsec protects data flows between the hosts or any network devices. IPsec supports network-level peer authentication, data integrity, and data confidentiality through encryption [4]. However, TCP/IP-based security solutions do not provide a strong foundation of security, as it allows devices to first communicate and then authenticate. In such situations, attackers get a chance to enter the data transmission process before authentication occurs. To overcome this situation, the Cloud Security Alliance (CSA) proposed a novel idea to authenticate first before communication happens, and this idea is called *software-defined perimeter* (*SDP*) [5]. Figure 1 shows the clear difference between traditional TCP/IP-based security and SDP.

> The client of the SDP architecture has an extensive range of functions, such as user-identity-to-device verification and routing local applications to remote ones.

According to the Gartner's Top 10 Strategic Technology Trends for 2017, adaptive security architecture is listed [6]. Security in cyberspace is challenging, and multilayered security, along with the use of user and entity behavior analytics, will become a requirement virtually for every enterprise in the future. Many security solutions work efficiently under the assumption that the devices and users are fully protected by traditional perimeter defense mechanisms [7]–[9]. However, several applications, the perimeter-based protections are not feasible, since network devices (i.e., sensors) are positioned in unattended environments [10]. Hence, a new security-designing approach is appropriate to avoid such circumstances. Recently, industries have advocated three promising proposals: 1) zero trust, 2) deperimeterization, and 3) SDP [10], [11].

Zero trust always follows the principle "never trust, always verify" to architecture the framework [12]. It allows no default trust for any entity (e.g., devices, applications, packets, and users) regardless of its type or whether it is on or related to the corporate network. Hence, zero trust is appropriate for securing devices and users. The term *deperimeterization* is defined as a hardened perimeter security strategy that is impossible to sustain within an agile business model [13]. By using encryption and dynamic-data-level authentication, deperimeterization secures user data on many levels. This multilevel approach fits to the most

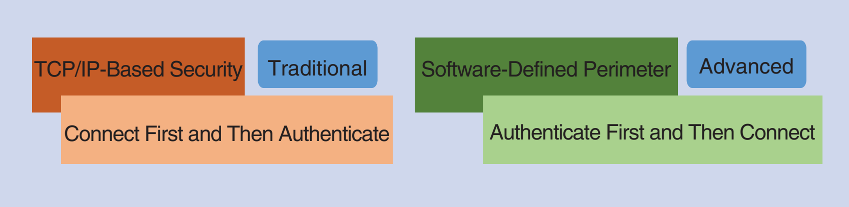
TCP/IP-Based Security
Traditional
Connect First and Then Authenticate
Software-Defined Perimeter
Advanced
Authenticate First and Then Connect

FIGURE 1. The difference between traditional security and a software-defined perimeter.

Digital Object Identifier 10.1109/MCE.2017.2714744
Date of publication: 22 September 2017

advanced computing systems such as the IoT, cloud computing, and edge computing, as this article tests with a multilayered architecture. The CSA launched the SDP research initiative in December 2013 with the goal of limiting network-based cyberattacks against the application infrastructure [14]. Currently, SDP is gaining a large amount of interest from global researchers by combining the cloud, edges, network devices, and users.

SDP BASICS

Current IT infrastructures are more hybrid and diversified. At present, they are moving from hardware-based to software-based infrastructure. Now, IT technology is changing from a static environment to a dynamic environment, where users obtain multiple services simultaneously and according to the individual requirement. There is a shift from network-centric security solutions to a user or an identity-centric approach. This approach gives a better way of ensuring security from a user perspective and not from a network perspective.

According to most advanced CSA surveys, 68% of organizations are concerned about security for several reasons, such as protecting systems, infrastructure, and economic goals. In fact, many industries are concerned about security. Additionally, one survey found that 80% of the cloud infrastructure is hybrid, so users can access both the public and private cloud to perform their tasks. The same survey revealed that 65% of IT resources are offside.

SDP alleviates cyberthreats in network-based cyberspace by building a simple and dynamic security perimeter in any space in the cyberdata center. To provide a basic level of security, the SDP begins at zero availability and zero visibility. The SDP powerfully constructs systems to authorize applications only after the client has been authenticated. Organizations use SDP to ensure the visibility of applications on the Internet, e.g., for cross-organization coordinated efforts, and their immigration to infrastructure as a service and software as a service. Organizations use SDP to protect secure inward-business basic applications for nonrepresentative and bring-your-own-device access in addition to isolated critical applications.

