



WHITE PAPER

Approaching Security

Joe Levy, CTO of Solera Networks



APPROACHING SECURITY

A \$20,000 firewall with every industry certification will not make you secure. VPNs, NAC, 2FA, DLP, FDE, and WAFs will not make you secure.¹ Achieving regulatory compliance or passing a vulnerability scan will not make you secure. So what will? Nothing. Literally. So long as we are trying to protect assets, we can anticipate that some combination of existing threats and vulnerabilities puts those assets at risk. Pursuing a goal of absolute security is untenable. Instead, security should not be considered a goal or a final destination, but rather a perpetually reiterative effort.

To institutionalize this, we should look at information security in terms of risk management—the practicable mitigation of risks to an acceptable level. Such an approach to security should be considered not only by CSOs and designers and implementers of information systems, but also by IT consultants and vendors en masse, as it may help to usher in systems where security is a natural and integral component rather than an afterthought.

INTRODUCTION TO RISK MANAGEMENT

Despite a lack of widespread awareness, risk management is not a new concept within information security. There is hardly a shortage of established risk management methodologies, or of risk-oriented information security management systems (ISMSs) and compliance programs (such as ISO 17799, ISO 27001, PCI, OCTAVE, NIST sp800-30, COBIT, ITIL, SOX, GLBA, HIPAA, and FISMA,² to name just a few). In fact, we see quite a bit of overlap between all of these, and it is at these areas of intersection where their commonalities begin to emerge. The foundations for these systems predate the explosion of the Internet age; although, we have a tendency to associate them with network security—an ironic symptom of our urge to both blame every problem on technology, and then try to solve every problem with technology. These systems and programs are based on methodologies that are decades old, and they share this essential constitution:

- Know what you are trying to protect.
- Know the events you are protecting your assets against.
- Know how those events might occur.
- Know what you are currently doing for protection.

¹ VPN: Virtual Private Network; NAC: Network Access Control; 2FA: Two Factor Authentication; DLP: Data Leakage Protection; FDE: Full Disk Encryption; and WAF: Web Application Firewall.

² ISO: International Standardization Organization; PCI: Payment Card Industry; OCTAVE: Operational Critical Threat Asset and Vulnerability Evaluation; NIST: National Institute of Standard and Technology; COBIT: Control Objective for Information and Related Technology; SOX: Sarbanes-Oxley Act of 2002; GLBA: Gramm-Leach-Bliley Act of 1999; HIPPA: Health Insurance Portability and Accountability Act of 1996; and FISMA: Federal Information Security Management Act of 2002.

Oversimplified as they may be, these four assertions are the heart of every risk management system. The aim of this article is not to reinvent these systems, but rather to describe their purpose in a consumable and relatable fashion, and to present their relationships to IT. After all, not everyone in IT has the time or inclination to read hundreds of pages of guidelines, best practices, or requirements documents, but everyone in IT should have an understanding of the role of risk in selecting and implementing tools—only one of which is technology—in the arms race of security.

Because of the cultural influence of IT, and our conditioning to expect symptomatic relief and instant results, as an industry we seem to be disposed to ask the question, “How does my firewall fit into this?” Precisely. Firewalls et al fit into this, rather than this fitting into firewalls. Perhaps that sounds absurd, but there is an observable and understandably natural tendency for many IT professionals to evaluate security from the perspective of the technology—to consider information security from the inside out. This tendency has a disorienting effect—one ranging from misconception to fear. Misconception is characterized by statements such as, “I run anti-virus and a personal firewall—I’m secure,” as well as questions like, “Which model firewall is PCI compliant?” Fear, on the other end of the spectrum, manifests in withdrawal or retreat from engagements involving compliance, frameworks, and metrics abstractions.

Lately, IT professionals have taken up the “security is process” mantra. This rings true. Security is, indeed, a process rather than an appliance, a technology, or a certain configuration—but before exploring the practicality of that truism; let’s examine some of its fundamental components.

ASSETS

Assets are what we are trying to protect. An asset is something of value, tangible or intangible, including our data centers, servers, customer database, PBX, network connectivity, and source code. These different types of assets have situationally different values. For example, an e-commerce site might value a primary Internet feed more highly than a retail store with a 3G backup for processing credit card transactions, but the retail store might value a single file-server more highly than the e-commerce site with its dozens of redundant servers. It is the contextual evaluation of assets that allows for a definition of objectives.

