Critical Applications Resilience (CAR)—the next phase in government’s effort to secure critical infrastructure—is a complementary program to Continuous Diagnostics and Mitigation (CDM). The causes of the prevalence of nonresilient software are technical and nontechnical; therefore, solutions must take several factors into account. Over the last decade, HP has pursued and promoted building in cybersecurity resilience.
Table of contents

Attack resilience.................................................................................................................................................................................. 1
Review executive and governance considerations ......................................................................................................................... 2
Build It Right; develop it right.................................................................................................................................................. 4
Continuously monitor operations........................................................................................................................................... 7
Learn how HP can help.......................................................................................................................................................... 8
About the authors ................................................................................................................................................................. 9
Gain cybersecurity survivability

CAR builds cybersecurity survivability into applications, making them resilient to attack, able to more readily defend themselves and recover operations. With CAR, a critical component can take a hit, recover, and return to normal operations in a known, bounded, and generally acceptable time period. This complements CDM-deployed applications and solutions.

Attack resilience

As cyber attacks grow in sophistication and effectiveness, those developing and deploying applications must enhance their understanding of the impact of associated failures on systems and applications availability and functionality. Steps need to be taken so systems can resist attack and rapidly recover from failures and outages—whatever their cause.

Industry-standard application development practices do not result in cybersecurity resilience. Most applications and other software today are riddled with vulnerabilities because current and past development practices failed to build in resiliency or sufficient load or stress testing to identify failure points in their applications or systems. With the growth in increasingly extended software development supply chains and competitive markets, the focus is more on functionality and low cost rather than resistance to attack and resiliency. The challenge is spreading the knowledge and incentives on how to design and build more secure and resilient systems. It’s essential to introduce mechanisms that reward security, resilience, and integrity rather than just functionality and speed to market.

The U.S. National Institute of Standards and Technology (NIST) recently released the initial public draft of the “Special Publication 800-160, Systems Security Engineering: An Integrated Approach to Building Trustworthy Resilient Systems.” It pursues the federal cybersecurity strategy of “Build It Right, Continuously Monitor” to “address security issues from a stakeholder requirements and protection needs perspective and to use established organizational processes to ensure that such requirements and needs are addressed early in and throughout the lifecycle of the system” as stated early in SP 800-160. In the same timeframe, the Department of Homeland Security is initiating a program to address Critical Applications Resilience as a complementary program to Continuous Monitoring. Critical Applications Resilience is designed to do for applications what the U.S. Department of Homeland Security’s (DHS’s) Continuous Diagnostics & Mitigation (CDM) program is doing for the IT infrastructure.

By introducing Critical Application Resilience, the U.S. DHS acknowledged the lack of prevalent application resilience and is beginning to take steps to address it. For decades, HP has acknowledged this need and has been protecting client infrastructure and developed proactive and resilient approaches for applications and system cybersecurity.

Attention to and investment in application resilience is welcome, as the gap between the threat environment and application’s ability to defend has grown gradually through the decades. At its root, the IT cybersecurity industry has yet to prevalently apply lessons from the IT quality field on building quality in and not expecting to merely test it out. There is some evidence of progress, for instance, the DHS/DoD/GSA Build Security in program, the Build Security In Maturity Model (BSIMM), the Open Software Assurance Maturity Model (OpenSAMM), and the Certified Secure Software Lifecycle Professional (CSSLP) certification. However, common practices remain focused on either testing vulnerabilities out or purely reactive security patching, with little focus on designing security in, resilience, or survivability.

In the 1950s, Edwards Deming and others introduced quality concepts such as Total Quality Management (TQM) to manufacturing. This first took hold in Japan, resulting in the Japanese automotive quality greatly surpassing U.S. automotive quality. Quality transformation arrived in the United States in the 1970s, and in the IT software arena in the 1980s—at HP with the then HP CEO John Young’s 10X quality initiative to improve software quality by an order of magnitude in a decade. These quality initiatives focused on repeatability, building in quality, managing quality, and going beyond merely testing.
HP has been gathering and using capabilities developed in-house—such as HP Comprehensive Applications Threat Analysis (CATA) and best-in-breed acquisitions such as HP Fortify, HP WebInspect, the Zero Day Initiative among others—to build cybersecurity resiliency into applications and other software, systems, and products.

