Managing Vulnerabilities in Networked Systems

Robert A. Martin
The MITRE Corp.

The Common Vulnerabilities and Exposures initiative, an international, community-based effort from industry, government, and academia, is collaborating on efforts to find and fix software product vulnerabilities more rapidly, predictably, and efficiently.

Most organizations recognize the importance of cyber security and are implementing various forms of protection. However, many are failing to find and fix known security problems in the software packages they use as the building blocks of their networks and systems, a vulnerability that a hacker can exploit to bypass all other efforts to secure the enterprise. Consider the following scenario:

> You would have thought that the firewalls, combined with filtering routers, password protection, encryption, and disciplined use of access controls and file permissions would have been protection enough. Yet an overlooked flaw in the company's Web server application version allowed a hacker to insert a series of ".." sequences into a URL. This modification let the hacker make the server navigate out of its document directories and retrieve a database of user names and encrypted passwords. Unfortunately, the passwords had only a weak encryption algorithm for protection. The hacker quickly decrypted the database and extracted the passwords. After logging into the server using one of the stolen passwords, the hacker exploited a known buffer overflow vulnerability in a system utility to obtain administrator-level access. From there it was easy for the hacker to scan and break into other machines within the company's intranet, crashing the payroll server with malformed inputs that did not comply with the standard for communications protocols. Once the hacker replaced the company's public Web pages with details of the hack and added a live video stream of an ongoing internal, private, and sensitive company meeting, no one could doubt how badly the company had been hacked.

To avoid such disasters and transform this area from a liability to a key asset in the fight to build and maintain secure systems, a broad spectrum of organizations in the information security and software products communities are participating in the Common Vulnerabilities and Exposures initiative. CVE, which began in 1999, seeks the adoption of a common naming practice for describing software vulnerabilities and including these names within security tools and services as well as on the fix sites of commercial and open source software package providers.

VULNERABILITIES AND EXPOSURES

Programmers know that they make mistakes when writing software, including typos, math errors, incomplete logic, or incorrect use of functions or commands. Sometimes mistakes occur even earlier in the development process, reflecting an oversight in the requirements guiding the design and coding of a par-

0018-9162/01/$17.00 © 2001 IEEE

ticular function or a software program's capability. Mistakes that have security implications become *vulnerabilities*, which hackers can use directly to access protected data, and *exposures*, which provide information or capabilities that can function as stepping-stones to direct access.

A hidden danger

All types of software are likely to contain mistakes that have security ramifications. Developers must evaluate large and complex software and smaller applications alike for errors that can threaten security integrity. Any of the various software elements comprising a system could be the one that compromises it.

In the past, organizations had stand-alone computer systems that interacted only with other internal systems. Only a few systems used tapes and file passing to exchange information with outside systems. This isolation meant that software errors usually had limited impact. The general public was unaware of most errors, crashes, and oversights, which at best caused occasional troubles for an organization's closest business partners.

Today, however, few organizations—whether in the private or government sector—have or build self-contained systems. It is the norm for employees, customers, business colleagues, and the general public to have some degree of access and visibility into the minute-by-minute health and performance of an organization's software environment. Delays in processing, mistakes in calculations, system downtime, and even slowdowns in response times attract notice and often result in critical comments.

An explosion in the different ways to access and use these systems accompanies this increased visibility. Web and application servers facilitate system interconnections and leverage Internet-based technologies. Access to Web sites, purchase sites, online help systems, and software delivery sites increases the visibility of organizations that own those sites. To better support business partners and employees working at remote locations, on the road, or from home, corporate intranets and extranets connect them to their backroom systems. Emerging technologies such as instant messaging, mobile code, and chat offer users effortless access across organizational boundaries.

The movement to highly accessible systems, driven by the need to save resources and improve efficiency as well as the reality of having to do more with less, has dramatically increased the impact of mistakes in commercial and open source software.

Consequences

Although errors in self-developed software can have a major impact on an organization's ability to function, vulnerabilities and exposures in the commercial and open source software packages used as system components create a bigger problem. A mistake giving access to an unauthorized individual can expose private information about customers and employees. Unauthorized access can enable hackers to change information or use the system to their own advantage, or it can shut down internal and publicly accessed systems without the organization's knowledge.

