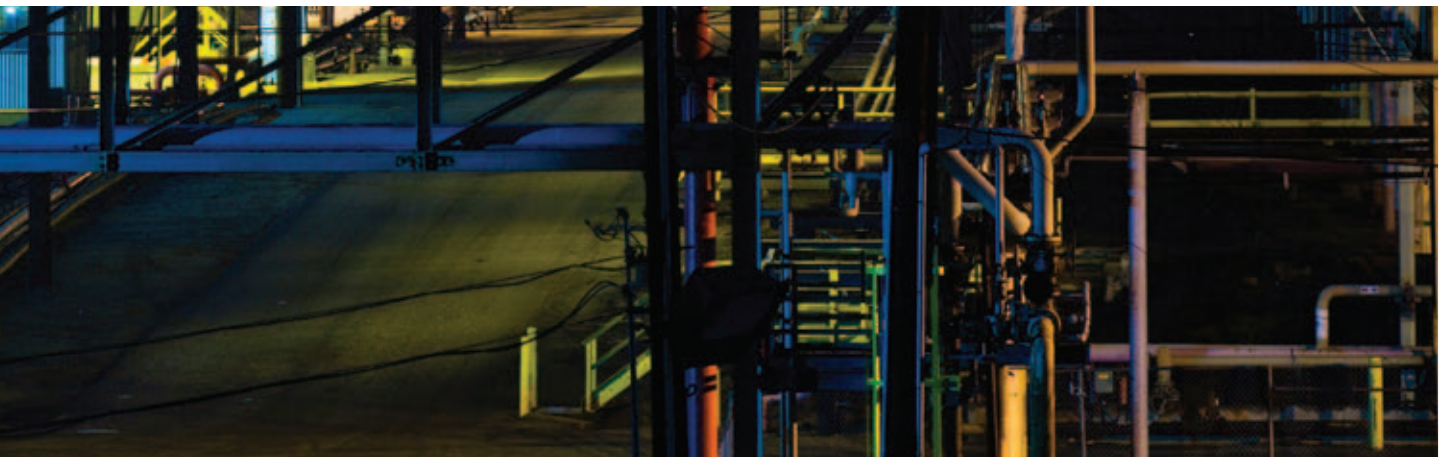
Confronting Cyber Risk in Critical Infrastructure

The National and Economic Benefits of Security Development Processes



GOOD HARBOR
CONSULTING, LLC

Table of Contents

Microsoft commissioned Good Harbor Consulting, chaired by former White House cybersecurity advisor Richard A. Clarke, to research and write this report. The report was prepared by Jacob Olcott, a recognized cybersecurity expert who served in senior professional staff positions in the United States Senate Committee on Commerce, Science, and Transportation and the House of Representatives Committee on Homeland Security, with the support of William Howerton and Emilian Papadopoulos. The team interviewed experts in critical infrastructure security, industrial control systems security, application security, cyber and national security policy, and code analysis to assess their opinions on the state of application security in critical infrastructure environments, its impact on the public and private sectors, and the priority of issues for public discussion. The opinions and conclusions expressed in the report are solely those of the authors or those quoted for the report. The report does not necessarily represent the opinion or position of Microsoft Corporation. The report is available online at http://www.goodharbor.net/media/books_and_reports.php#reports

Executive Summary

In recent decades, critical infrastructure has become dependent on complex software applications to perform vital societal functions that include energy distribution, banking and finance, and transportation. A cyber event affecting critical infrastructure systems or assets could not only harm the business but also potentially impact public health and safety, the economy, and national security.

The responsibility of ensuring safe and secure functioning of these increasingly complex systems has typically rested solely with critical infrastructure providers. Efforts to secure and defend networks were tactical and operational and largely consisted of the deployment of defensive technologies, including firewalls, antivirus, and intrusion detection systems. Far less attention was paid to the underlying code that makes applications vulnerable to begin with. Over the past decade, however, some application developers and vendors have begun accepting responsibility for building security into their development processes.

Since the launch in 2002 of Microsoft's Trustworthy Computing Initiative, developer efforts to build more robust applications have increased. Market pressures, driven both by 1) increased customer awareness and demand and 2) negative market sentiment associated with exploitable products, have led many developers, including critical infrastructure vendors, to utilize a security development process. Additionally, many developers can show real financial benefits by investing in a security development process. Systematically addressing security throughout the development process prevents late discovery of vulnerabilities that may significantly delay product release. Moreover, development teams are better poised to issue patches and updates subsequent to product release.

While market leaders are now systematically building security into the application development lifecycle, the practice needs to be adopted more broadly, particularly for applications used in the critical infrastructure environment. As end user and developer awareness of the operational and economic benefits of security development increases, there is reason to believe that adoption of widely accepted best practices will grow, minimizing the number and impact of vulnerabilities in the critical infrastructure application environment.

Introduction

Modern civil society is dependent upon critical infrastructure: the systems and assets that provide vital services like water, electricity, communications, and banking.

Like all modern businesses, critical infrastructure owners and operators have taken advantage of advances in information technology to reduce operational costs, increase productivity, and create new efficiencies. However, the deployment and networking of information technology in the critical infrastructure environment also has brought new risks. Intentional and unintentional cyber events can and do result in significant financial, operational, and reputational harm to businesses of all types. A cyber event affecting critical infrastructure systems or assets could not only harm the business but also potentially impact public health and safety, the economy, and national security.

What is "Critical" Infrastructure?

"Systems and assets, whether physical or virtual, so vital to the United States that the incapacity or destruction of such systems or assets would have a debilitating impact on security, national economic security, national public health or safety, or any combination of those matters."

U.S. Public Law 107-296

The Department of Homeland Security has designated 18 infrastructure sectors as "critical":

- Agriculture and Food
- Banking and Finance
- Chemical
- Commercial Facilities
- Communications
- Critical Manufacturing
- Dams
- Defense Industrial Base
- Emergency Services
- Energy
- Government Facilities
- Healthcare and Public Health
- Information Technology
- National Monuments and Icons
- Nuclear Reactors, Materials and Waste
- Postal and Shipping
- Transport Systems
- Water

Information technology developers and end users alike share the responsibility of managing and mitigating cyber risk. As Ed Amoroso, Chief Security Officer at AT&T, notes, "when it comes to cybersecurity, there must be a balance between the obligations of a developer and a provider of critical infrastructure."[1] One way that software developers and critical infrastructure owners and operators can mitigate cyber risk is by building and buying applications created using a meaningful security development process, a process whereby potential exploitable vulnerabilities in software are systematically addressed throughout development. While reducing vulnerabilities is a clear benefit of using a security development process, there is growing evidence that investing in a security development process can also create operational and economic efficiencies for developers and end users alike. This report, *Confronting Cyber Risk in Critical Infrastructure: The National and Economic Benefits of Security Development Processes,* highlights the efforts of some critical infrastructure software application developers who are realizing the benefits of a security development process and provides recommendations and resources for developers and end users. *Confronting Cyber Risk in Critical Infrastructure* also contains recommendations for policymakers.