Most commonly, the primary objectives in protecting assets are as follows:

- **Confidentiality** – Only authorized parties should have access (both physical and electronic) to the asset.
- **Integrity** – The asset should be protected against physical theft, impairment, destruction, and loss, and against unauthorized electronic modification and deletion.
- **Availability** – The asset should remain accessible and resiliently able to provide its services and value.

So, what’s the relevance of our earlier question: How does my firewall fit into this? A firewall is a technological tool that can protect, primarily, the

confidentiality of an asset. Within the context of a systematic approach to security, it becomes clear that a firewall is only a single example of a technological countermeasure that can be employed to manage risk. Outside of such a context, however, the firewall has become mythically synonymous with security. While this out-of-context perception has the simple benefit of making some kind of firewall a nearly ubiquitous network component, it also has the misfortune of creating a potentially dangerous illusion of security.

Organizations must also protect less concrete assets, such as the following:

- **Reputation** – Maintaining trust, reliability, irreproachability, and consumer confidence.
- **Accountability** – Identity-based responsibility and non-repudiation.
- **Compliance** – Safeguarding against fines, lawsuits, or disciplinary actions.

As broad as they may be, these components are generally acknowledged within the IT community as defining an asset. The ambiguity starts around the other terms such as threats, vulnerabilities, and risk. These terms are frequently used interchangeably when discussing security, but they are actually rather distinct.

THE ELEMENTS OF RISK

Threats

Threats are the events we are protecting our assets against. They can be intentional or unintentional events, agents, or actions that have the potential to harm an asset, including natural disasters, botnets, chemical spills, inadvertent data dissemination, espionage, and unauthorized database access. It's best to order threats based on frequency of occurrence: Extortion by a cyber-criminal threatening a Distributed Denial of Service (DDoS) is less likely than a user downloading a keylogger. If a threat has no appreciable frequency of occurrence, it should be filtered out.

Vulnerabilities

Vulnerabilities are how those events might occur. They are the susceptibilities to particular threats through flaws or weaknesses in design, implementation, or procedure. When considering vulnerabilities, we tend to focus on technology, such as unpatched systems, insufficient input validation, buffer overflows, rogue access points, and out of date anti-malware pattern files. This view too easily neglects the near mimetic wisdom that security is a combination of people, process, and technology.

Of these three components, process controls remain the most overlooked because of their seemingly overwhelming scope and complexity. Security policy, process, and assurance requirements can easily grow to hundreds of pages, but these definitions, as part of a perpetually reiterative security process, need not be all-or-nothing. Even a short and germane process definition is more effective than none at all.

The people component garners more awareness, if only because of many IT folks' penchant for citing "user error" as the source of security breaches. This derisive, almost adversarial approach can be counterproductive, but it does expose two problems that deserve remediation through education. First is the detrimental tendency for organizations to allow their IT staffs to operate with poor or isolated understandings of business processes and workflows, and second is the frequency with which organizations under-resource ongoing IT systems and security training for their employees.

Vulnerabilities can have multiple factors. For example, we almost immediately classify an unpatched server as a technology vulnerability, but we can also consider it as a process deficiency (lacking a documented definition such as: Servers must be patched every Wednesday during scheduled downtime between 3:00 a.m. and 4:00 a.m., or within four hours of a critical patch release), or a people deficiency (lacking adequate training necessary to complete a requirement, such as Larry was on patch duty on Wednesday, but he hasn't yet been trained on Ubuntu). Whatever the classification, the point is that vulnerabilities comprise more than merely technology.

DEFINING RISK

Risk is the probability that some threat will exercise a certain vulnerability and have a negative impact on an asset. This calculation should also take into account the current set of controls called countermeasures, or what we're currently doing for protection.

