Developing on the Net, Dealing with Software Vulnerabilities

presented by

Robert A. Martin

28-30 August 2001

Sponsored by

AFCEA International

Developing on the Net, Dealing with Software Vulnerabilities

Robert A. Martin, Member, AFCEA
The MITRE Corporation, B155
202 Burlington Road,
Bedford, MA 01730-1420, USA
(781) 271-3001
ramartin@mitre.org

One 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 enough protection. However, an overlooked flaw in their web server application version allowed a hacker to insert a series of ".." sequences into an URL which allowed the hacker to make the web server to navigate out of the web server's document directories and retrieve a database of user names and encrypted passwords. Unfortunately, the encryption algorithm on the passwords was weak, and the hacker was able to quickly decrypt the database and extract 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 and obtained administrator-level access. From there it was easy for the hacker to scan and break into other machines within their Intranet, crashing the payroll server with malformed inputs that did not comply with the standard communications protocols. Once the company's public web pages were replaced with details of the hack and the addition of a live video stream of an ongoing internal, private and sensitive company meeting, it left little room for doubt as to how badly they had been hacked.

1. Introduction

While most organizations have addressed the various aspects of implementing cyber security, many are failing to successfully address the one area of security that can allow someone to bypass all of their other efforts to secure the enterprise. That area is finding and fixing the known security problems in the software packages that they use as the building blocks of their networks and systems. But there may be an answer that will transform this area from a liability to a key asset in the fight to build and maintain secure systems. The answer rests in an initiative to adopt a common naming practice for describing the vulnerabilities, and the inclusion of those names within security tools and services as well as the software fix sites of the commercial software package vendors. The initiative has been in practice for more than a year across a broad spectrum of the information security and software products community: It is called the *Common Vulnerabilities and Exposures (CVE) Initiative.*

1.1 To err is human

Every programmer knows they make mistakes when writing software, whether it be a typo, a math error, incomplete logic, or incorrect use of a function or command. Sometimes the mistake is even earlier in the development process - reflecting an oversight in the requirements guiding the design and coding of a particular function or capability of a software program. When these mistakes have security implications, those with a security bent will often refer to them as *"vulnerabilities and exposures."*[1]

All types of software, from the large complex pieces to those that are small and focused, are likely to contain software mistakes with security ramifications. Large complex software like operating systems, database management systems, accounting systems, inventory management systems, as well as smaller applications like macros, applets, wizards, and servlets need to be evaluated for mistakes that can impact their security integrity. We must remember that when we put these various software products together to

[1] A vulnerability is a mistake that can be directly used by someone to gain access to things they are not supposed to have. An exposure is a mistake that gives that person access to information or capabilities which he or she can then make use of, as a stepping-stone, to gain access.

provide an overall system, any of the software elements that comprise the system could be the one that compromises it.

Things were different in the past, when an organization's computer systems were stand-alone and only interacted with other systems within the same organization. Only a few systems used tapes and file passing to exchange information with outside systems. The same holds true for government and military systems, including weapons. This isolation meant that errors in commercial or developed software usually had limited impact, at least from the public's point of view. In fact, most errors, crashes, and oversights went unnoticed by the general public. At most, these problems would cause occasional troubles for an organization's closest business partners.

1.2 You can't hide it…

The same is not true today. Very few of today's organizations, whether in the private sector or government, have or build self-contained systems. It is the norm for employees, customers, business partners, 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, even slow downs in response times are noticed and often result in critical comment.

Accompanying this increase in visibility is an explosion in the different ways that those systems are accessed and used. Web and application servers have been created to help make systems interconnect and leverage Internet-based technologies. Access to web sites, purchase sites, online help systems, and software delivery sites make the organizations that own the sites very visible. To better support business partners and employees working at remote locations, on the road, or from home, we have connected our back-room systems to the corporate intranet and extranet. New technologies have emerged, like instant messaging, mobile code, and chat, whose functionality requires effortless access by users across organizational boundaries. The movement to highly accessible systems, driven by the need to save time and make businesses more efficient and the reality of having to do more with less, has dramatically increased the impact of mistakes in commercial software.

While errors in self-developed software can still have a major impact on an organization's ability to function, it is the vulnerabilities and exposures in the commercial and open source software packages that they use as components of a system that creates the bigger problem. A mistake in a commercial or open source software package can open a front or a back door into situations that most organizations strive to avoid. A mistake permitting an unauthorized individual access can expose private information about customers and employees. It can allow them to change information or perform services with your systems to their own advantage. In addition, a vulnerability can allow them to shut down your internal and publicly accessed systems, sometimes, without your knowledge. In those cases where the vulnerability or exposure allows someone to make changes or bring down systems, or when the theft of services and information is eventually noticed,[2] there can be a huge impact to the organization's public image.[3] There can also be legal liability and direct operational impact.

