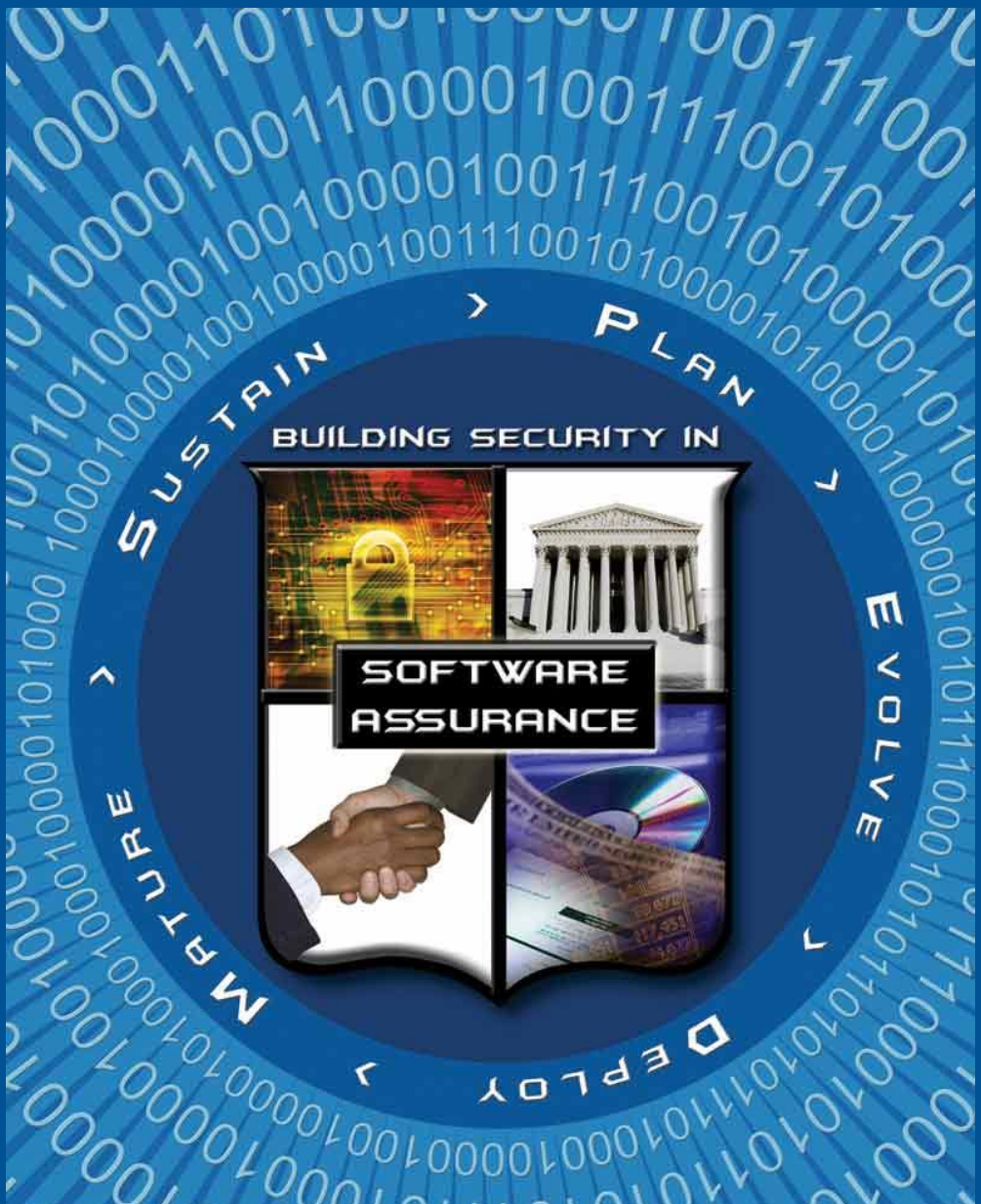
Software Supply Chain Risk Management & Due-Diligence

Software Assurance Pocket Guide Series:
Acquisition & Outsourcing, Volume II
Version 1.2, June 16, 2009



Software Assurance (SwA) Pocket Guide Resources

This is a resource for 'getting started' in selecting and adopting relevant practices for delivering secure software. As part of the Software Assurance (SwA) Pocket Guide series, this resource is offered for informative use only; it is not intended as directive or presented as being comprehensive since it references and summarizes material in the source documents that provide detailed information. When referencing any part of this document, please provide proper attribution and reference the source documents, when applicable.

This volume of the Software Assurance Pocket Guide series focuses on software supply chain risk management and due-diligence. Buyers and evaluators of software and services can gain security risk-based insight. They can put suppliers on notice that consumers are concerned about software security and the risks to their organizations that are attributable to exploitable software.

At the back of this pocket guide are references, limitation statements, and a listing of topics addressed in the SwA Pocket Guide series. All SwA Pocket Guides and SwA-related documents are freely available for download via the SwA Community Resources and Information Clearinghouse at http://buildsecurityin.us-cert.gov/swa.

Acknowledgements

The SwA Forum and Working Groups function as a stakeholder mega-community that welcomes additional participation in advancing software security and refining SwA-related information resources that are offered free for public use. Input to all SwA resources is encouraged. Please contact Software.Assurance@dhs.gov for comments and inquiries.

The SwA Forum is composed of government, industry, and academic members. The SwA Forum focuses on incorporating SwA considerations in acquisition and development processes relative to potential risk exposures that could be introduced by software and the software supply chain.

Participants in the SwA Forum's Acquisition & Outsourcing Working Group collaborated in developing the material used in this pocket guide as a step in raising awareness on how to incorporate SwA considerations throughout the acquisition process.

Information contained in this pocket guide is primarily derived from "**Software Assurance in Acquisition: Mitigating Risks to the Enterprise**" available through the SwA Community Resources and Information Clearinghouse at https://buildsecurityin.us-cert.gov/swa/acqact.html. The full document was also codeveloped with representatives from the Information Resources Management College (IRMC) http://www.ndu.edu/irmc/ and published through the National Defense University Press; so a copy can be accessed at http://www.ndu.edu/inss/press/NDUPress_Occasional_Papers.htm.

