**

*DEPARTMENT OF HOMELAND SECURITY*

*SCIENCE AND TECHNOLOGY DIRECTORATE*

Program Management Plan (PMP)

Software Assurance Program

Version 1.6

06/23/17

**PMP Approvals:**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_**

Kevin Greene Date

Project Manager

Homeland Security Advanced Research Projects Agency (HSARPA)

U.S. Department of Homeland Security (DHS),

Science & Technology (S&T) Directorate

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_**

Douglas Maughan Date

Division Director

Homeland Security Advanced Research Projects Agency (HSARPA)

U.S. Department of Homeland Security (DHS),

Science & Technology (S&T) Directorate

Record of Changes

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Version** | **Date** | **Figure # or Section #** | **A/M/D\*** | **Description of Change / Rationale** |
| 0.5 | 2/1/17 |  |  | Initial Draft |
| 1.0 | 2/27/17 |  | M | Draft for OSE Review/Comment |
| 1.5 | 3/15/17 |  | M | Updated with Org chart, transitions, schedule modifications |
| 1.6 | 4/28/17 |  | M | Updated with additional sections on transition pathways, stakeholder engagement strategy |

\* A-Added, M-Modified, D-Deleted

Contents

[1. Program Summary 1](#_Toc481474558)

[1.1 Background and Purpose 1](#_Toc481474559)

[1.2 Customer(s) 4](#_Toc481474560)

[1.3 1.3 Deliverables/Products 6](#_Toc481474561)

[1.4 Transition Plan 7](#_Toc481474562)

[1.4.1 Transition Pathways 8](#_Toc481474563)

[1.4.2 Project Transition 9](#_Toc481474564)

[1.5 Schedule and Budget Summary 10](#_Toc481474565)

[1.6 Project Organization 11](#_Toc481474566)

[1.6.1 Project Personnel 11](#_Toc481474567)

[1.6.2 Internal Structure 12](#_Toc481474568)

[1.7 Benefits to S&T 12](#_Toc481474569)

[1.7.1 New Capabilities / Technology 12](#_Toc481474570)

[1.7.2 New Intellectual Property 13](#_Toc481474571)

[1.7.3 Public Access of Federally Funded Scientific Research 13](#_Toc481474572)

[1.7.4 Data Management 13](#_Toc481474573)

[1.7.5 Other Benefits 13](#_Toc481474574)

[1.8 Assumptions 13](#_Toc481474575)

[2. Requirements 13](#_Toc481474576)

[2.1 Requirements Management Plan 13](#_Toc481474577)

[2.2 Customer Engagement 14](#_Toc481474578)

[2.3 Customer Requirements 15](#_Toc481474579)

[2.4 Key Functions and Desired Performance Requirements 19](#_Toc481474580)

[2.5 Other Metrics 21](#_Toc481474581)

[3. Technology and Innovation 21](#_Toc481474582)

[3.1 3.1 Core Technology 21](#_Toc481474583)

[3.2 Technology Scouting 21](#_Toc481474584)

[3.3 3.3 Technical Approach 22](#_Toc481474585)

[3.3.1 Trade Space Analysis 22](#_Toc481474586)

[3.3.2 Preferred Alternative 22](#_Toc481474587)

[3.3.3 Logistics Considerations 22](#_Toc481474588)

[4. Project Management 22](#_Toc481474589)

[4.1 Acquisition Management 22](#_Toc481474590)

[4.1.1 Program Acquisitions 22](#_Toc481474591)

[4.1.2 Acquisition Strategy 23](#_Toc481474592)

[4.1.3 Potential Sources 25](#_Toc481474593)

[4.2 Scope Management 25](#_Toc481474594)

[4.2.1 Work Breakdown Structure 25](#_Toc481474595)

[4.2.2 Scope Control Plan 26](#_Toc481474596)

[4.3 4.3 Schedule Management 26](#_Toc481474597)

[4.3.1 Milestones/Deliverables 26](#_Toc481474598)

[4.3.2 Schedule Control Plan 27](#_Toc481474599)

[4.4 Cost Management 27](#_Toc481474600)

[4.4.1 Cost Estimate 27](#_Toc481474601)

[4.4.2 Cost Baseline 27](#_Toc481474602)

[4.4.3 Cost Control Plan 28](#_Toc481474603)

[4.5 4.5 Status Reporting 28](#_Toc481474604)

[4.6 Cost/Schedule/Performance Trade-Offs 28](#_Toc481474605)

[5. Risk Management 28](#_Toc481474606)

[5.1 5.1 Risk Management Plan 28](#_Toc481474607)

[5.2 5.2 Risk Identification and Assessment 29](#_Toc481474608)

[5.2.1 Technical Risks 29](#_Toc481474609)

[5.2.2 Programmatic Risks 30](#_Toc481474610)

[5.2.3 Operational Risks 30](#_Toc481474611)

[5.3 5.3 Risk Register 30](#_Toc481474612)

# Program Summary

The Software Assurance (SWA) Program consists of 4 coordinated projects that together deliver a comprehensive Software Assurance capability, including: 1) improving software analysis techniques, tools and methods through the Software Quality Assurance (SQA) and Static Tools Analysis Modernization Project (STAMP), 2) Developing and improving automated capabilities that mimic behavior and activity of attackers to identify exposures and vulnerabilities in real-time to reduce the window of exposure through Application Security Threat Attack Modeling (ASTAM) and 3) a collaborative research environment designed to assist researchers, software developers, academia, and tool developers in finding new advancements to improve software assurance capabilities, called the Software Assurance Marketplace (SWAMP).

## Background and Purpose

Software increasingly underlies every aspect of our digital world. While the Internet of Things (IoT) garners the majority of headlines today, software is an element of all IoT as well as the non-IoT or legacy systems that powers our nation’s critical infrastructure. Gartner estimates that there will be 20.8 billion connected devices by 2020, all powered, connected and integrated by software. 90% of security breaches can be traced back to poorly developed and designed software. [[1]](#footnote-2)

The average code developed in the United States has 0.75 defects per function point or 6,000 defects per million lines of code (MLOC) for a high-level language, such as Java or C#. Very good levels would be 600 to 1,000 defects per MLOC, and exceptional levels would be below 600 defects per MLOC. As one example, the Boeing 878 Dreamliner has 14 MLOC, and if we assume all of it is exceptional code, then 8,400 defects remain in the code and approximately 420 vulnerabilities.[[2]](#footnote-3) In the cybersecurity business, the criminal or adversary only needs to find and exploit one defect, whereas the defender needs to be able to cover all potential attack vectors. Thus even with a robust systems development process such as Boeing uses, there are several thousand potential defects which could be exploited.

