Collaborating Across the Supply Chain to Address Taint and Counterfeit

Dan Reddy, EMC Corporation

Abstract. Before the community of acquirers and providers of technology can get to the heart of supply chain risk management regarding taint and counterfeit they must reach some consensus on two basics questions: 1) Where is the mitigation focus when we discuss supply chain?, and 2) Are we discussing both quality issues that occur in technology development or just products that have been tampered with along the supply chain?

Introduction

When one speaks of risks inherent in building Information and Communication Technology (ICT) products, is it really a supply chain discussion from soup to nuts? Should there be an equal discussion about the quality of the technology produced and whether the product has been deliberately tampered with during the product lifecycle process? These two questions can quickly derail conversations about secure software development and supply chain risk, two topics that are already complex enough. Framing these debates properly might allow the discussion to productively proceed to how the risks can be mitigated by applying best practices by the right party at the right juncture. The answers to these framing questions can help focus the discussion on where to effectively apply the controls to offset threats. Should the provider take the lead in applying controls only within its own organization? How should the provider ensure that best practices are applied throughout the rest of the supply chain?

What are we guarding against when we consider supply chain risk management? The first major threat is an attack that tampers with a product as it is being sourced, built, or delivered and potentially introduces capability or maliciously inserted code that the original provider of the product never designed or planned to deliver. This tampering related threat also extends to hardware where the attacker's planned substitution of faulty counterfeit components could undermine the manufacturer's planned capability, its performance or introduce new malicious capability. The second threat area is related to the quality of the product. Poor quality practices during the development of software or firmware could lead to bugs or errors that can be exploited by attackers before, during, or after installation, thus undermining another dimension of software assurance. There can be poor quality counterfeit components in the supply chain because someone wants to make money through a lower cost substitute. Therefore not all counterfeits are due to the introduction of malicious capability.

Yes, the customer who ultimately acquires the information technology should reasonably expect quality products without exploitable vulnerabilities stemming from known weaknesses or malware. This customer should expect that the operational environment in which the product is deployed is uncompromised. The customer should reasonably assume that it is the authentic product from the original provider and it is the high quality product that functions as the technology provider intended, no more and no less. Technology providers stand behind the product that they make and sell. They convey that it is the real deal and the product's integrity has been preserved along the complex creation and delivery journey to the customer. None of these expectations should be in doubt. Every component supplier along the way is inherently a provider in its own right and must stand behind its product in the same way, offering authenticity, integrity and security. These are the three elements of assurance as described by SAFECode, an industry led group formed in 2007 to focus on software assurance [1].

Figure 1 SAFECode Three Elements of Assurance [1]

Is it all Part of the Supply Chain? A Provider-centric View

To address this first pivotal question on the scope of supply chain, one could view the entire process from the concept of a product through its delivery to the customer as a series of complex supply chain interactions with a multitude of players and lose sight of where the primary ICT provider's role, activities and oversight come into play. The provider's role is strongest in what it directly controls in its own shop and more indirect when it relies on others to build and deliver hardware and software components. When they rely on other suppliers, providers can have strong expectations, tests, contracts, acceptance procedures and audits but they do not directly oversee and control many aspects of any one supplier's component in the same way they cover their own practices; they always rely on other parties to some extent.

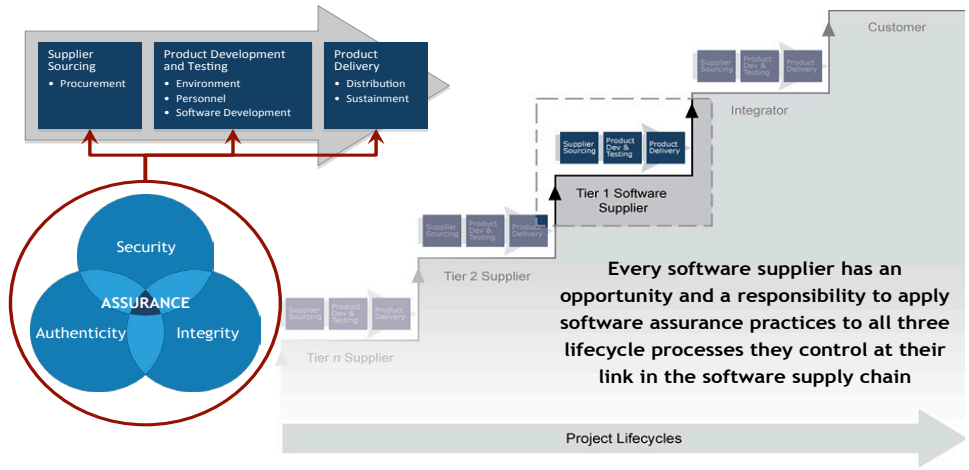

Figure 2 SAFECode Supply Chain Framework [3]