Special thanks to the Department of Homeland Security (DHS) National Cyber Security Division's Software Assurance team who provided much of the support to enable the successful completion of this guide and related SwA documents.

Overview

The SwA Pocket Guide Series is produced by collaborators within the SwA Forum and its working groups. The focus of this resource is to increase awareness for the need to include SwA and identify best practices in the acquisition of software. This pocket guide reflects collective information for managing risk in the software supply chain and performing SwA Due-Diligence. It aids in selecting from among other resources available on the various aspects of software security and assurance.

Using SwA Due-Diligence questionnaires puts suppliers on notice that consumers are concerned about the security of the software and risks to their organizations that are attributable to exploitable software. SwA Due-Diligence questionnaires can assist those involved with acquiring or purchasing software or outsourcing software development and support services. The questionnaires can assist in obtaining additional information about the software and its supply chain. In this context, due-diligence involves taking reasonable steps to ensure that software or a software-intensive system not only meets functional and technical requirements, but also addresses SwA concerns. The intent is to inform evaluators and acquirers of potential risks associated with the software they are considering for purchase. The questionnaires support those opting to exercise security-enhanced Due-Diligence; they provide a means to gather, in advance, some of the information needed to make risk-based decisions about the security of the software.

Due-Diligence Questionnaires' Objectives:

- *Enhance processes and practices for acquiring and delivering secure software,*
- *Assist in assessing software security and managing supply chain risks,*
- Aid in understanding risk exposure attributable to software and suppliers' processes.

The questionnaires are useful tools for those evaluating or purchasing software on behalf of enterprises and users, as well as for integrators who incorporate software from third-party suppliers as part of their offering of software as part of systems or services. The questionnaires can be used in whole or in part. Some questions may apply, some may not; some may be added, and some may be tailored. The questionnaires are a means for gathering relevant information to support decision making. The questionnaires support the exercise of SwA Due-Diligence by acquirers and others evaluating the software and the capabilities of the suppliers. Using SwA questions and the relevant responses (or lack of answers) help to identify potential risks. The questionnaires are tools. They are not checklists or complete listings of all possible software security concerns. Some examples when questionnaires may be effectively applied include:

- Conducting market research or developing vendor surveys for risk-based trade-off studies;
- Gathering information on given software products or suppliers to determine which software application to procure or which supplier to consider for contracting software-related services;
- Developing a request for information to gather information for a major software development program;
- Developing a request for proposal for building a critical software-intensive system, including an information system or embedded system platform;
- Developing work statements, contract language or evaluation criteria:
  - Some questions can be transformed into SwA requirements that are then included in a work statement.

- Some questions can be transformed into other contractual language, such as terms and conditions.
- A questionnaire can be incorporated as part of evaluation factors for award.

An Imperative for SwA in Acquisition

Often the common practice in acquisition is to accept software that satisfies functionality with little regard for specifying, determining or assuring security properties – increasing the risk exposure to users. Many purchasing organizations and acquirers continue to accept software riddled with exploitable flaws and other security vulnerabilities. This, in part, may be due to acquisition policies and procedures that do not ensure that security is a main concern of software.

In addition, acquirers may not be aware of the increased life cycle costs and increased risk exposure to the organization attributable to software that is not secure. Purchasing secure software might entail moderate upfront costs to the acquisition project (especially in dealing with suppliers who have not incorporated security in their development processes); however, the price paid in lost time and resources to continually fix or patch a vulnerable software component can run as much as three times the initial purchase of secure software. Many organizations fall behind in properly patching vulnerable software due to those exponential costs, leaving them exposed to attack. Dangers may be attributable to software errors or other vulnerabilities to include the unknowing acceptance of software that contains malicious code.

The Imperative. Rampant worldwide increase in exploitation of software vulnerabilities demands that purchasing organizations require that not only have checks for acceptable functionality been provided, but also that provisions are made to achieve acceptable software assurance. Although the prevailing practices of many software suppliers have failed to produce safe, secure, reliable, and dependable software, some segments of the software industry are moving toward rigorous software development practices to minimize software errors and other vulnerabilities. Shifts that focus on security being 'built in prior to being put into use' minimize opportunities for adversaries to exfiltrate data or deny and degrade services.

Purchasing decisions and the acquisition process can be leveraged to promote security-enhanced software development practices and to facilitate the delivery of trustworthy software. Most software security requirements decisions are made during the acquisition process, in addition to acceptance and implementation decisions. Security cannot be "bolted on" after the software product is delivered.

As the users' representative in dealing with suppliers, the acquirer is important in the line of defense to ensure that safe, secure, reliable, and dependable software is delivered. To that end, the acquirer should provide an acquisition process with practices that ensures the continuity of essential business operations across a wide range of potential emergencies and/or attempted exploitations.

Those purchasing software or evaluating suppliers, and determining if the software is 'fit for purpose,' have a Due-Diligence responsibility to account for the needs of users. As such, risk management must consider both risks to the acquisition project and residual risks that might be passed to the users as a result of not being adequately addressed in purchasing software and support services for software-intensive systems.