For example, a computer hacker broke into a hospital's computer network in the Seattle area and downloaded thousands of medical records. The hacker's activities went unnoticed by the hospital until he went public with his accomplishment, at which time the hospital initially denied his claims. The next day, the hospital confirmed the intrusion.[1]

Identifying the vulnerabilities in the software your organization uses requires research and, probably, spending some money.

In addition to tarnishing an organization's public image, the discovery of a vulnerability that enables making unauthorized changes or that enables the theft of services and information can have legal or financial implications that have a direct operational impact. For example, the recent Code Red and Nimda worm infections, which made use of several publicly known vulnerabilities in Microsoft's IIS Web server, seem to have prompted the Gartner Group to recommend that organizations immediately consider moving to a different Web server.[2]

ASSESSING THE THREAT

Determining the vulnerabilities and exposures embedded in commercial and open source software systems and networks is a critical first step. If you know what you need to fix and how to get the fix, a simple patch, upgrade, or configuration change could be sufficient to eliminate even the most serious vulnerability.

Identifying the vulnerabilities in the software your organization uses requires research and, probably, spending some money. Most commercial software customers have little or no insight into the implementation details of the software they purchase. At best, they may have an understanding of a package's general architecture and design philosophy. Commercial software vendors typically regard design details and software code as business-critical private information and, in a highly competitive environment, these vendors are often reluctant to share problems even with their customers.

Where to go

How do you find out about software vulnerabilities and exposures if the vendors will not tell you and looking for the problems yourself is impractical? During the past decade, three groups have emerged that share the same curiosity:

Table 1. Alert and advisory services.

Service	Type	Organization
Bugtraq	E-mail list	Bugtraq
Cassandra	Alerts	Center for Education and Research in Information and Security (CERIAS), Purdue University
CERT/CC Advisories	Advisory	Computer Emergency Response Team Coordination Center (CERT/CC)
CyberNotes	Monthly newsletter	National Infrastructure Protection Center
RAZOR	Advisory	BindView
SAFER	Monthly newsletter	The Relay Group
SANS NewsBites	E-mail list	Systems Administration, Networking, and Security (SANS) Institute
Security Alert Consensus	E-mail list	Network Computing/SANS
SecurityFocus Newsletter	Newsletter summary of Bugtraq e-mails	SecurityFocus
X-Force Alerts	Advisory	Internet Security Systems (ISS)

- *Hackers*. Originally used to describe prolific and inventive software programmers, in recent years the term has come to refer to those who circumvent information systems or network security mechanisms. So-called black-hat hackers, also known as crackers, seek to uncover software vulnerabilities and exposures for some malicious purpose and often share their information with like-minded individuals. White-hat hackers, on the other hand, usually help organizations to assess and understand the vulnerabilities and exposures in their systems.
- *Commercial interests*. These include software and network security companies that sell consultation services to find and assess the mistakes or tools—some of which are Internet based—that let you evaluate your systems' vulnerabilities and exposures by yourself.
- *Philanthropists*. This term describes security researchers in various government, academic, and nonprofit organizations as well as unaffiliated individuals who enjoy searching for these types of mistakes. They usually share their knowledge and tools freely.

All three groups find searching for software vulnerabilities and exposures challenging because new classes of software and ways to use them are constantly emerging in the marketplace. This mushrooming of software capabilities also requires organizations that use these systems to correctly configure and integrate various vendors' offerings without creating additional mistakes.

Information on vulnerabilities and exposures in software products is widely available. In response to the arduous task of tracking and reacting to new and changing errors, members of these three groups use Web sites, newsgroups, software and database update services, notification services such as e-mail lists, and advisory bulletins to keep constituents informed and current.

However, organizations or individuals who make a vulnerability discovery often act as if they are the only source of information about it and use their own approach for quantifying, naming, describing, and sharing information on what they find. Also, accompanying the introduction of new types of software products and networking are additional classes of vulnerabilities and exposures that must be described and categorized. Further, the software suppliers who create and maintain commercial and open source products do not always use the same descriptions and names as hackers, commercial interests, and philanthropists, making it difficult to locate the fix to a particular problem once it is detected.