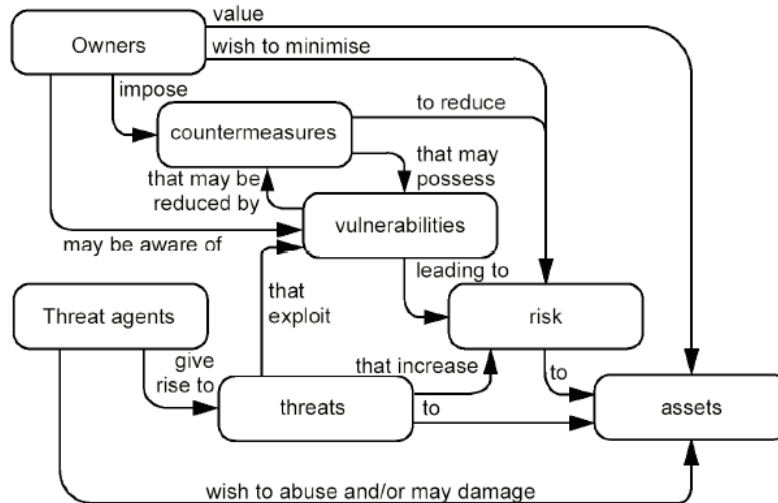


Figure 1 – Security Concepts and Relationship

The relationship between these concepts is well depicted in the ISO standards document ISO/IEC 15408-1:20053 (Figure 1), also known as the Common Criteria, Part 1, and is commonly represented by some variation of this formula:

Risk = Threat * (Vulnerability-Countermeasure) * Impact

The scale that is used in the valuation of these elements is relative, and we have many examples within the various risk management methodologies to choose from, but the essence is that it should consistently measure the frequency of the threat, the likelihood of the vulnerability, the effectiveness of the countermeasure, and the severity of the impact. To illustrate, we can use a scale of one to five (low to high) and some qualitative approximations for considering how a PC contracts a backdoor rootkit mailbot through a browser exploit:

- Frequency of Threat – 3.5
- Likelihood of Vulnerabilities – 2.7
 - User error (e.g. following a link from an unknown source) – 3
 - Unpatched browser – 3
 - Out of date anti-malware protection – 2
- Effectiveness of Countermeasures – 3.2
 - Users attend regular security training – 3
 - Users run a browser not subject to the exploit – 4
 - Client anti-virus installed and up-to-date – 3
 - Gateway anti-virus installed – 3
 - Unsanctioned SMTP activity is blocked/logged – 3
- Severity of Impact – 4
 - Rootkit makes detection / removal difficult – 4
 - Backdoor access to PC – 4
 - PC becomes a spamming mailbot – 3

Risk = Threat * (Vulnerability-Countermeasure) * Impact

Risk = 3.5 x (2.7 – 3.2) x 4 = -7 [negative]

So even if a certain threat has a high frequency of occurrence and a very high impact, if there is no effective vulnerability to the threat, then the risk could be negligible. Of course, the occurrence of the threat is still possible (for example, through some unknown vulnerability), but it is simply improbable and, therefore, likely worth accepting the risk. Similarly, if a certain threat has a high probability (for example, a high frequency combined with a high effective vulnerability) but insignificant impact, then that risk might also be worth accepting. Figure 2,

³ ISO International Standard ISO/IEC 15408-1:2005, Second Edition (October 2005). “Information Technology – Security Techniques – Evaluation Criteria for IT Security – Part 1: Introduction and General Model”.

below, from NIST Special Publication 800-30⁴ provides a simple decision chart to help determine the acceptability of risk.

