

Seeing your systems through a hacker's eyes

Attack Trees: It's a Jungle Out There

BY MICHAEL S. PALLOS

Computer security is an important aspect of any IT architecture. The requirement for security vigilance is especially critical, given the widespread availability of technology that potentially enables novice hackers to penetrate corporate IT defenses simply by using a tool available on the Internet.

Attacks can range from the theft of credit card or other sensitive customer information to embarrassing defacements of corporate Web sites. The ramifications of breached enterprises can negatively affect corporate valuation and brand equity, as well as customer and partner relations, not to mention the legal liabilities for security breaches.

Figure 1 outlines the types of malicious tools available and illustrates how their availability has lowered the bar for would-be hackers. The chart is presented by David Dittrich, senior security engineer, University Computing Services, University of Washington, and is based on information gathered by the Computer Emergency Response Team (CERT) Coordination Center (www.cert.org).

IBM WebSphere security at the application level requires a functional assessment as the application is being developed. IBM Redbooks, such as *IBM WebSphere V5.0 Security*, offer best practices for securing important WebSphere components, which include the Web server, application server, servlets, and Enterprise JavaBeans (EJBs). Another excellent cross-platform Redbook displaying how to integrate and secure the enter-

prise is *WebSphere MQ Security in an Enterprise Environment*. Once the application is deployed, a production risk assessment is also required to evaluate application security in real-world environments.

The WebSphere universe includes multiple third-party products to strengthen application infrastructure security. WebSphere MQ security products offered by Candle Corp. and other vendors, for example, help organizations secure messages before, during, and after transportation by leveraging authentication, authorization, nonrepudiation, and encryption techniques. This allows organizations to secure the back-end WebSphere Application Server and WebSphere MQ interfaces, as well as other aspects of the WebSphere environment.

"Security," according to security expert and *Secrets and Lies* author Bruce Schneier, "is only as strong as the weakest point." Often, the weakest point does not involve technological vulnerabilities, but instead may be related to enterprise procedures or physical security. Encryption algorithms are reasonably secure, and Public Key Infrastructure (PKI) Certificates, RSA key length, and

secure phone lines are not always the target penetration points for attackers.

One organization, for example, created three levels of physical security to protect its business-critical servers. The organization, however, had connected the servers to unsecured direct dial-in lines for technical support. These unsecured phone lines could provide hackers with open access to sensitive organizational data.

One approach for securing a system is to consider an attacker's viewpoint. Examining the "How can I penetrate this system?" mindset offers a perspective into the production system that is usually not considered by architects and developers. For example, would a potential attacker be able to retrieve sensitive information such as source code or network architecture blueprints from the corporate dumpster?

Identifying Risk

Attack tree analysis, created by Schneier, quantifies the security or vulnerability of a system based on the goals of the attacker. For example, if an information systems manager were responsible for an order fulfillment system containing credit card details and related customer information, an attacker may have the goal of stealing the customer credit card data. This goal – "steal credit card data" – is the starting point, or root node, of the attack tree. The attack tree is then extended, building branches down the tree to identify the different subgoals and penetration points available to the attacker. The branching process continues as the means of penetration are decomposed, or expanded, to the lowest level of intrusion, known as the leaves.

An attack tree can represent each opportunity for an attack against a computer system. Computer systems potentially contain numerous penetration points and vulnerabilities. A



ABOUT THE AUTHOR

Michael S. Pallos, is a senior solution architect for Candle Corp. (www.candle.com) with 19 years' experience in the IT industry. He is a consultant to some of Candle's largest corporate customers and a featured speaker at industry conferences. Michael is also an IBM-certified e-business designer, e-business technologist, and WebSphere specialist.

E-MAIL

michael_pallos@candle.com

single attack tree represents each single penetration point. The consolidation of numerous attack trees is known as an attack forest.

The attack-tree approach allows analysts to rethink system vulnerabilities from the attacker's perspective. This perspective expands the software or system vulnerabilities model, as defined by I.V. Krsul in his thesis, "Software Vulnerability Analysis," to include elements other than the software application. One example is the previous example of an attacker rummaging through a dumpster searching for password information and other sensitive documents.

