White Paper

Application Development, Test & Delivery

Micro Focus Fortify for PCI Compliance

MICRO FOCUS

Table of Contents

Introduction

Internet and network-accessible applications are some of the greatest tools that firms can use to reach their customers, while simultaneously exposing the firm to added risk just by the nature of those applications being accessible. Outside access to technology resources, such as applications or Web services, increases the attack surface that information security managers must defend against attacks by malicious users. As we continue to increase cashless payments throughout our companies, payment card information proliferates through these applications, making compliance with standards such as PCI DSS more challenging.

Application security is a critical element for the enterprise wishing to be PCI-compliant. Application attacks compromise the logic flow and data handling from within the application, affording access to sensitive data and more. Identifying and removing vulnerabilities during development and testing, and protecting vulnerabilities that may remain in production, is the most effective way to reduce these risks.

Compliance-Driven Security

Information security professionals will simultaneously applaud compliance as a driver that helps them achieve their goals of securing the enterprise while cursing its existence as a check-box method to prop up the baseline of information security we all hope protects us. With each passing year, more compliance initiatives appear and disappear, but the information security bar that we set stays relatively low. The types of attacks in the wild today suggest that this bar is something you trip over instead of leap over.

Compliance-driven security is great for firms or industries where information security is not a priority. In the case of retail, PCI DSS revolutionized how those firms built and managed security programs. There was no stick that forced the retail sector to think about their security posture, prior to PCI DSS. From 2004 to 2008, retailers stood up substantial programs in response to PCI DSS that created a new baseline of information security for that industry. Late in the 2000s, the stick got bigger with actual fines and adjustments to interchange that amounted to millions of dollars per year in some cases.

As great as this can be for some industries and information security laggards, it often creates a culture of "checking the box" in response to compliance initiatives. Managers of these firms will ask themselves, "What do I need to do to meet this compliance requirement?" instead of "What do I need to do to protect myself from the threats my firm faces today?" These questions do result in similar answers and actions at times, but compliance moves slowly—far more slowly than the invisible hackers do on the other side of the wire.

Strategic Countermeasures

Most executives and employees alike will say that they want to avoid a situation where their company suffers a breach. No one wants their company to suffer a breach. So how can a company full of diverse backgrounds pull off the seemingly impossible task of keeping the bad guys out?

There is no question that the odds are not in the favor of the defender in this war of bits and bytes. Attackers have seemingly unlimited resources and don't have to pay attention to the law. Defenders have budgets, boards of directors, and shareholders to answer to for every dollar they spend. You won't find a defender out there that isn't pining for some new tool or technology; that's why the real art of defense is knowing how to deploy your countermeasures in the most effective way.

Sun Tzu's *The Art of War* gives defenders many things to think about as they deal with their adversaries. One of his most profound teachings is to win the war without fighting. Attack the enemy's strategy instead of destroying the enemy himself. This is the essence of defense in a digital world, and this is where products like Micro Focus® Fortify can be an asset to the defenders.

Beyond the Checkbox

Compliance requirements can drive change in the information security field, but it's important for firms to deploy a diverse toolset that will cover more than just compliance. The Micro Focus Fortify suite does both by checking several compliance initiative requirements, such as PCI DSS, while bringing relevant and current countermeasures to information security professionals.

The Fortify application security suite has a long-standing history in application defense. The portfolio expands beyond static code analysis into a fully integrated suite of products that help meet several compliance initiatives and standards including PCI DSS, FISMA, HIPAA, NERC/FERC, and ISO 27000.

- **Fortify Static Code Analyzer (SCA)** is static application security testing (SAST) used by development groups and security professionals to analyze the source code of an application for security vulnerabilities. It reviews code and helps developers identify and resolve issues with less effort and in less time.
- **Fortify WebInspect** is a Web application security assessment solution designed to scan Web applications and Web services thoroughly for security vulnerabilities. It is essentially automated penetration testing.
- **Fortify Application Defender** is a runtime application self-protection solution (RASP). It deploys into Java and .NET applications by sitting in the byte code interpreters that run them. This allows for a quick deployment without changing source code and without a recompile. It can monitor and protect custom code or commercial off-the-shelf applications.

- **Fortify on Demand** (FoD) is an application security testing and program management solution that delivers Fortify's static and dynamic testing technologies with expert review and superior customer support. It enables customers to easily create, supplement, and expand a software security assurance program.