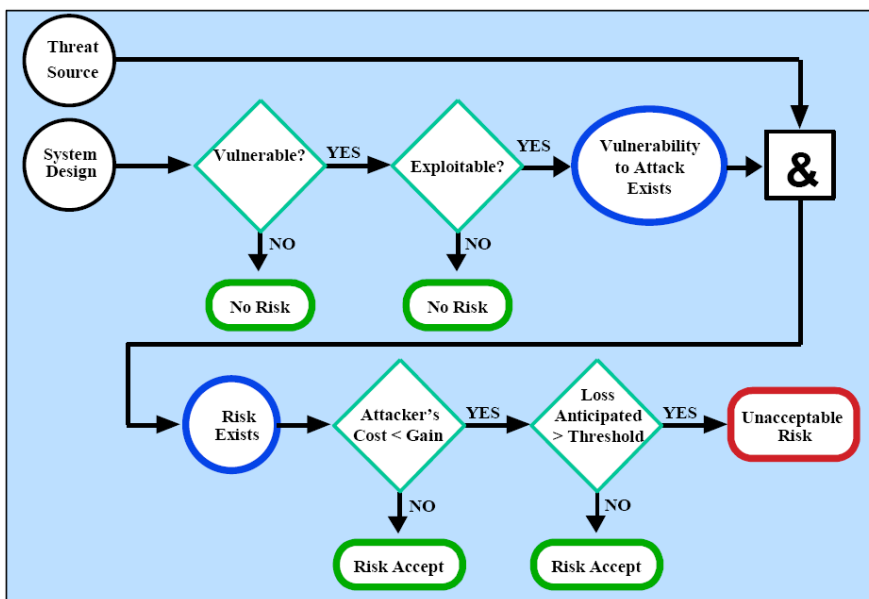


Figure 2 – Risk Mitigation Action Points

LIVING WITH RISK

With the elements of risk in mind, we can consider the opening description of risk management: the practicable mitigation of risk to an acceptable level. Despite the previous example, the idea that there can be an acceptable level of risk often seems counterintuitive; after all, wouldn't it be better to reduce every risk to its most minimal level? Given limitless resources, perhaps, but that is a luxury few have. Instead, we should invest just enough in mitigation to reduce risk to a point that we can accept. This is the notion of practicability: An investment we can afford in terms of time, money, energy and space consumption, human resources, tolerability, and sustainability. Anything in excess of this would either be untenable, or simply a wasteful pursuit of ever diminishing returns.

In addition to accepting risk and mitigating it to the point of acceptability, there are two other commonly cited methods of dealing with risk. The first, *avoidance*, is generally unrealistic, because it simply attempts to evade the risk altogether. For example, if we are worried about receiving an email-borne virus, we just stop using email. The second, *transference*, attempts to shift the risk to someone else—for example, outsourcing to a Managed Security Service Provider (MSSP), or buying network security insurance (an emerging market that will continue to grow as accountability for security breaches increases).

⁴ NIST Special Publication 800-30 (July 2002). "Risk Management Guide for Information Technology Systems".

Although risk could be measured quantitatively using estimated terms such as *Annualized Rate of Occurrence* and *Single Loss Expectancy* (particularly by those who continue to view security as an overhead expense), we possess an irresistible human tendency to evaluate risk qualitatively. For example, we naturally consider events or actions to be more risky if they have such qualities as being involuntary, beyond our personal control, epidemic, or dreaded. This is why security issues such as botnets and keyloggers tend to earn so much more attention than deficient end-user training or weak or non-existent process definitions. The effect sometimes places more or less emphasis on certain threats than they might warrant. A good example of this is the profusion of studies propounding that as much as 80 percent of security incidents come from insider threats.⁵ Even taking these figures with a dose of skepticism and entertaining that the percentages are perhaps two-fold exaggerations, the insider threat category is still one of the most formidable. Despite attempts by data-leakage protection solution providers to incite interest, the perceived danger is still too low to keep its budget and adoption at the same level as those of spyware, viruses, and spam.

While clarity and the presence of a threat alone without an accompanying sense of urgency might not be enough to spur the industry to widely adopt countermeasures, the history of IT still demonstrates that, as new threats emerge, information security vendors will respond with reactive countermeasures. When spyware became a threat, anti-spyware solutions appeared in the marketplace; when client-integrity became a threat, NAC solutions appeared; when data-leakage became a threat, DLP solutions appeared; and so on. Predictably, this trend will continue, and as new threats come to the fore, “anti-new threat” solutions will appear.

For example, let’s look at Cross Site Request Forgery (CSRF), or an attack where a victim is tricked into following a malicious link that performs an unwanted “trusted” action on a legitimate site to which they are already authenticated, begins to get the attention it deserves. Anti-CSRF solutions begin to appear, such as a “Unique Request Token Transparent Proxy Appliance.” Assuming these appliances start at \$20,000 and rise to \$100,000 based on capacity, organizations evaluating this risk must ask: Should we invest in this latest threat-mitigation technology? To answer the question, organizations must consider their existing effective vulnerability (for example, are trusted site actions performed using static URLs?) and the potential impact of exploit (for example, repudiation of actions, losses from fraudulent transactions, legal liabilities, and more).

