HOST-BASED IDS

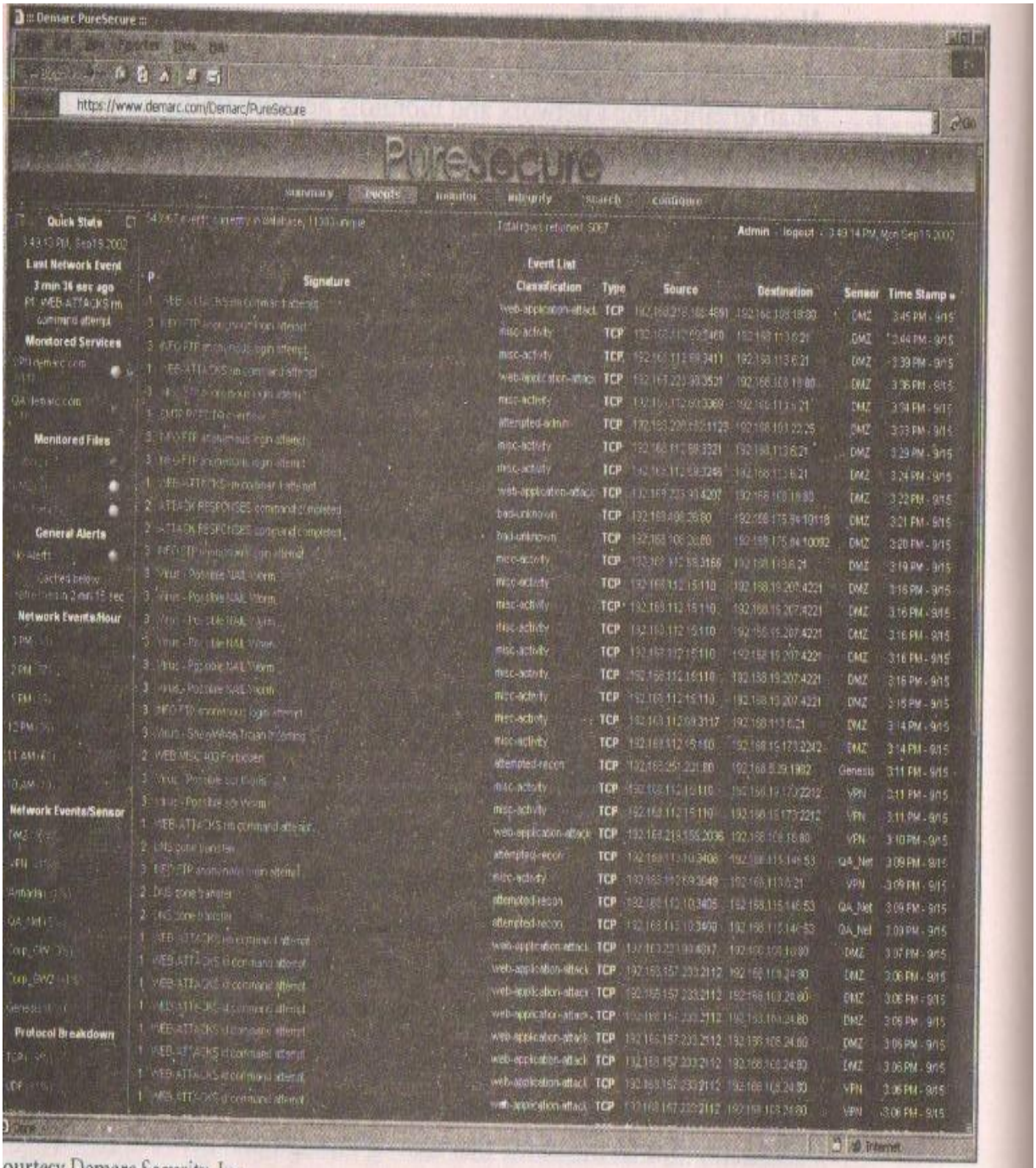
A host-based IDS (HIDS) works differently from a network-based version of IDS. While a network-based IDS resides on a network segment and monitors activities across that segment, a host-based IDS resides on a particular computer or server, known as the host, and monitors activity only on that system. HIDSs are also known as system integrity verifiers5 as they benchmark and monitor the status of key system files and detect when an intruder creates, modifies, or deletes monitored files. A HIDS is also capable of monitoring system configuration databases, such as Windows registries, in addition to stored configuration files like .ini, .cfg, and .dat files. Most HIDSs work on the principle of configuration or change management, which means they record the sizes, locations, and other attributes of system files. The HIDS then triggers an alert when one of the following changes occurs: file attributes change, new files are created, or existing files are deleted. A HIDS can also monitor systems logs for predefined events. The HIDS examines these files and logs to determine if an attack is
Underway or has occurred, and if the attack is succeeding or was successful. The HIDS will maintain its own log file so that even when hackers successfully modify files on the target system to cover their tracks, the HIDS can provide an independent audit trail of the attack.