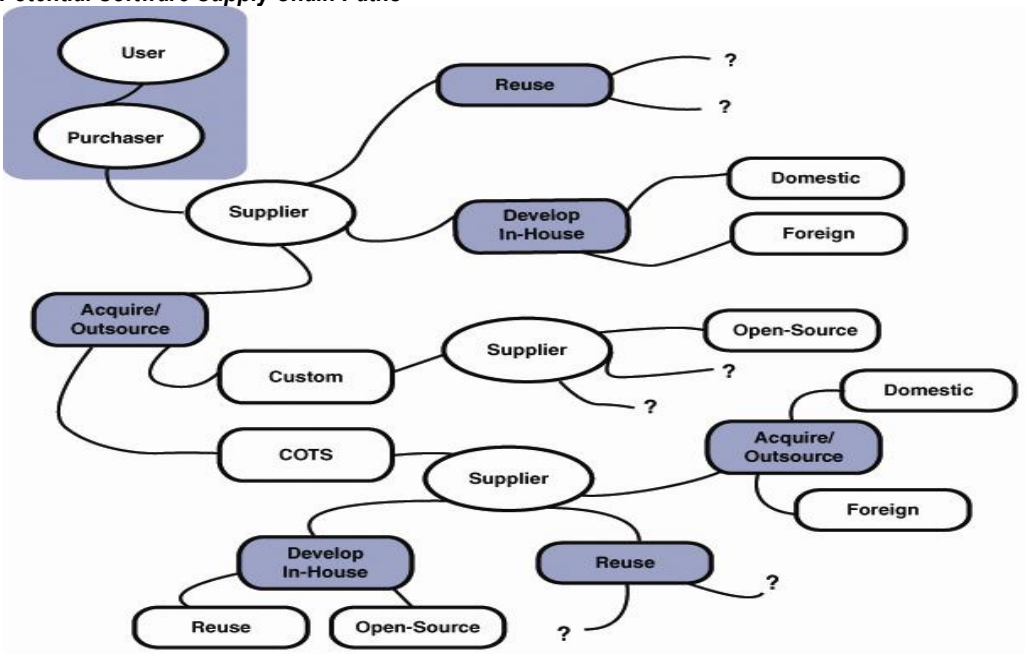
Software Vulnerabilities Side-Effects

- *Unintentional errors leading to faulty operations,*
- *Destruction of information or major disruption of operations,*
- *Insertion of malicious code,*
- *Theft of sensitive, personal or classified information,*
- *Changed product.*

Information Assurance vis-à-vis SwA. Information assurance relates to measures that protect and defend information and information systems by ensuring their availability, integrity, authentication, confidentiality, and non-repudiation. These measures include providing for restorations of information systems by incorporating protection, detection, and reaction capabilities. Information systems include the software that controls the system and processes data and information. Therefore, measures must be used to protect the systems from software vulnerabilities and unintended software processing that expose a system to compromises in availability, integrity, and other security properties. SwA provides those measures.

Figure 1 shows some of the potential paths software can take before it is acquired and put into use. Each organization in the supply chain path has an influence on the security or exploitability of the software. Knowing who produced the software and being able to determine if they use security-aware practices in producing software, can provide the requisite transparency for informed risk-based decision-making in purchasing software or contracting for software services.

Figure 1 – Potential Software Supply Chain Paths



SwA Concern Categories

To focus on mitigating risks attributable to software, and the suppliers of software and related services, the SwA Due-Diligence questionnaires have been collaboratively developed and organized to represent a logical grouping of SwA concerns. Table 1 relates SwA concern categories to a risk description and purpose for gathering the data. The identified risks are examples and are not intended to be a complete list of all risks.

Table 1 –SwA Concern Categories

SwA Concern Categories	Risks	Purpose for Questions
Software History and Licensing	The software supplier's development practice in using code of unknown origin may be unable to produce trustworthy software.	To address supply chain concerns and identify specific risks pertaining to the history/pedigree of the software during any and all phases of its life cycle that should have been considered by the supplier. This point addresses supply chain concerns.

Table 1 –SwA Concern Categories		
SwA Concern Categories	**Risks**	**Purpose for Questions**
Development Process Management	If supplier project management does not perceive the value of SwA and enforce best practices, they will not be consistently implemented.	To determine whether project management enforces software assurance–related best practices.
Software Security Training and Awareness	Developers unaware of software assurance best practices are likely to implement software with security flaws (making it more susceptible to attack).	To determine whether training of developers in SwA best practices is a supplier policy and practice.
Planning and Requirements	If nonfunctional requirements (security, quality, safety) are not specified, developers will not implement them.	To determine whether the supplier's requirements analysis process explicitly addresses SwA requirements.
Architecture and Design	The software may be designed without considering security or minimization of exploitable defects.	To determine how security is considered during the design phase.
Software Development	If developers lack qualified tools or if personnel are allowed to inappropriately access or change configuration items in the development environment, then delivered software might have unspecified features. The supplier might lack sufficient process capability to deliver secure products, systems or services.	To ascertain that the supplier has and enforces policies and SwA practices in the development of software that use secure software development environments to minimize risk exposures.
Built-in Software Defenses	The software may lack preventive measures to help it resist attack effectively and proactively.	To ensure that capabilities are designed to minimize the exposure of the software's vulnerabilities to external threats and to keep the software in a secure state regardless of the input and parameters it receives from its users or environment.
Component Assembly	Insufficient analysis of software components used to assemble larger software packages may introduce vulnerabilities to the overall package.	To ensure that the software components are thoroughly vetted for their security properties, secure behaviors, and known types of weaknesses that can lead to exploitable vulnerabilities.
Testing	Software released with insufficient testing may contain an unacceptable number of exploitable defects.	To determine whether the appropriate set of analyses, reviews, and tests are performed on the software throughout the life cycle which evaluate security criteria.
Software Manufacture and Packaging	Vulnerabilities or malicious code could be introduced in the manufacturing or packaging process.	To determine how the software goes through the manufacturing process, how it is packaged, and how it remains secure.
Installation	The software may not install as advertised and the acquirer may not get the software to function as expected.	To ensure the supplier provides an acceptable level of support during the installation process.
Assurance Claims and Evidence	Supplier assurance claims (with supporting evidence) may be non-existent or insufficiently verified.	To determine how suppliers communicate their claims of assurance; ascertain what the claims have been measured against, and identify at what levels they will be verified.
Support	Supplier ceases to supply patches and new releases prior to the acquirer ending use of software. Vulnerabilities may go unmitigated.	To ensure understanding of supplier policy for security fixes and when products are no longer supported.
Software Change Management	Weak change control procedures can corrupt software and introduce new security vulnerabilities.	To determine whether software changes are adequately assessed and verified by supplier management.
Timeliness of Vulnerability Mitigation	Sometimes it can be extremely difficult to make a software supplier take notice and repair software to mitigate reported vulnerabilities.	To ensure security defects and configuration errors are fixed properly and in a timely fashion.