But these organizations should also ask another question: Are there other ways for us to mitigate this risk? Often, the answer is yes. In the CSRF scenario, organizations could ask their IT staff to rewrite Web applications to always use non-predictable URLs for trusted actions. Depending on the architecture of the Web application, however, such a solution might prove infeasible. Another way organizations could address this threat would be to employ the principle of *complete mediation* (enforcing access controls on all accesses) through the relatively simple approach of re-credentialing. When an authenticated user

⁵ Chuvakin, Anton (August 2001), “Insider Attacks: The Doom of Information Security”. *Journal of Information Security*. CRC Press.

attempts to perform some kind of trusted action, the site would simply redirect the user to a page describing the action and requiring re-authentication in order to perform the action. This way, if the action were invoked through a CSRF attack, the user would know what was being attempted, and would be able to abort it. While this additional enforcement might prove a minor nuisance to users who are accustomed to single-session authentication, it is a perfectly reasonable and bearable cost for the solution it would provide.

INCIDENTS WILL OCCUR

Employing a risk management model helps to determine the thresholds for risk, a major factor of which are the resource costs (for example, direct, opportunity, and intangible) of countermeasures. But irrespective of the resources allocated to various controls to prevent threats from exercising vulnerabilities, it remains highly probable that most IT systems will experience some kind of a security incident, from something simple like a user installing spyware on a desktop, to a network-wide denial-of-service attack, to a full-blown database breach. Surprisingly, despite the inevitability of some sort of security incident, far too few operations voluntarily define an incident response process. Regulatory and compliance programs, including HIPAA, SOX, GLBA, California's Security Breach Notification Law SB 1386, and PCI, have recognized this, and all require incident response plans.

Foremost, an incident response plan helps contain the direct damage caused by a compromise. Additionally, a methodical and timely response also serves to control the indirect damage, such as injury to reputation, negative publicity, lost customer confidence, legal repercussions, and other fines or penalties. Finally, a systematic response plan helps identify and resolve the root causes of the incident so that repeat occurrences might be avoided. In general, an incident response plan should include some variation of the following steps:

1. Contain the damage.
2. Preserve/duplicate of the compromised system's state.
3. Contact law enforcement and legal agents.
4. Restore operations of compromised system.
5. Determine incident cause.
6. Document incident and recovery details.
7. Update control agents/implementation details accordingly.
8. Update incident response plan, as needed.

A good reference for incident response can be found in NIST Special Publication 800-61 "Computer Security Incident Handling Guide."⁶

We're seeing a trend toward the escalation of compliance and regulatory requirements. By this, I'm referring to activities such as the codification of the private sector's PCI Data Security Standard (DSS) standard into law by Minnesota's Plastic Card Security Act of 2007 and Texas' House Bill 3222 of 2007 (presumably the first of many states to adopt PCI DSS before it became

⁶ NIST Special Publication 800-61 (March 2008). "Computer Security Incident Handling Guide".

federal), and the Federal Notification of Risk to Personal Data Act (S.239) of 2007, which is the federal interpretation of California's SB 1386. The primary motivation for this escalation is governmental recognition that voluntary information security, on average, is ineffectual or non-existent. A secondary motivation or effect is that the regulation of a level of accountability (and by extension, liability) allows for the development of risk transference programs through traditional insurance models.

Understandably, this trend has been criticized because of the potential for "superseding" mandates to weaken existing controls. An often-cited example of this is the CAN-SPAM Act, a federal anti-spam law passed in 2003 that was largely derided because its provisions were weaker than many of the state laws it preempted. While the criticism is fitting in some cases, it should not be applied as a generality. We will encounter instances where codification strengthens controls rather than weakens them. This would be true in cases where there are no existing controls to weaken, and where the mandates are presented as benevolent requirements (which can always be exceeded by the ardent or ambitious) rather than illicit prohibitions (which, naturally, are ignored by criminals and probed for loopholes by the unscrupulous). Ultimately, yet perhaps idealistically, this practice can become even more effective as the construction of the laws matures to where they avoid the neutering effects of insufficiently restrictive specifications and sweeping exclusions or exemptions.