2. What can you do?

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

To find out about the vulnerabilities in the software used by your organization you have to do some research and probably spend some money. With commercial software, the customer has little or no insight into the implementation details. At the very best you may have an understanding of the general architec-

---

[2] A computer hacker broke into a hospital in the Seattle area and thousands of medical records were downloaded. The hacker's activities went unnoticed by the hospital and when the hacker went public with his accomplishment, his claims were initially denied. The next day, the hospital confirmed the intrusion [5].

[3] One of Microsoft's web sites was penetrated by a Dutch hacker through the web server's "IIS Unicode" vulnerability that let him copy files, execute commands, and change files [2].

ture and design philosophy of a package. Companies offering commercial software treat the design details and software code as business-critical private information. In addition, since most of these companies are highly competitive, commercial software vendors are sometimes reluctant to share their problems with even their customers.

2.1 Who knows?

So how do you find out about software vulnerabilities if the vendors aren't going to tell you and it would be impractical for you to look for the vulnerabilities yourself? During the last decade, three groups have emerged who share the same curiosity. For sake of discussion we will refer to these as the hackers,[4] the commercial interests, and the philanthropists. The hackers, unfortunately, want to find vulnerabilities and exposures so they can exploit them to gain access to systems. Those with commercial interests want to be hired to find the mistakes or they want you to buy their tools to help you find the vulnerabilities and exposures yourself. They offer their services in the form of consultants who will come and do an evaluation of your software systems, and in the form of tools that you can buy and run yourself. Some proffer the use of their tools as an Internet-based service. This group includes software and network security companies that provide security consulting services and vulnerability assessments, databases of vulnerabilities and exposures, and the tools for security services and vulnerability evaluations. The philanthropists include security researchers in various government, academic, and non-profit organizations, as well as unaffiliated individuals that enjoy searching for these types of mistakes, and usually share their knowledge and tools freely.

Each of the three groups have members focused on sharing the information that is found: among like-minded individuals in the hackers group, for a price in most cases for the commercial interests group, and generally for free in the philanthropists group.

For all three groups the search for vulnerabilities and exposures in software is challenging since the marketplace is constantly developing and offering new classes of software and new ways of using them. This mushrooming of software capabilities also creates an ever-changing challenge for organizations using these software systems to correctly configure and integrate the offerings of various vendors without opening additional vulnerabilities and exposures from configuration and permission mistakes.

2.2 How to find out

In response to the arduous task of tracking and reacting to new and changing vulnerabilities and exposures, the members of these three groups are using web sites, news groups, software and database update services, notification services like e-mail lists, and advisory bulletins to keep their constituents informed and current.

So the information on vulnerabilities in software products is available. Great, right? Well, not quite. There are several problems. The largest is that each organization (or individual) in these three groups has been pursuing their vulnerability discovery and sharing efforts as if they were "THE" source of information on vulnerabilities. Each uses its own approach for quantifying, naming, describing, and sharing information about the vulnerabilities that they find. Additionally, as new types of software products and networking are introduced, whole new classes of vulnerabilities and exposures have been created that require new ways of describing and categorizing them.

The other problem is that finding the vulnerabilities and exposures within systems is just the first step. What we really want to do is to take the list of vulnerabilities and get them fixed. This is the domain of software suppliers, who create and maintain our commercial and open source products. Unless they use the same descriptions and names as the hackers, commercial interests and philanthropists groups, we will have a difficult, confusing and frustrating time getting the fix for any particular problem that is found.

---

[4] Unlike its original meaning which referred to a hacker as a prolific and inventive software programmer, over the past few years hacking has come to refer to the act of circumventing security mechanisms of information systems or networks. "Black-hat" hackers are those intent on doing harm, as opposed to "white-hat" hackers, who are usually working in support of organizations to help them assess and understand the vulnerabilities and exposures in their systems. Black-hat hackers are sometimes referred to as crackers.

2.3 A closer look at who knows what

The Internet is the main conduit used by the hackers group for sharing information on vulnerabilities and how to exploit them. Different member organizations in the commercial interests group have their own mechanisms for sharing vulnerability information. For example, the tool vendors create vulnerability scanners that are driven by their own vulnerability databases. The Intrusion Detection System (IDS) vendors build different types of software systems for monitoring your network and systems for attacks. There are also scanner and IDS tools available from the philanthropists group as freeware. Both the scanner and IDS providers have to continuously update their tools with new information on what and how to look for problems. Examples of these organizations and tools are shown in Table I.