Any kind of malicious requests or queries are going to be dropped at the SDP gateway without reaching the data center.

SDP ARCHITECTURE

The threat against application infrastructure increases with the adoption of current critical cyberinfrastructure. Since traditional security mechanisms cannot protect the service provider and edge data center, SDP creates a cryptographic perimeter from a source device to the edges and cloud data center. SDP provides a user-centric security solution by creating a perimeter to enclose the source and destination within the perimeter. This also dynamically adjusts according to the user requirement. Figure 2 provides a clear example of SDP and user access.

SDP is designed with three major elements [5]: 1) a security model to verify the device identity, or for the users, the roles for access before granting access into endangered systems; 2) a cryptography technique to guarantee that the security model is fool proof; and 3) a security solution to address the prior mentioned issues to demonstrate in broad daylight space security controls.

CSA published the SDP version 1 concept in April 2014. In this design, an initiating host transmits user and device identity to a controller over a mutual transport layer security (TLS) connection. The controller thus associates with an issuing certification authority to an identity provider to confirm the client's identity after checking the hardware identity. Once confirmed, the controller would then arrange mutual TLS connections between the initiating host and the appropriate accepting hosts after proper verification. The controller would then arrange at least one common TLS association between the initiating host and the proper accepting hosts. More importantly, the SDP could avoid all forms of network attacks, including distributed denial-of-service (DDoS) and man-in-the-middle scenarios.

FIGURE 2. An overview of the procedure to generate a simple and dynamic perimeter.

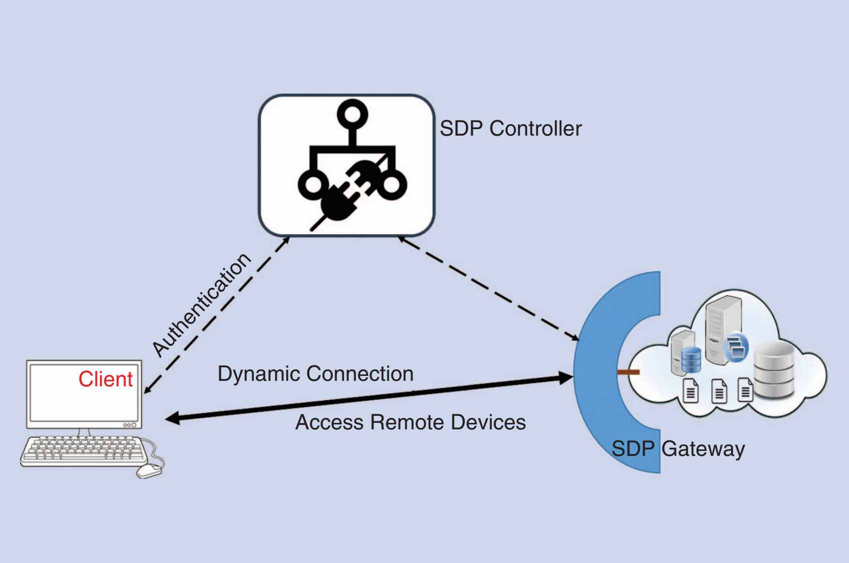

FIGURE 3. The authentication process used to draw a dynamic perimeter.

There are three major components that play a role in the SDP design, such as the client, the SDP controller, and a set of SDP gateways, as shown in Figure 3. The SDP controller initiates the hosts to become a client and a gateway. The client of the SDP architecture has an extensive range of functions, such as user-identity-to-device verification and routing local applications to remote ones. The client is formed progressively to guarantee the testament-based shared TLS virtual private network associates with services for which the client is approved. The SDP controller works as a trusted third party between the client and SDP gateway. This also provides services such as a certificate authority and identity provider to the client. The SDP controller continuously arranges both the client and SDP gateway as a mutual TLS association after verifying the client's authenticity. The SDP gateway is mainly deployed at the destination end, and the TLS connection from the client terminates at this point.