We consider CAR to have elements residing under three broad topic areas:

- Executive and governance considerations
- Build It Right and development
- Continuous Monitoring and operations

Of course, governance is important as it determines priorities, funding, and what gets measured and done. The other two sections are partitioned along the lines of the U.S. federal cyber security strategy mentioned in NIST 800-160: “Build It Right, Continuously Monitor.” Retrofitting security is insufficient, and needs to be combined with ongoing operational security.

### Review executive and governance considerations

It’s important to determine how resilience is prioritized and where it’s applied by suppliers and applications users, especially in governments and large enterprises. This is particularly true when determining the criticality of applications for resiliency prioritization and dynamics around the incentives for and against resilient software. Additional topics that round out the field include an introduction of how the Internet of Things (IoT) relates to resiliency and how Secure Software Engineering (NIST SP 800-160) ties into it. NIST SP 800-160 peppers the tie-in to stakeholders and governance throughout the document.

### Application criticality classification

One or more mechanisms must to be chosen to focus on application resilience only for critical applications. Prioritizing critical applications may prove necessary to determine allocation of limited budgets. Existing mechanisms that can be used include FIPS 199 classification and Mission Assurance Categories (MACs with a threshold such as FIPS High), which is used to designate critical applications. Mission-critical applications used to support critical infrastructure could be another potential determinant. In the commercial realm, some applications have high availability and rapid recovery time requirements or responsible for implementing critical business processes, which may be identified by a Business Impact Assessment. Those might include:

- Supporting critical business functions including a web storefront
- High-integrity requirements, such as financial and legal transactions
- High-confidentiality requirements including those protecting trade secrets, confidential financials, and others

On the other hand, all applications would benefit from an appropriate—usually enhanced beyond current—degree of resilience; industry standard application development practices generally result in low cybersecurity resilience applications, riddled with vulnerabilities and weaknesses. This often results in unknown, uncharacterized, and unmitigated risks.

### Economic signals and incentives for resilient software

In large part, the root cause of most software lacking cybersecurity resiliency is an economic issue first and a technical issue second. Solving technical issues won’t accomplish much if market incentives are not in place to cause vendors to prioritize resiliency sufficiently. Markets reward characteristics that can be used as buying criteria. Today, it’s very hard to tell if a system is or will be resilient or “secure.” Absolute security is not practical, but the observation applies to “secure enough,” also.

Other markets have solved this problem with simple market signals of quality or safety, for instance, UL listings for electrical devices and professional or peer reviews, such as Consumer Reports, Edmunds, National Highway Traffic Safety Administration (NHTSA), Insurance Institute for Highway Safety (IIHS) crash test and safety star ratings, and Amazon reviews. No equivalent exists as a simple measure of cybersecurity resiliency or security quality (CERT has created a Resilience Management Model—CERT-RMM, though that reflects organizational capability, not resilience of any specific software), and so the market is unable to reward secure applications over insecure ones. A very interesting early contender for such a signal is modeled on NHTSA crash test ratings, but is for automotive cybersecurity, called Five Star Automotive Cyber Safety Program. Common Criteria Certification does not provide this economic signal, with the limited, possible exception of smart cards. Most certifications are at EAL3 or below and often fairly weak security protection profiles, as high EALs and strong protection profiles are not generally commercially feasible for most products and vendors. Neither Protection Profiles nor Security Targets focus on security resilience. Until vendors are rewarded in the marketplace for secure products, attempts to fix the problem are likely to meet with limited success. In the past, public sector purchasing power has not proven sufficient to overcome this barrier.
“The most significant issues [in Common Criteria Certification] are the lack of criteria pertaining to the characteristics of software that are considered essential to its security, i.e., lack of exploitable vulnerabilities and weaknesses and ability to continue operating dependably when under attack. [...] STs [security targets] say nothing about the robustness against attacks of, and lack of vulnerabilities in, the software that implements those security functions. With this as its objective, it is not surprising that the CC evaluation does not include analysis or testing of the TOE for implementation faults that may lead to vulnerabilities. Unlike safety cases, ST documents (which are, in essence, security cases) are not intended to demonstrate the TOE’s own inherent security robustness, but only its conformance to security policy. For this reason, CC STs are considered by those who are developing standards for software security assurance cases (see Section 5.1.4) to provide only a limited amount of meaningful security evidence for supporting security claims made in the context of software security assurance cases.”