When SAFECode first published an article on software supply chain integrity [2] in July of 2009, they framed the software supply chain as being comprised of Supplier Sourcing, Product Development and Testing, and Product Delivery. This view is notably provider centric. It is the providers who should develop a program based on best practices as they engage with outside suppliers to source people and components for their products. They then may have their own in-house development and testing activities that govern the software as it is being built and tested to prepare for the final delivery phase which may either be under their own control or may be another opportunity to engage within the supply chain. The framework laid out in the SAFECode whitepaper envisioned a staircase effect comprised of n tiers of suppliers whereby each supplier in the chain would concentrate on applying the best practices for each of these three phases.

This model distributes the operational responsibility to make the practices work most effectively at each tier. It does not change the overall acquirer's or customer's expectation of the provider of the product. Providers can enforce controls in their own organizations while they focus on indirect verification when they engage suppliers. Just as the customer cannot effectively enforce the controls inside the provider's shop, the provider must turn to verifying that controls are in place within their suppliers. If the right players apply the right controls at the right spot, the industry will achieve overall scalability and accountability across the supply chain. It is better than trying to have each customer, each agency, each branch of service or each procurement officer create its own approach. That simply would not scale and is likely to become derailed in the pivotal discussions outlined here.

SAFECode in a later publication [4] outlined the specific set of controls that is applicable at each lifecycle process phase. For all dealings with suppliers and delivery partners who source people, services and components, one might describe the interactions to protect the supply chain as "engagement controls" such as writing contracts to set expectations or looking for measures of authenticity and integrity such as digitally signed code or verifying cryptographic checksums to validate a binary deliverable. Engagement controls begin in the provider's enterprise but extend out into the supply chain at various touch points. These controls include how the provider's enterprise brings on contractors for in-house work, how they accept delivery and test software components from a supplier, and how they determine who is an authorized service partner. Such controls are shaped first in the provider's organization and then come into play in preparation for engagement with outside suppliers. Since they are applied between organizations, they differ in their reach from the direct enforcement controls that a software provider should apply in-house with their own resources as they develop and test good software. The application, monitoring and governance of controls naturally will also require ongoing sustainment and recalibration.

The same approach can apply to avoiding counterfeits throughout the supply chain. One must first acknowledge that an attacker could create a perfect replica of a component that functions and performs well from a quality perspective—but it could have intentional malicious capability added. Testing hardware for meeting functional specifications alone is not sufficient for this abuse case. Knowing the strength of the chain of custody of the component and being alert for the potential impact of an inauthentic component in the architecture may inform proper negative testing or additional hardware tests for traceability and other characteristics that may still be required. However, testing hardware for quality can identify faulty counterfeit components that have entered the supply chain purely for someone's monetary gain. These different approaches can be applied both in-house where hardware may be involved and also by engaging suppliers in the chain when some of the hardware work is done through sourcing.

The best way to mitigate risk throughout the supply chain is for all ICT providers to adopt common industry best practices while delivering their own quality products and mitigating the risk of having a product tampered with or including counterfeit components along the way. If each supplier whose components, code, or assemblies along the supply chain subscribes to best practices that could be measured based on commonly defined outcomes, the customer could develop a deeper sense of trust down multiple levels into the supply chain. No single provider can reach down deep enough into tiers of suppliers to highlight where best practices do occur unless an overall ecosystem that includes provider, supplier and acquirer evolves to expect and measure compliance as each supplier comes into view. An ICT provider may have dozens or hundreds or more suppliers. Even if there is rigor and consistency in how the expectations are set, monitored and verified by the primary provider as component items are

specified, built, sourced and delivered across the supply chain, it will not convey the same confidence as it would if each supplier along the chain could also adopt global standards. Each supplier could pass along assurance as to their compliance so that the provider could summarize the results in the aggregate for their customers. Then the confidence would be evident and visible at each tier in the supply chain, at least for some basic assurance. The lens then needs to focus on each supplier to make sure its own house is in order. This is not to beg for a pass for the distant tiers of suppliers in the chain because they are remote, but to recognize that the law of physics works against having the same level of deep control throughout the chain as providers can when they supervise their own organizations.