Table I
Scanner and IDS Offering Examples

Product	Tool Type	Organization
Centrax	scanner/IDS	CyberSafe
CyberCop	scanner	Network Associates
Dragon	IDS	Network Security Wizards
HackerShield	scanner	BindView Corporation
LANPATROL	IDS	Network Security Systems
Nessus	freeware scanner	Renaud Deraison & Jordan Hrycaj
NetProwler	IDS	Symantec Corporation
QualysGuard	ASP-based scanner	Qualys
RealSecure	IDS	Internet Security Systems
SAINT	scanner	World Wide Digital Security
Secure IDS	IDS	Cisco Systems
STAT	scanner	Harris Corporation
SWARM	scanner	Hiverworld, Inc.

Scanners typically include tests that compare version information and configuration settings of software with an internal list of vulnerability data. They may also conduct their own scripted set of probes and penetration attempts. IDS products typically look for indications of actual attack activities, many of which can then be mapped to the specific vulnerabilities that these attacks could exploit. The scanner market recently developed a self-service-based capability. It uses remotely hosted vulnerability scanners on the Internet that you can hire to scan your Internet resident firewalls, routers, and hosts. The results of the scans are provided to you through a secure link, and you can usually run the scans whenever you want. These scans are shielded from everyone other than you, including the service provider. IDS capabilities are often available as part of a managed security service, where the organization contracts out the intrusion detection and monitoring to a security services vendor.

Both IDS and scanner tool providers harvest information about vulnerabilities and exposures from public information sites, hacker sites, newsletters, and advisories. They also have their own investigative researchers who continuously look for new vulnerability information that will make their company's offering better than the competition,[5] as well as providing them with the "free" advertising that comes with finding and publicly reporting new vulnerabilities and exposures. Typically, these researchers serve as consultants within the company, offering their services to evaluate an organization's systems and net-

[5] As an alternative to tracking and recording each update, patch, and upgrade that gets applied to each platform in the enterprise, the use of vulnerability scanners is an attractive choice for monitoring the health of your software applications. These tools are benefiting from the vigor of the marketplace's hunt for vulnerability information and the development of testing approaches that can turn up the presence of vulnerabilities or exposures in the "deployed" systems of an organization. However, due to "false positives," "false negatives" and incomplete coverage to-date, these tools are not a panacea.

works. Their parent companies also offer databases of vulnerabilities for a fee, although some also share the information openly as raw information on a web site.

Some members of the philanthropists group also offer very sophisticated search and notification services for free, but their veracity, quality, and level of effort vary considerably. Examples of vulnerability-sharing organizations are shown in Table II.

Table II
Vulnerability Sharing Examples

Site Name	Type	Organization
arachNIDS	free IDS database	Max Vision Network Security/Whitehats
CERIAS Vulnerability Database	database	CERIAS/Purdue University
Fyodor's Playhouse	hacker web site	Insecure.Org
Online Vulnerability Database	database	Ernst & Young's eSecurityOnline.com
ICAT Metabase	free web site	NIST
Bugtraq mailing list Database	mailing list database	SecurityFocus.com
PacketStorm	hacker web site	Securify, Inc.
SWAT Database	database	Symantec Corporation
Vigil@nce AQL	database	Alliance Qualité Logiciel
X-Force Database	free web site	Internet Security Systems

In addition to freeware scanner tools, IDS tools, and vulnerability databases, the philanthropists group government and academic members offer several announcement, alert, and advisory services that are widely used and highly valued. Some commercial interests group companies offer these types of free services as well. Examples are shown in Table III.

Table III
Alert and Advisory Services

Service	Type	Organization
Bugtraq	e-mail list	Bugtraq
Casandra	alerts	CERIAS/Purdue University
CERT Advisories	advisory	CERT Coordination Center
CyberNotes	monthly newsletter	NIPC
Razor	advisory	Bindview Corporation
S.A.F.E.R.	monthly newsletter	The Relay Group
SANS NewsBites	e-mail list	SANS Institute
Security Alert Consensus	e-mail list	Network Computing and SANS
SecurityFocus Newsletter	newsletter summary of Bugtraq e-mails	SecurityFocus.com
SWAT Alerts	alerts	AXENT Technologies
X-Force Alert	advisory	Internet Security Systems

So, there are numerous venues for finding out what vulnerabilities and exposures exist in your organization's commercial software systems, as well as many tools and service providers willing to help you figure out which vulnerabilities and exposures you have. The three groups we've covered … hackers, commercial interests, and philanthropists … all address locating the vulnerabilities and exposures in the commercial software that forms the base of your live systems and networks.