Vulnerability tools

Hackers use the Internet as their main conduit for sharing information about exploiting software vulnerabilities and exposures. Member organizations among commercial interests create and continually update their own mechanisms. Some vendors market vulnerability scanners driven by their own databases such as Network Associates' CyberCop Scanner, BindView's bv-Control, and Qualys's QualysGuard. Other vendors sell different types of intrusion detection systems that monitor networks and systems for attacks, including Enterasys Networks' Dragon, Cisco Systems' Secure IDS, and Symantec's NetProwler. Philanthropists also make both types of tools available as freeware.

Scanners include tests that compare software version information and configuration settings with an internal list of vulnerability data. They can also conduct their own scripted set of probes and penetration attempts. The market recently developed a self-service-based capability using remotely hosted vulnerability scanners that scan Internet-resident firewalls, routers, and hosts. Scan results are provided to customers via a secure link and shielded from everyone else—including the service provider. Once set up, the scans can be rerun whenever customers want.

As an alternative to tracking and recording every update, patch, and upgrade applied to each platform in the enterprise, vulnerability scanners offer an

attractive choice for monitoring the health of software applications. These tools benefit from the market's vigorous hunt for errors and the development of testing approaches that can reveal vulnerabilities or exposures in an organization's deployed systems. However, because they can yield false positives and negatives and thus far do not provide complete coverage, they are no panacea.

IDS products look for indications of actual attack activities, many of which can then be mapped to specific vulnerabilities that these attacks could exploit. IDS capabilities are often available as part of a managed security service, in which the organization contracts out the intrusion detection and monitoring to a security services vendor.

Both scanner and IDS tool providers harvest information about vulnerabilities and exposures from public information sites, hacker sites, newsletters, and advisories. They also have their own investigative consultants who continuously evaluate an organization's systems and networks for new software mistakes. The vendors' parent companies offer vulnerability databases for a fee, but many—including Symantic and Internet Security Systems—also openly share raw information on a Web site. Some philanthropists likewise provide very sophisticated search and tailored notification services for free but their veracity, quality, and the level of effort required to make use of them vary considerably.

Alert and advisory services

Government and academic philanthropists, and some companies, offer several widely used and highly valued announcement, alert, and advisory services for free. Table 1 lists several examples.

The product suppliers who make the software that contain these vulnerabilities also provide solutions for them. Many have their own methods of providing customers with software fixes and updates. Until recently, most software suppliers were not proactive in distributing patches and updates outside of their normal development cycle. Now, however, many major vendors—including IBM, Microsoft, SGI, and Sun Microsystems—provide alerts and advisories concerning security problems, fixes, and updates.

Vulnerability Tower of Babel

If you tried to make use of these various vulnerability services, tools, and databases—along with the software suppliers' update announcements—to assess, manage, and fix your own vulnerabilities and exposures just a few years ago, you would have faced a cacophony of naming standards and methods for defining individual security problems in software. For example, Table 2 shows how in 1998 each of a dozen leading organizations referred to the same vulnerability by a different name. Such confusion made it hard to understand what vulnerabilities you faced and which ones each tool was looking for—or not looking for. To get a fix, you then had to map the vulnerability or exposure to the software supplier's name for the problem.

Table 2. Vulnerability Tower of Babel, 1998.

Organization	Name referring to vulnerability
AXENT (now Symantec)	phf CGI allows remote command execution
BindView	#107—cgi-phf
Bugtraq	PHF Attacks—fun and games for the whole family
CERIAS	http_escshellcmd
CERT/CC	CA-96.06.cgi_example_code
Cisco Systems	HTTP—cgi-ph
CyberSafe	Network: HTTP 'phf' attack
DARPA	0x00000025 = HTTP PHF attack
IBM ERS	ERS-SVA-E01-1996:002.1
ISS	http—cgi-phf
Symantec	#180 HTTP server CGI example code compromises http server
SecurityFocus	#629—phf Remote Command Execution Vulnerability