Growing developer awareness of cost savings and consumer demand for security development makes it likely that, over time, an increasing number of application developers will respond to market pressures and incentives to implement security development processes. The acceleration of this trend would more rapidly increase the security of critical infrastructure against cyber attacks.

The Evolution

Reliance on Software Applications

Most of what this paper defines as critical infrastructure has been present in one form or other for a century or more; it is only recently that critical infrastructure became dependent on information technology, including software applications.

Like most businesses, the average critical infrastructure information technology environment today uses software applications to manage client-side business processes (customer care, human resources, data compilation and analytics, accounting/billing, word processing) and customer-facing web services (online bill-pay). In addition, however, they also use software to control sensitive processes and physical functions.

Today, the average critical infrastructure application environment is a complex amalgam of networked applications created by **internal** and **outsourced** developers, including commercial vendors, integrators, and developers who provide unique, proprietary solutions. While the dependence on software is prevalent in every critical infrastructure sector, the provenance of software varies by industry. In some industries, such as electric power, a small number of developers supply the control software throughout the world, creating substantial homogeneity. In others, such as the banking and finance sector, many companies develop a significant amount of applications in-house. In general, however, most critical infrastructure companies rely heavily on vendors for their software application needs.

The software development ecosystem is further complicated by the fact that many developers themselves rely on external developers to create and service products. This was not always the case. Decades ago, large critical infrastructure companies with significant internal engineering organizations were more likely to develop customized applications entirely in-house. Control system applications, for example, were once largely homegrown. Programming languages were not extremely complex, and for many engineers application development was a significant aspect of the job. With the growth of the information technology industry, however, the complexity of the underlying code and the displacement of custom hardware solutions by sophisticated software solutions dramatically increased the reliance on outsourced developers in many of the process control industries.

Modern businesses also take advantage of the cost savings and efficiencies created by networked communications, and critical infrastructure companies are no exception. Companies that use process control technologies, for instance, rely on short and long-range communications capabilities to operate and service remote facilities. In recent years, connections to Internet Protocol-enabled equipment and instrumentation has provided greater visibility and control into all aspects of a critical infrastructure environment, allowing ease of communications, repairs, and monitoring of businesses operations in a way that was unimaginable decades ago. The convergence of information and process control technologies was not without complication, however, as it created tensions between engineers and IT professionals, two groups that had previously operated independently.

Convergence Creates Tensions Between Engineers and IT

The growth of information technology in the critical infrastructure operating environment created tension between the engineering and information technology communities, particularly in sectors that use process control systems. In the early days of convergence, rivalries "between control engineers and administrators began immediately," recalls Bryan Owen, cybersecurity manager of OSISoft, a control systems application development firm. "Engineers used to give presentations at conferences explicitly on 'How To Keep IT Out' of control systems operations." Both sides had different backgrounds and technical understandings, and disagreed on the use of IT in critical environments. Although the relationship between the two communities has improved, "much work remains to be done," notes Eric Cosman, Co-Chair of the Industrial Automation and Control Systems Security Section of the International Society of Automation.

With the Switch Came Risk

The Hidden Connections

As critical infrastructure companies increasingly adopt software applications and networked computing to perform essential functions, the risk of harm from an intentional or unintentional cyber incident, from worms and viruses to denial of service attacks and targeted exploitation, steadily increases.

Growing complexities of applications can create unintended vulnerabilities in systems that can compromise the confidentiality, integrity, and availability of critical infrastructure resources. As AT&T's Ed Amoroso observes, "the complexity of our infrastructure and its dependency on software increased as software became embedded. This increased complexity and dependency has significantly heightened our application security risk."[2]

External actors are increasingly targeting the application layer in their attacks against business networks. Threat surveys indicate that the majority of exploitable vulnerabilities in networked systems are now found in the application layer, and the most likely point of compromise of a networked system is through exploiting an error in the source code of an application. In the *2012 Data Breach Investigations Report*, Verizon Business and the U.S. Secret Service noted that web application hacking was the primary hacking attack vector of organizations of at least 1,000 employees, accounting for 54 percent of incidents and 39 percent of compromised records.[3] Additionally, Microsoft's *Security Intelligence Report* notes that vulnerabilities in applications are substantially more widespread than vulnerabilities in operating systems or web browsers.[4] Chris Wysopal, co-founder and Chief Technology Officer of application security testing company Veracode, suggests that the same shift is evident in attacker trends in recent years. Previously, attackers focused on stealing personally identifiable information by attacking web applications, but they have shifted focus to using the application layer as a way to breach network security perimeters to cause greater damage.[5] An October 2011 report by the Office of the U.S. National Counterintelligence Executive (NCIX) warned that malicious actors would increasingly exploit complex software as a way to gain access to corporate networks.[6]

Increased network connectivity between systems can increase risk to critical infrastructure owners and operators who use process control systems. Sensitive applications once designed to run process control systems on closed, proprietary networks are now, in many cases, logically connected to corporate and public-facing networks.