It’s challenging for vendors to balance security against time-to-market, functionality, and good enough when consumers, governments, and enterprises tend to favor buying features and functions. This is especially true when cybersecurity resilience signals are not available. And yet, if the underlying incentives aren’t addressed, the commercial off-the-shelf (COTS) applications upon which systems rely will continue to lack the needed resiliency. And without this resiliency built in, the cost of compensating controls to increase assurance to required levels either becomes considerably more expensive than if designed in, or out of reach—technically or financially.

Expanding beyond applications—IoT and systems resilience

Applications are a critical component in the enterprise, but it’s also vital to monitor emerging technologies and trends as they are adopted. IoT is just beginning to be understood, and the growth associated with it will be staggering. With the explosion of devices happening now, sensors, radio frequency identification (RFID) tags, wearables, and other “smart” Internet-connected “things” are becoming commonplace.

What should be the concern in this new environment? Embedded systems and software, which lack the ability for user or device authentication, are requiring device-to-device communications, automation, and proprietary systems. These IoT devices provide embedded functionality; therefore, device manufacturers need to provide secure solutions up front—something there is little control over and limited capability to monitor on an enormous scale. This presents issues that have to be analyzed, security programs that have to be adapted to identify the risk, and mitigation strategies created to provide necessary security.

Gartner predicts that the installed base of “things,” excluding PCs, tablets, and smartphones, will grow to 26 billion units in 2020, which is almost a 30-fold increase from 0.9 billion units in 2009. The component cost of [Internet of Things] IoT-enabling consumer devices will approach $1, and “ghost” devices with unused connectivity will be common.

IoT will provide value to public and private sector enterprises, but risk must be balanced while integrating value that benefits the mission. It’s necessary to be cognizant of personally identifiable information (PII) protection, the increased attack surface that the abundance of devices will create, and the launch point that IoT vulnerabilities may create into critical systems. With a proper security strategy and Continuous Monitoring, participation of IoT in today’s environment can be accommodated. It’s imperative to continuously re-evaluate your position as this emerging trend matures and is deployed in environments.

Resiliency and scale can sometimes be improved by architectural choices, for instance, centralizing some data collection and control to limit the excessive scale IoT sensors might impose, and consolidating some problem resolutions—resiliency among them.

The interdependency or interconnectivity brought on by IoT adds to complexity due to increased interconnected and interdependent components. Therefore, achieving resiliency becomes a greater task. In order to protect against this situation, applications may require routines that preserve integrity and operational continuity even if other components were to fail, while remaining as simple as practical (economy of mechanism).

---

2 www.gartner.com/newsroom/id/2844317
Alignment with NIST Special Publication 800-160

NIST SP 800-160, “Systems Security Engineering: An Integrated Approach to Building Trustworthy Resilient Systems,” takes a broader approach than applications, though much of the publication applies to applications, or applications within a system. It also does not focus only on “critical” systems. Much of what is highlighted in the publication tracks well with approaches HP has been developing and refining over the last decade. This includes:

• Attention to security implications of architecture and design
• Viewing security in part as a quality property
• Focus on security relevance
• Trust and trust relationships
• Security throughout the lifecycle

HP has leading application security resilience capabilities in key phases of the lifecycle, such as security requirements gap analysis, and architectural threat modeling with our CATA—described under Secure Software Development Lifecycle (SDLC), implemented with HP Fortify, and verified and validated with HP WebInspect.