In addition to the increasing volume of software being developed, the tools used for software assurance are not suited for today’s environment, specifically:

* Wider spread adoption of DevOps ("software Development" and "information technology Operations), provides more rapid building, testing and release of software, but also introduces the potential for vulnerability acceleration
* Dynamic programming languages introduce new challenges to the existing challenges with static programming languages
* Tool developers tend to err on the side of caution, which can lead to high false positive rates and consequently increase the cost of assurance through additional, potentially unnecessary investigation, re-testing, etc.
* Software development focuses on customer functionality needs primarily and does not consider software security, especially in the design phase

To address the challenges associated with static analysis tools, the SWA program launched the Static Tools Analysis Modernization Project (STAMP). There are a host of free and open-source static analysis tools that fall into the category of neglected and/or underperforming and are prime candidates for modernization. These static analysis tools overall don’t perform very well and offer marginal value. Modernization of these static analysis tools is needed to help advance and improve software analysis capabilities. For example, no software analysis tool was able to find the weakness or flaw in OpenSSL that exposed the Heartbleed vulnerability. Innovation in software analysis capabilities is needed to keep pace with the evolution in software systems; to improve static analysis tools there must be advancements in research and development to discover new techniques, methods, services, and capabilities in testing and evaluating software for critical weaknesses and vulnerabilities. This project is focused on closing the gaps in two key areas: research and development of new tools and techniques for software analysis and applying new and improved capabilities in testing and evaluation activities. STAMP will provide several benefits to the federal organizations and the software assurance community at large:

• Help reduce costs of formalizing and deploying an enterprise software analysis capability by leveraging open-source tools

• Savings on maintenance costs associated with procuring a commercial software analysis tool

• Improved tool coverage for better analysis results and reduction of false-positives to help achieve security at-speed in agile environments.

• Provides customer awareness regarding software analysis tool acquisition that outlines the strengths of tools

The goal of the Application Security Threat and Attack Modeling (ASTAM) Project is to analyze software systems and applications to identify potential risks, security threats, and exposures to the system environment, and then develop appropriate countermeasures to prevent, or mitigate the effects of threats to the system environment by bring together independent assessment activities to build better situational awareness regarding potential threats. Software programmers and developers tend to design, develop, and maintain software systems and applications with a focus on the customer needs, and neglect to fundamentally understand ways in which software systems and applications can be exploited and compromised by potential attackers. This project will model all of the information that affects the security (confidentiality, integrity, and availability) of software systems and applications to provide a view of threats in the environment from both a risk management and security perspective. ASTAM will provide several benefits to the federal organizations and the software assurance community at large:

• Provides real-time awareness of control weaknesses and exposures from an application security perspective through all phases of the software development lifecycle

• Cost savings – product agnostic, best of breed technologies to reduce risk.

• Improves efficiencies through automation with application security context for better situational awareness

• Provides organizations the capability to deploy real-time mitigation responses to attacks and vulnerabilities in application systems

This technology will allow organizations to better achieve continuous monitoring and risk assessment activities more efficiently and effectively by automating and integrating software assurance activities that model threats of the targeted systems in real-time, detecting weaknesses and exposures preemptively.

The SWA program establishes DHS S&T as a leader in the software assurance space in several aspects:

* The SWAMP is the first of its kind resource for the software and software assurance tool development community where the power of multiple tools
* Modernizing static analysis tools improves the software analysis capability overall, reducing false positive rates, encouraging software assurance in line with “at speed” DevOps and critically, leveraging existing static analysis tools.
* The UTM capability will enable software analysis and testing capabilities throughout the software development lifecycle, helping identify and mitigate software vulnerabilities earlier in the development cycle at a lower cost.
* Leverages diverse relationships with academia, government, and the private sector transition technology into operational environments to help organizations formalize software assurance.

Under the 2014 DHS Quadrennial Homeland Security Review (QHSR), Mission 4 is to “Safeguard and Secure Cyberspace” and sets forth priorities to reduce national cyber-related risks and enhance critical infrastructure security and resilience. In addition, the QHSR calls to strengthen the security and resilience of critical infrastructure against cyberattacks and other hazards; secure the federal civilian government information technology enterprise; advance cybersecurity law enforcement, incident response, and reporting capabilities; and strengthen the cybersecurity ecosystem. Secure software contributes to all of these QHSR objectives.

The SWA program also aligns with the 2016 Federal Cybersecurity Research and Development Strategic Plan (RDSP). The 4 main elements of the Plan are listed below:

1. Deter. The ability to efficiently discourage malicious cybersecurity activities by measuring and increasing costs to adversaries carrying out such activities, diminishing the spoils, and increasing risks and uncertainty for potential adversaries. **SWA alignment**: Higher quality, secure software, with fewer vulnerabilities, can increase the cost for attackers.

2. Protect. The ability of components, systems, users, and critical infrastructure to efficiently resist malicious cyber activities and to ensure confidentiality, integrity, availability, and accountability. **SWA alignment**: Higher quality, secure software is more resistant to malicious activities.

3. Detect. The ability to efficiently detect, and even anticipate, adversary decisions and activities, given that perfect security is not possible and systems should be assumed to be vulnerable to malicious cyber activities. **SWA alignment**: Continuous monitoring and Assessment program elements provide ongoing awareness of their security status, ongoing threats and vulnerabilities and can thus detect changes in their environment sooner, limiting the window of exposure.

4. Adapt. The ability of defenders, defenses, and infrastructure to dynamically adapt to malicious cyber activities, by efficiently reacting to disruption, recovering from damage, maintaining operations while completing restoration, and adjusting to thwart similar future activity. **SWA Alignment**: Rapid change detection, enabled through continuous monitoring of software security status, enables cyber security operators to respond quickly to changes and adjust or adapt their security measures and protocols.