Although corporate officials appear often to believe that their internal control networks cannot be accessed from the public Internet, that is a false assumption in the vast majority of cases. Appearing before the House of Representatives in May 2011, Sean McGurk, National Cybersecurity and Communications Integration Center Director at the Department of Homeland Security, testified that during vulnerability assessments for critical infrastructure owners and operators, the agency has *always* discovered connections between the enterprise network and the operations network:

"In our experience in conducting hundreds of vulnerability assessments in the private sector, in no case have we ever found the operations network, the SCADA system or energy management system separated from the enterprise network. On average, we see 11 direct connections between those networks. In some extreme cases, we have identified up to 250 connections between the actual producing network and the enterprise network." [7]

Sometimes these critical systems are even connected directly to the Internet. At a conference in January 2012, a security researcher presented research finding over 10,000 industrial control systems connected to the public Internet.[8] These connections subject critical infrastructure owners and operators to the same kinds of operational disruptions caused by worms, viruses, and denial of service attacks that other business face, with potentially graver consequences. In November 2011, for instance, the Federal Bureau of Investigation stated that it had discovered that critical infrastructure networks in three U.S. cities had been remotely access by hackers who maintained the capability to cause power outages and disrupt water and sanitary services.[9]

Even infrastructure that is isolated by design is vulnerable. Solomon Tessema, former Chief Information Security Officer at Southern California Edison, believes that "traditional security through air gaps and obscurity is no longer possible."[10] The Stuxnet worm illustrated this phenomenon in dramatic fashion when it reportedly damaged industrial control systems after being introduced through an infected thumb drive that bridged the air gap and propagated through the local area network.

Critical infrastructure owners and operators are on the front line in the battle to keep their systems secure. According to Michael Assante, former Chief Security Officer at the North American Electric Reliability Corporation, critical infrastructure owners ought to think of their networks as "contested territory."[11] One thing is certain: the contested territory is expanding. Smart grid technology in the energy sector, for instance, represents a revolutionary way to manage energy efficiency and introduce new sources of energy into the grid. This includes adding digital communication technology, sensor, and control devices to the electric grid and home environments. However, it also opens up multiple new attack vectors, and expands the network perimeter of an electric company from the control room to every single home with a smart meter. The growing use of mobile devices by businesses of all types, including critical infrastructure owners and operators, also presents new opportunities for attackers and increasingly difficult challenges for network defenders.

"You have to know that the bad guys are going to get inside. There will never be another war where critical infrastructure is not a target for cyber attack."

Dickie George, former Technical Director of the Information Assurance Directorate, National Security Agency

Defending with Band-Aids

Outdated, Patchwork Defenses

As cyber risks to critical infrastructure grow, many end users focus their efforts on operational and tactical methods to mitigate application security risks.

Operational methods in this context means protecting existing applications through network-level defenses, including firewalls, antivirus software, data encryption and data loss prevention tools, and intrusion detection systems. Many end users, particularly security staff at smaller businesses with more limited resources, are most focused on these pressing operational or tactical security matters, in part because many regulatory and compliance mandates specifically target these types of efforts. This is particularly true among owners of critical infrastructure. According to Eric Cosman, smaller operators of critical infrastructure "are more concerned about operational security right now,"[12] a sentiment echoed by Michael Assante, who believes that "many [critical infrastructure] owners are focused primarily on operational issues."[13]

While operational security initiatives play a crucial role in mitigating risks to applications, there are inherent limitations to defensive technologies

Regulatory Focus on Operational Issues Can Be Limiting

Some commentators are concerned about the role that regulations play in emphasizing operational security (actions taken by end users to defend applications and networks) over strategic risk management approaches, including secure application development. Indeed, as Forrester noted in a 2011 survey, "whenever compliance is the main driver, organizations tend to do the bare minimum needed to become compliant, rather than focusing on best practices and long-term objectives."

There are some compliance regimes – like the private sector-driven Payment Card Industry Data Security Standards (PCI-DSS), which establish mandatory cybersecurity requirements on merchants who accept credit cards – that do contain requirements for entities to develop and maintain secure systems and applications. PCI-DSS Requirement 6, for instance, contains specific requirements for merchants to follow a secure application development process.

that have led some to conclude that deployment of these systems alone cannot adequately address the threat. First, the most commonly deployed defensive technologies rely on exclusion list logic, which tests against a known set of malicious vectors. In the context of defensive technologies, this results in protection against known attack methods targeting specific assets but will not stop unknown attacks or attacks designed to avoid the exclusion list. Second, defensive technologies can only protect known assets. Contemporary network environments, particularly in critical infrastructure, contain a degree of complexity that renders full mapping difficult if not impossible. As a result, the known perimeter of a network may be difficult to determine, thereby limiting the effectiveness of defensive technologies. Third, operational initiatives do not address the core weaknesses that allow attacks to occur in the first place. Aggressive deployment of defensive technologies without a commensurately serious approach to improving the underlying code is like "securing a grass hut with a steel door," says Doug Cavit, an executive in Trustworthy Computing at Microsoft.[14] The obvious point of entry may be secure, but the structure itself remains vulnerable. As Richard Clarke, former White House cybersecurity czar, notes, "the strategy of hiding the application as best you can instead of dealing with the weaknesses inherent in the application isn't good risk management. It's a recipe for failure."[15]

Assessing the Underlying Problem

Given the inherent shortcomings of operational security initiatives, strategic application risk management efforts designed to improve the inherent security of the underlying application deserve increased attention by developers and end users alike.

A strategic approach to addressing application risk integrates security practices into each phase of the application development process. This approach begins with **training** development teams to stay educated on security basics and recent trends. Subsequently, developers establish security and privacy **requirements** for the application to act as benchmarks against which code can be measured. They also conduct risk assessments that identify functional aspects of the application requiring in-depth review. During the **design** phase, teams set the security and design specifications to meet the previously identified standards and also model threats to parts of the application with meaningful security risks. In the **development** phase, teams use approved tools and functions and employ static code analysis to ensure that security requirements are respected. During the **testing** phase, teams perform dynamic analysis based on risk areas identified, test the application, and review the

Elements of a Strategic Approach to Application Risk

The Microsoft Security Development Lifecycle (SDL) is an example of a strategic approach to application security. The SDL incorporates the following phases:

1. **Training:** Core security training; secure coding techniques; security testing
2. **Requirements:** Establish security requirements; create quality gates; security/privacy risk assessment
3. **Design:** Establish design requirements; analyze attack surface; threat modeling
4. **Development:** Use approved tools; deprecate unsafe functions; static analysis
5. **Testing:** Dynamic analysis; fuzz testing; attack surface review
6. **Distribution:** Incident response plan; final security review; release

Limits of Tactical Approaches to Code Review

Like operational defenses in a network environment, the operational tools utilized by developers to detect errors and vulnerabilities also have their own shortcomings. Developer tools can have significant value for an organization but are bounded by a tester's expertise, the tool's capabilities, and the scope and time of the review.