We will now address finding the "fixes." The solutions for vulnerabilities are provided by the product suppliers who make the software in which these vulnerabilities were found. Many of them have their own methods of providing their customers with software fixes and updates. Until recently, most software suppliers were not very proactive in distributing patches and updates outside of their normal software de-

velopment cycle. This has improved considerably. Now, many major suppliers provide alerts and advisories concerning security problems, fixes, and updates (see Table IV).

Table IV
Vendor Alert and Advisory Services

Service	Type	Organization
IBM ERS	advisory	IBM
Microsoft Product Security Notification Service	advisory	Microsoft Corporation
SGI Security Advisory	advisory	Silicon Graphics, Inc.
Sun-alert	alert	Sun Microsystems, Inc.

But can these various vulnerability services, tools, and databases, along with the software suppliers' update announcements, effectively combine to help you assess, manage, and fix your vulnerabilities and exposures? The short answer is that it used to be very difficult, but now a way to do it seems to be at hand. So what was wrong and what changed?

3. The tower of Babel

In 1998, if you tried to make use of these various tools, services, and databases you were faced with a problem rooted in the heritage of each product and effort. Each had developed its own naming standards and methods for defining individual entries in their respective vulnerability data stores. Table V shows how the same vulnerability was referred to by twelve different names by twelve leading organizations. With such confusion, it was very hard to understand what vulnerabilities you faced, what vulnerabilities were being looked for … or *not* looked for … by each tool. And then you still had to map the vulnerability or exposure to the software supplier's name for the problem to get a fix.

Table V
The Vulnerability Tower of Babel

Organization	Name used to refer to vulnerability
AXENT	phf CGI allows remote command execution
BindView	#107 – cgi-phf
Bugtraq	PHF Attacks – Fun and games for the whole family
CERIAS	http_escshellcmd
CERT	CA-96.06.cgi_example_code
Cisco Systems	HTTP - cgi-phf
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
Security Focus	#629 - phf Remote Command Execution Vulnerability

3.1 A solution

Driven by our own attempts to develop an integrated picture of what was happening in our own networks and in trying to select some new tools, The MITRE Corporation[6] started to design a method for working through the confusion of vulnerability and exposure information. This method was based on the

[6] MITRE, working in partnership with government, is an independent, not-for-profit corporation working in the public interest.

creation of a reference list of unique names that would then be mapped to the appropriate items in each tool and database. In January, 1999, the first public statement of our idea for a reference list of unique names for vulnerabilities and exposures was presented at the 2nd Workshop on Research with Security Vulnerability Databases, held at Purdue University. MITRE presented a paper [3] at this conference that outlined the basic ideas and approach for what is today called the *Common Vulnerabilities and Exposures (CVE) Initiative* [cve.mitre.org/].

Our vision for CVE was for it to provide a mechanism for linking together vulnerability-related databases or concepts – and nothing more (see Figure 1). Rather than viewing this narrow scope as a limitation, we saw it as an advantage. By agreeing to limit the use of CVE to the role of a logical bridge, we could avoid competing with exiting and future commercial efforts. This was important, since it was critical that the information security community concur with the CVE concept and proceed to incorporate the common names into their various products and services.