Once properly configured, a HIDS is very reliable. The only time a HIDS produces a false positive alert is when an authorized change occurs for a monitored file. This action can be quickly reviewed by an administrator and dismissed as acceptable. The administrator may choose then to disregard subsequent changes to the same set of files. If properly configured, a HIDS can also detect when an individual user attempts to modify or exceed his or her access authorization and give him or herself higher privileges.

A HIDS has an advantage over NIDS in that it can usually be installed in such a way that it can access information that is encrypted when traveling over the network. For this reason, a HIDS is able to use the content of otherwise encrypted communications to make decisions about possible or successful attacks. Since the HIDS has a mission to detect intrusion activity on one computer system, all the traffic it needs to make that decision is coming to the system where the HIDS is running. The nature of the network packet delivery, whether switched or in a shared-collision domain, is not a factor.

A HIDS relies on the classification of files into various categories and then applies various notification actions, depending on the rules in the HIDS configuration. Most HIDSs provide only a few general levels of alert notification. For example, an administrator can configure a HIDS to treat the following types of changes as reportable security events: changes in a system folder (e.g., in C:\Windows or C:\WINNT); and changes within a security-related application (such as C:\TripWire). In other words, administrators can configure the system to alert on any changes within a critical data folder. The configuration rules may classify changes to a specific application folder (e.g., C:\Program Files\Office) as being normal, and hence unreportable. Administrators can configure the system to log all activity but to page them or e-mail them only if a reportable security event occurs. Although this change-based system seems simplistic, it seems to suit most administrators, who, in general, become concerned only if unauthorized changes occur in specific and sensitive areas of the host file system. Applications frequently modity their internal files, such as dictionaries and configuration templates, and users are constantly updating their data files. Unless a HIDS is very specifically configured, these actions can generate a large volume of false alarms.

Managed HIDSs can monitor multiple computers simultaneously. They do this by creating a configuration fiJe on each monitored host and by making each HIDS report back to a master console system, which is usually located on the system administrator's computer. This master console monitors the information provided from the managed hosts and notifies the administrator when it senseS recognizable attack conditions. Figure 7-3 provides a sample screen from Tripwire, a popular host-based IDS (see www.tripwire.com).



ourtesy Demarc Security, Inc.

FIGURE 7-2 Demarc Pure Secure Total Intrusion Detection

In configuring a HIDS. the system administrator must begin by identifying and categorizing folders and files. One of the most common methods is to designate folders using a pattern of red, yellow, and green categories. Critical systems components are coded red, and usually include the system registry and any folders containing the OS kernel, and application software. Critically important data should also be included in the red category. Support components, such as device drivers and other relatively important files, are generally coded yellow; and user data is usually coded green. This is not to suggest that user data is unimportant, but in practical and strategic terms, monitoring changes to user data does have a lower priority. One reason for this is that users are often assigned storage space that they are expected to use routinely to maintain and back up their documents, files, and images; another reason is that user data files are expected to change frequently-that is, as users make modifications. Systems kernel files, on the other hand, should only change during upgrades or installations. Categorizing critical systems components at a different level from less important files will ensure that the level of response to change will be in proportion to the level of priority. Should the three-tier system be overly simplistic for an organization, there are systems that allow for an alternative scale of 0-100, with 100 being most mission-critical and zero being unimportant. It is not unusual, however, for these types of scales to be overly refined and result in confusion regarding, for example, the prioritization of responses to level 67 and 68 intrusions. Sometimes simpler is better.

Advantages and Disadvantages of HIDSs: The following is a summary, taken from Bace and Mell, of the advantages and disadvantages of HIDSs:

Advantages:

1. A HIDS can detect local events on host systems and also detect attacks that may elude a network-based IDS.
2. A HIDS functions on the host system, where encrypted traffic will have been decrypted and is available for processing.
3. The use of switched network protocols does not affect a HIDS.
4. A HIDS can detect inconsistencies in how applications and systems programs were used by examining the records stored in audit logs. This can enable it to detect some types of attacks, including Trojan Horse programs.6

Disadvantages:

1. HIDSs pose more management issues since they are configured and managed on each monitored host. This means that it will require more management effort to install, configure, and operate a HIDS than a comparably sized NIDS solution.

2. A HIDS is vulnerable both to direct attacks and to attacks against the host operating system. Either circumstance can result in the compromise and/or loss of HIDS functionality.

3. A HIDS is not optimized to detect multi-host scanning, nor is it able to detect the scanning of non-host network devices, such as routers or switches. Unless complex correlation analysis is provided, the HIDS will not be aware of attacks that span multiple devices in the network.

4. A HIDS is susceptible to some denial-of-service attacks.

5. A HIDS can use large amounts of disk space to retain the host as audit logs; and to function properly, it may require disk capacity to be added to the system.

6. A HIDS can inflict a performance overhead on its host systems, and in some cases may reduce system performance below acceptable levels.7