application threat models. Finally, prior to **distribution**, teams develop incident response plans that detail how to remediate exploitable vulnerabilities discovered once the application is in the field. They also subject the application to a final security review.[16]

Unfortunately, the use of strategic security techniques by developers today remains largely immature.[17] Dickie George likens application development to "the Wild West, and there is no sheriff in town."[18] Market surveys echo this sentiment, including a series of reports from Forrester Research from 2009 to 2011. One of the key conclusions of the Q3, 2009 *TechRadar for Application Security* report asserts that many developers remain focused on the most tactical application security measures, including penetration testing and application scanning, while adoption of preventive and strategic measures like secure architecture design and code-level analysis lagged behind.[19] In a subsequent security survey published a year later, Forrester found that little had changed; only 12 percent of respondents had adopted code-level analysis technologies, and 16 percent reported the use of security architecture consulting services.[20] Thereafter, the 2011 *State of Application Security* report revealed that 47 percent of respondents did not perform acceptance tests for third-party code, 30 percent were not using static analysis or manual code review, and 27 percent were not practicing threat modeling and usage scenario review.[21]

Flaws in Conventional Wisdom

Is It Really More Expensive?

Conventional wisdom suggests that developers do not implement security development techniques because they are cost-prohibitive and time-intensive.

It is believed that internal and third party application developers are incentivized to build applications quickly and fix security vulnerabilities later because any code delay in the development cycle will increase production costs and compromise completion dates.[22] End users sometimes reinforce this message by purchasing products from developers who do not use a fully transparent security development process. Some have concluded that the current market simply does not incentivize developers to incorporate security into their processes.

There is, however, growing evidence to suggest that implementing a security development process does provide real economic benefit. As with most design and engineering disciplines, early vulnerability mitigation lowers the overall development cost of a product. By considering security throughout the lifecycle, vulnerabilities can be fixed earlier in the process and are less likely to create significant, costly delays because they can be more easily corrected. A Microsoft/iSEC Partners report from 2009 suggests that investment in security throughout the development process is an investment that pays dividends over the development lifecycle. According to the report, development teams will find that "incorporating proactive security efforts at each phase adds a cost to the schedule, but that cost is predictable and is frequently quite small. Security issues that flow down through several levels increase in complexity rapidly...and the effect of a single architectural vulnerability may be larger than the total cost of several preventative security practices."[23]

Security development processes create more predictability in development, allowing developers to avoid "show-stopping vulnerabilities" later in the lifecycle, which can be very expensive for developers to fix.[24] For instance, a study performed by the National Institute of Standards and Technology (NIST) found that fixing an application vulnerability at the "acceptance" stage costs a developer up to 30 times more than if it was fixed during the design of the product.[25] Indeed, executives at Oracle Corporation have noted "it is very much in our interest to build robust, secure software, because it is enormously expensive for us to fix defects – especially security defects – after we ship software."[26] The ability to save money and to deliver a product in a timely manner has clear benefits not only to developers, but also to the end users who are paying for the products.

Investing in strategic application risk management has been shown to produce a stronger return on investment for developers than efforts focused solely on tactical security initiatives deployed late in the development process. According to a 2011 study by Aberdeen, companies that incorporate security throughout the development process, as compared to those that waited until the end of the process to

Benefits of Secure Development to End Users

The benefits of early vulnerability mitigation implemented by developers also extend to end users. One security officer at a Fortune 500 critical infrastructure company put it simply: "if you're buying a product that is inherently more secure, it follows that you will spend less time on integration, configuration, and engineering to add security after the fact."

Leading The Field: Fixing Vulnerabilities

According to Aberdeen Group, "best in class" companies remediate nearly 90 percent of vulnerabilities before deployment versus "laggards," who only remediate around 75 percent of vulnerabilities.

perform technical reviews using application-testing tools, realized four times the return on their annual investments in application security, higher than those who adopted strictly operational approaches.[27] Other studies suggest that the amount of time developers who use security development processes spend on correcting and remediating bugs in the post-development phase is less than those who do not use a security development process.

Mitigating vulnerabilities during the development process also saves the developer money over the lifetime of the product. The 2011 Aberdeen study compares proactive and reactive security expenditures. That study estimates that the average cost of remediating an actual application security-related vulnerability is around $300,000 *per incident*, but the average annual investment that developers make in strategic application security initiatives, including people, processes, and training, totals $400,000.[28] When faced with the prospect of investing in security development up-front versus having to pay significantly more later, the best developers choose to mitigate vulnerabilities earlier using a security development process, in part because it is more cost effective.

OSISoft's Transformation

Indeed, commercial application developers of varying sizes in the critical infrastructure environment are already realizing the benefits of a security development process.

The experience of OSISoft, a privately held control systems data management application development company of 800 employees, offers an interesting case study. In the early 2000s a handful of senior engineers from the company consulted with security experts at Idaho National Labs (INL) on the role of application security in networked control systems. In a full-scale simulation, researchers from INL demonstrated that even in environments with "demilitarized zones" between corporate and control system networks, exploiting errors in applications could lead to significant disruption of control systems. This experience was eye opening for the OSISoft engineers, who started to think about balancing a more secure coding practice with the business imperatives of bringing new products to market in a timely fashion.

The OSISoft experience at INL happened shortly after the launch of Microsoft's Trustworthy Computing initiative in 2002. OSISoft realized that in the complex network environments of control systems, their applications would only be as secure as their partners, and subsequently adopted Microsoft's Security Development Lifecycle (SDL) process and modified it for their unique needs in 2005. The timing was right: OSISoft began to find itself in meetings with end users who asked explicitly what their measures were for strategic application security, whether they were outsourcing development, and if they were using a security development process. Using the SDL allows OSISoft to transparently describe their process to end users, avoiding unnecessary auditing and code reviews. As part of their SDL practice, OSISoft now sends key developers to receive intensive training on a variety of security issues. According to Bryan Owen, cybersecurity manager for OSISoft, secure application development "can't just be lip service. Customers place a great deal of trust in our software. We have everything to lose by not building security into our product."[29]

Trustworthy Computing: Microsoft's Security Development Lifecycle

As malware proliferated in the late 1990s and early 2000s, users of Microsoft software increasingly found their systems exploited. Widespread infection rates led to widespread customer dissatisfaction, and Microsoft's reputation suffered accordingly

While individual, operational efforts to perform basic security checks on products prior to release existed at Microsoft, there were no mandatory programs, or tests comprised of a holistic security process.