SECURITY IS A *PROCESS*

An earlier claim that technology employed on its own as a security solution creates a potentially dangerous illusion of security deserves some elaboration. This refers to our predilection to try to solve problems in as scant and expeditious a manner as possible. Why diet and exercise when you can just order a \$25 bottle of "Change Your Life Today?" Why go through an intensive and ongoing process of assessment, planning, implementation, and review when you can just order a \$500 firewall and a \$25 PCI scan? Why? Because the conceit that a better-built mousetrap could provide a total security solution would only cease to be delusional if we were battling mice rather than a well-organized, distributed, and motivated malware marketplace. Operating under such an illusion of security turns dangerous when the misapprehension becomes a substitute for sound practices, or worse, when it emboldens those using information technology to behave more recklessly than they might in the absence of the illusion.

Misperceptions extend to compliance and standards, as well. PCI does not certify a product as (relatively) secure, it certifies an environment. The obligation of a product within the context of PCI is primarily to not preclude the implementation of the prescriptions—so long as it can do that, then the important part is configuring it to meet functional and assurance requirements. Just because one network found to be compliant by a Qualified Security Assessor was using firewall brand X, model Y, running firmware Z, does not ensure that another network using that same device will be found compliant.

Within IT, the Common Criteria (CC) international standard (ISO 15408), is regarded as perhaps the most revered certification within the industry. Yet,

even CC is often demoted to little more than an item on a checklist. This is good because it raises the bar for product quality through the rigor and intensiveness of the evaluation, and bad because many uncritically assume that a product with a CC certification is somehow more secure than a product without a CC certification.

Any of us who have participated in or reviewed the CC process know that CC does not specify security features or requirements. Nowhere is it required to claim that a security solution will protect against denial-of-service attacks, or viruses, or rogue access points. Instead, CC allows for the vendor submitting the product to define the boundaries of the Target of Evaluation (what functional areas of the product will/won't be included in the evaluated configuration), the Protection Profile (an optionally conformed to set of features and functions that are common to a certain class of product, like an operating system or a firewall), and the Security Functional Requirements (a generic collection of malleable security related functions, like "... protects the stored audit records from unauthorized deletion"). The Security Target then wraps up these elements, defines threats and security objectives (for example, limiting access or logging), lists the assurance measures (how the target meets the designated Evaluation Assurance Level, for example, EAL-4), and presents rationale that the Target of Evaluation (ToE) provides effective countermeasures by mapping objectives to the threats, assumptions, and functional requirements. That's the simple part.

The larger effort is in the assurance requirements. *Assurance* is intended to provide documentation and testing evidence that the ToE achieves the claimed security objectives. As the evaluation assurance level increases, so does the burden of proof, as illustrated in Figure 3.⁷ Sparing the full detail, there are assurance elements that most assume to be part of security certification (functional specifications, high- and low-level design documents, administrator guidance, functional testing, and strength of security functions). But a significant part of the assurance evidence includes descriptions of the source code repositories that are used: How version control for code check-ins is performed; how access to the source code is controlled; how firmware is delivered to manufacturing; and how products are shipped to end users.

⁷ ISO International Standard ISO/IEC 15408-3:2005, Second Edition (October 2005). "Information Technology – Security Techniques – Evaluation Criteria for IT Security – Part 3: Security Assurance Requirements".

Assurance class	Assurance Family	Assurance Components by						
		Evaluation Assurance Level						
		EAL1	EAL2	EAL3	EAL4	EAL5	EAL6	EAL7
Configuration management	ACM_AUT				1	1	2	2
	ACM_CAP	1	2	3	4	4	5	5
	ACM_SCP			1	2	3	3	3
Delivery and operation	ADO_DEL		1	1	2	2	2	3
	ADO_IGS	1	1	1	1	1	1	1
	ADV_FSP	1	1	1	2	3	3	4
Development	ADV_HLD		1	2	2	3	4	5
	ADV_IMP				1	2	3	3
	ADV_INT					1	2	3
	ADV_LLD				1	1	2	2
	ADV_RCR	1	1	1	1	2	2	3
	ADV_SPM				1	3	3	3
	AGD_ADM	1	1	1	1	1	1	1
Guidance documents	AGD_USR	1	1	1	1	1	1	1
	ALC_DVS			1	1	1	2	2
Life cycle support	ALC_FLR							
	ALC_LCD				1	2	2	3
	ALC_TAT				1	2	3	3
	ATE_COV		1	2	2	2	3	3
Tests	ATE_DPT			1	1	2	2	3
	ATE_FUN		1	1	1	1	2	2
	ATE_IND	1	2	2	2	2	2	3
	AVA_CCA					1	2	2
Vulnerability assessment	AVA_MSU			1	2	2	3	3
	AVA_SOF		1	1	1	1	1	1
	AVA_VLA		1	1	2	3	4	4

Figure 3 – Evaluation Assurance Level Summary

So while CC might not certify that a product will protect against spyware, it does certify that the product was systematically protected from exposures created through outsourcing, insertions of backdoors by rogue developers, or tampering during the manufacturing process. The moral: Common Criteria deserves its revered reputation as a security certification, but perhaps for reasons other than those that are prevalent, given that its security assurances are at least as concerned with process as they are with technology.