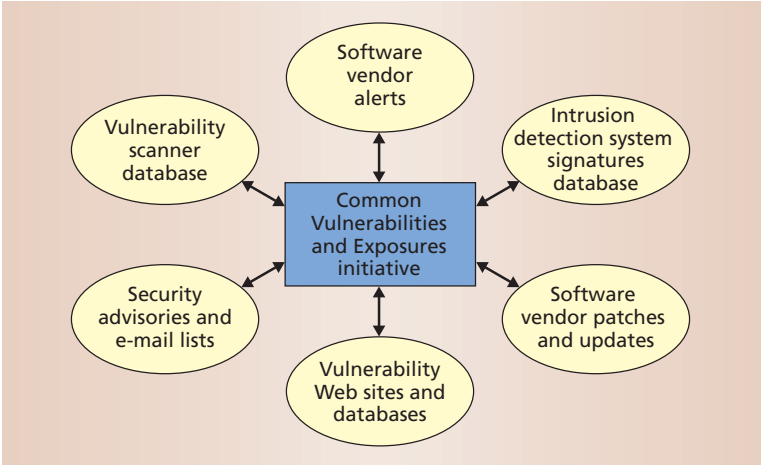

Figure 1. Common Vulnerabilities and Exposures initiative—a bridge linking databases and concepts.

THE CVE SOLUTION

Driven by our desire to develop an integrated picture of what was happening in our own networks and to select some new tools, the MITRE Corporation (http://www.mitre.org) began designing a method to sort through the confusion. It involved creating a reference list of unique vulnerability and exposure names and mapping them to appropriate items in each tool and database. In January 1999, we presented a paper[3] at the 2nd Workshop on Research with Security Vulnerability Databases at Purdue University that outlined the basic ideas and approach of the CVE Initiative (http://cve.mitre.org).

A logical bridge

As Figure 1 shows, we envisioned CVE to begin as simply a mechanism for linking vulnerability-related databases or concepts—nothing more. Because we felt it was critical for the information-security community

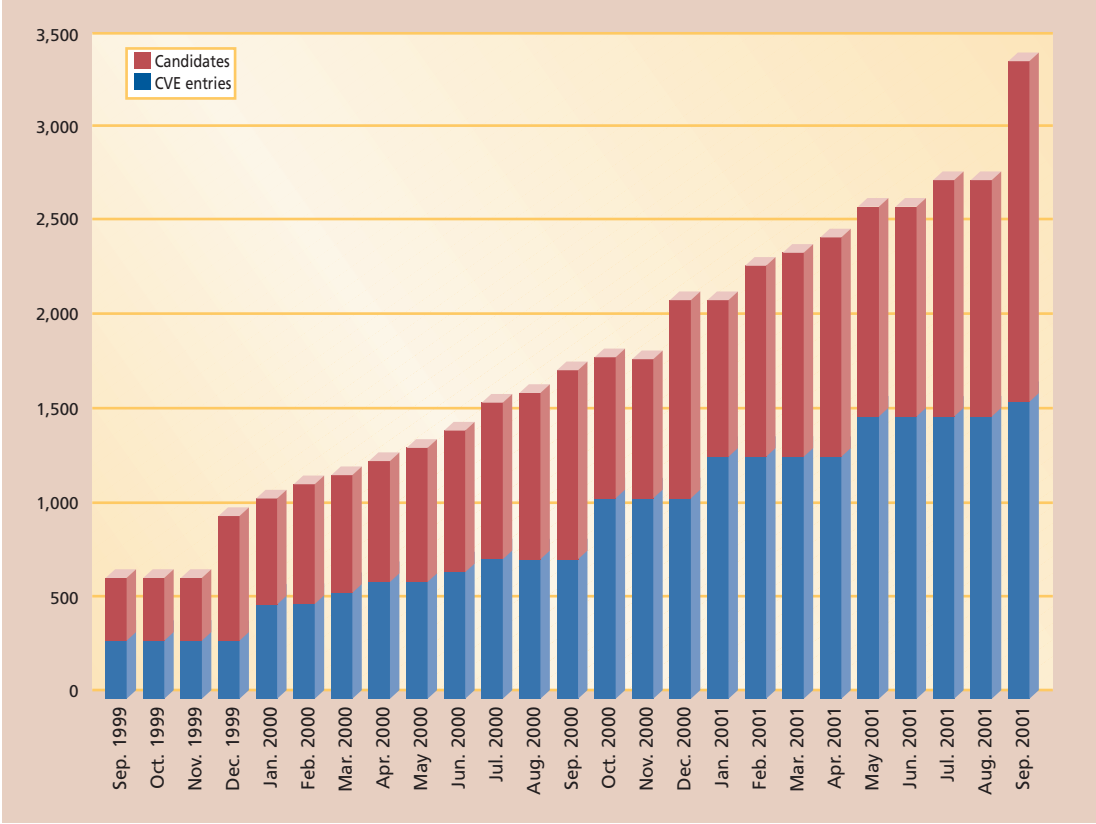

Figure 2. CVE growth over time. Since the CVE initiative began in September 1999, the number of list entries has grown by 400 percent, and the number of candidates has increased by 460 percent.

to concur with the concept and begin incorporating the common names into their various products and services, we limited CVE's role to a logical bridge to avoid competing with existing and future commercial efforts.