Table 1 –SwA Concern Categories		
SwA Concern Categories	**Risks**	**Purpose for Questions**
Individual Malicious Behavior	A developer purposely inserts malicious code, and the supplier lacks procedures to mitigate risks from insider threats within the supply chain.	To determine whether the supplier has and enforces policies to minimize individual malicious behavior.
Security "Track Record"	A software supplier that is unresponsive to known software vulnerabilities may not mitigate/patch vulnerabilities in a timely manner.	To establish insight into whether the supplier places a high priority on security issues and will be responsive to vulnerabilities they will need to mitigate.
Financial History and Status	A software supplier that goes out of business will be unable to provide support or mitigate product defects and vulnerabilities.	To identify documented financial conditions or actions of the supplier that may impact its viability and stability, such as mergers, sell-offs, lawsuits, and financial losses.
Organizational History	There may be conflicting circumstances or competing interests within the organization that may lead to increased risk in the software development.	To understand the supplier's organizational background, roles, and relationships that might have an impact on supporting the software.
Foreign Interests and Influences	There may be controlling foreign interests (among organization officers or from countries) with malicious intent to the users' country or organization planning to use the software.	To help identify supplier companies that may have individuals with competing interests or malicious intent to a domestic buyer/user.
Service Confidentiality Policies	Without policies to enforce client data confidentiality/ privacy, acquirer's data could be at risk without service supplier liability.	To determine the service provider's confidentiality and privacy policies and ensure their enforcement.
Operating Environment for Services	Operating environment for the services may not be hardened or otherwise secure.	To understand the controls the supplier has established to operate the software securely.
Security Services and Monitoring	Insufficient security monitoring may allow attacks to impact services.	To ensure software and its operating environment are regularly reviewed for adherence to SwA requirements through periodic testing and evaluation.

Software Assurance Due-Diligence Questionnaires

The following contains example software assurance Due-Diligence questionnaire for several types of software. Acquirers and those evaluating software and suppliers may use a questionnaire as a means for gathering relevant information to support decision making. *When using the questionnaires, acquirers and evaluators should tailor the questions to suit their particular situations* since not all questions are applicable and other acquisition-specific questions may be more appropriate. *The questionnaires are intended to solicit information from the suppliers. They are not checklists, nor are they a complete listing of all possible SwA/security concerns.* Expertise in software, acquisition, information assurance, and the user environment – as well as common sense – is critical to risk-based decisions regarding the acquisition of software. Questions should be reviewed prior to submission, and responses assessed, by knowledgeable SwA subject matter experts or other appropriate functional experts.

In addition, when using questionnaires as a tool, acquirers should ensure that they solicit evidence (or can gain access to evidence) to support supplier responses when applicable. In some cases, suppliers might be unable or unwilling to provide answers (or supporting evidence) to all questions that might be asked by purchasing organizations. In such instances that lack of transparency might represent a risk to the using organization because they would lack sufficient information to make better risk-based decisions. In some cases, suppliers have answered similar sets of questions; so acquirers and evaluators should consider the 'reuse' of those responses.

Acquirers and evaluators should only use Due-Diligence questions for eliciting information that contribute to the purchasing decisions that factor in the security needs of users. If a response would not influence purchasing decisions, then the question should not be posed to the supplier.

Table 2 lists questions to consider asking when evaluating suppliers and sources of Commercial-Off-The-Shelf (COTS) software (both Proprietary and Open-Source), Government-provided (often referred as GOTS) software, and Custom-developed software and the suppliers of the software.

Table 2- Questions for COTS (Proprietary & Open Source), GOTS, and Custom Software

No.	Question	COTS Proprietary	COTS Open-Source	GOTS	Custom
1	Can the pedigree of the software be established? Briefly explain what is known of the people and processes that created the software.	✓	✓	✓	✓
2	Explain the change management procedure that identifies the type and extent of changes conducted on the software throughout its life cycle.	✓		✓	✓
3	What type of license(s) are available for the open source software? Is it compatible with other software components in use? Is indemnification provided, and will the supplier indemnify the purchasing organization from any issues in the license agreement? Explain.	✓	✓		✓
4	Is there a clear chain of licensing from original author to latest modifier? Describe the chain of licensing.	✓			
5	What assurances are provided that the licensed software does not infringe upon any copyright or patent? Explain.	✓		✓	✓
6	Does the company have corporate policies and management controls in place to ensure that only corporate-approved (licensed and vetted) software components are used during the development process? Explain.	✓			✓
7	Are licensed software components still valid for the intended use?	✓		✓	
8	Is the software in question original source or a modified version?		✓		
9	Has the software been reviewed to confirm that it does not infringe upon any copyright or patent?	✓	✓		✓
10	How long has the software source been available? Is there an active user community providing peer review and actively evolving the software?	✓	✓		
11	Does the license/contract restrict the licensee from discovering flaws or disclosing details about software defects or weaknesses with others (e.g., is there a "gag rule" or limits on sharing information about discovered flaws)?	✓			✓
12	Does the license/contract restrict communications or limit the licensee in any potential communication with third-party advisors about provisions for support (e.g., is there a "gag rule" or limits placed on the licensee that affect ability to discuss contractual terms or breaches) regarding the licensed or contracted product or service?	✓			✓
13	Does software have a positive reputation? Does software have a positive reputation relative to security? Are there reviews that recommend it?	✓	✓		
14	Is the level of security where the software was developed the same as where the software will operate?			✓	✓
	Development Process Management				
15	What are the processes (e.g., ISO 9000, CMMI, etc.), methods, tools (e.g., IDEs, compilers), techniques, etc. used to produce and transform the software (brief summary response)?	✓		✓	✓
16	What security measurement practices and data does the company use to assist product planning?	✓			✓
17	Is software assurance considered in all phases of development? Explain.	✓		✓	✓
18	How is software risk managed? Are anticipated threats identified, assessed, and prioritized?	✓		✓	✓
	Software Security Training and Awareness				
19	Describe the training the company offers related to defining security requirements, secure architecture and design, secure coding practices, and security testing.	✓			✓
20	Does the company have developers that possess software security related certifications (e.g., the ISC2 CSSLP, SANS SSI secure coding certifications, etc.)?	✓			✓
21	Describe the company's policy and process for professional certifications and ensuring certifications are valid and up-to-date.	✓			✓
	Planning and Requirements				
22	Are there some requirements for security that are "structured" as part of general releasability of a product and others that are "as needed" or "custom" for a particular release?	✓			✓