SWA also addresses three of the critical dependencies outlined in the RDSP:

1) Scientific Foundations: Developing “sound mathematical and scientific foundations with clear objectives, comprehensive theories (e.g. of defense, systems and adversaries)”[[3]](#footnote-4). **SWA alignment:** Will produce an application threat modeling engine and analysis platform. In addition, SWA will also develop quantifiable test cases using real software programs and a compressive scoring framework to assess the performance of software quality assurance tools.

2) Transition to Practice: Bridging the gap between the research community and the operational communities. **SWA alignment**: Tools developed as part of SWA are targeted for integration into the SWAMP, a fully operational software testing and evaluation capability.

3) Workforce Development: Developing and retaining a skilled cybersecurity workforce.is a key challenge. **SWA alignment**: SWAMP is a no-cost resource that educators are using to help today’s students learn more about software security and acquire valuable skills before entering the workforce.

SWA integration with CSD research. As part of each CSD issued BAA, there is a requirement for each proposal to identify how software security auditing will be accomplished. There is also a stated expectation that offerors will follow software design best practices and also encourages the use of the SWAMP.

## Customer(s)

Since software in ubiquitous across the government and the private sector, the potential customer set is very expansive. The Software Assurance Program works closely with the following organizations:

|  |  |  |  |
| --- | --- | --- | --- |
| **SWA Project** | **Customer Organization** | **Program/Office Name** | **Customer POC** |
| SQA | National Institute of Standards and technology (NIST | Software Assurance Best Practices related to FISMA | Ron Ross |
| SQA (Code Dx) | Domestic Nuclear Detection Office (DNDO) | DNDO ISSO | Doug Falls |
| SQA (Code Dx) | DHS S&T | Chief Information Security Office (CISO) | Michael Smith |
| SWAMP | Bowie State University | Department of Computer Science | Dr. Lethia Jackson |
| SWAMP | University of Indiana | School of Informatics and Computing | Von Welch |
| SWAMP | NIST | Software Assurance Metrics and Tool Evaluation (SAMATE) | Paul Black |
| SWAMP: Swamp-in-a-Box (SIB) | Raytheon, Army-CERDEC, Northrup Grumman | Beta pilot deployments | Multiple |

One of the main elements of the SWA program is the Software Assurance Marketplace (SWAMP) project. This is a research capability for entire software development community, which includes software assurance tool developers, software assurance researchers, software developers, infrastructure operators and software users and educators and students. The SWA approach to meeting the diverse and evolving needs of these stakeholders is outlined below:

* Software Assurance Tool developers: SWAMP provides the ability for assurance tool developers to develop their own work flows, assess multiple software packages and have the results delivered rapidly to the tool developers.
* Software Assurance Researchers: Software assurance tool developers are one part of the broader SWA research community, which is also looking more broadly at methodologies, standards and software development techniques. SWAMP facilitates this research in several ways, including but not limited to: providing a standardized analysis tool output and integrating software analysis tools with software packages.
* Software Developers: SWAMP provides software developers to create workflows to test their software using one or more of the software assurance testing tools resident within the SWAMP. This capability provides them immediate feedback and allows near real time change to ongoing development work, thus helping speed the transition of software capability to market as well as increasing the quality of the software.
* Software end users: This stakeholder segment will be able to create their own workflows to assess software of interest to them.
* Educators and students: The SWAMP provides educators and students a valuable resource for their academic programs, allowing for an analysis of software and analysis tools against the range of SWAMP housed software packages. Students and educators will be able to receive timely and valuable feedback. This capability will help drive software development practices into the educational system.

As of March 2017, the SWAMP has close to 1500 registered users from all of the above stakeholder/customer groups. The SWAMP runs approximately 1500 evaluations/week, and has a registered user base of approaching 1500 organizations, ranging from industry, academia and other members of the research community. In addition to the customer listed above, the Software Assurance program interacts with a broad and diverse set of stakeholders, detailed below.

Under the DHS Integrated Product Team (IPT) structure, the Cyber Security Division acts as the Executive Secretariat for the Secure Cyberspace IPT. Within the Secure Cyberspace IPT, the SWA Program Manager chairs the Software Security Assurance Delivery (SSAD) sub IPT. This sub-IPT prioritizes software assurance gaps from DHS components and other participants, including, United States Citizenship and Immigration Services (USCIS), National Aeronautics and Space Administration (NASA), and Health and Human Services (HHS).

In addition, the SWA program is involved in several federal and industry working groups and fora, including but not limited to: the Networking and Information Technology Research and Development Program (NITRD) Software Productivity, Sustainability, and Quality Interagency Working Group (SPSQ IWG) and the Software and Supply Chain Assurance Forum.

The SWA program manager also participates in program reviews from other federal agencies, as well as participates and collaborates with Open Web Application Security Project (OWASP). The frequency of interactions vary from quarterly meetings to ad-hoc meetings.

Evidence of customer support for the Software Assurance program includes:

* Installation by several parties of “SWAMP-in-a-box” for beta testing in their respective environments
* Participation in the SSAD Sub IPT by inter agency partners, including HHS, NASA and DHS components
* Bowie State and Indiana University customers have integrated software assurance as part of their coding and software development curriculum; student run their class projects through the SWAMP

## 1.3 Deliverables/Products

This section describes the products and services that will be delivered as part of the SWA program.

**1.3.1 Product or Service Descriptions**

The SWA Program delivers a wide range of software assurance tools and services.

The primary service offering is through the Software Assurance Marketplace Program (SWAMP). SWAMP provides a collaborative research environment for researchers and tool inventors to improve software assurance tools, critical to protecting the systems that power much of our interconnected, internet dependent economy and critical infrastructures. SWAMP provides two service options:

* the ready-to-use cloud computing platform at mir-swamp.org
* The SWAMP-in-a-Box (SiB) open-source distribution that is downloadable from GitHub.

Both are available at no-cost to the software development community and include an array of open-source and commercial software security testing tools and a comprehensive results viewer to simplify vulnerability remediation. Each also supports an API, allowing for integration of the SWAMP into existing software development workflows.