CVE has since evolved into an international, 32-organization, cross-industry effort to create and maintain a standard list—known as the CVE list—of vulnerabilities and exposures. It is gradually approaching its goal of uniquely naming every publicly known security-relevant software mistake. More than half are now either listed or under review, and 39 organizations are presently building more than 60 products or services that use common CVE names.

How CVE works

Several key tenets underlie the CVE initiative:

- Each vulnerability or exposure should have one name and one standardized description.
- The CVE list should exist as a dictionary rather than as a database. Further, the list and relevant information about CVE activities should be publicly accessible via the Internet for downloading and reviewing.
- Industry endorsement should occur via a CVE editorial board and the development of CVE-compatible products and services.

The common names in the CVE list result from open and collaborative discussions of the CVE editorial board. The board includes prominent information security specialists from numerous information-security-related organizations around the world, including commercial security-tool vendors, academic and research institutions, and government agencies. The board invites other information-security experts to participate on an as-needed basis, based on recommendations from board members. Archives of board meetings and discussions are available at the CVE Web site.

With MITRE'S support, the board identifies which vulnerabilities or exposures to include on the CVE list and then determines the common name, description, and references for each entry. For example, CVE-1999-0067 is an encoding of the year and a unique number *N* for the *N*th name assigned that year. MITRE maintains the CVE list and Web site, moderates editorial board discussions, analyzes submitted items, and provides guidance throughout the process to ensure that CVE remains objective and continues to serve the public interest.

Building the CVE list

MITRE analyzes vulnerabilities and exposures identified prior to the initiative as well as newly discovered ones for possible inclusion in the CVE list. MITRE has thus far received approximately 8,400 pre-CVE submissions from Axent (now Symantec), BindView, Harris Corporation, Cisco Systems, Purdue University's Center for Education and Research in Information and Security, Hiverworld (now nCircle), SecurityFocus, Internet Security Systems (ISS), Network Associates, L3 (now Symantec), and the Nessus Project (Renaud Deraison and Jordan Hrycaj).