Recognizing the importance of making trustworthy software for consumers, then-CEO Bill Gates wrote a memo in January 2002 outlining the company's goals for "Trustworthy Computing." The memo defined Trustworthy Computing as "computing that is as available, reliable and secure as electricity, water services and telephony" and made this the top priority for the company.

In response to this challenge, an internal team developed a plan for a mandatory security process that would be integrated into the development of all Microsoft products. The result, the Security Development Lifecycle (SDL), has been in place since January 2004.

The SDL is a 16-part process that covers application development from start to finish and is currently in its ninth iteration, SDL 5.1, released on October 1st, 2010.

Microsoft is now recognized as an industry leader in secure application development. The incidence of "critical" flaws has significantly decreased in Microsoft products, and the SDL is regarded as the industry standard process for secure development. In order to promote improved coding generally, Microsoft offers a free version of the SDL to the public. Other leading software developers, notably Cisco and Adobe, have recognized the value of the SDL and have adapted it to fit their unique needs.

Invensys

A Developer's Experience

The experience of Invensys, a publicly traded control systems application development company with over 20,000 employees, follows a similar narrative.

In 2002 the senior management of Invensys came together to form a strategic corporate agenda to address cybersecurity. A comprehensive code review of their applications suggested that the company faced challenges in improving the security of their existing installed products, as well as those under development. Based on this review, Invensys created an application security practice with the mission to ensure that Invensys applications were subject to systematic and holistic security development procedures. Realizing that they were not in the position to build unique security functionality for control systems, the company partnered with Microsoft to develop their security development program.

While Invensys had employed elements of the program previously, in 2008 the company officially adopted the SDL methodology as its application security risk management framework, further refining it each year. According to Ernest Rakaczky, Program Manager for Control Systems Cybersecurity at Invensys, "every product has to go through the SDL." Additionally, the company invests heavily in training of its developers. "All of our developers around the world are now in continual training programs," says Rakaczky. "They come in with the skills to write code, but we help train them in security."

"All of our developers around the world are now in continual training programs… They come in with the skills to write code, but we help train them in security."

Ernest Rakaczky, Program Manager for Control Systems Cybersecurity at Invensys

Security Development as Market Differentiator

Some commercial vendors are beginning to recognize that the use of a security development process is a potential market differentiator from developers who do not utilize similar procedures. As Mary Ann Davidson of Oracle notes, "if you build buggy, crappy software that performs poorly and is expensive to maintain, you will lose customers to competitors, who love to point at your deficiencies if customers have not already found them." According to a security officer at a major chemical manufacturer, "what's changing now is that with more press coverage and more general awareness, vendors are saying that they don't want to be the last in line – they don't want to be lowest guy on the pole."

This is a particularly important message for small and medium sized software development businesses to understand, because larger companies are leveraging secure application development as a key criterion during the vendor selection process.

Invensys is currently in the process of collecting data to make precise investment return calculations on its application security program, but the company believes in the value of their effort and in its importance. Rakaczky believes that "there really was no way for us to get better without the SDL. Because we are writing our own code, we need to do everything that we can to give it due diligence. But I can go to bed at night knowing that our applications are going through a process that creates best-in-class products."

Invensys is also intensively reviewing existing code to develop best practices for future generations of applications. The company believes that their security development process is a market differentiator and that they will realize financial benefit from its implementation with the release of applications that are more secure, of higher quality, and positioned to meet their customers' procurement requirements.[30]

MidAmerican Energy

An Operator's Experience

Commercial vendors are not the only developers who benefit from using a security development process.

The experience of MidAmerican Energy Company suggests that critical infrastructure owners and operators with their own internal application development teams have experienced real gains from implementing a security development process. In 2008, MidAmerican experienced a series of Structured Query Language (SQL) injection attacks on an old webpage that inserted malicious code that propagated throughout the site. The site had to be taken down in order for the vulnerability to be remediated and for guidance to be issued to other portfolio companies that were similarly exposed. Despite significant efforts to manually correct the vulnerability, the same attack exploited a single page that escaped review. As a result, MidAmerican issued a review of over 900,000 lines of proprietary code running applications across their network infrastructure, including websites, customer care, energy trading applications, and even the systems involved in

"This isn't just an issue for the big players. If you're not using a security development process today, you're putting your company and your customers in jeopardy."

Richard Clarke

power generation and distribution. In order to properly manage the process, the company adopted a variant of the SDL, which gave them a framework to model threats and design their applications accordingly. MidAmerican now subjects all their proprietary code to the SDL, building security into the application. While precise figures are not available, estimates place the company's realized gain in productivity for technical staff at up to 20 percent.[31]

Although implementing a security development process requires financial investment, most developers suggest that the biggest change required is in the culture and business processes of the development organization. Changing the culture means holding developers accountable for security. In many development organizations, however, secure coding is not an evaluated skill; a 2011 report by Forrester Research found that nearly three-quarters of developers are not measured with security-related metrics.[32] Dan Brewer, a development manager for MidAmerican, called the code review process "bracing... to say the least. In the beginning it was easy to dump on the developers whose code needed to be refactored. But the issues were so widespread, and the cultural change required was so deep, that before long all developers saw their code flunk one security review or another."[33] A Chief Information Security Officer of an international financial services company confirms that reforming the organizational culture is an important step in implementing a security development process: "success starts with the developers... not in punishing them or in trying to change their incentives but in enhancing their skill sets. It's about deputizing them to care about security in the code they build."[34]

The Public Policy Concern

For developers and end users, adopting a strategic approach towards security development may not only generate real cost savings, but it can send a positive message to policymakers concerned about the national security impact of malicious attacks against key owners and operators of critical infrastructure.

Some critics consider software developed without due regard for security to be faulty products with negative effects from which the public should be in some way protected. U.S. Congressman James R. Langevin, a respected voice on critical infrastructure cybersecurity issues, has been "deeply concerned for years about the growing cyber risk to critical infrastructure, and what many of us on the Hill perceive as a lack of urgency on the part of developers and end users in taking responsibility to reduce these risks."[35] Many have expressed concern that the market does not adequately incentivize investments in cybersecurity or penalize the haphazard efforts of developers and end users. To address these and related concerns, the U.S. Senate and House of Representatives are considering legislation creating new cybersecurity regulations for certain owners and operators of critical infrastructure.