Fortify PCI Matrix

The following section has a matrix of all of the PCI DSS requirements that the Fortify solutions cover, as well as guidance for assessors who may be evaluating these tools for compliance. In some cases, the products can be used to justify compliance (as in, to verify that accounts with blank passwords do not exist). The comments and guidance only apply to in-scope applications and systems. The descriptions have been edited for brevity; the PCI Security Standards Council publishes the full version of PCI DSS on their website.

Fortify on Demand is software-as-a-service that can be used to meet all of the requirements met below by Fortify SCA and Fortify WebInspect. Assessors should review specific contracted services in conjunction with the in-scope applications to understand which elements may be used to justify compliance with PCI DSS or other security standards.

PCI Requirement	Fortify Solution	Assessor Guidance
1.3.7—Do not disclose private IP addresses and routing information to unauthorized parties.	(E) WebInspect will highlight areas in applications that expose private IP addressing.	Look for this finding in a WebInspect report to see if it was detected and present.
2.1—Always change vendor-supplied defaults and remove or disable unnecessary default accounts.	(E) WebInspect can be used to validate that Web applications do not have default passwords and accounts available for use. This is most useful for commercial off-the-shelf applications.	Firms can use WebInspect to demonstrate compliance with this requirement by scanning their commercial off-the-shelf applications and internally developed applications to look for default accounts. Scans should be reviewed and remediated before the applications are put into production.
2.2.4—Configure system security parameters to prevent misuse	(A) Both SCA and WebInspect can scan their targets to find configurations that would be deemed insecure. For example, WebInspect can find Web server configurations that may be insecure. SCA can look for insecure handing of information or connections to other systems or applications that use insecure protocols. (A) Application Defender can be deployed to protect applications from abuse of misconfigured system parameters in real time.	SCA and WebInspect can be used in support of this requirement and should be deployed as part of the go-live process. Application Defender can monitor and actively protect systems from misuse as a compensating control for production applications.

Continued on the next page

PCI Requirement	Fortify Solution	Assessor Guidance
3.2 (and sub requirements)—Do not store sensitive authentication data after authorization	(E) SCA analyzes a model of the source code and identifies potential locations in the code where sensitive information is being stored. Issues that may result in sensitive authentication data storage are highlighted for the developer to address.	Firms should use SCA and WebInspect in addition to manual reviews to ensure that there are no business process rules that may enable storage in certain situations after authorization.
3.4—Render PAN unreadable anywhere it is stored (including on portable digital media, backup media, and in logs)	(E) SCA analyzes a model of the source code and identifies potential locations in the code where sensitive information is stored in an insecure manner. Issues that may result in insecure PAN storage are highlighted for the developer to address. (E) WebInspect scans application network traffic looking for possible disclosure of sensitive information such as passwords, credit card numbers, and SSNs. Some of this sensitive information can also be used as PAN.	Firms should use SCA and WebInspect in addition to manual reviews.
3.6.1—Generation of strong cryptographic keys	(E) SCA analyzes situations whereby cryptographic keys are generated by an application using APIs. Any misuse will be detected and highlighted for a developer to address.	In the case that crypto key generation is required, SCA will help ensure that the code generating those keys uses methods to generate strong keys.
4.1—Use strong cryptography and security protocols to safeguard sensitive data during transmission.	(E) SCA can detect usage of insecure cryptographic algorithms and protocols in the source code. (E) WebInspect scans applications to help ensure they are using secure protocols for their communications. These products can be used for those applications that rely on software controls to enforce secure protocols and encryption. These products will identify any solutions that could be considered risky and flag them for follow-up.	Firms can use SCA or WebInspect in conjunction with encryption tools to ensure applications are not relying on insecure protocols or algorithms. For additional information, download our Crypto Manifesto whitepaper. The absence of strong cryptography at the source-code or application layer does not conclude the absence of it for the system. For example, mitigation may be applied for the entire system through a hardware-based control, which encrypts/decrypts all data leaving/entering the system.
4.2—Never send unprotected PANs by end-user messaging technologies	(E) SCA maps how data flows throughout an application, and in the case that the application has end-user messaging capability, will determine if sensitive information (like a PAN) can be dropped into those messages. (E) WebInspect scans application network traffic looking for possible disclosure of sensitive information such as passwords, credit card numbers, and SSNs. This data can ultimately find its way to a file system for storage, system console, printers etc. where it can be compromised.	As data flows through the various functions in an application, certain interactions could lead to data disclosure over these messaging systems. SCA and WebInspect can help ensure that data is not transmitted over end-user messaging technologies as part of the logic flow within an application.