The CVE Content Team—a collaboration of MITRE

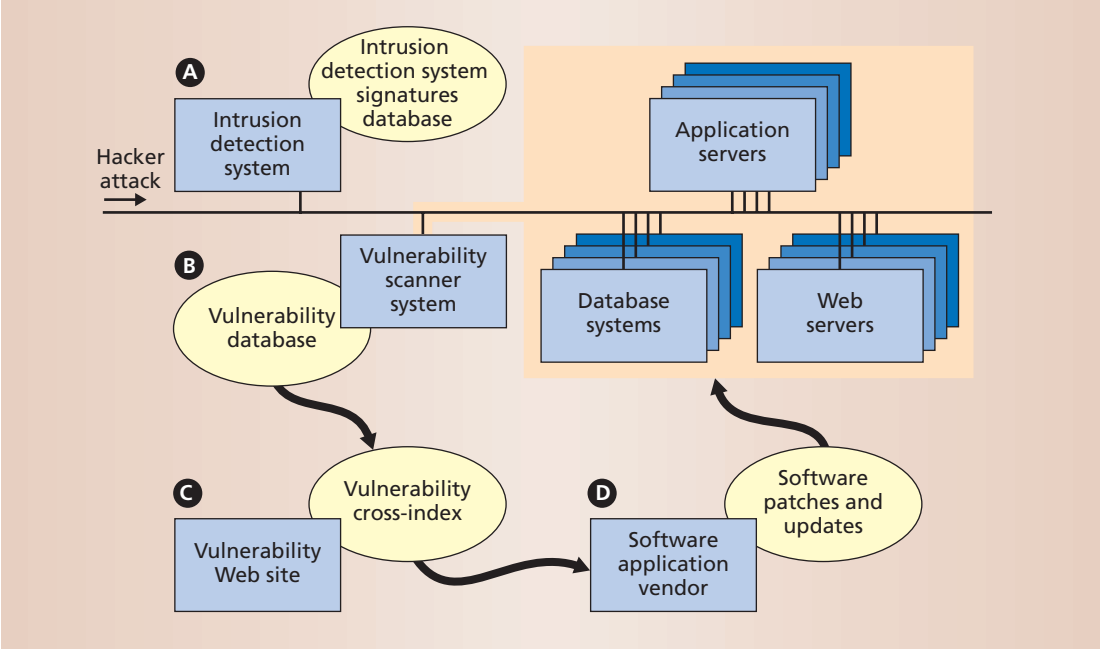

Figure 3. Using a CVE-compatible intrusion detection system. (A) An attack report identifies vulnerabilities to attack. (B) You compare this information to recent scanner output to determine if an attack can exploit the vulnerabilities or exposures. If it can, (C) you use a vulnerability Web site to identify the location of a CVE-compatible fix and (D) obtain the fix from the software product vendor.

security analysts and researchers who sift through related submissions to create uniquely named candidates that meet CVE criteria and have sufficient reference and descriptive data to establish each candidate's uniqueness—has just completed the first round of analysis. It has thus far eliminated 2,500 legacy submissions that duplicated existing candidates or entries or did not meet the CVE definition of a vulnerability or exposure. An additional 3,900 require additional information from the source that provided them, and 1,100 have been set aside for more detailed examination and study. The remaining 900 legacy submissions formed the basis of 563 CVE candidates.

Each month, MITRE receives between 150 and 300 new submissions from ISS, SecurityFocus, Neohapsis, and the National Infrastructure Protection Center. It also assigns five to 15 specific candidates to new vulnerabilities and exposures within the initial public announcements to the world via vendor and security community member alerts and advisories. To date, ISS, Rain Forest Puppy, BindView, Compaq, SGI, IBM, the Computer Emergency Response Team Coordination Center, Microsoft, Hewlett-Packard, and Cisco Systems have included CVE names in their alerts or advisories.

As Figure 2 shows, the number of entries in the CVE list increased from 321 in September 1999 to approximately 1,600 entries in October 2001, while candidates have increased from 320 to nearly 1,800. Including the current CVE list, recently added legacy candidates, and the ongoing generation of new candidates from recent discoveries, the CVE Web site now tracks some 3,400 uniquely named vulnerabilities and exposures.

BENEFITS OF CVE COMPATIBILITY

A CVE-compatible tool, Web site, database, or service uses CVE names in a way that lets users cross-link its information with other repositories, tools, and services that also use CVE names. CVE compatibility entails meeting three basic requirements:

- Customers who use CVE names to inquire about scope, content, or coverage must receive any related information.
- Output must include all related CVE names.
- Any vulnerability repository used by a CVE-compatible item must be provided to MITRE with a mapping relative to a specific CVE list version. The item's owner must make a good-faith effort to ensure that the item's mappings remain accurate as the CVE list and the compatible item are updated over time.

Various products, services, and repositories address different portions of the CVE list. For example, some deal with Unix while others focus on Windows NT. Users must evaluate CVE-compatible items against their organization's specific needs in terms of platform and software product coverage.

Integrating vulnerability services, databases, Web sites, and tools that incorporate CVE names will provide more complete security coverage. For example, a report from a vulnerability scanning tool that uses CVE names will let an organization quickly and accurately locate fix information in one or more of the CVE-compatible databases and Web sites—such as the ICAT Metabase (http://icat.nist.gov/icat.cfm), a searchable index of computer vulnerabilities and exposures that links users who want to find and fix existing system problems to a variety of publicly available vulnerability databases and patch sites. Figure 3 shows how an organization could detect and react to an ongoing hacker attack with CVE-compatible products.

Organizations can determine exactly what each CVE-compatible tool covers because the CVE list provides a baseline. The organization can simply determine how many of the CVE entries apply to its

platforms, operating systems, and commercial software packages and then compare this subset to the tool's coverage.