SWA also deploys a range of software testing applications for use in the marketplace. The tools that have been transitioned into use are listed below:

* Code Pulse (Open Source)
* Code Dx
* Code Sonar
* ThreadFix (Open Source)
* TOIF (Open Source)
* Code Hawk
* RevealDroid (Open Source)
* SWAMP-in-a-box (Open Source)

Under the ASTAM project, SWA will deliver a Unified Threat Management (UTM). This product is a system of tools and a continuous monitoring environment that will allow cyber security operators to monitor and manage a wide range of security applications and infrastructure components through a single management console/interface.

The STAMP project will produce a set of updated static analysis tools (products) with higher quality and performance that provide better coverage against open source and more modern software analysis tools. It will also provide services in the form of realistic test cases, and a capability to benchmark, score and label various software analysis tools.

The goal of the SWA program is to commercialize or make available through open source, all of its capabilities. In general, the deployment choice of commercialization or open source is left to the developer. Each SWA solicitation includes a requirement to describe the offerors transition plan as part of their technical approach. Specifically, the transition plans should include strategies for transitioning tools to the SWAMP, identification and a targeted list of potential transition partners, commercialization plans and a detailed description of how the transition execution plan. The PM works with the performers to develop commercialization plans and helps them develop use cases for specific transition partners, and provides funding to support targeted pilot deployments.

**1.3.2 Capability or Performance**

CSD issued a Broad Agency Announcement 11-02 with Technical Topic Area #14 seeking proposals for a national level Software Assurance Marketplace. The goal was to establish an evaluation environment where researchers and developers of analysis tools can continuously test and evaluate their SWA tools against a growing set of reference software and real software products, as well as comparing and contrasting their results against other tools and approaches. This environment, called the SWAMP, achieved Initial Operating Capability (IOC) in Q2, FY 14. DHS expects the SWAMP to become a national level R&D resource in software assurance for open security technologies, used across civilian agencies and their communities as both a research platform and core component supporting US Government supported software development activities. The SWAMP’s competitive advantage is provides the assessment infrastructure for users; they don’t have to worry about calibrating or configuring tools, and they can focus on selecting the right tools together to improve analysis of results. The SWAMP lowers the barrier to entry to help formalize software assurance. The SWAMP is one of the few, if only research infrastructures dedicated to improving software security through collaboration.

## Transition Plan

With 4 projects under the SWA program, and multiple contracted performers under each project, a single transition plan that applies to all of SWA is not desirable or realistic. The SWA Program establishes processes and foundational elements across the program that will help products transition, but each performer is expected to be heavily invested in their own product transition. These processes and elements include, but are not limited to:

* Requiring, as part of a proposal solicitation, that each offeror provide their transition plan (e.g. strategies for transitioning tools to the SWAMP, identification and a targeted list of potential transition partners, commercialization plans and a detailed description of how to execute the transition plan).
* Where needed and appropriate, help the offers develop solid use cases for their tools
* Connect performers with potential transition partners via conferences, brokered meetings, etc.
* Leveraging CSD resources such as the Assessment and Evaluation program to conduct red teaming analysis and other testing (e.g. functional) to help improve their products and increase the likelihood of transition
* Providing funding as part of the contract award to support pilot deployments for their technologies to increase the likelihood of tech transition

### Transition Pathways

There are multiple pathways that the SWA program utilizes in order to transition its’ tools, capabilities and techniques into operational use in the market space. The SWA program has utilized each approach for broader adoption of CSD funded R&D.

#### Commercialization. The performers need to follow their own business development approach strategy to commercialize their products. There is no “one size fits all” approach as each company’s situation is different and each product may have a different set of factors driving business decisions. However, there are some basic elements to the commercialization process which most companies should follow. These include: identify the need(s) of the target market, build the value proposition for the user; the business model to be employed; and how the company will enter and reach the market, including identification and engagement of partnering and collaboration entities. The SWA program does not direct these activities, but will advise performers as needed. For example, CSD provides training to Principal Investigators (PI) on how to market and frame their product around key elements a potential customer wants to know, e.g. What is the Need, what is the approach, what are the benefits and what are competitive or alternative approaches. For most SWA programs, commercialization will be the preferred pathway because it can reach a broader set of customers than by targeted transitions to a specific agency or organization.

#### Open Source. Open source as a transition path provides several advantages, including but not limited to: essentially crowd sourcing additional development and updates to a software capability, the ability for users to tailor software capabilities to their specific needs, anticipating new threats, responding to continuously changing requirements, and supporting software reliability and security efforts. The SWA program, particularly through the establishment of the SWAMP and more recently the piloting on SWAMP-in-a-Box, will provide a platform for open source tools to be exercised.

#### Transition to DHS Component Use. If a SWA technology is planned to transition to a DHS Acquisition program, the SWA PM will work closely with the component acquisition executive and the component requirement champion to identify component specific requirements needed to ensure a successful transition. This may include: the major program objectives, Current phase of the acquisition life cycle and which phase of the life cycle the SWA capability will be inserted into desired Initial Operational Capability (IOC) date, and Projected life-cycle costs, including operations and support The DHS components require software assurance tools and capabilities for their environments and may use SWA capabilities as additional tools in their cybersecurity operational environment. This may fall outside of a formal DHS acquisition program. In these cases, the SWA program will work closely with the respective component CIO/CISO to ensure the capability undergoes the appropriate level of test and integration into the component’s environment.

#### Other/Knowledge Product. Part of the SWA program produces research reports and knowledge products which provide the software assurance research and user community described in Section 1.2 with valuable insights, techniques, standards and approaches for addressing the software security challenge. For example, the SWA program funded the mapping of the NIST Special Publication (SP) 800-53, Security and Privacy Controls for Federal Information Systems and Organizations. This document identifies and extracts applicable software security controls in order to bridge the gap between Certification & Accreditation security controls and software weaknesses, such that the risks can be understood and effectively addressed. Elements of this work are incorporated into NISTP (Draft) 800-160, Systems Security Engineering Considerations for a Multidisciplinary Approach in the Engineering of Trustworthy Secure Systems.