The SDP controller is online with the network to protect any applications. It will also be where a central authentication point and policy are stored, and it always connects to the specific policy model, such as a public key infrastructure and IM. In addition, when users connect to the network, the connection is authenticated by the SDP controller, followed by the controller to evaluate the policies to find whether the requested services are open to that specific user at that time.

> As SDP adopts a technique to authenticate first and then communicate, users never get the chance to assess properties of security without being authenticated first.

When the user starts communicating or accesses resources from data centers, a secure tunnel is established between the user and the SDP gateway. Then the SDP gateway evaluates the second level of policy in real time to determine the condition when the user is allowed to access these resources. There are three possible situations that arise from this real-time policy evaluation: 1) users are allowed to access the resources, 2) users are blocked and are not allowed to access the resources, and 3) users need more authentication steps to obtain access. All of these situations ensure that the SDP gateway performs the real-time access control needed to protect against unauthorized access. SDP works with consistent and meaningful policies.

The network traffic is encrypted from user devices to the gateway by creating a secure tunnel. In the process, a high level of security is enforced in the network traffic by maintaining data integrity and confidentiality. The SDP gateway is always dynamic in nature; it checks the user access as well as the data center resources and whether they are allowed or disallowed for user access. SDP protects the highly sensitive data by deciding which data should be accessible to which user based on user-access control policies.

CYBERTHREATS ANALYSIS OF SDP

In the network system's potential threats, there are three major possibilities: 1) server exploitations, 2) credential theft from users, and 3) attacks during communication [14]. Vidder's report describes the defeating attacks on network-based cyberspace, as shown in Figure 3. The major possible attacks associated with severe exploitations are DDoS, misconfiguration, and vulnerabilities, where attackers from the Internet try to compromise the server. The SDP gateway deployed at the data center always checks the identity and policies associated with that specific user. SDP isolates the data center and protects it by using single-packet authorization and dynamic firewall functionalities. Consequently, any kind of malicious requests or queries are going to be dropped at the SDP gateway without reaching the data center.

Similarly, examples of threats to obtain the user's or client's credentials are phishing and brute force attacks. In SDP architecture, a combination of mutual TLS and the client's fingerprint (the client's own secret key) is utilized as transparent multifactor authentication. Therefore, no one gets the client's credentials without receiving help from the SDP controller.

Finally, the most common attack space in a network system is the attack on communication, where man-in-the-middle, certificate forgery, and domain name system (DNS) spoofing attacks are some of the common cases [15]. The SDP architecture authenticates first and then connects by creating a secure tunnel between the client and SDP gateway before allowing communication, as described in the "SDP Architecture" section. The SDP controller provides IP addresses instead of a DNS server, so contaminating the DNS is not possible. Therefore, it is quite difficult for an attacker to break the security perimeter. Hence, it is evident that SDP provides a new level of security to protect networked systems and defends against cyberattacks.

CONCLUSIONS

SDP provides a simple and user-centric security solution instead of network or data-centric solutions. In an organization, not everyone can see all of the network resources; instead, he/she can only see the resources made available for him/her. Since network resources are not visible to outsiders, this acts as a significant benefit. As SDP adopts a technique to authenticate first and then communicate, users never get the chance to assess properties of security without being authenticated first. Therefore, attackers have very limited information for network-based attacks to be successful.

ABOUT THE AUTHORS

Deepak Puthal (Deepak.Puthal@uts.edu.au) earned his Ph.D. degree in computer and information systems from the University of Technology Sydney (UTS), Australia. He is a lecturer in the School of Computing and Communications at UTS. His research interests include cybersecurity, the Internet of Things, distributed computing, and wireless communications. He has published in several international conferences and journals, including IEEE and Association for Computing Machinery publications.

Saraju P. Mohanty (saraju.mohanty@unt.edu) is a professor in the Department of Computer Science and Engineering at the University of North Texas, Denton. He is an inventor of four U.S. patents and is an author of 220 peer-reviewed research articles and three books. He is currently the editor-in-chief of *IEEE Consumer Electronics Magazine*. He is on the editorial board of peer-reviewed journals including *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems* and *ACM Journal on Emerging Technologies in Computing Systems*. He has been the chair of the Technical Committee on Very Large Scale Integration, IEEE Computer Society, and overseen a dozen IEEE conferences. He serves on the steering, organizing, and program committees of several international conferences.