Build It Right; develop it right

Several topics fall in this area, starting with security within and throughout the software development cycle. All too often, security is an afterthought and applied only near the lifecycle’s end. Here, implications on resilience of new vs. legacy code on application modernization and transformation are considered. Fault tolerance and high availability and how they can support resilience are also considered. We must also think of the role of application instrumentation in providing audit logging support, needed during operations, to provide additional defenses for missed vulnerabilities.

Secure Software Development Life Cycle (SDLC)

Typical industry approaches to application security have been reactive, and have failed to apply lessons from the quality field. The two most prevalent approaches are:

• “Bury head in sand”—Reactive security patching—for example, CVEs—with nothing to avoid or minimize vulnerability introduction, most often seen in industry segments with minimal security-relevant regulatory burden.
• “Test security in”—Nothing to build resiliency into applications, with effort applied to find and fix vulnerabilities during testing in combination with security patching. This approach appears more commonly in the public sector, such as Federal Information Security Management Act (FISMA) driven, and security-regulated industries, such as financial, for example, Payment Card Industry Data Security Standard (PCI-DSS), and healthcare including the Health Information Portability and Accountability Act (HIPAA) and Health Information Technology for Economic and Clinical Health (HITECH).

However, security as a quality attribute, needs to be applied at every stage of the lifecycle, in the same way the quality field learned it decades ago:

• Quality can’t be tested in; it has to be designed and built it in, and then tested.
• Its orders of magnitude less expensive to find defects—in this case security defects and vulnerabilities—early, rather than late, in the lifecycle.
For example, using the lifecycle phases referenced in NIST SP 800-160 (from ISO/IEC/IEEE 15288:2008):

- **Requirements Analysis**—HP CATA Security Requirements Gap Analysis identifies security requirements and controls gaps.
- **Architectural Design**—HP CATA Architectural Threat Analysis, including threat modeling, identifies opportunities to build architectural and design resiliency in, and supports proper error handling, including graceful degradation.
- **Implementation**—Security Source Code Analysis, such as HP Fortify Static code analysis and human expert code review, identifies vulnerabilities in source code.
- **Verification & Validation**—HP WebInspect and human expert vulnerability assessment and penetration testing identify vulnerabilities in run-time behavior. Performance, including stress testing, operational testing, and fail-over, is crucial to maximizing resilience.

**Figure 1.** Cost/benefit comparison

HP CATA is an industry best practice. It proactively builds in security quality—reductions in vulnerabilities—at the earliest stages while leveraging the highest layers of abstraction of the lifecycle—requirements, architecture, and high-level design. This is accomplished via security requirements gap analysis and architectural threat analysis. This methodology is designed to significantly reduce the problem of undiscovered vulnerabilities. With conventional development, only a small fraction of vulnerabilities are discovered and fixed pre-release, with many discovered and corrected at a high-cost post release, and worse yet, many remain undiscovered.

HP provides comprehensive findings that detail potential vulnerabilities and weaknesses discovered during the assessment with recommended mitigation approaches for each. This review provides an enhanced level of security for the application by identifying the need and opportunity to strengthen security requirements, and architectural resilience to limit and reduce exposure to known and unknown vulnerabilities.

CATA elements:

- **The Security Requirements Gap Analysis** determines security challenges for deploying an application in compliance with common standards such as NIST SP 800-53, PCI-DSS, Gramm–Leach Bliley US Law (GLB), and others.
- **The Architectural Threat Analysis** is an architecture-level review and threat modeling analysis, including a deeper look into the security properties of underlying components and in-depth recommendations for mitigating the highest risk areas. This review helps enable resilient designs to greatly reduce the likelihood and severity of vulnerabilities, despite inevitable coding defects that may be present in any nontrivial application or system.