Table 2- Questions for COTS (Proprietary & Open Source), GOTS, and Custom Software					
No.	Question	COTS Proprietary	COTS Open-Source	GOTS	Custom
23	What process is utilized by the company to prioritize security-related enhancement requests?	✓			✓
24	What review processes are implemented to ensure that nonfunctional security requirements are unambiguous, traceable and testable throughout the entire Software Development Life Cycle (SDLC)?	✓			✓
26	Are security requirements developed independently of the rest of the requirements engineering activities, or are they integrated into the mainstream requirements activities?	✓			✓
26	Are misuse/abuse cases derived from the application requirements? Are relevant attack patterns used to identify and document potential threats?	✓			✓
27	What tool(s) does the company use for requirements management?				✓
28	If an agile development method is used, how formally are requirements documented?	✓		✓	✓
29	Were security and quality requirements included in the requirements analysis process?	✓		✓	✓
	Architecture and Design				
30	What threat assumptions were made, if any, when designing protections for the software and information assets processed?	✓		✓	✓
31	What security design and security architecture documents are prepared as part of the SDLC process?	✓		✓	✓
32	How are design documents for completed software applications archived?	✓			✓
33	What threat modeling process, if any, is used when designing the software protections?	✓			✓
34	What analysis, design, and construction tools are used by the software design teams?				✓
34	Are design documents for the software archived and available? Are software interfaces described in published documentation?			✓	✓
36	How are confidentiality, availability, and integrity addressed in the software design?			✓	✓
	Software Development				
37	What are/were the languages and non-developmental components used to produce the software (brief summary response)?	✓		✓	✓
38	What secure development standards and/or guidelines are provided to developers?	✓		✓	✓
39	Are tools provided to help developers verify that the software they have produced has a minimal number of weaknesses that could lead to exploitable vulnerabilities? What are the tools, and how have they been qualified? What is the breadth of common software weaknesses covered (e.g., specific CWEs)?	✓			✓
40	In preparation for release, are undocumented functions in the software disabled, test/debug code removed, and source code comments sanitized?	✓		✓	✓
41	Does the company have formal coding standards for each language in use? If yes, how are they enforced? How often are these standards and practices reviewed and revised?	✓		✓	✓
42	Does the software development plan include security peer reviews?	✓			✓
43	Does the organization incorporate security risk management activities as part of the software development methodology? If yes, will a copy of the documentation of this methodology be available or information on how to obtain it from a publicly accessible source?	✓			✓
44	Does the organization establish contractually binding agreements with their own developers and/or with their third-party developers regarding the ownership and/or licensing of intellectual property?				✓
45	Does the software use closed-source Application Programming Interfaces (APIs) that have undocumented functions?	✓			✓
46	Are there contractual recourses that the organization can take if a third-party developer delivers software that contains malicious code?				✓
47	Does the organization ever perform site inspections/policy compliance audits of its domestic development facilities? Of its foreign-based facilities? Of the facilities of its third-party developers? If yes, how often do these inspections/audits occur? Are they periodic or triggered by events (or both)? If triggered by events, provide examples of "trigger" events.				✓
48	Are configuration/change controls in place to prevent unauthorized modifications or additions to source code and related documentation? Do these controls detect and report unexpected modifications/additions to source code? Do they aid in rolling back an affected artifact to a pre-modified version?			✓	✓
49	Does the software's exception-handling mechanism prevent all faults from leaving the software, its resources, and its data (in memory and on disk) in a vulnerable state? Does the exception-handling mechanism provide more than one option for responding to a fault? If so, can the exception-handling options be configured by the administrator or overridden?			✓	✓

Table 2- Questions for COTS (Proprietary & Open Source), GOTS, and Custom Software					
No.	Question	COTS Proprietary	COTS Open-Source	GOTS	Custom
	Built-in Software Defenses				
50	Does the software validate (e.g., filter with white listing) inputs from potentially untrusted sources before being used?	✓	✓	✓	✓
51	Has the software been designed to execute within a constrained execution environment (e.g., virtual machine, sandbox, chroot jail, single-purpose pseudo-user, etc.) and is it designed to isolate and minimize the extent of damage possible by a successful attack?	✓	✓	✓	✓
52	Does the documentation explain how to install, configure, and/or use the software securely? Does it identify options that should not normally be used because they create security weaknesses?	✓	✓	✓	✓
53	Where applicable, does the program use run-time infrastructure defenses (such as address space randomization, stack overflow protection, preventing execution from data memory, and taint checking)?	✓	✓	✓	✓
54	How does the company minimize the risk of reverse engineering of binaries? Are source code obfuscation techniques used? Are legal agreements in place to protect against potential liabilities of non-secure software?	✓		✓	✓
55	How does the software's exception handling mechanism prevent faults from leaving the software, its resources, and its data (in memory and on disk) in a vulnerable state?	✓	✓	✓	✓
56	Does the exception-handling mechanism provide more than one option for responding to a fault? If so, can the exception-handling options be configured by the administrator or overridden?	✓	✓	✓	✓
57	Does the software default to requiring the administrator (or user of a single-user software package) to expressly approve the automatic installation of patches/upgrades, downloading of files, execution of plug-ins or other "helper" applications, and downloading and execution of mobile code?	✓	✓	✓	✓
	Component Assembly				
58	What security criteria, if any, are considered when selecting third-party suppliers?	✓			✓
59	Is the software required/regularized to conform to coding or API standards in any way? Explain.	✓		✓	✓
60	Does the software have any security critical dependencies or need additional controls from other software (e.g., operating system, directory service, applications), firmware, or hardware? If yes, please describe.				✓
61	Is delivery of demonstrably secure software a contractual requirement for third-party developed software? If yes, what criteria are used to operationally define "secure software"?				✓
62	Are additional risk management measures in place in the software's design to mitigate risks posed by use of third-party components?				✓
63	Does the software include content produced by suppliers other than the primary developer? If so, who?			✓	✓
64	What are the policies and procedures for verifying the quality and security of non-developmental components used?			✓	✓
	Testing				
65	What types of functional tests are/were performed on the software during its development (e.g., spot checking, component-level testing, integrated testing)?	✓		✓	✓
66	Who and when are security tests performed on the product? Are tests performed by an internal test team, by an independent third party, or by both?	✓			✓
67	What degree of code coverage does testing provide?	✓		✓	✓
68	Are misuse test cases included to exercise potential abuse scenarios of the software?	✓		✓	✓
69	Are security-specific regression tests performed during the development process? If yes, how frequently are the tests performed?	✓			✓
70	What release criteria does the company have for its products with regard to security?	✓			✓
71	Does the company's defect classification scheme include security categories? During testing what proportion of identified defects relate to security?	✓		✓	✓
72	When does security testing occur during the SDLC (e.g., unit level, subsystem, system, certification and accreditation)?	✓		✓	✓
73	Are regression test scripts available?			✓	✓
	Software Manufacture and Packaging				
74	What security measures are in place for the software packaging facility?	✓			✓
75	What controls are in place to ensure that only the accepted/released software is placed on media for distribution?	✓			✓
76	How is the software packaged (e.g. Zipped, Linux RPM etc) and distributed?	✓	✓		✓