Recognizing the security and economic benefits, some critical infrastructure sectors have developed initiatives in recent years designed to spur greater adoption of security development processes. For example, in February 2012, BITS, a division of the Financial Services Roundtable, released a Software Assurance Framework for software used within the financial services industry.[36] The Framework provides a high-level outline of the various components of a mature, strategic program for secure software development. It can be applied to software developed directly by financial institutions or by third parties for the use of financial institutions. In its 2011 *Roadmap to Achieve Energy Delivery Systems Cybersecurity*, the Energy Sector Control Systems Working Group alludes to near- and mid-term goals for the sector for safe code development and software assurance.[37]

Even in the absence of new regulations affecting end users, there is reason to believe that the financial benefits and increasing market pressure will incentivize more developers to adopt secure application development processes. Developers who consider the cost savings of utilizing a security development process and increased demand for security development from end users, along with the potential costs of new regulations, will likely conclude that the proactive adoption of security development policies provides their company with a clear and obvious financial benefit. Policymakers have offered encouragement for developers to initiate these efforts independent of regulatory mandates that might compel such activity. Indeed, as Congressman Langevin adds, "those companies who are implementing an application security program are taking an important step towards better security, and I hope that more vendors and end users move in that direction."[38]

Do Some Government Policies Unintentionally Create New Cyber Risk?

Policymakers should be wary about creating unintended cyber risks with the ambitious adoption of new technology policies and programs in critical infrastructure. Smart grid technology, for instance, has the potential to deliver electricity more efficiently using computer-based and remote technology. Recognizing the importance of this technology, government leaders at the federal and state level developed policies and provided billions of dollars in grants to encourage private sector adoption across the United States. But, according to the Department of Energy Inspector General's Audit Report: The Department's Management of the Smart Grid Investment Grant Program, released in January 2012, grant recipients often did not adequately demonstrate cyber risk mitigations prior to their receipt of an award. In some cases, the accelerated process required developers and end users to work on implementing technology solutions far before the existence of mature products, security requirements, or standards.

Shortcomings in Higher Education

Policymakers can play an important part in aligning incentives for security development, perhaps nowhere more effectively than in supporting secure programming initiatives at educational institutions.

Industry executives and security experts alike have become increasingly vocal in their frustration with what they perceive to be a lack of security-aware computer science graduates. In a disturbing trend, it is possible today to get a computer science degree from many of the finest universities in the nation without ever having studied computer security or secure code development. Policymakers may be able to use existing grant programs to incentivize computer science departments to integrate secure coding into their curricula and evaluate students on secure coding practices. Though government-sponsored cybersecurity research is traditionally a scientific and technical discipline, research into the economic benefits of adopting cybersecurity measures like security development are equally valuable in helping make the business case for security investments.

A Trend in Industry Toward Strategic Security

As more developers and end users realize their value, efforts to implement strategic security development are likely to increase in the years to come.

According to an April 2011 Forrester Research report, nearly a quarter of respondents in 2010 indicated that their investments in secure application design would increase between 5 and 10 percent in 2011, a marked increase from the 15 percent of respondents who fell into this category in 2009.[39]

Organizations seeking to implement a strategic security development process have a variety of resources available to them at no cost. Distinguishing between the types of information available becomes important. Microsoft's SDL process, including templates, tools, and other resources, is known as a **prescriptive** model for security development because it makes deliberate value judgments on security practices based on their real world effectiveness. Microsoft makes the SDL available for free. The Software Assurance Forum for Excellence in Code (SAFECode), a technology industry-driven non-profit organization dedicated to promoting

Be a Discriminating Consumer

Steven Lipner, Senior Director of Security Engineering Strategy at Microsoft and leader of Microsoft's Security Development Lifecycle (SDL) team encourages end users to not only examine the details of their developer's security initiatives, but how committed they are to the process: "There's a difference between security initiatives that emphasize 'might, may, and could' versus those that say 'shall, stop shipment, and fire,'" says Lipner.

secure coding practices, also makes prescriptive security development information available to the public. SAFECode's Fundamental Practices for Secure Software Development paper represents an ongoing effort to identify and recommend secure development activities shown to be effective in real-world implementations by its members. The SDL, SAFECode, and similar prescriptive efforts are critical initiatives for organizations building or refining their strategic approaches to security development to consider.

Descriptive resources like the Building Security in Maturity Model (BSIMM), on the other hand, allow organizations with security development initiatives to compare themselves against others. By design, descriptive resources only describe existing security development practices and do not measure their effectiveness. Mature organizations or those seeking to become more mature will build a prescriptive internal process and use descriptive resources and checklists for informational purposes. Developers and end users can find a list of resources in the appendix of this report.

In the critical infrastructure context, there are signs that more commercial developers are establishing and publicizing their security development processes. For instance, outside of the described efforts by vendors like Invensys and OSISoft, critical infrastructure vendor Siemens AG recently joined SAFECode.[40] A growing number of large commercial developers are hiring "chief product security officers" to work on their critical infrastructure applications. To work effectively within an organization, these positions should have a mandate to work across the product lines to address security

challenges and should be charged with implementing a corporate-wide, repeatable, security development process. Tactical initiatives, which are important components of an overall strategic approach to security development, may also be increasing in the critical infrastructure environment. Chris Wysopal of Veracode notes that while certain critical infrastructure sectors have not yet fully embraced application security testing, he thinks that "this is coming this year or next. Some sectors are simply faster moving than others."[41]

Though developers are responsible for adopting security development processes, end users play a major role in creating the demand for these processes. End users must not only enhance the maturity of their own internal software development processes but should also use their procurement processes to favor third-party products created using a security development process. In recent years, larger end users have been increasingly effective at communicating their security needs to commercial developers. This has been true in critical infrastructure as well, where larger companies tend to have strong internal engineering organizations and are more likely to interact and operate with developers on a peer-to-peer basis.[42] Smaller critical infrastructure companies and companies who have decentralized operations, on the other hand, may be less familiar with security development methodologies or perceive a lack of leverage in their interactions with developers. For all companies, there are resources available, including sample procurement and contractual language, to help end users ensure that their vendors are using a security development process.[43]

Committing to Accelerating the Trend

There are, to be sure, unique challenges in implementing a security development lifecycle in the critical infrastructure environment.