The assessment can also function very efficiently and effectively—like an Independent Verification and Validation (IV&V) of security requirements and architectural security resilience analysis for applications development projects.
Legacy code vs. new code
Legacy code and new or “greenfield” code bring different challenges and opportunities to application resiliency. For instance, legacy code was developed at a much “safer” time, in many cases when the Internet was not considered a hostile environment. As a result, applications were developed with minimal consideration for defending themselves or the data entrusted to them. As such, they have become a weak link in enterprise security, often with greater than 70% of successful exploits and vulnerabilities found at the application layer.

For instance, 2H2013 Microsoft® Security Intelligence Report shows 83% of reported vulnerabilities in applications and only 17% in core operating systems. It’s worth noting that infrastructure and endpoint security blocks a high percentage of attack attempts. Legacy applications need to be modernized and transformed to support emerging architectures and technologies, such as cloud, mobile, and others, and upgrading cybersecurity resilience is an essential element of that modernization or transformation.

Many of these legacy applications reside on End-of-Life (EOL) components, and organizations are stuck with components that are known to be vulnerable, but are needed for mission-critical applications. Modernization can be expensive and span multiple years, though it often has higher return on investment when compared to other options. Maintaining cyber resiliency in the face of this reality is difficult. Mitigating controls may be required, which can involve network isolation and multi-layered defenses, such as adding HP Application Defender and/or virtual patching from HP TippingPoint.

Legacy systems that remain in maintenance mode, with no security transformation or modernization investment, represent a significant risk as they were not developed to be resilient to the greater risks we have today. When proactive investment to improve application cybersecurity resilience is out of reach—it can only be addressed reactively, for instance, with security patch management. This reliance on infrastructure security is meant to compensate for weaknesses in application resiliency.

Even new code is rarely purely greenfield; rather it’s more often closer to a modernization or transformation project as modules are reused or retrofitted to meet a current need. As software and its implementation become increasingly complex and dependent on diverse infrastructures, including EOL components, it’s essential for those designing and developing computer applications to be aware of evermore challenging environments into which software is installed.

Regardless, new applications afford the opportunity to “Build it Right” and take into account the modern threat environment. Of course, if the application becomes a legacy application in a decade, the threat environment may change again, and the cycle continues. Building it right really requires direct attention, investment, and even acculturation to cybersecurity resilience as a key development consideration. It cannot be relegated to an afterthought and expect to achieve any degree of security resilience. It must be designed in. At HP, we have a Secure Product Development program to guide product generation through building security in from the earliest lifecycle stages.

Fault tolerance, high availability
Concepts of fault tolerance and high availability have existed outside the cybersecurity realm for decades. An example is HP’s NonStop High Availability systems, formerly Tandem Computers, or high-availability software, such as HP Serviceguard, or other clustering solutions. Important cybersecurity resiliency concepts carry over from high availability and fault tolerant designs. For instance, the security design principle of multi-layered security provides a way to design the no Single Point of Failure (SPOF) fault tolerant design principle in the cybersecurity realm. Fault tolerance and recovery make Denial of Service (DOS) attacks more difficult and less effective.

Cybersecurity fault tolerance is especially important for cybersecurity resiliency for at least three reasons:

1. Vulnerabilities are prevalent; and even with significantly improved secure SDLC development practices, software developers are human and will make mistakes (defects) in anything but trivial applications. Cybersecurity fault tolerance reduces the chance of those defects becoming exploitable vulnerabilities.
2. The ability to resist attacks—and recover quickly to an acceptable performance level after failure due to successful exploits, unintended damaging actions, or accidents—is crucial for applications running in many organizations.
3. The concept of “failsafe” or failing securely is directly applicable and important to apply from fault tolerance to cybersecurity.