Attack trees are represented graphically and textually. A graphical representation is usually built with the root node, or goal, on the top. The tree then descends branches and subgoals until the leaves are finally reached at the bottom level. Figure 2 is the conceptual model of an attack tree represented in a graphical format.

The textual representation of an attack tree follows a numeric outline

structure. The root node, or goal, is represented at the first level with no indentation. Each subgoal is then numbered accordingly and indented one unit per level of decomposition. The representation below presents the textual view using the same example content found in Figure 2.

1. Goal (root node)
 - 1.1 Leaf 1
 - 1.2 Sub-Goal
 - 1.2.1 Leaf 2
 - 1.2.2 Leaf 3

When applying attack-tree logic to a production WebSphere application, you can gain insights into the potential penetration points that an attacker may leverage. You start by building the higher-level nodes, then expand downward.

Utilizing "The Twenty Most Critical Internet Security Vulnerabilities – The Experts' Consensus," developed by the SysAdmin, Audit, Network, Security (SANS) Institute, you can build an attack forest by focusing on the Unix vulnerabilities illustrated in

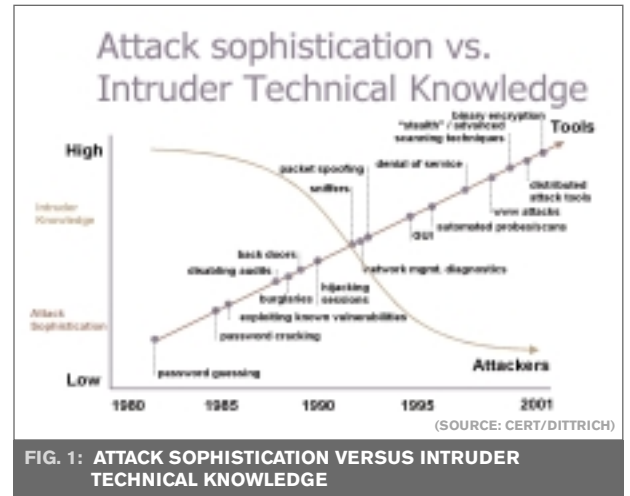


FIG. 1: ATTACK SOPHISTICATION VERSUS INTRUDER TECHNICAL KNOWLEDGE

Figure 2. Each of the nodes in Figure 3 can be extended, creating a unique attack tree.

Figure 4 represents a graphical attack tree that examines one of the Unix vulnerabilities reported by SANS. According to the experts, password vulnerabilities for users, systems administrators, and applications include accounts with no passwords,

Once you're in it...

...reprint it!

- Wireless Business & Technology
- Java Developer's Journal
- XML Journal
- ColdFusion Developer's Journal
- PowerBuilder Developer's Journal