For instance, the abundance of mission-critical legacy applications and systems makes it difficult for developers to retroactively test and remediate applications that are already in place. Improving the security of these applications through modification or upgrade while implementing a security development lifecycle will further help to reduce network vulnerability and improve security over time. In process control industries, where the product lifecycle is significantly longer than in a traditional business environment, it means that the control system must be incrementally upgraded regularly, taking great care to respect that the facility must always be running. Therefore, operators should prioritize their security investments based on risk assessments of their systems to address the most critical needs first. Moreover, due to the unique services provided by critical infrastructure owners and operators, software developers may respond more slowly to customer demands than they would to those from other industries. As Ed Amoroso notes, critical infrastructure operators have a "unique" application security risk profile because of the limited number of vendors producing a limited number of applications that are valuable for a limited number of customers.[44] These and other factors may contribute to the delay in secure application adoption.

Cybersecurity challenges in the critical infrastructure environment may be difficult to address, but their consequences are too great to ignore. Efforts to build and purchase more robust software must be a top priority for all application developers and end users; unfortunately, too few recognize the importance and value of these efforts today. Growing concern about the cyber risk to critical infrastructure may lead policymakers to take action that may lead to negative and unintended consequences for developers and end users. If end users and developers want to avoid government intervention and costly failures, while achieving real cost savings, the time is now to show their commitment to software products developed using a security development process. Moreover, as end user and developer awareness of the operational and economic benefits of security development increases, there is reason to believe that adoption of widely accepted best practices will grow, minimizing the number and impact of vulnerabilities in the critical infrastructure application environment.

Appendix: Application Security Resources

Industry/Non-profit Organizations

Software Assurance Forum for Excellence in Code (SAFECode

SAFECode is a global, industry-led effort to identify and promote best practices for developing and delivering more secure and reliable software, hardware and services. SAFECode publishes software security guidance and best practices based on the real-world experiences of its members, which include Adobe Systems Incorporated, EMC Corporation, Juniper Networks, Inc., Microsoft Corp., Nokia, SAP AG, Siemens AG and Symantec Corp.
www.safecode.org

OWASP (The Open Web Application Security Project)

OWASP is a global not-for-profit organization focused on improving the security of application software. OWASP projects cover many aspects of application security, and it publishes a wide variety of free resources including documents, tools, teaching environments, guidelines, checklists and more.
www.owasp.org

Training and Certification Resources

GIAC (Global Information Assurance Certification) Secure Software Programmer (GSSP) Certification

The GIAC Secure Software Programmer (GSSP) Certification Exam was developed in a joint effort involving the SANS Institute, CERT/CC, several US government agencies, and leading companies in the US, Japan, India, and Germany. It allows candidates to demonstrate mastery of the security knowledge and skills needed to deal with common programming errors that lead to most security problems.
www.giac.org

(ISC)2 (International Information Systems Security Certification Consortium, Inc.) Certified Secure Software Lifecycle Professional (CSSLP)

Developed by (ISC)2, the Certified Secure Software Lifecycle Professional (CSSLP) is a certification designed to validate secure software development knowledge and expertise.
www.isc2.org

The International Council of E-Commerce Consultants (EC-Council) Certified Secure Programmer (ECSP) and Certified Secure Application Developer (CSAD)

The EC-Council is a member-based organization that certifies individuals in various e-business and information security skills. Its Certified Secure Programmer (ECSP) and Certified Secure Application Developer (CSAD) programs aim to provide the essential and fundamental skills to programmers and application developers in secure programming.
www.eccouncil.org

Publications, Tools and Standards

Microsoft Security Development Lifecycle (SDL)

Developed by Microsoft, the Security Development Lifecycle (SDL) is a software development security assurance process consisting of security practices grouped by seven phases: training, requirements, design, implementation, verification, release and response. Microsoft provides a large number of SDL templates, tools and other resources at no cost to assist other companies wishing to improve the security of the software they develop.
www.microsoft.com/security/sdl/default.aspx

ISO (International Organization for Standardization) 27034

ISO is a global developer and publisher of International Standards across a broad range of industries. It recently published a new standard as part of its information security-focused 27000 series that focuses on application security. ISO/IEC 27034 provides guidance to assist organizations in integrating security into the processes used for managing their applications. It introduces definitions, concepts, principles and processes involved in application security.
www.iso.org/iso/home.htm

BITS Software Assurance Framework

BITS, the technology policy division of the Financial Services Roundtable, published a Software Assurance Framework to provide an overview of the components of a mature, strategic software development program for financial institutions. The paper offers practices and principles to apply at all stages of software development, including education and training, threat modeling, coding practices and security testing, among others.
www.bits.org

Building Security in Maturity Model (BSIMM)

The BSIMM is an inventory of existing software security practices from over 40 large-scale, IT dependent organizations across seven business vertical categories. The BSIMM is useful for comparing an organization's software security activities to the activities observed among the 40 firms that have participated in the BSIMM.
www.bsimm.com

The 2011 CWE/SANS Top 25 Most Dangerous Software Errors

The 2011 CWE/SANS Top 25 Most Dangerous Software Errors is a list of the most widespread and critical errors that can lead to vulnerabilities in software.
www.sans.org/top25-software-errors/