Note that cybersecurity resilience benefits from operational resilience and vice versa; however, addressing one will not by itself adequately address the other. HP CATA assesses for and encourages security fault tolerant designs.
Application instrumentation
During development, it’s important to instrument applications with logging and audit support of security-relevant events. These log or audit records need to be continuously monitored during operations. Resilience instrumentation can also provide a second layer of defense against the possible presence of some known vulnerabilities, such as provided by HP Application Defender.

Continuously monitor operations
Some of the key operational and monitoring aspects related to application resiliency include software asset and configuration management, SIEM, dynamic/active defenses, and Continuous Monitoring.

Software asset and configuration management
Knowing what to analyze and protect is vital to critical applications resilience. There have been a multitude of applications added to most production environments over many years and decades, so it’s easy to lose track. Therefore, rigorous software asset and configuration management must be applied to keep an accurate inventory of all applications in service. With a comprehensive inventory, governance principles can be applied to determine individual application characteristics, understand life expectancy, resolve any functional duplication issues of multiple systems, and provide a prioritized list of recommendations for application sustainment. These include security patching, refinement, or retirement, and resiliency analysis and investment. Change controls need to be well managed. A major reason for application failures include inadequate configuration or change management, system sizing or capacity planning problems, IT staff errors, and patch management issues.

Frequently updated software represents its own challenges, as new software revisions may not only fix vulnerabilities in previous revisions, but may also introduce new ones. If updates are too frequent, time to test for vulnerabilities may not be sufficient, which is an especially challenging problem for rapid mobile application development cycles.

Software Asset Management needs to be applied continuously. As systems age, their usefulness to achieve mission objectives, and their protection against vulnerabilities and threats can continually change. An enterprise must review applications not only at the individual app perspective but also holistically to best guard against current and emerging threats to their environments. A reliable and criticality-prioritized software asset inventory can enable prioritizing resiliency investment as warranted by impact to mission risk and other impact factors.

Consumption of application instrumentation
Of course, application instrumentation—see instrumenting applications under the Build it Right section—provides no value if it’s never examined. Often logs and audit trail data are collected in fragmented logs, and never looked at.

At a minimum, log data needs to be used to gather information once a security incident occurs, to help characterize what happened, and determine a suitable response.

However, the clues to discovering incidents during or before occurrence are often in the logs—if only they were aggregated and continuously monitored. Combinations of proper tuning, policy, tool automation, and human monitoring may be required to get the most value from audit logs. SIEM tools, such as HP ArcSight, can help aggregate and correlate security event data and automatically raise alarms.

Active defense
Resilience can come in passive or active forms. While mostly passive defenses have been covered, active defense can include self-healing and Intrusion Detection and Intrusion Prevention Systems (IDS/IPS). An example of self-healing is HP Sure Start BIOS protection, which detects BIOS corruption and automatically restores a known, good BIOS in cases of infection by BIOS-corrupting malware. IDS/IPS systems exist today, of course, such as HP TippingPoint, and similar functionality can be built directly into applications. A partially active approach is one taken by HP Application Defender, which blocks attacks from within the application. Examples of such systems include OWASP’s AppSensor and Mykonos Web Security, which embed “bad actor” sensors into generated code that can trigger different responses.

However, active defense comes with its own risks and complications. For instance, response is often to shut down suspicious connections, but that can provide a new and easy DoS path to an attacker—trigger the active defense to force a DoS on demand. That may be the lesser evil if it prevents a worse attack.

Secondly, intrusion detection is usually heuristic in nature, which means it isn’t 100% accurate. False negatives and false positives have to be balanced. False negatives mean the system may not be sufficiently effective, and false positives may cause the active response to be too disruptive. Intrusion detection usually cannot be tuned to avoid both false negatives and false positives.
Also, building active response directly into applications adds complexity versus keeping active response in separate systems, such as an IPS. Complexity often introduces defects and vulnerabilities.

