

Layer 7 Application Firewalls

Introduction

The firewall, the first line of defense in many network security plans, has existed for decades. The purpose of the firewall is straightforward; permit authorized data to enter and exit the trusted network while preventing unauthorized traffic from doing so. Interestingly, the makeup and methodology of the firewall has not changed as rapidly as the systems and applications it intends to protect. Over the years, enhancements certainly have occurred within perimeter security; however, the overall methodology of the firewall has not radically changed.

Present-day threats in information security require a new firewall technology called application layer filtering, otherwise known deep packet inspection. The purpose of this paper is to describe why application firewalls are quickly becoming a requirement for all corporate computer networks.

Traditional Firewalls

Before detailing application firewall functions and capabilities, it is important to understand how traditional firewalls behave. Since their inception, firewalls have been driven by rules and policies codified by a communications engineer. Most firewalls in use today operate by the session (sender / recipient), and by the packet. Firewalls receive packets of data, which is comprised of header information, and the payload, which is the actual data intended for transmission. Traditional firewalls read information from the packet header that determines what the firewall ought to do with the packet (Ingham, 2002). The rules that determine what the packet filtering firewall is going to do- the “policies”- are largely static in nature.

Put into information technology terms, traditional firewalls operate at the data link, network, and transport layers of the Open System Interconnection (OSI) model (Proch, 2009). Accordingly, a traditional firewall can look into the header of a packet and determine basic information; such as where the packet came from, where it wants to go, and what port it intends to travel on. For example, a packet coming into a firewall might have a header that indicates that it originated from a specific IP number ‘A’, intends to deliver its payload to IP number ‘B’ and is traversing transmission port protocol (TCP) number twenty-three (Clarke, 2009). Based on this information, the firewall then decides either to allow or deny access. This relatively simple rule-based system has worked for a long time, but is quickly becoming only the starting point in gateway threat protection.

The New Threat

Today, “malware”, such as worms and bots, are entering networks on ports typically left open by most firewalls. Executable code can launch in a web browser over http port eighty, or via instant messenger using a remote procedure call (RPC). These ports provide services that cannot be turned off by most network administrators, thus leaving the network or security administrator in a precarious position.

Structured Query Language (SQL) injection attacks are a good example of this situation. These attacks to a database back end occur from a website front end. The website front end invariably has insecure code that could be exploited by a hacker. Traditional firewalls have no way to fend off this attack, as the entire purpose of the site is to receive web hypertext end user requests via port 80 and secure socket layer (SSL - encrypted) port 443. Older generation firewalls will see these attacks as normal permissible surfing.

A second example is the case of the Blaster Worm. This malicious code launched a distributed denial of service against Microsoft’s Windows Update Service. Blaster travelled over TCP port 135. This port is legitimately used for many Microsoft software applications (Greene, 2004). At the time of the attack, the prescribed remedy, shutting down the port, was not acceptable for many organizations. They needed 135 open.

Enter Application Firewalls

Application firewalls identify malicious traffic that traditional firewalls cannot. They accomplish this by performing “deep packet inspection” on the payload data contained in every packet. This data supplies a signature from which the firewall database determines to the application to which the data belongs. From this data, the firewall has an understanding of permissible actions this packet might take. The firewall can also determine if the packet is a “wolf in sheep’s clothing”- malicious code masquerading as typical data.

Applications firewalls offer a wide range of functions. In addition to the traditional analysis of layer two to layer four packet headers (traditional firewall rules), application firewalls should support all network protocol layers along with full packet payload analysis. These devices must be able to identify applications with static, dynamic, and negotiated protocol and port fields (Magalhaes, 2008). In other words, they must be capable of scanning the payload content to determine application type without reliance on headers that can be misleading. Application firewalls must keep a running, and constantly updated, list of application signatures logged for reference on the firewall

(Proch, 2009). Finally, the devices need to keep up with the network interface. This is a real-time active scanning solution. The process of inspecting packets cannot interfere with data transmission.

Comparison to Network Intrusion / Prevention Devices

Readers may be asking themselves, what can this device do that a network intrusion / prevention device (NIDS / NIPS) cannot? These devices scan data in much in the same manner as application firewalls. The difference lies in the simplicity of the application firewalls over other content-inspection solutions. NIDS / NIPS devices require complex configurations, ongoing maintenance, and regular tuning to stay effective. Smaller shops may not have the expertise, manpower, or both required to deploy and maintain this sort of solution. Left unmaintained, very expensive NIDS / NIPS devices are nearly worthless.

Conversely, application firewalls typically take immediate, automatic action once they identify a threat. Tim Green of Network World explains (2004), "Intrusion-detection systems already do this [inspect payload data for malware], but their response is to trigger alerts for network administrators to decide whether suspicious traffic means an attack is really underway. Deep packet inspection firewalls differ in that they automatically take steps to block the attacks they detect" (Green, 2004, p. 1). This automated function, in and of itself, most likely pays for itself in resource strapped IT departments.

Conclusion

Traditional firewalls have worked well for many years. They operate on basic rules that rely on packet header information. Historically, this information alone was enough to filter out those with a malicious intent. The device did not need to analyze payload data to make this determination.

Unfortunately, the internet threat environment has irreversibly changed. Identification of data type by packet header and TCP port is no longer reliable. As demonstrated in this article, malicious code can easily masquerade itself as benign web traffic, or other legitimate web-based services. Preventing malicious code from entering the trusted network requires active inspection of data. Application aware firewalls identify data type by application signature. Armed with this intelligence, the device is in a superior position to determine how data should behave. Data that does not conform to the parameters of the device is not permitted to enter the trusted network. Application firewalls will quickly overtake traditional firewalls as the mandatory method for preventing network intrusions.

References

Bradbury, D. (2009). Securing e-business with web application firewalls. *Computer Weekly* , 18.

Clarke, J. (2009). Resilience under attack: Techniques for continuing online business in the face of security compromise. *Journal of Business Continuity & Emergency Planning* , 3(3), 222-226. Retrieved October 17, 2009 from Business Source Complete.

Greene, T. (2004, February 2). *The evolution of application layer firewalls* . Retrieved October 17, 2009, from Network World:
<http://www.networkworld.com/news/2004/0202specialfocus.html>

Ingham, K. F. (2002). A History and Survey of Network Firewalls. *The University of New Mexico, Computer Science Department* , 1-42. Retrieved October 17, 2009 from Business Source Complete.

Magalhaes, R. (2008, June 25). *The Difference Between Application and Session Layer Firewalls*. Retrieved October 17, 2009, from WindowsSecurity.com:
<http://www.windowsecurity.com/articles/Difference-Between-Application-Session-Layer-Firewalls.html>

Proch, D. &. (2009). Plumb The Depths Of Deep Packet Inspection. *Electronic Design* , 57(16), 47-50. Retrieved October 17, 2009 from Business Source Complete.