Continued on the next page

PCI Requirement	Fortify Solution	Assessor Guidance
6.1—Establish a process to identify security vulnerabilities	(A) WebInspect and SCA can all be used to identify vulnerabilities before the applications enter production. Fortify on Demand can perform static and dynamic scans to identify vulnerabilities, as a service. Each product has its own use case and applicability to this requirement (E) Application Defender can identify vulnerabilities in production applications. It is especially helpful for applications that have not undergone previous security testing. Used properly, the Fortify suite can be used to meet this requirement entirely for any application, even where you do not have the source code (internally developed, third-party developed, code escrow, open source, etc.).	SCA should be used to examine application code in its entirety before it is pushed to production. WebInspect is used to scan applications to look for common vulnerabilities caused by vulnerable code or incorrect configuration. Application Defender can also protect vulnerabilities from exploitation, providing a compensating control. Processes should be put in place to ensure application security tools are applied consistently.
6.2—Ensure that all system components and software are protected from known vulnerabilities.	(A) WebInspect will identify known vulnerabilities in protocols or implementations that may exist outside the code of the custom application. For example, an application that leverages an insecure method of communication or a Web server that is misconfigured would be flagged for follow-up. (A) Application Defender can be used additionally to protect applications in production, and is especially helpful where you may not have the original source code. Application Defender detects and protects in real time from known vulnerabilities as well as Zero-Days in the applications.	WebInspect should be used to augment other existing methods of identifying vulnerabilities, specifically as it relates to custom code and other commercial off the shelf products. Firms deploying this technology should not rely solely on WebInspect, but use it to fill in the gaps other tools, such as NMAP, Nessus, Qualys, and Rapid7 miss. In addition, integrations with Sonatype and Black Duck enrich reporting data. Application Defender can be used to prevent vulnerabilities from being exploited before official patches or in-house fixes are released and deployed.
6.3—Develop internal and external software applications securely	(A) SCA supports this requirement by reviewing the source code for vulnerabilities that would violate PCI DSS or other security frameworks.	SCA, when used in conjunction with other secure development practices, can be used to identify vulnerabilities as a developer writes code, thus preventing them from showing up in production environments. This product demonstrates compliance with the requirement by checking code against various standards and security best practices.
6.3.1—Remove development, text and/or custom application accounts, user IDs, and passwords before applications become active.	(A) SCA and WebInspect will scan for poor authentication methods, such as hardcoded or weak passwords, which could end up in a production application.	SCA should be used in conjunction with other processes and methods, such as manual reviews of database entries, to help ensure that test or development credentials do not end up in deployed code.

Continued on the next page

PCI Requirement	Fortify Solution	Assessor Guidance
6.3.2—Review custom code prior to release	(A) SCA should be included in the go-live process for any custom code to help ensure that vulnerabilities are caught early. This process will augment existing code-review processes to bring scale to larger deployments.	SCA and WebInspect can be used to support this requirement as part of a larger program for custom code review. In addition, any business logic code that is intended to implement security-like functionality must be reviewed manually by domain experts.
	(A) WebInspect checks that Web code, when introduced into its deployed environment, does not have additional runtime issues only introduced there.	
6.5—Address common coding vulnerabilities in software-development processes.	(A) SCA scans source code to identify common coding vulnerabilities. All vulnerabilities will be flagged for follow-up and remediation advice provided.	SCA is used to scan pre-production source code to satisfy this requirement. Exceptions should be reviewed, patches deployed, and rescanned to verify the problems are fixed. Removing vulnerabilities early is far more efficient than later in the lifecycle.
	(A) WebInspect identifies violation of common problems such as the vulnerabilities represented in the OWASP Top 10 by directly mapping to each of the sub requirements in 6.5.	WebInspect should be used to scan working applications, ideally pre-production, so security flaws are removed before there is risk of exploit. Vulnerabilities can be resolved directly in the application or externally through additional controls.
	(A) Application Defender can continuously monitor applications in production for vulnerabilities not present when code was deployed. In addition, when vulnerabilities exist prior to patches or fixes, Application Defender will prevent those vulnerabilities from being exploited.	Using a RASP solution, like Application Defender, can help you immediately protect vulnerabilities in production applications. Vulnerabilities may exist due to lack of access to third-party code, because the vulnerability has not yet been remediated, or because new attack methods have created a new vulnerability. Review the configuration to ensure it is deployed on in-scope applications and protections are set to meet each of the requirements in 6.5.
6.5.1—Injection flaws, particularly SQL injection. Also consider OS Command Injection, LDAP, and XPath injection flaws, as well as other injection flaws.	(A) SCA scans through code to find potential injection flaws, and then provides the developer with guidance on how to address the vulnerability.	Injection flaws allow a hacker to inject a malicious query or other code, which can alter the logic flow and/or data query in the application. This can result in data loss or other unintended outcomes.
	(A) WebInspect scans through Web applications to look for various injection issues. Items found are highlighted for developers to address.	Ideally, vulnerabilities should be removed early, during development (SCA) and/or test (SCA and WebInspect). Sometimes vulnerabilities exist in production applications. Using a RASP solution, like Application Defender, can help you protect the vulnerability that you cannot eliminate.
	(A) Application Defender will protect applications from injection attacks by preventing the attack from successfully exploiting the vulnerable code.	
6.5.2—Buffer overflows	(A) SCA scans through code to find potential buffer overflow situations, and then provides the developer with guidance on how to address the vulnerability.	Buffer overflow occurs when a program, while writing data to a buffer, reads or writes past the buffer's boundary, writing over adjacent memory. This causes errors to occur that usually end execution of the application in an unexpected way.
	(A) WebInspect scans through Web applications to look for overflow opportunities. Items found are highlighted for developers to address.	SCA and WebInspect help you comply with this requirement by identifying potential buffer overflow so it may be corrected.