**Reputation-based resilience and rapid response**

DHS described Critical Application Resilience in January 2014 as a means to have applications mimic human immune responses by being better able to detect dangerous/low reputation network traffic and defend against it rapidly without human intervention. The idea is for network traffic to carry reputation information, to enable applications to recognize suspect or malicious traffic. HP already does something at the IDS/IPS layer with HP TippingPoint ThreatDV (formerly known as Reputation Digital Vaccine-RepDV). In a sense, this is more easily accomplished out-of-band. This doesn’t require new standards to generate and transmit reputation information directly with the network traffic, whereas in-band would likely require creation and adoption of new standards—and doesn’t require modifying applications, as IPS can be retrofitted onto a network. On the other hand, modified applications could behave in a more context aware application-specific manner than an independent IPS.

**Continuous Monitoring**

NIST 800-160 introduces itself by referencing the government strategy, “Build it Right; Continuous Monitoring.” This is certainly a strategy long supported at HP—such as building it right with our CATA process, which has been in use since 2003, and our Secure Product Development Program that has been in place since 2008. Once a resilient application is developed, assessed, authorized, and deployed, it requires Continuous Monitoring of its security posture, to apply newly released vulnerability patches, monitor audit logs, software assets, and others. Also, once an application is developed or altered to a “final” state that is secure and resilient, its known good state is mapped and stored in a universal Configuration Management DataBase (uCMDB). With continuous monitoring, the deployed application is constantly monitored for changes against that known good state, and risk notifications are created when the two states are not identical. This prevents unwanted modifications to software settings, detecting changes that could indicate viruses or advanced persistent threats.

**Learn how HP can help**

HP offers products and services to help in most aspects of Critical Applications Resilience, including:

1. **Requirements analysis, architectural design**—HP Comprehensive Applications Threat Analysis (CATA) assesses and guides resiliency of security requirements, architecture, and design.

2. **Implementation**—HP Fortify and human expert code review find vulnerabilities in source code through static security analysis.

3. **Verification and validation**—HP WebInspect and human expert vulnerability assessment and penetration testing find vulnerabilities in run-time behaviors through dynamic security testing.

4. **Continuous Monitoring**—The NIST Risk Management Framework (see NIST Special Publication 800-37 rev. 1) requires system owners to monitor—on a regular and on-going basis (continuously)—all of the security controls identified for systems. The HP Continuous Monitoring offering provides a “best-of-breed” approach to those controls that can be monitored and controlled in an automated manner, improving the overall security posture of systems and applications. HP Continuous Monitoring offerings help agencies meet their continuous monitoring requirements, strengthen the underlying information technology infrastructure by applying state-of-the-art architectural and engineering solutions, and leverage automation to support implementation of a Risk Management Framework. Enterprises are responsible for implementing and maintaining their individual Information Assurance (IA) programs to assess organizational compliance with laws, regulations, and policies, among others. Until recently, these IA programs were based on a periodic checklist—or a snapshot in time of an organization’s security posture. By implementing a CM program to replace the checklist-based periodic IA program, CM provides:

   • Continuous feedback on the effectiveness of risk management activities and responses
   • Real-time identification of changes to information systems and operational environments and correlation between those changes, and how they affect your risk tolerance
   • Verification of compliance to federal, state, and local legislation; executive orders; directives; policies; and standards and guidelines
   • System and application maintenance costs reduced

5. **Operational security**—HP ArcSight, HP TippingPoint, HP Application Defender, HP Networking

6. **Application modernization and transformation**—HP offers a robust set of services around application modernization and transformation, which are recommended to include to enhance application cybersecurity resiliency, such as with CATA.
About the authors

John Diamant, HP Distinguished Technologist, Secure Product Development Strategist, CISSP, CSSLP
Jeff Misustin, HP Security Services Marketing
Steve Lazerowich, HP CISM, FITSP-Manager
Stan Wisseman, HP ESP Security Strategist, CISSP, CISM, CSSLP

Learn more at
hp.com/go/applicationsecurity
hp.com/enterprise/security
hp.com/go/esp