Securing HP-UX

By Angela D. Patterson
Hewlett-Packard Company
20 Perimeter Summit Blvd.
MS 1906
Atlanta, GA 30319
Phone: 404.648.7044
Fax: 404.648.5450
Angela_Patterson@hp.com

Computing has become a major component in our lives and will continue to be a significant force for years to come. Even if we are not particularly savvy PC users, computing affects us because records of our life and life activities are managed by and reside on computers.

Back when computers were in their infancy, few people had intimate knowledge of their inter-workings. Vital information about our lives was not stored on a disk. It wasn't until their popularity grew that a larger population sought to become more technically proficient, and our method of capturing and storing information evolved from paper-driven to paperless.

As computer technology has emerged, the need for security has also risen. Since the personal information that is on computers is typically viewed as private or valuable, security of the data should be of paramount importance to the individuals responsible for maintaining the data. The average person (typically by no specific choice of her own) must trust a great number of entities to keep personal information secure.

Securing an operating system is a multi-faceted task. It involves more than simply performing operating system commands; they handle only the technical aspects of security. Other non-technical aspects of security are equally important but many people are unaware of them. We'll discuss these now and then talk about the more technical items after.

Each individual who uses a system containing private data should be concerned about its security. Therefore, the user community should make every effort to be trained sufficiently on all applications. Over half of the data loss that occurs in companies is due to poor training of personnel. Management must make application training essential for all employees. It's one proactive measure that is likely to result in increased security of data in the computing environment.

Computer hackers have many methods for gathering information about a system that they are targeting for sabotage. Social engineering and dumpster diving are two non-technical approaches that hackers use to collect data for their mission. Social engineering is a con game. A hacker masquerades as a technical expert, luring an unsuspecting user into sharing (what appears to be) harmless information about the system. This can include the system's name or its 800 dial-in number. In some cases, a hacker does *such* an effective job at gaining the confidence of the individual that he or she may reveal his username or password. This provides a clear path of invasion for the hacker. Users should be extremely cautious about with whom they share information about the system – even if it is a piece of information that seems insignificant.

Dumpster diving involves a hacker pursuing output that has been generated by the system. Reports are regularly produced which contain data from computers. Some will get filed others will be tossed out. When reports are thrown out it is usually done without thought as to what data is on the report. Users don't normally destroy reports when they are no longer needed. They often end up in recycle bins or garbage cans. Anyone who has easy access to a trash dumpster can retrieve anything from it. A hacker might find valuable information this way. Shredding or manually tearing up unwanted documents will significantly inhibit a hacker's chances of discovering a fruitful paper trail from the target machine.

Obtaining a valid password is another way in which a hacker can gain access to a system. The user community needs to be very careful about managing their passwords. For starters, chosen passwords should be difficult for a hacker to guess. Using something like a first name or a spouse's name is not considered secure. Selecting an uncommon word combined with a number is a good for a password. **Crack** is a widely-known tool that can be used to determine poor passwords. This is best used by an administrator to discover vulnerable passwords on his system and coach users in selecting more secure passwords. The downside of **Crack** is that a hacker can get a hold of it and run it to determine passwords on a system for intrusive purposes.

In addition to non-technical methods of protection against hackers, there are also many technical tasks that a system administrator can do to secure a system. The items listed in this paper are referencing the HP-UX operating system. Much of this can be applied to other UNIX implementations and the general concepts are applicable in most computing environments.

The first thing a hacker will often do is attempt to gather information about the target machine. If the system is on a network to which the hacker has access, he has the ability to invoke a number of commands from his machine to get information about the other machine. These commands include: **telnet**, **finger**, **showmount**, and **rwho**. The fortunate thing is that these commands can be secured to a certain extent. A good system administrator should know how to make these commands more secure.

Proactive account management is also an activity that system administrators can do to decrease the likelihood of intrusion by hackers. When an employee leaves a company or perhaps changes departments, the administrator should remove or disable her account. Leaving it active is considered "dormant" and is an easy target for hackers. Accounts without passwords are also highly vulnerable to attack.

A system administrator has the ability to convert his system to "trusted". The HP-UX trusted system facility is available through the graphical administration tool (**SAM**). It offers a much stricter level of security for the machine. Policies can be set for the entire system or by individual user. Features include login management capabilities, enhanced password management, enhanced terminal security, process auditing, and password history.

Various log files on a computer system reflect different types of activity that occur on the system. Viewing these files can reveal suspicious activity. The standard set of HP-UX log files includes tracking of successful logins and unsuccessful login attempts over time, currently logged in users, shutdown activity, scheduled activity, network services, and web server activity. Log file entries can be monitored through execution of the **swatch** tool.

Basic UNIX security is built upon a paradigm of access (read, write, and execute) being provided for the owner of a file, the group members of the file and any user that does not fit into one of the two former categories. Specific file owners and the system administrator are the only individuals capable of setting permissions for specific files. Access Control Lists provide an even more granular approach toward file security. Poor file permission settings, especially open write access, can render a hacker the opportunity to create “backdoors”. Backdoors allow a hacker to login to the machine at his convenience, usually unbeknownst to the administrator. **Tripwire** is an excellent tool that monitors file modifications. The **COPS** utility checks inappropriate file permissions.

Most computer systems are connected to an internal network and many are also connected to the Internet. Networked computers are significantly more susceptible to attack than standalone configurations. This is unfortunate, however, because networking is essential for the way business is conducted these days. Networking provides file sharing and print sharing among other numerous benefits.

There are a number of precautions which an administrator can take to secure network connections. **Inetd** is a process that continuously runs on HP-UX. It is responsible for “front ending” the majority of network service requests that come in to a machine. It evaluates the security of the request and if the request passes evaluation, then it is passed on to the specific network service to complete the request.

As much as we would like to believe that our popular name service resolution techniques (e.g., NIS and DNS) are secure, they are actually fundamentally unsecure pieces of network infrastructure. Configuration files can be added to these environments to decrease the vulnerability of the system.

Hackers have methods for spoofing IP addresses and also capturing packets that travel across a network. Double-Reverse Name Resolution, authentication, and encryption are alternatives that assist in the effort to secure a network. **SATAN** is a well-known tool that can be used to determine network vulnerabilities on a system or a subnet.

In the event that a security breach occurs on a computer, the approach toward resolution and recovery should be fairly standard across any operating system. The damage should be assessed immediately and contained. If the hacker is still on the system, then tracing his command history will give the administrator an idea of what the hacker has already done. Whether the hacker is on or off the machine is only one factor that should be considered when deciding on what to do after a break in is discovered. Disconnecting a system from the network or putting the system into single user mode might be initial options for the administrator.

The course of action following a break in depends on the extent of the damage and also the security policy of the organization. Recovery procedures should be implemented and once complete, business should proceed as usual. Having each user change his or her password as part of recovery is a must.

The development of a computer security policy is an investment that all companies should make. It defines the standards by which security will be enforced throughout the company. Responsible parties should be identified and more importantly, a clear plan of action should be outlined to address any security issues that arise.

Operating system security is extremely important in a computing environment. Unfortunately, it is not embraced to the extent that other protective measures are (e.g., mirroring of disks, CPU redundancy, UPS). Reaction to a break in typically results in more attention being directed toward security. One point that cannot be underscored enough, is that security is **not** the sole responsibility of a system administrator, it is the combined responsibility of *everyone* who uses the machine.