Continued on the next page

PCI Requirement	Fortify Solution	Assessor Guidance
6.5.3—Insecure cryptographic storage	(E) WebInspect finds situations where sensitive information is stored in locations without strong cryptography. (E) Application Defender can log cryptography exceptions to be examined by security operations.	Cryptographic storage external to the application will not be discovered. Firms using this product should ensure a proper manual review of those controls.
6.5.4—Insecure communications	(A) SCA scans source code to find insecure communications, and then provides the developer with guidance on how to address the vulnerability. (A) WebInspect scans through Web applications to look for insecure communications. Items found are highlighted for developers to address.	Insecure communication can happen in multiple areas of the application. WebInspect and SCA will highlight obvious areas. However, security teams should also perform a manual review to ensure the data they consider sensitive is communicated securely.
6.5.5—Improper error handling	(A) SCA scans source code to find poor error handling, and then provides the developer with guidance on how to address the vulnerability. (A) WebInspect scans through Web applications to look for error handling problems that can lead to the disclosure of underlying code, stability issues, and potential remote exploits. Items found are highlighted for developers to address. (A) Application Defender can protect unhandled exceptions from being exploited.	Proper error handling helps ensure that when things go wrong in an application, they go wrong safely and don't give an attacker information about your application, or a method by which they can break in. SCA and WebInspect can identify improper error handling for remediation. Application Defender can block exploits of improper error handling in production applications where this vulnerability remains.
6.5.6—All "high-risk" vulnerabilities identified in the vulnerability identification process	(A) SCA scans through code to find all high-risk vulnerabilities, and then provides the developer with guidance on how to address the vulnerability. (A) WebInspect scans through Web applications and identifies high-risk vulnerabilities for developers to address. (A) Application Defender will protects applications' high-risk vulnerabilities by preventing the attack from exploiting vulnerable code. It allows protection of newly discovered high-risk vulnerabilities without waiting for remediation and rescanning or patches from vendors. (A) In addition to static and dynamic scans as-a-service, Fortify on Demand offers a Continuous Monitoring capability that looks for new (unscanned) applications and new high-risk vulnerabilities.	Fortify software and services (Fortify on Demand) prioritize vulnerabilities by type of threat/risk. Be sure that high-risk vulnerabilities are remediated by developers or vendors who are responsible for the application's source code. The product used is determined by the type of application and by the unique processes used by your enterprise to deploy it. Ideally, you should remove vulnerabilities early, during development and/or test. When using third-party code, you may not have access to the source code for remediation. In this case, using a RASP solution, like Application Defender, can help you protect the vulnerability that you cannot eliminate.