Table 2- Questions for COTS (Proprietary & Open Source), GOTS, and Custom Software					
No.	Question	COTS Proprietary	COTS Open-Source	GOTS	Custom
77	How is the integrity of downloaded software (if an option) protected?	✓	✓		✓
78	For the released software "object", how many "files" does it contain? How are they related?	✓	✓		✓
	Installation				
79	Is a validation test suite or diagnostic available to validate that the application software is operating correctly and in a secure configuration following installation?	✓		✓	✓
80	What training programs, if any, are available or provided through the supplier for the software? Does the company offer certification programs for software integrators? Does the company offer training materials, books, computer-based training, online educational forums, or sponsor conferences related to the software?	✓			✓
81	If the company is responsible for installing the software, is this done by the supplier or through third-party consultants?				✓
82	Are instructions available to securely configure the application?	✓	✓	✓	✓
	Assurance Claims and Evidence				
83	How has the software been measured/assessed for its resistance to identified, relevant attack patterns? Are Common Vulnerabilities & Exposures (CVEs) or Common Weakness Enumerations (CWEs) used? How have exploitable flaws been tracked and mitigated?	✓	✓	✓	✓
84	Has the software been evaluated against the Common Criteria, FIPS 140-2, or other formal evaluation process? What evaluation assurance was achieved? If the product claims conformance to a protection profile, which one(s)? Are the security target and evaluation report available?	✓	✓		✓
85	Are static or dynamic software security analysis tools used to identify weaknesses in the software that can lead to exploitable vulnerabilities? If yes, which tools are used? What classes of weaknesses are covered? When in the SDLC are these scans performed? Are SwA experts involved in the analysis of the scan results?	✓	✓	✓	✓
86	Does the software contain open-source or third-party developed components? If yes, are those components scanned by a static code analysis tool?	✓	✓		✓
87	Has the product undergone any penetration testing? When? By whom? Are the test reports available under a nondisclosure agreement? How have the findings been mitigated?	✓	✓	✓	✓
88	Are there current publicly-known vulnerabilities in the software (e.g., an unrepaired CWE entry)?	✓	✓	✓	✓
89	Has security testing been performed on the software with posted results?		✓		
90	Does the company develop security measurement objectives for phases of the SDLC? Has the company identified specific statistical and/or qualitative analytical techniques for measuring attainment of security measures?				✓
91	How is the assurance of software produced by third-party developers assessed?				✓
92	Has the software been certified and accredited? What release/version/configuration? When? By whom? What criteria or scheme was used to evaluate and accredit the software?	✓	✓	✓	✓
	Support				
93	Is there a Support Life cycle Policy within the organization for the software in question? Does it outline and establish a consistent and predictable support timeline?	✓			✓
94	How will patches and/or Service Packs be distributed to the purchasing/using organization?	✓			✓
95	What services does the help desk, support center, or (if applicable) online support system offer?	✓			
96	Are multiple tiers of support contracts available? If yes, please describe the support plans available.				✓
97	How are trouble tickets submitted? How are support issues escalated, particularly those related to security?				✓
98	Are help desk or support center personnel internal company resources or are these services outsourced to third parties?				✓
99	If help desk or support center services are outsourced to third parties, are they located in foreign countries?	✓			✓
	Software Change Management				
100	How extensively are patches and Service Packs tested before they are released?	✓		✓	✓
101	Can patches and Service Packs be uninstalled? Are the procedures for uninstalling a patch or Service Pack automated or manual?	✓	✓	✓	✓
102	Will configuration changes (if needed for the installation to be completed) be reset to pre-patch configuration state where the change was not made explicitly to close a vulnerability?	✓		✓	✓
103	How are reports of defects, vulnerabilities, and security incidents involving the software collected, tracked, and prioritized?	✓	✓	✓	✓