SYNERGISTIC SOLUTIONS

Most security products provide ad-hoc point solutions to security threats: anti-virus, NAC, content filtering, and so on. While these remain essential tools to be employed in a layered security implementation, many of them focus on treating a technology concern, often neglecting the people and process components, or treating them only as a side-effect or a symptom. Acknowledging that security is combination of people, process, and technology, we should constantly be asking the question: How do we integrate the people and process elements of security into the information technologies we build so that they solve more than just technology problems?

One of the best examples of this was recently shared with me by David Raikow, security expert and senior editor of *VARBusiness*. He described the combined logical/physical access control scheme in the form of a common converged smart card system. Simply, it's a system where access to building and doors is granted when a card reader is presented with an authorized smart card, and where network access is granted only when a station-attached card-reader has an authorized smart card inserted. In addition to providing the apparent benefits of identity-based access controls, unified user management for physical and network access, and the strength of a second factor of authentication, the technology transparently provides a compounded benefit: It remedies a flawed human behavior. Specifically, it solves the all-too-common problem of people walking away from workstations without logging out. By requiring users to

remove the smart card from the workstation reader in order to have physical access to other areas of the building, it mitigates the vulnerability of an unauthorized user easily and undetectably piggybacking on unattended authenticated sessions. And the enforcement, transparent as it is, does more than just treat a symptom—it addresses the cause by retraining the user. Conditioning sounds Orwellian, but this is evidence that properly designed technologies can reinforce the people and process aspects of security.

Other illustrations, however inelegant they might be by comparison, can be readily found by examining other people or process vulnerabilities. For example, consider the common problem of failure to revoke access from employees in the event of termination. Interfaces should exist between HR management systems and user directories (AD/LDAP) so that when a termination is processed in the HR system, it automatically disables the affected account, while providing notification to HR and IT teams. There are a number of automated user de-provisioning products on the market—but whether off-the-shelf or custom-developed, this type of process control is not considered a security product, and therefore fails to get the attention and use that it should.

This approach is not new, but rather an application of the principle of Defense in Depth. Until the rapturous time when we imbue data with the ability to protect itself, as described by such visionaries as Van Jacobson and the Open Group's Jericho Forum, Defense in Depth will remain one of the better methods of protecting data. As is illustrated previously, the principle does not suggest installing multiple firewalls, but rather it advocates building a layered, complementary, synergistic approach to security. The value is neatly supported by Bayes' Theorem:

*If one layer is 60 percent effective, the system will fail 40 percent of the time. Add a second layer, 50 percent effective, and the system will fail 20 percent of the time (.4 * .5 = .20). Add a third layer, 40 percent effective, and the system will fail 12 percent of the time (.4 * .5 * .6 = .12).*

Obviously, for this to be effective, the layers must be complimentary rather than redundant. Having a primary and secondary Internet connection from the same service provider would be no more effective a security measure than running two instances of the same anti-virus client.

APPLIED PRINCIPLES

Anyone implementing any kind of system, from developers to systems integrators, should also consider Saltzer and Schroeder's eight design principles,⁸ which summarily prescribe:

⁸ Saltzer, Jerome H.; Schroeder, Michael D. (September 1975). "The Protection of Information in Computer Systems". *Proceedings of the IEEE*, **63** (9): 1278-1308. IEEE 75CH1050-4.