Priyadarsi Nanda (Priyadarsi.Nanda@uts.edu.au) earned his Ph.D. degree from the University of Technology Sydney, Australia. He is a senior lecturer in the School of Computing and Communications at University of Technology Sydney, Australia. He has more than 26 years of experience in the area of cybersecurity, Internet of Things security, networks quality of service, assisted health care using sensor networks, and wireless sensor networks. He has published more than 70 refereed research articles.

Uma Choppali (umachoppali@gmail.com) earned her M.S. degree from the Indian Institute of Technology Bombay, India, and her Ph.D. degree from the University of North Texas, Denton, in 2006. She is currently an adjunct faculty member at Brookhaven College, Dallas. She has authored a dozen peer-reviewed articles.

REFERENCES

[1] S. P. Mohanty, U. Choppali, and E. Kougianos, "Everything you wanted to know about smart cities," *IEEE Consum. Electron. Mag.*, vol. 6, no. 3, pp. 60–70, July 2016.

[2] M. L. Rajaram, E. Kougianos, S. P. Mohanty, and U. Choppali, "Wireless sensor network simulation frameworks: A tutorial review," *IEEE Consum. Electron. Mag.*, vol. 6, no. 2, pp. 63–69, Apr. 2016.

[3] S. Sharma, D. Puthal, S. Jena, A. Zomaya, and R. Ranjan, "Rendezvous based routing protocol for wireless sensor networks with mobile sink," *J. Supercomputing*, vol. 73, no. 3, pp. 1168–1188, 2017.

[4] S. Raza, S. Duquennoy, J. Höglund, U. Roedig, and T. Voigt, "Secure communication for the Internet of things—A comparison of link-layer security and IPsec for 6LoWPAN," *Security Commun. Networks*, vol. 7, no. 12, pp. 2654–2668, 2014.

[5] Cloud Security Alliance (CSA). [Online]. Available: https://cloudsecurityalliance.org/group/software-defined-perimeter

[6] (2017, May 15). Gartner's top 10 strategic technology trends for 2017. [Online]. Available: http://www.gartner.com/smarterwithgartner/gartners-top-10-technology-trends-2017

[7] D. Puthal, S. Nepal, R. Ranjan, and J. Chen, "DLSeF: A dynamic key-length-based efficient real-time security verification model for big data stream," *ACM Trans. Embedded Comput. Syst.*, vol. 16, no. 2, p. 51, 2017.

[8] D. Puthal, X. Wu, S. Nepal, R. Ranjan, and J. Chen, "SEEN: A selective encryption method to ensure confidentiality for big sensing data streams," *IEEE Trans. Big Data*, 2017. doi: 10.1109/TBDATA.2017.2702172.

[9] A. Nanda, P. Nanda, X. He, A. Jamdagni, and D. Puthal, "SecureGLOR: An adaptive secure routing protocol for dynamic wireless mesh network," in *Proc. 16th IEEE Int. Conf. Trust, Security and Privacy in Computing and Communications (IEEE TrustCom-17)*, 2017.

[10] D. Puthal, S. Nepal, R. Ranjan, and J. Chen, "Threats to networking cloud and edge datacenters in the Internet of things," *IEEE Cloud Comput.*, vol. 3, no. 3, pp. 64–71, 2016.

[11] E. Bertino, K.-K. R. Choo, D. Georgakopolous, and S. Nepal, "Internet of Things (IoT): Smart and secure service delivery," *ACM Trans. Internet Technol.*, vol. 16, no. 4, p. 22, 2016.

[12] National Institute of Standards and Technology, "Developing a framework to improve critical infrastructure cybersecurity," NIST RFI 130208119 -3119-01, submitted 4 Aug. 2013.

[13] Jericho Forum. (2017, May 15). De-perimeterization. [Online]. Available: https://collaboration.opengroup.org/jericho/presentations.htm

[14] Vidder, Software Defined Perimeter 2016. [Online]. Available: www.vidder.com/why-vidder/software-defined -perimeter.html

[15] D. Puthal, "Secure data collection and critical data transmission technique in mobile sink wireless sensor networks," M.S. thesis, National Institute of Technology, Rourkela, 2012.