Contact Carrie Gebert
201 802-3026
carrieg@sys-con.com

RePrints

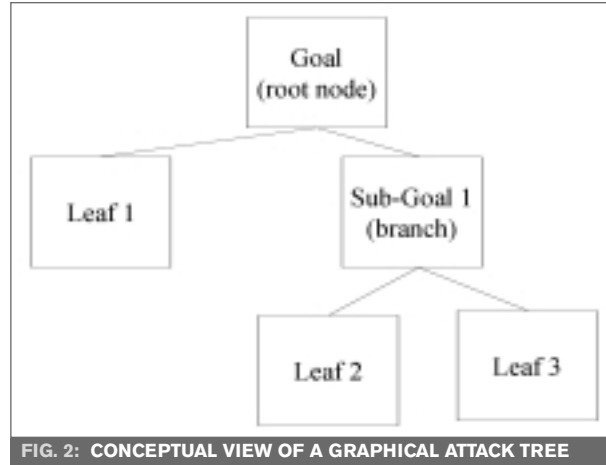


FIG. 2: CONCEPTUAL VIEW OF A GRAPHICAL ATTACK TREE

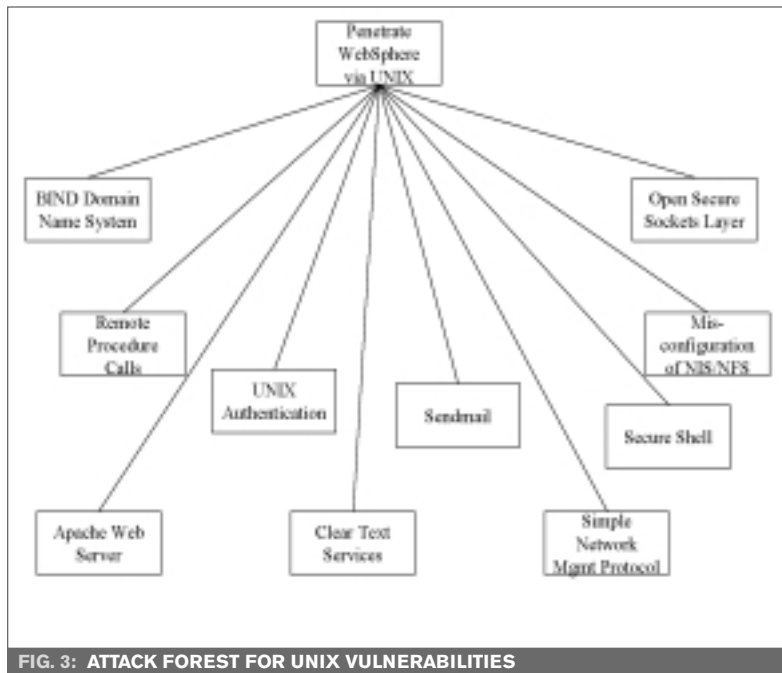


FIG. 3: ATTACK FOREST FOR UNIX VULNERABILITIES

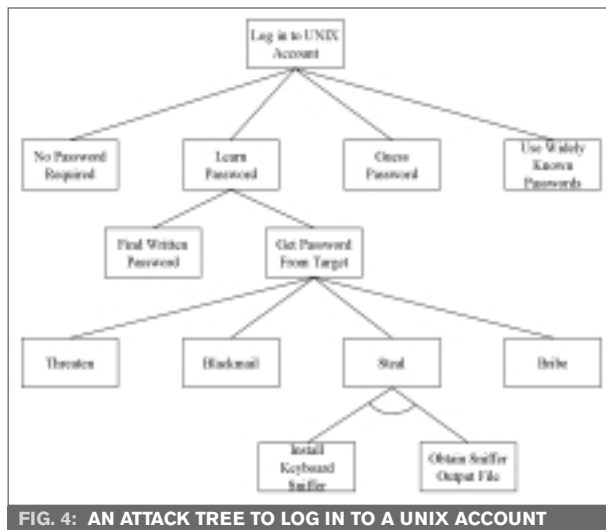


FIG. 4: AN ATTACK TREE TO LOG IN TO A UNIX ACCOUNT

weak passwords, commonly known passwords (such as your company name), and weak hash algorithms.

The attack tree in Figure 4 also aids you in considering alternative ways in which a node can be achieved. Analysts are forced to ask themselves questions from an attacker's perspective, such as "How can I steal passwords?" By taking a broader view of information security, WebSphere Application Server security expands from permissions granted on an EJB to the possibility of installing keyboard sniffers on WebSphere Application Server administrators' computers. This perspective is far different from a WebSphere developer's perspective for designing and building secure EJBs.

Comprehensive WebSphere security encompasses more than the specific WebSphere Application Server application environment. Enterprise architects, information system managers, system administrators, security experts, and WebSphere team members must consider additional aspects of vulnerabilities and penetration points that computer attackers can exploit outside of the WebSphere framework. Attack tree analysis offers a systematic methodology for identifying penetration points and system vulnerabilities not considered from the application design perspective. 🌐

Resources:

- Davies, S., et al. (2003). *WebSphere MQ Security in an Enterprise Environment*: www.redbooks.ibm.com/redbooks/SG246814.html
- Kovari, P., et al. (2003). *IBM WebSphere V5.0 Security*: www.redbooks.ibm.com/redbooks/SG246573.html
- *Software Vulnerability Analysis*: www.acis.ufl.edu/~ivan/articles/main.pdf
- *Attack Modeling for Information Security and Survivability*: www.sei.cmu.edu/publications/documents/01.reports/01tn001.html
- *The Twenty Most Critical Internet Security Vulnerabilities*: www.sans.org/top20
- Schneier, B. (2000). *Secrets & Lies: Digital Security in a Networked World*. John Wiley & Sons.