The ideal in the fully evolved ecosystem is to have best practices occur both within the sphere of the provider's own shop and also at each tier in the supply chain. In order to make such an ecosystem scalable and viable, there must be practical methods to achieve and measure common outcomes. The most effective method is to have each supplier along the complex supply chain be evaluated against a global standard by a qualified assessor who can perform such an assessment in a reliably consistent manner. Then the provider does not have to sustain a unique conversation with each supplier as to expected good software development practices, good anti-counterfeiting practices along with good practices to prevent acquiring products that have been tampered with at any point along the lifecycle. The global Open Group's Open Trusted Technology Provider Standard (O-TTPS) [5] to mitigate maliciously tainted and counterfeit products is designed to enable all ICT providers to be evaluated by recognized third-party assessors. O-TTPS includes more than 50 requirements relating to how products are developed, how secure engineering is applied, and how supply chain security risks for maliciously tainted and counterfeit are addressed. Ideally then in the future state of the ecosystem each provider should be able to expect that their own preferred suppliers would have gone through their own process of becoming accredited and be listed in the Open Group's Trusted Technology Provider registry of accredited organizations. Each customer or acquirer would then be able to identify the associated set of products from each accredited organization that conforms to the best practices.

The O-TTPS outlines a distinction between those requirements that are specifically related to the provider's own shop (considered as part of Technology Development) and those requirements that involve an engagement with suppliers (considered as part of Supply Chain). In fact Figure 3 shows an example of mottled shading over the blocks depicting the stages of the lifecycle in relation to the technology development and supply chain. This reflects the reality that the number of touch points may vary between the provider and various suppliers that are engaged in any particular product's development lifecycle. Some products have a high internal development profile and others may have more touch points with external supplier organizations that contribute to the product along the lifecycle. All of the requirements must be met by the provider, but the O-TTPS does reference a best practice whereby providers seek qualified suppliers that follow the same set of practices as those embodied in O-TTPS. This recursive requirement should help facilitate the ecosystem in reaching its potential.

Addressing Software Assurance, Quality, and Tampering

Quality begins at home in the provider's own shop, regardless of whether they sell to an end-user or act as a supplier to another ICT provider. Good software development with security in mind should follow an array of good practices to avoid common mistakes so that the ultimate software produced is less subject to bugs or weaknesses that can be exploited during an attack. Following a software development lifecycle (SDLC) with security in mind is a discipline unto itself. A product development lifecycle imbued with secure engineering starts (like the supply chain discussion above) with making sure that developers in the provider's own shop are well trained, focused on activities like secure design, threat modeling, secure coding, proper iterative testing, ensuring a hardened state of all components and good documentation for the ultimate customer concerning the correct usage of the security related configurations. These are a few of the everyday practices that are considered quality related on the part of the developer in the provider's shop. As an industry-led organization for sharing best practices, SAFECode has outlined how to securely develop software in its whitepaper [6]. If an organization wants to model its own secure software development practices on those of the industry, this whitepaper would be a great place to start for some detailed recommendations. The software development organization's first obligation is to do the right thing from the beginning. It is no longer acceptable for a developer to say, "I did not know about the most dangerous errors to avoid while building software; I will be better next time."

Once these practices become routine for the provider and institutional knowledge is strengthened through ongoing measurement, adjustment and oversight, the bar is raised and development teams need to tackle the next challenges in building quality products that are resilient. For the moment let us assume that coding errors that are found are due to insufficient software design and hygiene being applied while building a technology

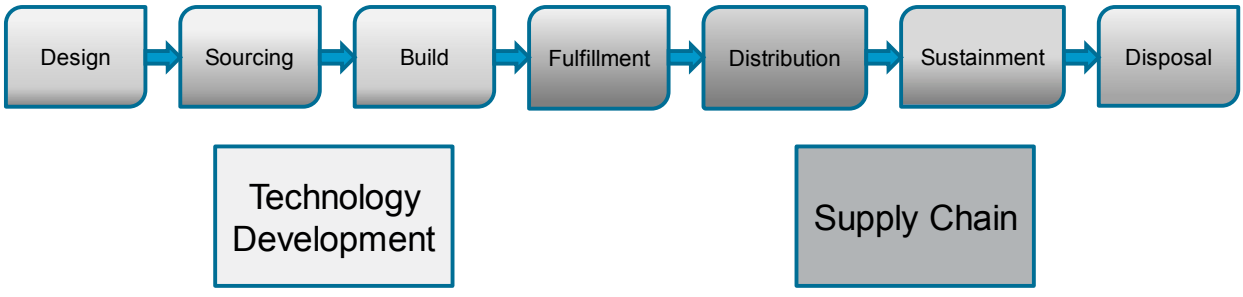

Figure 3. Sample view of the relationship between Technology Development and Supply Chain in O-TTPS [4]

product. These errors may lead to vulnerabilities that can in turn be exploited by attackers. Such errors, bugs and vulnerabilities are a fact of life in the world of software development; they can be reduced, but they are not going away any time soon. The immediate customer priority is a requirement for the product organization to have a mature process to respond to known problems and address them effectively, maintaining close linkages to both the customer community and engineering teams. The next challenge is building a sustainable means of avoiding errors or bugs in the future to the extent possible. Good software development and support from the provider's internal governance process can reduce obvious gaps through the diligent application of such best practices by each provider across the supply chain.

