StormWatch, Correlating Actions at the Host and System Level to Prevent New Intrusions

One of the values and competitive advantages of StormWatch is the ability to correlate actions. This applies at the agent level where events from the four different interceptors are correlated as well as the network-wide level where events from multiple agents are also correlated. This interrelationship of events, and the resulting pro-active actions taken by StormWatch, is crucial when preventing against new and unknown attacks.

Agent Rules Engine

The StormWatch agent, because it intercepts file, network, registry, and COM operations of an application and compares them to policy, is able to track the state of that application. Combinations and sequences of file, registry, network, and COM operations constitute the behavior of that application. When an application attempts an operation, StormWatch checks the operation against its access control policy and also correlates the policy for this operation against the maintained state of the application. This enables the agent to make real-time allow or deny decisions within the context of the overall application behavior and reduce the number of false positives associated with traditional non-correlating behavior blocking schemes.

For example, catching new Trojan horses is often difficult. A pattern matching approach that scans executables will not catch a new Trojan; maintaining inventories of acceptable executables with version and patch upgrades is a difficult task for IT. Stormwatch instead looks for actions that are commonly exhibited by Trojan programs and correlates these to make the determination that a given application is a Trojan. Examples of such actions include writing into other processes' address space to make themselves invisible in the process table, monitoring keystrokes to capture passwords, receiving UDP packets on high numbered ports. StormWatch then proactively prevents the executable from executing its intrusion.

Management Rules Engine

In the case of a new attack such as an e-mail worm, StormWatch may detect the malicious nature of the worm only after a sequence of file, network, registry, or COM operations has occurred. Once detection has occurred, an event is then sent to the management console. Ideally we want to be able to detect and stop the malicious code at other servers and desktops earlier in its operation. The management console solves this problem by correlating the events sent from various distributed agents. If the worm has been introduced to other agents on the network, these other agents will send similar events that are correlated by the management server. A policy is pro-actively updated which tells all agents not to open the offending file, effectively quarantining it and preventing further damage. The result is that IT is then faced with only a few desktops that need to be rebuilt, rather than a whole network.

StormWatch network-wide correlation provides unique functionality in the detection of distributed port scans. Low-level port scans are used by hackers to systematically scan single ports on single agents in an alternating fashion in order to map a network. For example server 1 would be scanned on port 1, server 2 on port 2, and so on. Each agent reports the scan activity to the Management Console. With its ability to

71 Second Avenue, Waltham, MA 02451
Tel: 781-209-3200 Fax: 781-209-3199
E-mail:info@okena.com
www.okena.com

6905 Rockledge Drive, Suite 600
Bethesda, MD 20817
301-896-9388

correlate events from distributed agents StormWatch is able to discern that a distributed scan is taking place where firewalls did not register the threat.

71 Second Avenue, Waltham, MA 02451
Tel: 781-209-3200 Fax: 781-209-3199
E-mail:info@okena.com
www.okena.com

6905 Rockledge Drive, Suite 600
Bethesda, MD 20817
301-896-9388