Continued on the next page

PCI Requirement	Fortify Solution	Assessor Guidance
6.5.7—Cross-site scripting (XSS)	(A) SCA scans through code to find potential XSS flaws, and then provides the developer with guidance on how to address the vulnerability. (A) WebInspect scans through Web applications to look for various XSS issues. Items found are highlighted for developers to address. (A) Application Defender will protect applications from XSS attacks by preventing the attack from exploiting vulnerable code.	XSS enables attackers to inject script into Web pages causing them to behave in a different, potentially malicious way. Ideally, vulnerabilities should be removed early, during development (SCA) and/or test (SCA and WebInspect). Sometimes vulnerabilities exist in production applications. Using a RASP solution, like Application Defender, can help you protect the vulnerability that you cannot eliminate.
6.5.8—Improper access control	(A) SCA scans through code to find improper access controls, and then provides the developer with guidance on how to address the vulnerability. (A) WebInspect scans through Web applications to look for various access control issues, such as directory traversal, restricted URL access, and private IP leakage. Items found are highlighted for developers to address.	Improper access control should be identified and removed early, during development (SCA) and/or test (SCA and WebInspect).
6.5.9—Cross-site request forgery (CSRF)	(A) SCA scans through code to find potential CSRF flaws, and then provides the developer with guidance on how to address the vulnerability. (A) WebInspect scans through Web applications to look for various CSRF issues. Items found are highlighted for developers to address.	In CSRF, malicious commands are transmitted from a user that the website trusts. Like XSS, it causes the Web application to behave in an unexpected way. CSRF should be identified and removed early, during development (SCA) and/or test (SCA and WebInspect).
6.5.10—Broken authentication and session management	(A) SCA scans through code to find potential authentication and session management flaws, and then provides the developer with guidance on how to address the vulnerability. (A) WebInspect scans through Web applications to look for various authentication and session management issues. Items found are highlighted for developers to address. (A) Application Defender will protect applications from session-related attacks (such as session hijacking).	Firms using these products should ensure they are choosing the right tool for the right applications to maximize effectiveness. Ideally, vulnerabilities should be removed early, during development (SCA) and/or test (SCA and WebInspect). Using a RASP solution, like Application Defender, can help you protect the vulnerability that you cannot eliminate.

Continued on the next page

PCI Requirement	Fortify Solution	Assessor Guidance
6.6—For public-facing Web applications, address new threats and vulnerabilities on an ongoing basis and ensure these applications are protected against known attacks.	(A) WebInspect is one of our products that can be used to scan applications for this requirement. The simplest way to meet the requirement with this tool is to include it in your go-live process, such that any time changes are deployed to your app, the app is scanned. Ensure all identified items are fixed before the code progresses to the next phase. (A) Application Defender protects any Java or .NET applications in their runtime environment in order to identify new and existing vulnerabilities. It monitors and protects against known and unknown vulnerabilities in real time. As new exploits create new vulnerabilities, Application Defender provides continuing protection without needing to change code or recompile.	Firms using these technologies need to be explicit in how and where they are deployed for security and compliance purposes. For example, Application Defender should not be listed as a control for a Web-based PHP app, but it could be used to protect a Java-based Web server interpreting the PHP code or WebInspect could be used. Either of these products will cover any application deployed publicly. Frequent updates to the rule packs will help ensure that new vulnerabilities will be detected early.
7.1.2—Restrict access to privileged user IDs to least privileges necessary to perform job responsibilities.	(E) Some programming APIs provide excessive privileges, which could lead to a failure to restrict certain functions to a User ID. SCA can detect and alert on the usage of these APIs.	Most software relies on third-party APIs for functionality. SCA will help ensure that those APIs are properly restricted to programmatically enable compliance to this requirement.
7.2—Establish an access control system(s) for systems components that restricts access based on a user's need to know, and is set to "deny all" unless specifically allowed.	(E) SCA identifies code that may override, weaken, or be vulnerable in ways that reduces the assumed level of access control in an application.	SCA should be used to help detect software vulnerabilities that could lead to weakened access control.
8.1.8—If a session has been idle for more than 15 minutes, require the user to re-authenticate to re-activate the terminal or session.	(E) SCA and WebInspect will test session timeouts in applications to determine if this requirement is met.	Leveraging these tools adds an extra layer of security to your environment by ensuring that the application times out even if the terminal does not.
8.2.1—Using strong cryptography, render all authentication credentials (such as passwords/phrases) unreadable during transmission and storage on all system components.	(E) SCA and WebInspect can detect the misuse of cryptographic APIs that detect poor key management and stored passwords. They can identify storing sensitive information in clear text and storing sensitive information in storage locations that do not have strong crypto.	Weak cryptographic usage can spring from misconfiguration or outdated code. SCA validates that the application uses cryptographic code properly.