Notes

1. Ed Amoroso, interview by Good Harbor Consulting, January 9, 2012.
2. Ed Amoroso, interview by Good Harbor Consulting, January 9, 2012.
3. Verizon Business & US Secret Service, "2012 Data Breach Investigations Report," (March 22, 2012) http://www.verizonbusiness.com/resources/reports/rp_data-breach-investigations-report-2012_en_xg.pdf (accessed March 22, 2012), pp. 32-33.
4. Microsoft Corporation, "Microsoft Security Intelligence Report – Worldwide Threat Assessment," (2011), http://www.microsoft.com/security/sir/keyfindings/default.aspx#!section_2_3 (accessed January 11, 2011).
5. Chris Wysopal, interview by Good Harbor Consulting, January 29, 2012.
6. Office of the National Counterintelligence Executive, "Foreign Spies Stealing US Economic Secrets in Cyberspace," (October 2011) http://www.ncix.gov/publications/reports/fecie_all/Foreign_Economic_Collection_2011.pdf (accessed November 3, 2011), p. 6.
7. Sean McGurk, testimony before the House of Representatives Oversight Committee Subcommittee on National Security, Homeland Defense, and Foreign Operations, May 25, 2011, http://www.gpo.gov/fdsys/pkg/CHRG-112hhrg70676/html/CHRG-112hhrg70676.htm (accessed January 24, 2012).
8. Kim Zetter, "10K Reasons to Worry About Critical Infrastructure," Wired.com, January 24, 2012, http://www.wired.com/threatlevel/2012/01/10000-control-systems-online/ (accessed January 24, 2012).
9. Hal Hodson, "Hackers accessed city infrastructure via SCADA – FBI," InformationAge, November 29, 2011, http://www.information-age.com/channels/security-and-continuity/news/1676243/hackers-accessed-city-infrastructure-via-scada-fbi.thtml (accessed January 24, 2012).
10. Solomon Tessema, interview by Good Harbor Consulting, January 13, 2012.
11. Michael Assante, interview by Good Harbor Consulting, December 28, 2011.
12. Eric Cosman, interview by Good Harbor Consulting, January 5, 2012.
13. Michael Assante, interview by Good Harbor Consulting, December 28, 2011.
14. Doug Cavit, interview by Good Harbor Consulting, January 24, 2012.
15. Richard Clarke, interview by Good Harbor Consulting, January 5, 2012.
16. Microsoft Corporation, "Simplified Implementation of the Microsoft SDL," (November 4, 2010), http://www.microsoft.com/download/en/details.aspx?displaylang=en&id=12379 (accessed December 16, 2011).
17. Forrester Research commissioned by Microsoft Corporation, "State of Application Security: Immature Practices Fuel Inefficiencies, But Positive ROI Is Attainable," (January 19, 2011), http://www.microsoft.com/download/en/details.aspx?displaylang=en&id=2629 (accessed December 16, 2011), p. 6.
18. Dickie George, interview by Good Harbor Consulting, December 28, 2011.
19. Definition from Forrester Research, "State of Application Security," p. 5: "A security practice is mature when it is well defined and able to respond proactively to emerging threats, with established technologies, well-known practices, and established metrics to measure and track performance."
20. Forrsights Security Survey, Q3 2010, cited by Chenxi Wang, "Application Security: 2011 And Beyond," Forrester Research (April 12, 2011), http://go.microsoft.com/?linkid=9777219 (accessed December 16, 2011), p. 6.
21. Forrester Research, "State of Application Security," p. 6
22. Wang, "Application Security."

23. Microsoft Corporation & iSec Partners, "Microsoft SDL: Return on Investment," (September 15, 2009), http://www.microsoft.com/download/en/details.aspx?displaylang=en&id=8873 (accessed December 16, 2011), p. 5.

24. Microsoft Corporation & iSec Partners, "Microsoft SDL," p.3.

25. RTI for National Institute of Standards and Technology, "The Economic Impacts of Inadequate Infrastructure for Software Testing," (May 2002), www.nist.gov/director/planning/upload/report02-3.pdf (accessed December 16, 2011).

26. Mary Ann Davidson, "The Root of the Problem (Mary Ann Davidson Blog)," Mary Ann Davidson Blog, September 2, 2010, https://blogs.oracle.com/maryanndavidson/entry/the_root_of_the_problem (accessed February 15, 2012).

27. Derek E. Brink, "Security and the Software Development Lifecycle: Secure at the Source," Aberdeen Research (December 2010), http://www.microsoft.com/download/en/details.aspx?displaylang=en&id=6968 (accessed December 16, 2011).

28. Brink, "Security and the Software Development Lifecycle."

29. Bryan Owen, interview by Good Harbor Consulting, January 12, 2012.

30. Ernest Rakaczky, interview by Good Harbor Consulting, February 8, 2012.

31. Microsoft Corporation, "MidAmerican SDL Chronicles: An inside look at one company's 273-day quest to transform its process and culture on the way toward making its software more secure," (March 30, 2011), http://www.microsoft.com/download/en/details.aspx?displaylang=en&id=2881 (accessed December 16, 2011).

32. Forrester Research, "State of Application Security."

33. Microsoft Corporation, "MidAmerican SDL Chronicles," p. 6.

34. Brink, "Security and the Software Development Lifecycle," p. 11.

35. James Langevin, interview by Good Harbor Consulting, January 26, 2012.

36. BITS, "Software Assurance Framework," (February 1, 2012), http://www.bits.org/publications/security/BITSSoftwareAssurance0112.pdf (accessed February 1, 2012).

37. Energy Sector Control Systems Working Group, "Roadmap to Achieve Energy Delivery Systems Cybersecurity," (September 2011), http://energy.gov/sites/prod/files/Energy%20Delivery%20Systems%20Cybersecurity%20Roadmap_finalweb.pdf (accessed January 31, 2012).

38. James Langevin, interview by Good Harbor Consulting, January 26, 2012.

39. Wang, "Application Security,," p. 5.

40. Software Assurance Forum for Excellence in Code, "SAFECode Adds Siemens as Newest Member," (November 8, 2011), http://www.safecode.org/press/SAFECode_Siemens_PR_110811.pdf (accessed December 16, 2011). Chris Wysopal, interview by Good Harbor Consulting, January 29, 2012.

41. Chris Wysopal, interview by Good Harbor Consulting, January 29, 2012.

42. Eric Cosman, interview by Good Harbor Consulting, January 5, 2012.

43. US Department of Homeland Security, "Cyber Security Procurement Language for Control Systems," (September 2009), http://www.us-cert.gov/control_systems/pdf/FINAL-Procurement_Language_Rev4_100809.pdf (accessed January 31, 2012).

44. Ed Amoroso, interview by Good Harbor Consulting, January 9, 2012.

Good Harbor Consulting provides strategic risk management services to corporate and government clients around the world. Good Harbor's Cyber Risk team works with senior corporate executives, investment professionals, and government leaders to develop cybersecurity programs that mitigate organizational risk in the face of advanced cyber threats. Good Harbor's consulting services include threat awareness, risk assessment, strategy and governance, crisis management and communications, regulatory and policy analysis, thought leadership, and investment diligence. The firm is led by Chairman Richard A. Clarke, a former, senior White House advisor on cybersecurity, counterterrorism, and national security, and the author of *Cyber War: The Next Threat to National Security and What To Do About It.*

To learn more about Good Harbor, visit http://www.goodharbor.net

GOOD HARBOR
CONSULTING, LLC