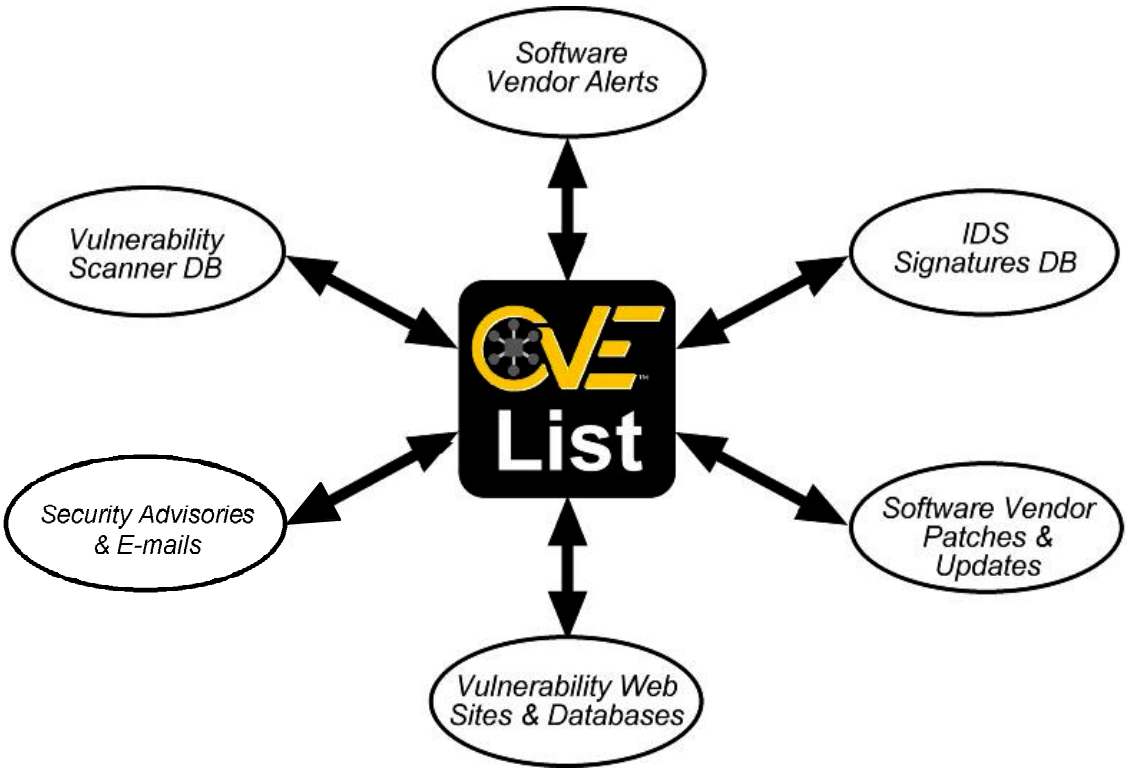

Figure 1. CVE List as a Bridge

By June 2001, the CVE effort evolved into a 31-organization, cross-industry effort, creating and maintaining a standard list of vulnerabilities and exposures. Almost half of the known vulnerabilities and exposures are either listed or under review, and presently, 32 organizations are building over 50 products or services that use the common CVE names.

3.2 The CVE List Today

The Common Vulnerabilities and Exposures Initiative is an international, community activity focused on developing a list that provides common names for publicly known information security vulnerabilities and exposures. The CVE List and information about the CVE effort are available on the CVE web site at [cve.mitre.org/cve/].

As mentioned above, the CVE Initiative is growing toward its goal of uniquely naming every known vulnerability and exposure. As shown in Figure 2, the two different portions[7] of CVE have grown from their initial level of 321 CVE entries (also called "names") and 320 candidates (CANs) in September 1999 to the current level of 1,510 CVE entries and 1,120 CANs. On average, there have been 100 new candidates added each month. The vast majority of the CVE List to-date has been based[8] on vulnerabilities that have been newly discovered. There are also approximately 10,0000 submissions[9] from industry vulnerability databases of pre-CVE (i.e., pre-1999) items. These legacy submissions are being analyzed for addition to the CVE List. The current estimate is that these pre-CVE submissions will result in approximately 4,500 uniquely named CVE candidates.