Table 2- Questions for COTS (Proprietary & Open Source), GOTS, and Custom Software					
No.	Question	COTS Proprietary	COTS Open-Source	GOTS	Custom
104	Does the company determine relative severity of defects, and does that drive processes that influence how fast the company resolves issues with software?	✓			✓
105	What are the policies and practices for reviewing design and architecture security impacts in relation to deploying patches?	✓		✓	✓
106	Are the version control and configuration management policies and procedures the same throughout the entire organization and for all products? How are they enforced? Are third-party developers contractually required to follow these policies and procedures?	✓			✓
107	What policies and processes does the company use to verify that software components do not contain unintended, "dead", or malicious code? What tools are used?	✓		✓	✓
108	How is the software provenance verified (e.g. any checksums or signatures)?	✓	✓	✓	✓
109	Which open-source repository is used, if open source software is included in the delivered software?	✓	✓		✓
110	How are patches distributed?	✓	✓		✓
111	How frequently are major versions of the software released?	✓	✓		✓
112	What are the policies and procedures for maintaining development documents, including requirements, design and architecture documents, source code, binaries, and user documentation?				✓
113	Are configuration/change controls in place to prevent unauthorized modifications or additions to source code and related documentation? Do these controls detect and report unexpected modifications/additions to source code? Do they aid in rolling back an affected artifact to a pre-modified version?				✓
114	Are there any undocumented features present not intended for use by end users, but available for use by the supplier for technical support and development?	✓			✓
115	Does the organization have policies and procedures in place to monitor and audit the transmission of its technology-related intellectual property to third parties, and to prevent unauthorized transmission of that intellectual property?				✓
116	Is a process utilized by the company that can be used for documenting and analyzing the security aspects of fielded systems and for steering future improvements and modifications to those systems?				✓
	Timeliness of Vulnerability Mitigation				
117	Does the company have a vulnerability management and reporting policy? Is it available for review?	✓			✓
118	Does the company publish a security section on its Web site? If so, do security researchers have the ability to report security issues?	✓			✓
	Individual Malicious Behavior				
119	Does the company perform background checks on members of the software development team? If so, are there any additional "vetting" checks done on people who work on critical application components, such as security? Explain.	✓		✓	✓
120	Does the company have formally defined security policies associated with clearly defined roles and responsibilities for personnel working within the software development life cycle, along with management oversight and enforcement? Explain.				✓
121	What training is available to the development staff to help them identify malicious behavior? Are there formal policies for reporting malicious behavior?				✓
	Security "Track Record"				
122	Has civil legal action ever been filed against the company for delivering or failing to correct defective software? Explain.				✓
123	Does the company have an executive-level officer responsible for the security of the company's software products and/or processes?	✓			✓
	Financial History and Status				
124	Does the company have established policies and procedures for dealing with the contractual obligations of third-party developers that go out of business?	✓			✓
125	Does the company have policies and procedures for periodically reviewing the financial health of the third-party entities with which it contracts for software development, maintenance, or support services?	✓			✓
126	Has the company ever filed for Recompany under U.S. Code Chapter 11? If so, please provide dates for each incident and describe the outcome.	✓			✓
	Organizational History				
127	Please summarize the company's history of ownership, acquisitions, and mergers (both those performed by the company and those to which the company was subjected).	✓			✓

Table 2- Questions for COTS (Proprietary & Open Source), GOTS, and Custom Software					
No.	Question	COTS Proprietary	COTS Open-Source	GOTS	Custom
128	Please provide a list of the names and dates of service of the following executive officers: Chairman of the Board (COB), Chief Executive Officer (CEO), President (if different from CEO), Vice President(s), and Chief Financial Officer (CFO).	✓			✓
129	How many employees does the company have: In country (domestic)? Worldwide?	✓			✓
	Foreign Interests and Influences				
130	Is the controlling share (51+%) of the company owned by a foreign (non-domestic) entity? If so, for U.S. Government procurements, the supply company must complete Standard Form 328, Certificate Pertaining to Foreign Interests.	✓			✓
131	Is the company an entity of a larger "parent" company? If yes" does that "parent" company include any subsidiaries or other sub-entities that are 51+% foreign owned? If so, please identify those subsidiaries or sub-entities.	✓			✓
132	Please provide company names of all third-party entities with whom the supplier contracts software development or support services related to this procurement.	✓			✓

Increasingly, software support and software applications are provided as a service and supported by someone other than the purchasing or using organization. Application service providers host the software that support the applications in data centers and provide different levels of service, including security-related services. Users remotely access the applications. Suppliers of software-related services should also ask software development questions for both internal and outsourced work for the appropriate software type to augment the example questions indicated in Table 3 for Hosted Applications.

Table 3 - Questions for Hosted Applications	
No.	Questions
	Service Confidentiality Policies
1	What are the customer confidentiality policies? How are they enforced?
2	What are the customer privacy policies? How are they enforced?
3	What are the policies and procedures used to protect sensitive information from unauthorized access? How are the policies enforced?
4	What are the set of controls to ensure separation of data and security information between different customers that are physically located in the same data center? On the same host server?
	Operating Environment for Services
5	Who configures and deploys the servers? Are the configuration procedures available for review, including documentation for all registry settings?
6	What are the policies and procedures for hardening servers?
7	What are the data backup policies and procedures? How frequently are the backup procedures verified?
8	What are the procedures for evaluating any vendor security alerts and installing patches and Service Packs?
9	How are vendor patches and Services Packs applied?
10	Is testing done after changes are made to servers? What are the rollback procedures in the event of problems resulting from installing a patch or Service Pack?
11	What are the agents or scripts executing on servers of hosted applications? Are there procedures for reviewing the security of these scripts or agents?
12	What are the procedures and policies used to approve, grant, monitor and revoke access to the servers? Are audit logs maintained?
13	What are the procedures and policies for handling and destroying sensitive data on electronic and printed media?

Table 3 - Questions for Hosted Applications	
No.	Questions
14	Does the service provider have a formal disaster recovery plan? What actions will be taken to recover from a disaster? Are warm or hot backups available?
15	What are the procedures used to approve, grant, monitor, and revoke file permissions for production data and executable code?
16	Is two-factor authentication used for administrative control of all security devices and critical information systems?
Security Services and Monitoring	
17	What are the types of information security services provided by the company?
18	How are virus prevention, detection, correction, and updates handled for the products?
19	What type of firewalls (or application gateways) are used? How are they monitored/managed?
20	What type of Intrusion Detection System/Intrusion Protection Systems (IDS/IPS) are used? How are they monitored/managed?
21	Is the system and network architecture based on a high availability design that includes redundant firewalls, routers, switches and IDS, and load balanced or clustered servers?
22	Does the company perform regular reviews of system and network logs for security issues?
23	Does the company have an automated security event management system?
24	What are the procedures for intrusion detection, incident response, and incident investigation/escalation?
25	Will the company provide on-site support 24x7 to resolve security incidents?
26	Does the company provide write-once technology for storing audit trails and security logs?
27	How does the company control physical and electronic access to the log files? Are log files consolidated to single servers?
28	Does the company provide security performance measures to the customer at regular intervals?
Assurance Claims and Evidence	
29	Has functional security testing been performed on the services?
30	Does the company perform penetration testing of the service? If yes, how frequently are penetration tests performed? Are the tests performed by internal resources or by a third party?
31	Does the company provide automated vulnerability testing of the service? If yes, how frequently are the tests performed? Are the tests performed by internal resources or by a third party?