The question is not whether the provider needs to be concerned with the quality issues. It is a matter of how the provider focuses to engage and verify what they receive from each supplier. Software is often delivered from its original supplier to a provider who in turn embeds the software as a component (perhaps as firmware) in an overall product. The supplier can be a commercial entity or an open source community. The provider cannot effectively go in and manage the SDLC process for the supplier or run all of the same tests that the original developer can run. For example, assume that the software development team uses threat modeling during design and again for later testing and verification. Let us also assume that their static source code is analyzed, triaged and fixed on an iterative basis during ongoing development and updates. The provider can reasonably expect to determine if the original development organization follows such practices and conducts them with growing competence and repeatability. It is not reasonable or scalable to assume that the provider will literally inspect or oversee such activities in someone else's original development organization. Instead, if each development organization could be accredited for having and following good product development and secure engineering along with supply chain practices, then the unique conversations between each tier of technology provider and the acquirer of the technology can be reduced. The provider's contracts can then require demonstrated adherence to measureable global standards such as the Open Group's recently announced Open Trusted Technology Provider accreditation process.

With such a foundation of development practices to guard against exploitable software quality weaknesses, each organization can then focus on guarding against tampering with a product. Tampering could occur during lapses in custodial care in the original development organization or elsewhere throughout the rest of the cycle among supply chain players. Each supplier must make sure that the integrity and authenticity of the end product

are strong and evident during the entire development cycle and afterwards throughout the cycle of being installed at a customer site and updated over its lifetime. Supplier and provider alike can require that proof of authenticity and integrity are evident as they exchange packages. In addition to these checks, the provider can test to make sure that no known malware resides in the received package. If such malware is found, is it likely to have been maliciously inserted along the way either in the original development shop or somewhere in the rest of the supply chain. Could it have spread to the environment by malicious design or could the contamination have been somewhat inadvertent? All of these best practices can be tied to the achievement of accreditation against a global standard and thereby define an important foundational stratum of capability within a development organization.

The public expects ICT providers to produce quality results delivered with provable and intact authenticity and integrity along the way. Each provider and supplier in the ecosystem must do its share to deliver these results. The improvement cycle starts with the global definition of industry practices that can be shared by providers to achieve security, integrity and authenticity of the software and hardware components they supply. Then the baseline of industry practice needs to be complemented by an accreditation regime that can measure and report how well these controls are being applied to each provider involved in the ICT supply chain. With those elements now in place the industry can move forward and leverage these defined practices and measure basic adherence to them instead of spending energy debating whether the customer and provider are referring to the same risks.

ABOUT THE AUTHOR

Dan Reddy leads Supply Chain Assurance in EMC's Product Security Office. He was co-chair for SAFECode's whitepaper on Supply Chain Integrity Controls. He's co-chair of the Open Group's Trusted Technology Forum's Acquisition workstream. Dan spent 15 years at New England Electric, a major utility with critical infrastructure. He teaches CIS at Quinsigamond Community College. Dan graduated from Tufts and holds M. Ed. degrees from Worcester State University in education and computer science.

EMC Corporation
171 South Street
Hopkinton, MA 01748
Phone: 508-435-1000
E-mail: dan.reddy@emc.com
E-mail: dan.reddy@gmail.com

REFERENCES

1. "Overview of Software Integrity Controls An Assurance-Based Approach to Minimizing Risks in the Software Supply Chain" June 2010: 1. pag. SAFECode. SAFECode, June 2010. Web. <www.safecode.org>
2. "Framework for Software Supply Chain Integrity July 2009: n. pag. SAFECode. The Software Assurance Forum for Excellence in Code, July 2009. Web. <www.safecode.org>
3. Copyrighted combined graphic from SAFECode Supply Chain Integrity presentation - The Software Assurance Forum for Excellence in Code, July 2010)
4. "Overview of Software Integrity Controls An Assurance-Based Approach to Minimizing Risks in the Software Supply Chain June 2010: n. pag. SAFECode. SAFECode, June 2010. Web. <www.safecode.org>
5. "Open Trusted Technology Provider Standard (O-TTPS)™"Version 1.0, Mitigating Maliciously Tainted and Counterfeit Products" 15. pag. Open Group. The Open Group, April 2013. Web. <www.opengroup.org>.
6. "Fundamental Practices for Secure Software Development 2nd Edition." SAFECode. The Software Assurance Forum for Excellence in Code, Feb. 2011. Web. <www.safecode.org>.