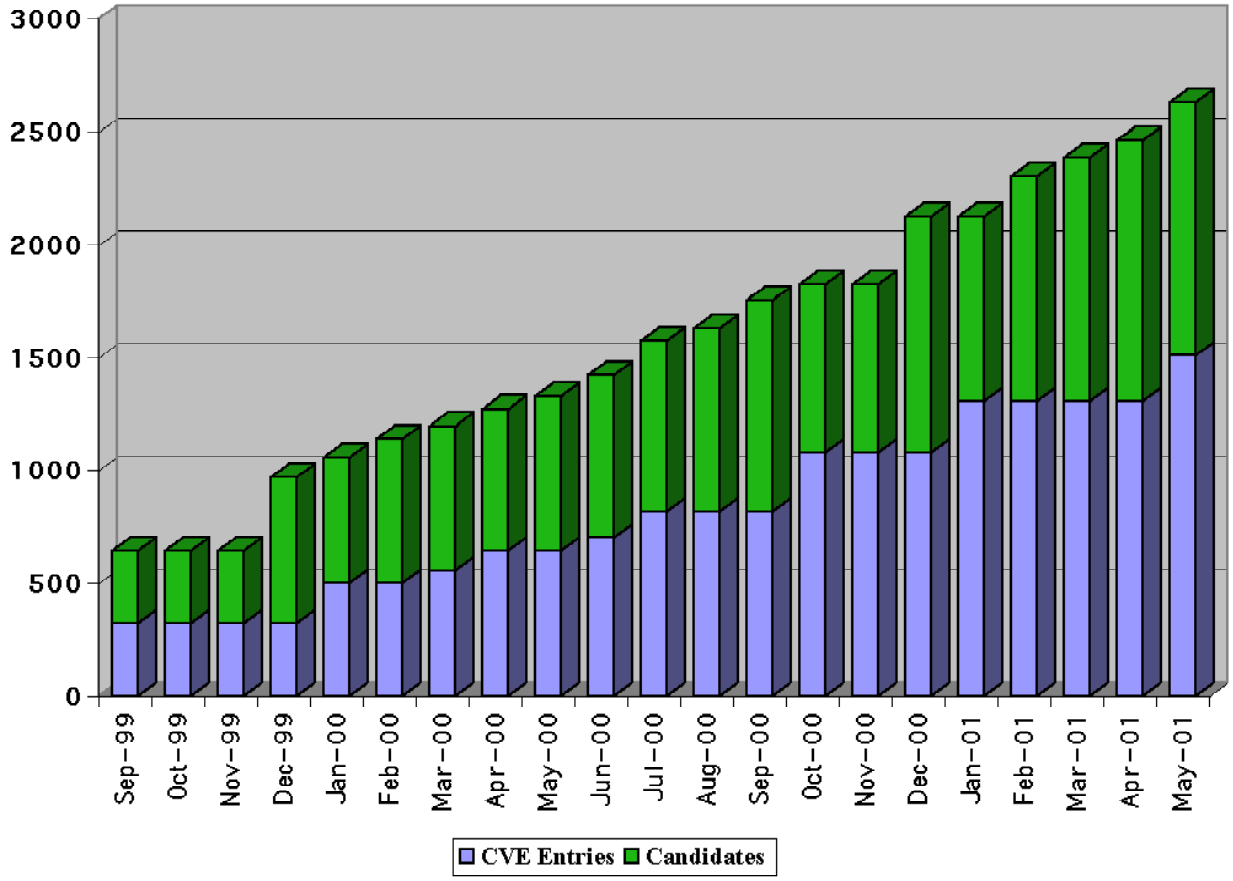

Figure 2. CVE Growth Over Time

3.3 How CVE works

The common names in CVE result from open and collaborative discussions of the CVE Editorial Board. This Board, as shown in Table VI, includes members from numerous information security-related organizations around the world including commercial security tool vendors, members of academia, research institutions, government agencies, and other prominent information security experts. The Board

[7] The CVE List includes entries awaiting approval by the CVE Editorial Board called candidates (CANs) and entries (official CVE entries) that have already been approved.

[8] CVE receives new vulnerabilities from ISS, SecurityFocus, Neohapsis, and NIPC CyberNotes.

[9] CVE received vulnerability databases from AXENT (now Symantec), BindView, Harris, Cisco, Perdue's CERIAS, Hiverworld (now nCircle), SecurityFocus, ISS, NAI, L3 (now Symantec), and Nessus with vulnerabilities that were discovered prior to the CVE Initiative.

identifies which vulnerabilities or exposures will be included in CVE, then determines the common name, description, and references for each entry. The CVE name, for example CVE-1999-0067, is an encoding of the year that the name was assigned and a unique number N for the Nth name assigned that year.

Table VI
CVE Editorial Board Composition

Area of Expertise	Organizations
Academic/Educational	UC Davis, SANS, CERIAS
Network Security Analysts	Vista IT, Genuity
Other Security Experts	IBM Research, NSA, MITRE, Zero-Knowledge Systems
Intrusion Detection Experts	Silicon Defense, SANS
Tool Vendors	The Nessus Project, ISS, PGP Security-Network Associates, BindView, AXENT, CyberSafe, Symantec, NFR, nCircle, Harris, Cisco
Software Vendors	IBM, Sun Microsystems, Microsoft
Incident Response Teams	CanCERT, CERT/CC, DOD-CERT
Information Providers	NTBugtraq, Security Focus, National Institute of Standards and Technology (NIST), Ernst & Young, eSecurityOnline.com

MITRE maintains the CVE List and web site, moderates Editorial Board discussions, and provides guidance throughout the process to ensure that CVE remains objective and continues to serve the public interest. Archives of Board meetings and discussions are available for review on the CVE web site at [cve.mitre.org/board/archives/]. Other information security experts are invited to participate on the Board on an as-needed basis, based upon recommendations from Board members.

The key tenets of the CVE Initiative are:

- One name for one vulnerability or exposure
- One standardized description for each vulnerability or exposure
- Existence as a dictionary rather than a database
- Publicly accessible for review or download from the Internet
  Industry-endorsed via the CVE Editorial Board and CVE-compatible products

4. What does it mean to be CVE-compatible?

"CVE-compatible" is a phrase that indicates that a tool, web site, database, or service uses CVE names in a way that allows a user to cross-link its information with other repositories, tools, and services that use CVE names. To be CVE-compatible, the product, service, database, or web site must meet the following three requirements:

- CVE Searchable: The user (or customer) can use CVE names to ask questions about scope, content, or coverage and they will be supplied with any related information.
- CVE Output: When information is presented it includes the related CVE name(s).
- Mapping: Any repository of vulnerabilities used has been provided to CVE with a mapping relative to a specific version of CVE, and a good faith effort has been made to ensure accuracy of that mapping.