### Project Transition

The goal of the SWA program is to commercialize or make available through open source, all of its capabilities. In general, the deployment choice of commercialization or open source is left to the developer. Part of each SWA solicitation includes a requirement for each offeror to describe their transition plan as part of their technical approach. Specifically, the transition plans should include strategies for transitioning tools to the SWAMP, identification and a targeted list of potential transition partners, commercialization plans and a detailed description of the transition execution plan. The table below highlights products that have transitioned under the SWA program:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Project** | **Tool/Technology** | **Transition Type (Commercialization, Open Source, Transition to DHS Use, Other Government Use)** | **Performer** | **Expected Transition Date** | **Impact** |
| SQA | Code Pulse | Open Source | Secure Decisions | 05/2014 (Completed) | 1500+ downloads, incorporated into Secure Decisions Code Ray Hybrid Analysis Suite. & Denim Group’s Thread Fix platform as part of ASTAM. |
| SQA | Code Dx | Transition (DNDO, S&T CIO, CERDEC) | Secure Decisions | 2013 (Completed) | $83K in kind funding from a financial bank. Part of SWAMP |
| SQA | RevealDroid | Transition/Knowledge Product |  | 03/2016 | Currently used in the SWAMP |
| SQA | Tool Output Integration Framework (TOIF) | Open Source & Commercialization (integrated KDM Analytics Workbench – customers DRDC) |  | 10/2012 | TOIF is integrated into the KDM Analytics Workbench. In use by Defense Research and Development Canada (DRDC) |
| SWAMP | SWAMP IOC | Operational Capability |  | 2/1/2014 |  |
| SWAMP | SWAMP in a Box | Open Source |  | 01/2017 | In pilot testing with several customers; deploys SWAMP capability within Enterprise networks |
| SQA | Code Sonar | Commercialization (Numerous commercial sector applications – US Army, FDA, WMS Gaming Inc.,) | Grammatech | 10/2013 | Improved their own internally developed SWA tools (% market share). Part of SWAMP |
| SQA | HAM - ThreadFix | Open Source & Commercial Version | Denim Group | 10/2016 | % market share (in Gartner’s magic quadrant). Part of SWAMP |
| SQA | Scan Project (Static Analysis Tool) | Commercialization | Coverity | 10/2012 | Part of SWAMP |
| SQA | NIST Mapping of 800-53 | Knowledge Product | University Of Nebraska Omaha | 03/2015 | Identify NIST 800-53 security controls with software assurance applicability that will help guide SW developers in implementing security controls to protect federal systems |
| SQA | Code Hawk – Binary Analyzer | Commercialized | Kestrel, Inc. | 06/2016 | In use by 3 Large Scale Integrators and SOC |
| SQA | Application Health Check | Commercialized | Sonatype | 02/2017 | Implemented and hosted in SWAMP |
| SQA | Reducing False Positives | Knowledge Product | Trustees of Indiana | 03/2017 | Being used in STAMP |

## Schedule and Budget Summary

**1.5.1 Project Schedule and Budget Summary**

The SWA program level budget and schedule are provided below:



**Figure 1: High-Level Milestone and Budget Summary**





**Table 1: High-Level Budget Summary**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Funding Source** | **FY15** | **FY16** | **FY17** | **FY18** | **FY19** | **FY 20** | **Total** |
| Internal (S&T) | **$ 5.56** | **$ 8.51** | **$ 12.7** | **$ 14.5** | **$ 8.85** | **$ 5.73** | **$55.85** |
| External | N/A | N/A | N/A | N/A | N/A | N/A |  |
| External | N/A | N/A | N/A | N/A | N/A | N/A |  |
| **Total** | **$ 5.56** | **$ 8.51** | **$ 12.7** | **$ 14.5** | **$ 8.85** | **$ 5.73** | **$55.85** |