Continued on the next page

PCI Requirement	Fortify Solution	Assessor Guidance
8.2.3—Passwords/ passphrases must meet minimum length and entropy requirements.	(E) WebInspect will test applications to determine if a password can be entered with less than the minimum entropy requirements of PCI DSS.	Assessors can rely on the output from WebInspect as a source to test how an application enforces entropy requirements.
10.2.1—All individual user accesses to cardholder data	(E) For select examples, SCA will identify areas of cardholder data access that do not translate to an accompanying log entry.	SCA has tests that will show missing log entries. Developers should ensure that logging processes are intact when routines access payment card data. Assessors can use evidence of this product as a checkpoint for this requirement.
10.2.4—Invalid logical access attempts	(E) SCA will advise developers when invalid access attempts do not result in a logged entry. For apps that are handling logical access control internally, this check will help ensure that the app is performing according to requirements by looking for log entries on invalid logical access attempts.	Malicious users often will try multiple ways to gain access to information or functionality they are not permitted to access. Early warning detection systems require valid log entries to detect bad behavior and alert on it. Assessors can use evidence of this product as a checkpoint for this requirement.
10.3.4—Success or failure indication	(E) SCA will help ensure that success and/or failure are properly logged as part of the normal event log. Oftentimes, developers log failed attempts but miss successful ones.	Log entries by themselves must indicate success or failure to assist in early detection and for compliance validation. Assessors can use evidence of this product as a checkpoint for this requirement.
10.5.2—Protect audit trail files from unauthorized modifications	(E) SCA can detect software issues that may allow log-forging attacks to be successful, which may allow unauthorized modification to occur.	Log integrity is critical if you need to use those logs to find a breach or an insider. An attack that modifies the audit trail can cover the tracks of an attacker.
11.3.1—Perform external penetration testing at least annually and after any significant infrastructure or application upgrade or modification.	(A) WebInspect automates penetration testing performed against the application. The deep scanning capability helps penetration testers identify potentially weak spots for further manual testing. (A) As a potential compensating control, SCA can be considered a valid method to meet this requirement as well, as long as all of the code the application relies on is included in the scan.	External penetration testing firms may use either product to satisfy this requirement. Assessors should review how the tool is used in support of these efforts to ensure its completeness.
11.3.2—Perform internal penetration testing at least annually and after any significant infrastructure or application upgrade or modification.	(A) WebInspect automates penetration testing performed against the application. The deep scanning capability helps penetration testers identify potentially weak spots for further manual testing. (A) As a potential compensating control, SCA can be considered a valid method to meet this requirement as well, as long as all of the code the application relies on is included in the scan.	Internal penetration testing groups may use either product to satisfy this requirement. Assessors should review how the tool is used in support of these efforts to ensure its completeness.
12.2—Implement a risk-assessment process	(E) While the products here are not risk-assessment tools, they can all be used in support of a risk-assessment methodology. Fortify products have the ability to show risk levels of specific vulnerabilities and exploit events. The results should feed into your risk-assessment process.	Outputs from the products should be incorporated into the risk-assessment process that the firm uses for this requirement.

Table 1. PCI requirements matrix for PCI DSS v3.2

Table Index:

(A): Directly ADDRESSES PCI DSS requirement.

(B): EVIDENCE from the product can be used to demonstrate PCI DSS compliance.

Notes:

(1): Fortify on Demand can be used as an on-demand alternative to on-premises SCA or WebInspect.

(2): PCI requirements that do not map to a Fortify solution are not listed.

Fortify Solution Overview

Static Code Analyzer (SCA)

The first, and classic, product in the Fortify suite is the Static Code Analyzer (SCA). Development groups and security professionals use SCA to analyze the source code of an application for security issues. The speed by which new code is built, tested, and pushed into production naturally creates an opportunity for mistakes to end up in production. SCA identifies root causes of software security vulnerabilities and delivers results with line-of-code remediation guidance—making it easy for your team to prioritize their fixes to solve the biggest problems first.

WebInspect

WebInspect does for Web applications what SCA does for compiled applications. It is designed to interact with Web applications in the same manner that users do. Developers and security professionals use it to run detailed automated testing against Web applications faster than they can do it manually. WebInspect goes deep into all of the available pathways that a Web application might take, and performs testing to find security vulnerabilities that an attacker could take advantage of to ultimately cause harm to the data behind the application.