Different products, services, and repositories address different portions of the complete CVE List. For example, some might deal with UNIX, while others focus on Windows NT. When looking at CVE-compatible items, you will need to evaluate them against your organization's specific needs in terms of coverage of the platforms and the software products that you use.

4.1 Why would you want CVE-compatible products?

In the first place, CVE compatibility makes it possible for you to use your vulnerability services, databases, web sites, and tools together since they can "speak" to each other through their shared use of CVE names. For example, if a report from a vulnerability scanning tool incorporates CVE names, you can quickly and accurately locate fix information in one or more of the separate CVE-compatible databases and web sites to determine how to fix the problems identified by the vulnerability scanner. Also, with CVE-compatible tools, you'll know exactly what each tool covers because the CVE List provides you with a baseline. You simply determine how many of the CVE entries are applicable for your platforms, operating systems, and commercial software packages, and use this subset to compare against the tool's coverage. Before the use of common names, it was extremely difficult to identify the vulnerabilities of your systems,[10] or to determine whether a particular tool or set of tools covered them.

4.2 Improving the process

The CVE effort is changing the way organizations use security tools and data sources to address their operational security posture. One example is shown in Figure 3, where an organization is able to detect an ongoing attack with its CVE-compatible IDS system (A). In a CVE-compatible IDS, specific vulnerabilities that are susceptible to the detected attack are provided as part of the attack report. This information can then be used to compare against the latest vulnerability scan by your CVE-compatible scanner (B) to determine whether your enterprise has one of the vulnerabilities or exposures that can be exploited by the attack. If it does, you can then turn to a CVE-compatible fix database at the vendor of the software product that has the problem or you can avail yourself of the services of a vulnerability web site like the ICAT Metabase,[11] which lets you identify (C) the location of the fix for a CVE entry (D), if one exists.

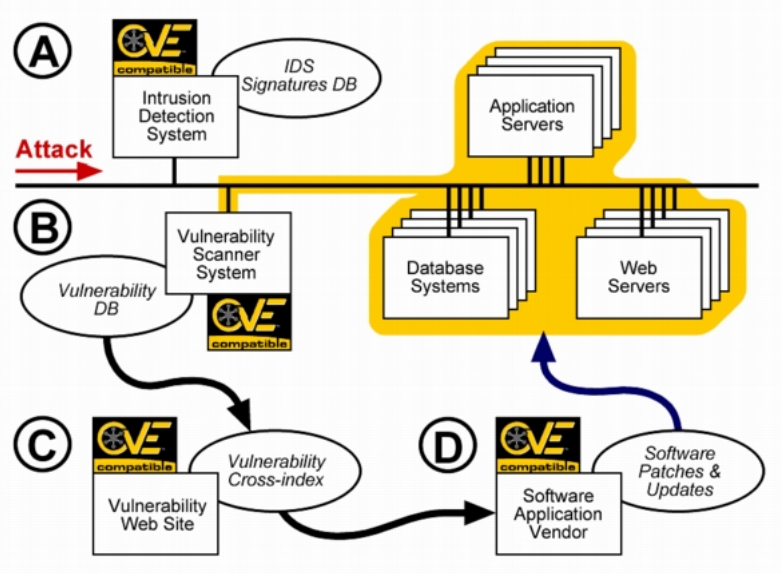

Figure 3. A CVE-enabled process

[10] The CVE Initiative is in the process of analyzing all of the "legacy" vulnerabilities and exposures, and assigning them CVE numbers. Numerous members of the security vulnerabilities reporting and tracking community have contributed their legacy databases to the CVE effort to support this effort.

[11] The ICAT Metabase is a searchable index of computer vulnerabilities and exposures. ICAT links users into a variety of publicly available vulnerability databases and patch sites, thus enabling one to find and fix the problems existing on their systems. ICAT is not itself a vulnerability and exposure database, but is instead a searchable index leading one to vulnerability resources and patch information [4].

4.3 Identifying your risk

Another thing you can accomplish with CVE-compatible products, that would be hard if not impossible to do before common names were adopted, is improving how your organization responds to security advisories. If the advisory is CVE-compatible it will include CVE entries. With that information you can see if your scanners check for these vulnerabilities, and determine whether your IDS has appropriate attack signatures for the alert. Additionally, for systems that you build or maintain for customers, the CVE compatibility of advisories and announcements will help you to directly identify any fixes from the vendors of the commercial software products in those systems (if the vendor fix site is CVE-compatible). This is a much more structured and predictable process for handling advisories than most organizations currently possess.