Network and security trade journals already refer to CVE name support as a desirable feature in their product reviews and comparisons of scanners and IDS devices.[4,5] Last year, a group of concerned security professionals composed a list[6] of the 10 most common, critical Internet security threats. Orchestrated by the Systems Administration, Networking, and Security Institute, the effort represented the consensus of a wide variety of security experts. To help ensure specificity and make the recommendations actionable, each suggestion included the appropriate CVE names—totaling 68—and detailed the specific issue areas for a variety of platforms and products. Now cosponsored by the Federal Bureau of Investigation, the recently updated SANS list of top Internet security holes has grown to 20 and includes 125 CVE names.[7]

An organization can also use CVE-compatible products to improve its response to security advisories. CVE-compatible advisories include CVE entries of vulnerabilities that scanners can check for, and you can determine whether your IDS has appropriate attack signatures for the alert. In addition, for systems that an organization builds or maintains for its customers, CVE-compatible advisories and announcements can help directly identify any need for software fixes from those systems' commercial vendors. This approach provides a more structured and predictable process for handling advisories than most organizations currently possess.

Vulnerabilities and exposures will always be a part of our systems, as will the groups that find and share information about commercial and open source software errors. The ability to apply all known security fixes and patches offers a robust way to keep an organization's software infrastructure healthy. The common-names-integration and cross-referencing abilities now emerging in vulnerability and exposure tools, Web sites, and databases make it possible to deal with security-relevant mistakes and improve systems' security. CVE's adoption and support within commercial and academic communities across the globe are facilitating a more systematic and predictable handling of security incidents.

An increasing number of organizations—all listed on the CVE Web site—are developing CVE-compatible products and services. Several members of each type of tool, service, repository, and announcement capability now support CVE names, with vendor announcement and fix sites the only underrepresented areas. However, many organizations are actively discussing adding CVE names to their announcements and alerts, and we anticipate that software patch and update sites will follow. As vendors respond to more user requests for CVE-compatible fix sites, securing the enterprise will gradually include the complete cycle of finding, analyzing, and fixing vulnerabilities. ✸

Acknowledgment

The summary work contained in this article was funded by the MITRE Corporation and is based on the composite effort of all those working on the Common Vulnerabilities and Exposures Initiative. An early version of the article was published in *CrossTalk, The Journal of Defense Software Engineering*.

References

1. B. Sullivan, "Hospital Confirms Hack Incident," *MSNBC,* 9 Dec. 2000; http://stacks.msnbc.com/news/499856.asp (current Oct. 2001).
2. J. Pescatore, "Nimda Worm Shows You Can't Always Patch Fast Enough," *Gartner FirstTake (FT-14-5524),* 19 Sept. 2001; http://www.gartner.com/DisplayDocument?doc_cd=101034 (current Oct. 2001).
3. D.E. Mann and S.M. Christey, "Towards a Common Enumeration of Vulnerabilities," *2nd Workshop Research with Security Vulnerability Databases*, Purdue University, West Lafayette, Ind., 1999; http://cve.mitre.org/docs/cerias.html (current Oct. 2001).
4. J. Forristal and G. Shipley, "Vulnerability Assessment Scanners," *Network Computing,* 8 Jan. 2001; http://www.networkcomputing.com/1201/1201f1b2.html (current Oct. 2001).
5. P. Mueller and G. Shipley, "To Catch a Thief," *Network Computing,* 20 Aug. 2001; http://www.networkcomputing.com/1217/1217f1.html (current Oct. 2001).
6. W. Jackson, "Top 10 System Security Threats Are Familiar Foes," *Government Computer News,* Aug. 2000; http://www.gcn.com/state/vol6_no8/news/812-1.html (current Oct. 2001).
7. S. Bonisteel, "'Top 10' List of Net Security Holes Grows to 20," *Newsbytes,* 2 Oct. 2001; http://www.newsbytes.com/news/01/170713.html (current Oct. 2001).

Robert A. Martin *is the primary point of contact for CVE compatibility efforts, a co-lead for MITRE's Cyber Resource Center Web site, and a principal engineer in MITRE's Information Technologies Directorate. His research focuses on the interplay of cyber security, critical infrastructure protection, and e-business technologies and services. Martin received an MSEE from Rensselaer Polytechnic Institute and an MBA from Babson College. He is a member of the IEEE Computer Society, the IEEE, the ACM, the NDIA, and AFCEA. Contact him at ramartin@mitre.org.*