WebInspect tests the dynamic behavior of running applications to identify configuration issues and security vulnerabilities. It mimics real-world security attacks and provides comprehensive dynamic analysis of complex Web applications and services.

Application Defender

Application Defender rounds out the Security offerings by monitoring and protecting production applications to prevent attacks from successfully exploiting software vulnerabilities. Undetected vulnerabilities put you at risk for cross-site scripting, SQL injection, malware, customer data theft, and other attacks that force your applications to reveal data and do harm. Finding the vulnerabilities is only half of the battle—fixing them can be challenging or impossible. Application Defender helps you monitor and protect your software vulnerabilities. This is particularly useful for applications that you don't have source code for or as a compensating control. Additionally, Application Defender can instrument applications to provide logs of application and user behavior and security exceptions. These logs can be sent to a Security Operations Center, Log Manager, or any system capable of ingesting syslog files.

Fortify on Demand (FoD)

Fortify on Demand brings the power of all of these solutions directly to you to be consumed on a service basis. This is great for companies who don't have the in-house expertise or resources to do detailed testing of applications, or those who need to get started right away. Firms wanting to understand their threat exposure and associated risk can leverage Fortify on Demand's Continuous Application Monitoring service to discover, profile, and assess external Web apps based on an initial seed item such as a URL, brand-related keywords, or an IP range.

Static Analysis

Fortify on Demand is a critical part of a secure SDLC process that takes an internally developed application through rigorous testing as part of the go-live process, ensuring that once applications do get to production their attack surface is at a minimum. Development teams can upload the source code or binaries of the application to the on-demand portal. Once received, both automated and manual tests are performed followed by a review of the results. Detailed results including line-level detail are available in standard or custom reports. For firms that need guidance on how to solve the issues, the Fortify on Demand account management team is there to support their findings after the engagement. Premium support services are also available depending on the need. Analysis can also be done on third-party and mobile applications.

Dynamic Analysis

For Web applications, three levels of security tests can be performed by providing a publicly-accessible URL to the Web-based application. Thorough tests include both automated and manual scanning of the application's security resilience, and note areas where the application has security risks to be addressed. Performing analysis on both functioning applications and source code will ensure that every problem is properly diagnosed and documented for repair. The same support options are available for dynamic applications as well as premium support services, depending on the need. Take advantage of continuous monitoring and our digital patching capabilities to secure your perimeter while you remediate..

Which Tool for the Job?

The completeness of security protection offers from often leads security professionals to ask, "What product is the right one for my needs?" While all of the products used together will provide maximum protection, some use cases may favor one over another, or perhaps a custom bundle. This section will provide guidance to prospective buyers to understand how best to use each tool to meet their needs.

Use Cases

In-House Web Applications

Web applications often rely on logic provided by third-party Web server code, database code, compiled and interpreted logic code, and presentation code. WebInspect on-premises or Fortify on Demand works through the Web-accessible layer of the application to find and document security flaws and is the standard first step. If there are compiled objects that your Web application leverages, be sure to run those through SCA to ensure those vulnerabilities are covered. Finally, Application Defender will prevent any known or unknown vulnerabilities from being exploited in production.

Compiled Applications

In-house applications that are compiled, such as thick client, desktop, or mobile applications, should leverage SCA on premise or Fortify on Demand to find and fix vulnerabilities in those codebases.

Third-Party Web Applications

For third-party Web applications where you may not have access to all of the source code or the developers, WebInspect will find and document security flaws visible to an attacker. And, as with in-house applications, Application Defender will prevent any known or unknown vulnerabilities from being exploited in production. For a full-service option, Fortify on Demand's vendor management service provides an easy-to-use approach that doesn't require source code and allows the vendor to test applications, resolve issues, and then publish a report to the procurer. Fortify on Demand serves as an independent third party and system of record for conducting a consistent, unbiased analysis.

Getting Started or Ad-Hoc Requests

If you lack security expertise or resources, consider Fortify on Demand. This service leverages everything you see above and more, but is consumed as a service. It is ideal to get started with application security or to manage your complete application security program. Contact your Micro Focus representative to get a custom quote that will help you with your specific needs.

Learn More At

www.microfocus.com/appsecurity

Additional contact information and office locations:
www.microfocus.com

www.microfocus.com