As previously noted, questionnaires and the relevant responses (or lack of answers) help to identify potential risks. In some cases, suppliers might be unable or unwilling to provide answers (or supporting evidence) to all questions that might be asked by purchasing organizations. In such instances any lack of transparency represents a risk to the using organization because they would lack sufficient information to make better risk-based decisions. Acquirers and evaluators should only use Due-Diligence questions for eliciting information that contribute to the purchasing decisions that factor in the security needs of users. If an answer would not influence purchasing decisions (based on an understanding of user needs), then the question should not be posed to the supplier.

Next Steps. Once risks are identified and a determination has been made about software and suppliers, the acquisition organization should consider the contract needs. Those considerations are covered in "Contract Language for Integrating Software Security into the Acquisition Life Cycle," Acquisition & Outsourcing, Volume 1 pocket guide and in "Software Assurance in Acquisition: Mitigating Risks to the Enterprise" available through the SwA Community Resources and Information Clearinghouse at https://buildsecurityin.us-cert.gov/swa/acqact.html . The source document is also available through the National Defense University Press at http://www.ndu.edu/inss/press/NDUPress_Occasional_Papers.htm .

On-line Resources

- SwA Community Resources and Information Clearinghouse at https://buildsecurityin.us-cert.gov/swa.
- National Institute of Standards and Technology (NIST) information security publications at http://csrc.nist.gov/publications/PubsSPs.html.
- *Information Resources Management College (IRMC) at http://www.ndu.edu/irmc/.*
- *National Defense University Press at http://www.ndu.edu/inss/press/NDUPress_Occasional_Papers.htm.*

Conclusion

This pocket guide compiles example SwA due-diligence questions for acquires and software evaluators as a means for gathering relevant information to support decision making. For the latest updates and details, visit the web sites listed in the preceding pages and resource box.

The Software Assurance Pocket Guide Series is developed in collaboration with the SwA Forum and Working Groups and provides summary material in a more consumable format. The series provides informative material for SwA initiatives that seek to reduce software vulnerabilities, minimize exploitation, and address ways to improve the routine development, acquisition and deployment of trustworthy software products. Together, these activities will enable more secure and reliable software that supports mission requirements across enterprises and the critical infrastructure.

For additional information or contribution to future material and/or enhancements of this pocket guide, please consider joining any of the SwA Working Groups and/or send comments to Software.Assurance@dhs.gov. SwA Forums are open to all participants and free of charge. Please visit https://buildsecurityin.us-cert.gov for further information.

No Warranty

This material is furnished on an "as-is" basis for information only. The authors, contributors, and participants of the SwA Forum and Working Groups, their employers, the U.S. Government, other participating organizations, all other entities associated with this information resource, and entities and products mentioned within this pocket guide make no warranties of any kind, either expressed or implied, as to any matter including, but not limited to, warranty of fitness for purpose, completeness or merchantability, exclusivity, or results obtained from use of the material. No warranty of any kind is made with respect to freedom from patent, trademark, or copyright infringement. Reference or use of any trademarks is not intended in any way to infringe on the rights of the trademark holder. No warranty is made that use of the information in this pocket guide will result in software that is secure. Examples are for illustrative purposes and are not intended to be used as is or without undergoing analysis.

Reprints

Any Software Assurance Pocket Guide may be reproduced and/or redistributed in its original configuration, within normal distribution channels (including but not limited to on-demand Internet downloads or in various archived/compressed formats).

Anyone making further distribution of these pocket guides via reprints may indicate on the pocket guide that their organization made the reprints of the document, but the pocket guide should not be otherwise altered.

These resources have been developed for information purposes and should be available to all with interests in software security.

For more information, including recommendations for modification of SwA pocket guides, please contact Software.Assurance@dhs.gov or visit the Software Assurance Community Resources and Information Clearinghouse: https://buildsecurityin.us-cert.gov/swa to download this document either format (4"x8" or 8.5"x11").

Software Assurance (SwA) Pocket Guide Series

SwA is primarily focused on software security and mitigating risks attributable to software; better enabling resilience in operations. SwA Pocket Guides are provided; with some yet to be published. All are offered as informative resources; not comprehensive in coverage. All are intended as resources for 'getting started' with various aspects of software assurance. The planned coverage of topics in the SwA Pocket Guide Series is listed:

SwA in Acquisition & Outsourcing

I. Software Assurance in Acquisition and Contract Language
II. Software Supply Chain Risk Management & Due-Diligence

SwA in Development

I. Integrating Security into the Software Development Life Cycle
II. Key Practices for Mitigating the Most Egregious Exploitable Software Weaknesses
III. Risk-based Software Security Testing
IV. Requirements & Analysis for Secure Software
V. Architecture & Design Considerations for Secure Software
VI. Secure Coding & Software Construction
VII. Security Considerations for Technologies, Methodologies & Languages

SwA Life Cycle Support

I. SwA in Education, Training & Certification
II. Secure Software Distribution, Deployment, & Operations
III. Code Transparency & Software Labels
IV. Assurance Case Management
V. Assurance Process Improvement & Benchmarking
VI. Secure Software Environment & Assurance Ecosystem

SwA Measurement & Information Needs

I. Making Software Security Measurable
II. Practical Measurement Framework for SwA & InfoSec
III. SwA Business Case & Return on Investment

SwA Pocket Guides and related documents are freely available for download via the DHS NCSD Software Assurance Community Resources and Information Clearinghouse at https://buildsecurityin.us-cert.gov/swa.