**Table 2: Budget Source Summary****($M)**

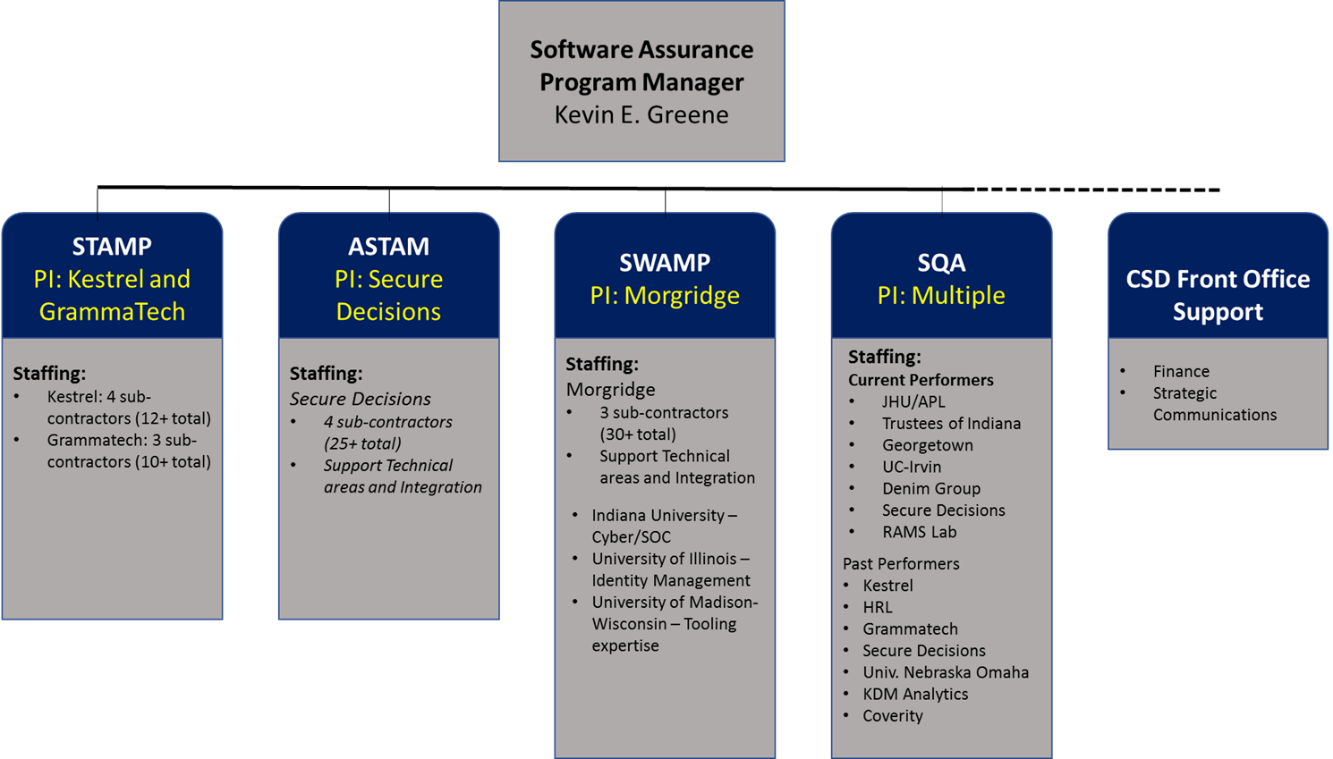
## Project Organization

### Project Personnel

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | **Organization** | **Title/Role** | **Email** | **Phone** |
| Kevin Greene | DHS S&T/HSARPA/CSD | Program Manager | [Kevin.Greene@hq.dhs.gov](mailto:Kevin.Greene@hq.dhs.gov) | 202-254-6877 |
| TBD | DHS S&T/HSARPA/CSD | SETA | TBD |  |
| Michael Costello | DHS S&T/HSARPA/CSD | Financial Support SETA | [Michael.Costello@associates.hq.dhs.gov](mailto:Michael.Costello@associates.hq.dhs.gov) | 202-254-5722 |
| Shelby Smith | DHS S&T/HSARPA/CSD | Communications Support SETA | [Shelby.Smith@associates.hq.dhs.gov](mailto:Shelby.Smith@associates.hq.dhs.gov) | 202-254-6796 |

Figure 3: Project Personnel

### Internal Structure



The SWA PM is ultimately responsible for all aspects of the Program. The SWA SETA supports the PM in all aspects of Program Management (e.g. cost, schedule performance management, program support activities, including but not limited to program strategy development, program review execution, coordinating with other elements of CSD). CSD Comms provides communications support to the SWA program as a matrixed capability. The office helps with messaging, press releases, graphics support and other communications support. The CSD Finance team support the PM and SETA team by reviewing/QC of PR packages, ensuring spend plan execution is adhered to, etc.

## Benefits to S&T

### New Capabilities / Technology

New capabilities that SWA has or will develop and deliver include:

* The SWAMP provides a high visibility capability that benefits the entire software development and software assurance communities.
* Mathematically proving code locations are safe from memory safety bugs similar to the bug that exposed the Heartbleed vulnerability
* Hybrid Analysis Mapping (HAM) that brings together independent testing activities/disparate tools; Static Application Security Testing (SAST) and Dynamic Application Security Testing (DAST) to improve security testing.
* The use of capture and recordings to perform reverse execution and crash analysis on software programs to uncover and detect malicious software.
* Development of a compositional verification technique called COVERT that combines both program and formal methods for finding exploitable bugs in mobile apps.
* Study and pattern analysis (big data analytics and machine learning) of software to determine what triggers false-positives in tools.
* Automated penetration testing, threat modeling, and attack surface analysis to mimic activity of potential attackers in real-time.
* Tunable information flow policy that significantly reduce run-time overhead (dynamic analysis), improve precision in finding information flow type weaknesses in software.
* Dynamic tracing and augmentation to provide insight into real-time code coverage of black box testing activities with visualization and analysis of attack surface.

### New Intellectual Property

No new intellectual property is expected to be produced by this effort. However, in the course of project execution, it becomes apparent that new IP will be produced, the SWA PM will coordinate with Lavanya “Elle” Ratnam at [Lavanya.ratnam@hq.dhs.gov](mailto:Lavanya.ratnam@hq.dhs.gov) at S&T.s Office of General Council for assistance with all IP questions and issues.

### Public Access of Federally Funded Scientific Research

DHS is developing a Plan to Support Increased Public Access to the Results of Research Funded by the Federal Government, currently in draft, and will likely require mechanisms to provide public access to peer reviewed journal articles that are funded by federal R&D after a prescribed embargo period. (Note: This plan will be posted on Knowledge Management and Process Improvement SharePoint site [*https://collaborate.st.dhs.gov/OUS/kpo/default.aspx*](https://collaborate.st.dhs.gov/OUS/kpo/default.aspx) when available.

### Data Management

The to-be-issued DHS Plan to Support Increased Public Access to the Results of Research Funded by the Federal Government is expected to require the creation of data management plans by those R&D projects creating electronic data. It will likely require mechanisms to provide public access to peer reviewed journal articles that are funded by federal R&D after a prescribed embargo period.

### Other Benefits

No additional benefits for S&T are expected to be produced by this effort.

## Assumptions

None.

# Requirements

## Requirements Management Plan

Requirements are managed by the SWA PM using MS Office products (i.e. Excel). Requirements are collected from a variety of sources, including but not limited to: The DHS IPT process (i.e. the SSAD Sub IPT), the previously mentioned fora that the SWA PM participates in (e.g. the Networking and Information Technology Research and Development Program (NITRD) Software Productivity, Sustainability, and Quality Interagency Working Group (SPSQ IWG) and the Software and Supply Chain Assurance Working Group.

Once requirements are collected, they are prioritized on an annual basis. In 2016, the existing requirements were prioritized using the Decision Lens tool. The SWA program will be able to follow this process in FY 17, if that is the path that S&T and DHS elects to take. The SWA requirements are validated through a review at the Secure Cyberspace IPT and based on the prioritization at that level, can be advanced to the Science Research Council (SRC) for prioritization. In the 2016 IPT process, one of the Software Assurance technology gaps was designated a high priority gap at the SRC level.

**2.1.1 Test and Evaluation**

The SWA program provides an evaluation service through the SWAMP project for any members of the software development and/or software assurance tool development communities.

Individual tools developed as part of the SQA, ASTAM and STAMP projects will be subject to internal developmental testing. The SWA Program Manager has the opportunity to review T&E plans for each project activity. Software Assurance tools developed under the SWA program are run through the SWAMP and use the curated data sets there to evaluate their tool effectiveness. Once final transition to the SWAMP occurs, the assurance tools become part of the SWAMP tool repository and are available to the SWAMP user community to conduct testing.

Another form of testing and evaluation is achieved through the deployment of technologies via open source (e.g. GitHub). This transition path leverages the power of crowd sourcing to identify issues (functional, security) and allows for the tool code to be tailored for different end user applications.