1. Economy of Mechanism – “Perfection is achieved, not when there is nothing left to add, but when there is nothing left to remove.” – Antoine de Saint-Exupery and WCR.
2. Fail-Safe Defaults – Access should be denied by default, and only allowed when an explicit condition is met.
3. Complete Mediation – Every access to every object should be checked.
4. Open Design – The strength of the mechanism should not come from its secrecy. Protection mechanisms should be separate from the protection keys. Don’t rely on security through obscurity.
5. Separation of Privilege – When feasible, access should only be granted when multiple conditions are met.
6. Least Privilege – Only the minimum necessary access privilege should be granted to users, programs, or any other entity.
7. Least Common Mechanism – As few entities (functions, programs, and users) as possible should share the protection mechanism to minimize the chance or extent of rejection or corruption.
8. Psychological Acceptability – Increases the chances that the protection mechanisms being used are implemented correctly.

Consider the effect that simple default-deny behaviors could have on common problems (especially where default-allow is hardly valid) such as the practice of openly allowing outbound SMTP or NetBIOS traffic from the LAN to the WAN on a gateway (particularly residential). If all gateway vendors configured the default behavior of their devices to block these classes of traffic, it could significantly stanch botnet-sourced spam and virus proliferation (and this could easily be extended to other suspect-traffic cases like inbound HTTP, HTTPS, and DNS—also common components of botnets). If ISPs were to do the same, the combination might have the effect of incapacitating a large segment of the botnet economy. So while we might see an increased support cost for vendors and ISPs to handle those cases where there is a legitimate need to allow outbound SMTP, the net effect would actually be a savings to the ISP through decreased bandwidth consumption. Further, vendors could moderate the impact by designing automated methods of detecting and handling attempts to send outbound SMTP.

Another area for more security through fail-safe design would be stricter formatting requirements on ubiquitous user applications. Specifically, I am referring to Web browsers, but it can be any application that consumes or renders data. Developers of Web browsers currently go to great lengths to compensate for poorly structured HTML. One of the effects of this is the ease with which Cross Site Scripting (XSS) attacks can be launched. Briefly, XSS is a vulnerability that exists in many Web applications wherein an attacker can inject malicious content into an otherwise innocuous Web for later viewing by (and targeting of) visitors to that page. Website developers are generally faulted for XSS vulnerabilities where insufficient content validation and sanitization allows the attacker to inject the malicious code. While this is certainly a problem, a visit to the “XSS Cheat Sheet” (found at the ha.ckers blog: <http://ha.ckers.org/xss.html>) quickly illustrates the boundless lengths to which a Web developer would have to go to validate and sanitize all effective permutations of injection. While this won’t stop XSS attacks, validation efforts would be much simpler and, thus, more successful if Web browsers didn’t accommodate errors such as errant line-breaks, missing delimiters, or unclosed

tags. And if the default behavior of the HTML parsers in the browsers was intolerance of malformed data, it would emphasize the need for “good content” to be properly formed and validated—thus reinforcing good development habits.

PERSPECTIVE SHIFT

The goal of implementing information or risk management systems and of compliance should not be to be compliant, but rather to be secure. Producers and consumers of information systems must adopt security as an ideology, and must apply this attitude in natural and continuous practice. So long as the perspective toward security remains something costly, inconvenient, and onerous that we must do, it will never achieve pervasiveness. It will be scarcely more than a punitive afterthought performed to a minimally acceptable and marginally effective level.

About the Author

Joe Levy, Solera Networks CTO, is a recognized expert in networking and network security and has nearly fifteen years of experience in the industry. He is known for guiding SonicWALL through many company-defining moments in his role as CTO. He has participated in various industry certification and design consortiums including the ICSA and IETF, and, along with his development teams, has authored a number of networking and security patents. Prior to SonicWALL, Joe spent six years with OneNet, Inc. where he served as VP of Technology Services, directing the Professional and Managed Services teams. Levy has authored his first book with several colleagues on the topic of wireless network security, covering many of the proprietary wireless enhancements designed by Levy and the SonicWALL architectural team, as well as industry standards best-practices. Levy's well-known "Worth a Glance" blog covers his views on a broad range of network security news, issues, and trends, and can be found at <http://blog.illurity.com>.

About Solera Networks

Solera Networks' DS Series is a line of high-performance network forensics appliances, including software-only virtual appliances, which capture, record and archive 100% of network traffic at speeds up to 10Gbps. The data is then accessible instantly via Solera Networks' search, alert and archive interfaces, or via any standards-based security, forensics, compliance, analytics or network management application. For more information on Solera Networks, visit <http://www.soleranetworks.com>.