4.4 Making guidance actionable

Earlier this year, a group of concerned security professionals put together a "Top 10" list [1] that outlined the 10 most common, critical Internet security threats. The effort was orchestrated by the System Administration, Networking, and Security (SANS) Institute and brought together a consensus list from a wide variety of security experts. To help bring specificity and make the recommendations actionable, each of the top 10 suggestions had the appropriate CVE names, detailing each of the specific issue areas for a variety of platforms and products. A total of 68 CVE names were called out in the list of 10 threats covered in the SANS list.

4.5 So who is CVE-compatible?

While the list of organizations with CVE-compatible products is expanding, at this writing the 34 listed in Table 7 are those working toward compatibility. For a current list visit the CVE web site at [cve.mitre.org/compatible/].

Table VII
Organizations Developing CVE-Compatible Products

Advanced Research Corporation	nCircle
Alliance Qualité Logiciel	NSecure Software (P) Ltd.
AXENT Technologies, Inc.	NIST
BindView Development	The Nessus Project
CERIAS/Purdue University	Network Security Systems
CERT Coordination Center	Network Security Wizards
Cisco Systems	NTBugtraq
Computer Security Laboratory, UC Davis	PGP Security, Network Associates
CyberSafe	Penta Security Systems, Inc.
CYRANO	Qualys
Ernst & Young	SANS
Harris Corporation	Security Focus, Inc.
Intranode	Security Watch
Intrusion.com	spiDYNAMICS
Internet Security Systems, Inc.	Symantec
LURHQ Corporation	Tivoli Systems Inc.
Max Vision Network Security/Whitehats	World Wide Digital Security

CVE has grown significantly over the past 18 months and, as of the writing of this paper, 9 more organizations are preparing to announce their support of CVE in their security products or services. Already, there are several members of each type of tool, service, repository, and announcement capability that support CVE names. The only areas that are under represented are vendor announcement and vendor

fix sites; however, several vendors are actively discussing adding CVE names to their announcements and we hope they will follow-up by adding CVE names to their software patch and update sites. By the time you read this article there should be several vendors using CVE names in their announcements and alerts. In addition, like the CVE Editorial Board, the list of organizations working on or delivering CVE-compatible products has become international in scope.

5. Conclusion

The application of all known security fixes and patches is the complement of standard security protection mechanisms. Keeping current on fixes offers a robust method for keeping the commercial software that makes up your organization's software infrastructure healthy. Vulnerabilities and exposures will always be a part of our systems, as will the groups that find and share information about vulnerabilities and exposures in commercial software. With the common names integration and cross-referencing abilities emerging in vulnerability and exposure tools, web sites, and databases it is becoming possible to deal with these mistakes and improve the security of our systems. The changes in tools and databases, brought about by the adoption and support of CVE within the commercial and academic communities, are allowing for more systematic and predictable handling of security incidents. As vendors respond to user requests for CVE-compatible fix sites, the complete cycle of finding, analyzing, and fixing vulnerabilities will be addressed. The adoption and support of CVE by more products, services, and vendors is moving this part of securing the enterprise from art to science.

References

[1] Jackson, William, "Top 10 system security threats are familiar foes," *Government Computer News,* Jun. 12, 2000.

[2] Lemos, Robert, "Power Play: Electric Company Hacked," *ZDNet News,* Dec. 15, 2000.

[3] Mann, David E. and Christey, Steven M., "Towards a Common Enumeration of Vulnerabilities," *2nd Workshop on Research with Security Vulnerability Databases, Purdue University, West Lafayette, Indiana,* Jan. 21-22, 1999.

[4] Mell, Peter, "The ICAT Metabase," *Computer Security Division at the National Institute of Standards and Technology. (http://icat.nist.gov/icat.taf),* Dec. 19, 2000.

[5] Sullivan, Bob, "Hospital confirms hack incident," *MSNBC,* Dec. 9, 2000.

Acknowledgments

The summary work contained in this article was funded by The MITRE Corporation and is based on the composite effort of all of the individuals working on the Common Vulnerabilities and Exposures Initiative.

Author Biography

Robert A. Martin is a co-lead for MITRE's Cyber Resource Center web-site and a Principal Engineer in MITRE's Information Technologies Directorate. At the culmination of his five years of Y2K leadership and coordination efforts, Mr. Martin served as the Operations Manager of the Cyber Assurance National Information Center, a 24x7 cyber security watch center within the President's Y2K Information Coordination Center. Today, Martin's efforts are focused on the interplay of cyber security, critical infrastructure protection, and e-Business technologies and services. Martin received a bachelor's degree and a master's degree in electrical engineering from Rensselaer Polytechnic Institute and a master's of business degree from Babson College. He is a member of the ACM, AFCEA, IEEE, and the IEEE Computer Society.