The SWA PM will also work with the CSD Program Manager for Assessments and Evaluations (A&E) to conduct selected testing on different project technologies under the SWA program. The A&E testing and evaluation options include:

* Functional Testing: Intended to evaluate the overall utility of the technology and factors in the details of the potential deployment site
* Operational/Pilot Testing: Consists of functional testing plus determination of the technologies operational readiness with respect to a particular end user or pilot environment
* Red Teaming: This consists of a comprehensive security assessment using adversarial techniques to identify security issues. The SWA program conducted a red team assessment of the SWAMP in 2014, which provided valuable insights for the SWAMP team, both from a digital and physical security perspective.

## Customer Engagement

The primary SWA customers are outlined in Section 1.2. The SWA program manager uses a variety of methods and techniques to communicate and engage with SWA customers and stakeholders. These include:

* Invitations for customers to attend CSD sponsored events, including but not limited to the annual R&D Showcase and Technology Workshop,
* The SWA PM also interacts with customers at fora of common interest (the Networking and Information Technology Research and Development Program (NITRD) Software Productivity, Sustainability, and Quality Interagency Working Group (SPSQ IWG) and the Software and Supply Chain Assurance Forum)
* Several customers are actively engaged in piloting SWA capabilities, primarily, the SWAMP-in-a-box. These organizations are also listed in Section 1.2 For example, SIB pilot is complete and is now available on GitHub for download.
* The SWA PM interacts with the DHS SSAD Sub IPT on at least a semiannual basis as part of the annual DHS IPT process.
* The SWA PM and CSD engage with the broader research community as well, using a variety of methods, including publications (e.g. blogs) and social media (Facebook, Twitter)

## Customer Requirements

The table below captures the high level customer requirements captured during the 2016 IPT Process through the SSAD Sub IPT. These were validated at the SSAD sub IPT level in October 2016 through the voting and prioritization process that the voting members of the SSAD had the opportunity to participate in. Software Assurance does not have an Operational Requirements Document (ORD) associated with it. SWA is an underlying function across multiple operating missions and multiple domains.

|  |  |  |  |
| --- | --- | --- | --- |
| Gap ID# | Title | Description | Disposition |
| G-FY16/17-CS-SSAD-8 | Secure Coding | Research in the area of secure coding is required to help eliminate egregious bugs by reinforcing good coding practices through standards, policy, and best practices. Improve the identification, coverage and reporting of software weaknesses. This will result in the development of federal compliance standards for secure coding; identify improvements and expansion in Common Weakness Enumerations (CWE) for reporting weaknesses in software; compliance mappings to NIST 800-53 that overlays coding violations, CWEs, and security controls; and a framework that can be used by static analysis tools to help identify violations in standards and report on them. | Being accomplished though Trustees of Indiana(SQA) and through S2ERC through data analysis processes to identify patterns and trends in software development issues |
| G-FY16/17-CS-SSAD-001a,b,c | Open-Source Component/Library Management | Lack mechanisms for documentation/permissions approval.  This includes:  1) Largely a manual (and often hardcopy/paper) process to request access to systems or services with long delays and no visibility into the approval cycle.  2) A system providing built-in workflow and electronic/digital approvals with a dashboard (or e-mail) view of the approval process would be ideal  3) Document storage might be “per user” so a complete spectrum of “requests”, “approvals”, or prerequisite “training certificates”, etc. might be maintained.  4) Published directory/registry of vetted shared libraries (similar to Forge.mil)  5)Digitally signed libraries  6) Limited capability to support of C/C++ code when performing origin analysis. Need more tools that perform this type of analysis.  7) Also need ability to analyze source code (not just binaries) for use of open source code snippets. | Partially addressed by SonaType Application Life Cycle check into the SWAMP. |
| G-FY16/17-CS-SSAD-002 | Vulnerability Management/Patch Management & Continuous Monitoring | Current capability gaps in the area of Vulnerability/Patch Management include:  1) Mechanisms for storing disparate vulnerability data into a ‘universal container’ for vulnerability trending, remediation, ISVM/CVE implications, Risk identification, etc.  2) Ability to scan container configurations (e.g. Docker, EC2)  3) Plugins/tools to scan for known indicators in STIX/CyBOX/MAEC  4) Additional DHS hardened application configurations Apache 2.x, Tomcat 7.x / 8.x, NoSQL Databases (i.e. MongoDB, Hadoop, Amazon Aurora)  5) Continuous patching verification capability via automated agents, based on criticality and exposure | Future research topic (Software Vulnerability Discussion, FY 19 new start) |
| G-FY16/17-CS-SSAD-003a, b, c,d | Penetration Testing | Lack the capability for robust, automated and on demand Pen Testing capability, to include Record & Scheduling System: :   1. Formal record of Penetration test approvals, Rules of Engagement, Team Members involved, Scope of Test, etc. 2. Test results 3. Remediation efforts or POA&M creation 4. Lessons Learned 5. Training indicators (i.e. if SOC missed a noisy penetration test, addition training may be indicated to resolve)   • Consolidated scan results between commercial responses (similar to DHS S&T ThreadFix)  • Exportable Pen Test results to STIX/CyBOX/MAEC to compare against Threat Analysis/Intelligence software  o Penetration testing capability must be able to be scripted to be able to: Run against lists of specific IP addresses, entire ranges of IP addresses, exclude specific IP addresses within ranges of IP addresses  o Penetration testing capability must be able to spoof IP addresses and MAC addresses  o Penetration testing capability should be able to set specific characteristics within attack signatures to allow their addition to whitelisting tools (more of a want then a requirement).  o Penetration testing capability must be able to generate output in standardized formats that can be imported into other industry tools (e.g. Nessus)  o Penetration testing capability must be able to be updated to take advantage of zero-day vulnerabilities  o Automated penetration testing. Technology that could monitor network nodes and perform some level of penetration testing as node come online. Depending on the type of device it could launch an automated set of tests against it when the device comes on the network. | This capability will be automated and be available on demand through the ASTAM project (FY 19-20) |
| G-FY16/17-CS-SSAD-0004 | Static Application Security Testing (SAST) | Lack the capability to scan Ruby on Rails and Python applications with dependencies | This gap will be addressed by the STAMP project |
| G-FY16/17-CS-SSAD-005 | Dynamic Application Security Testing (DAST) | Lack the capability to perform Jenkins scanning and integrate the results into CI/CD pipelines  Need a method to standardize severity results (similar to DHS S&T ThreadFix) | This gap will be addressed by the ASTAM project by \_\_\_ |
| G-FY16/17-CS-SSAD-006a,b,c | Malware Analysis | Currently lack the capability to scan applications for known Indicators of Compromise (IOC)  Lack the capability to detect malware in web traffic traversing from systems within the environment. This capability shall, at a minimum, be:  - Capable of detecting and blocking advanced zero-day exploits and outbound callbacks  -Able to detect exploits using both signature and signature-less engine(s)  -Able to monitor assets without the need to deploy software agents  -Capable of supporting auto-configured test environments to safely execute and inspect suspected malware payloads  -Able to dynamically generate actionable malware intelligence  -Able to detect and stop web-based attacks  o The malware analysis capability must be able to be deployed at strategic network egress points to ensure complete coverage.  o The malware analysis capability alert data must be able to be sent to a SEIM system.  o The connection must be accomplished with either a network tap or with the use of a span port on the infrastructure router.  o The malware detection capability should be able to be placed in-line to allow for active blocking of malicious activity or only on a tap/span port for detection purposes.  o Malware analysis system must be able to connect to third-party intelligence sources.  o Malware analysis system must be able to thoroughly trace the impact of suspected payload.  o Malware analysis system must be able to extract potential malware into a sandbox environment where it can be executed  o Malware analysis system must have robust reporting capabilities  o The malware analysis capability must be able to generate indicators of compromise | Kestrel CodeHawk binary analysis tool. Planning pilots with key stakeholders (Gold Standard) |
| G-FY16/17-CS-SSAD-008 | Continuous Monitoring (from an AppSec perspective) | • SCAP rulesets for automated secure application development checklists and scanning  o Federal NIST 800-53 v4 security controls  o OWASP Top 10s  o DHS Sensitive Systems security controls  o DoD Application and Security Development STIG  • Ability to scan deployed applications against a good known for configuration/compromise change detection | This gap will be addressed by the ASTAM project |
| G-FY16/17-CS-SSAD-020 | Vulnerability Management/Patch Management | o The Vulnerability Management capability must be able to scan assets for known vulnerabilities, such as missing patches and improper access controls, and software weaknesses.  o The Vulnerability Management tool must provide CVE feeds and:  o Discover and locate known security vulnerabilities  o Discover and locate software weaknesses in software applications and source code  o Support awareness and understanding of exposure risks associated with software weaknesses  o The Vulnerability Management tool must be able to export a format that can be ingested by other tools (e.g., RSA Archer EGRC tool, RedSeal Network Analyzer network mapping and risk context analyzer tool)  o The Vulnerability Management tool must score (assign a numerical value to) deviations for purposes of computing risk. Scoring includes:  - Scores for all possible values of a configuration setting.  - Standard Federal score for deviations from accepted benchmarks.  - Custom scoring models as defined by the Agency | This gap will be addressed by proposed new start in FY 19, Software Vulnerability Detection System (SVDS) |

