Chapter 4
The Last Line of Defense: A Comparison of Windows and Linux Authentication and Authorization Features

Art Taylor
Rider University, USA

ABSTRACT

With the rise of the Internet, computer systems appear to be more vulnerable than ever from security attacks. Much attention has been focused on the role of the network in security attacks, but evidence suggests that the computer server and its operating system deserve closer examination since it is ultimately the operating system and its core defense mechanisms of authentication and authorization which are compromised in an attack. This chapter provides an exploratory and evaluative discussion of the authentication and authorization features of two widely used server operating systems: Windows and Linux.

THE LAST LINE OF DEFENSE: THE OPERATING SYSTEM

The number of computer security incidents reported from various forms of attacks has increased significantly since the introduction of the Internet (CERT; Yegneswaran, Barford, & Ullrich, 2003). Though it is clear that the introduction of the Internet coupled with the decreased cost of networking has helped to pave the way for attackers, the end result of most malicious attacks is the alteration of the host operating system. This alteration is often with the intent of propagating the malicious program and continuing the attack (virus, Trojan horse) or potentially damaging, stealing or altering some content on the host machine. While this type of attack may be aided by the network and security weaknesses therein, the attack could not be successful without ultimately compromising the host operating system. While much attention has focused on securing the network, since it is ultimately the operating system which is compromised, a closer examination of the defense mecha-
nisms of the operating system may be warranted (Losocco, Smalley, Mucklebauer, Taylor, Turner, & Farrell, 1998). Security weaknesses in host operating systems are therefore a major concern for the IT practitioner. If unwanted modification of the host system can be prevented, then the attack may be thwarted despite any weaknesses in the network which allows the attacker to contact the host machine.

There has been a distinction drawn in research between application security and operating system security. It has become increasingly clear, however, that such a distinction is academic and that in practice malicious programs and the individuals who create them make no such distinction. Malware such as Code Red exploited weaknesses in both application and operating system security (Staniford, Paxson, & Weaver, 2002). What is required is an end-to-end solution, one that considers not only the distributed nature of the current computing environment and the network, but the close relationship between the application program and the operating system (Howell & Kotz, 2000; Saltzer, Reed, & Clark, 1981; Thompson, 1984). Recent focus on the concept of endpoint security represents an additional effort to understand and mediate these risks (Kadrich, 2007).

This chapter will examine specific security features of the host operating system in a descriptive and exploratory manner. By understanding the security controls available at the operating system level and the security weaknesses in those systems it is possible understand how to better prevent attacks on these systems.

Operating systems and their underlying security mechanisms are clearly a varied landscape which over time can be quite fluid. This chapter will focus on two common server operating systems: Microsoft Windows Server 2003 and Red Hat Enterprise Linux Server 5. Rather than refer to specific versions of these operating systems, this chapter will use the terms Windows and Linux to refer to Windows Server 2003 and Red Hat Enterprise Linux Server 5 respectively. (As this chapter goes to press, the next version of Windows server operating system, Windows Server 2008, is in Beta 3; the updates to security features in this release considered relevant to this discussion will be identified and evaluated in this chapter.)

**Security and Operating System Security Architecture**

Early computers operated in closed environments with experienced and generally trusted personnel. The introduction of time-sharing with multiple concurrent processes required the consideration of how to manage the resources of the computer relative to the processes using the computer. Initial computer security concerns had focused on protecting executing tasks or processes from each other. Lampson (1974) expanded on that with a broad definition of protection that involved all resources under control of the operating system: memory, CPU registers, files (disk resources) and password security. Lampson proposed *protection domains* to define access rights and objects and associated object access lists. Under this paradigm, access to objects is enforced in relation to the protection domain of the user. The evaluation of operating system security requires a firm definition of the somewhat nebulous concept of the functionality and purpose of a computer operating system. An expansive definition could complicate qualitative comparisons with other operating systems which have not been developed with such a broad definition. For this reason, this discussion will consider the operating system the set of software which controls access to the hardware resources (CPU, disk, peripherals) of the server and will focus on specific operating system functionality which is of central importance to security, in particular, *authentication* and *authorization* (Lampson, 2004). The definitions used here identify *authenticating* principals as those which involve the process of determining which security principal made a request, and