The direction following the 2016 IPT process was to focus on addressing current capability gaps rather than soliciting for new gaps. To achieve this, the SWA PM will work in conjunction with the SSAD sub IPT over FY 17, Q2 and Q3 to map existing SWA programs against these validated capability gaps.

## Key Functions and Desired Performance Requirements

Measuring the impact of software assurance improvements on software development activities is challenging largely because current state-of-the-art technologies have not kept pace with the evolution of software development. Software development is quickly evolving with new approaches such as DevOps, Continuous Integration (CI), and Continuous Deployment (CD) to get software to the market faster by taking advantage of improved collaboration between business stakeholders, application development, and operations teams.  To achieve security “at-speed’ organizations need better performing tools and capabilities. Below you will find key performance measure that will help create forward leaning capabilities to increase the adoption rate of software assurance tools early in the software development process.

The key performance parameters are outlined in the table below:

|  |  |  |  |
| --- | --- | --- | --- |
| KPP# | Desired KPP | Threshold | Objective |
| 1 | Improve applications security testing coverage with Hybrid analysis that will allow users to identify and reduce false-positives, and find those security issues that matters. | Improve testing coverage for web applications to cover at least 58% of the attack surface. | Expand testing coverage for web applications to cover 85% of the attack surface. |
| 2 | Enable the detection of advanced malware and malicious software | Detect advanced malware techniques such as obfuscation techniques and polymorphic/encrypted malware with a precision rate of 53%. | Expand capabilities used in advance malware detection with precision rate of 96%. |
| 3 | Identify false-positive patterns in static code analysis that impact the performance of software quality assurance tools. | Identify and collect nearly 35% of false-positive patterns that show a large number of warnings in the Juliet test case suite | Increase to nearly 70% of false-positive patterns that show a large number of warning in the Juliet test case suite and real software (open-source). |
| 4 | Enable the detection of obfuscated and polymorphic malware | Detect advanced malware techniques, such as obfuscation and polymorphic, without using virtual machine (VM) for advanced analysis. Most technologies use VM introspection and modern malware is designed to go to sleep if detected running inside a VM. | Expand capabilities used in malware analysis to use abstract interpretation which is designed to improve the performance and detection of malicious software. Most solutions use symbolic execution which is known to suffer from path explosion and performance issues. Analyze over 2,000 malware samples from various sources with a detection rate near 97%. |
| 3 | Software Assurance Mappings to NIST 800-53 controls | Assess all NIST 800-53 controls to determine software assurance applicability. Provide mappings to Common Weakness Enumerations (CWE) to help guide developers and assessors in understanding impact. | Develop a methodology to evaluate and test software assurance as part of the certification and accreditation (C&A) process as well as help developers address security early in the development and design phase of the systems. |
| 5 | Develop an ability to effectively measure static analysis tools in memory safely | Mathematically prove at least 70% of relevant code locations are deemed safe from memory safety issues | Mathematically prove near 99% of code locations are free from memory safety issues |
| 6 | Detection of Android Malware in Mobile devices | Develop capability to find at least 75% of malware in Android mobile applications using the RevealDroid platform | Increase detection capability to find 90% of malware in Android mobile applications using RevealDroid platform |
|  | Develop a hybrid analysis capability that correlates and normalizes disparate Static Analysis Security Testing and Dynamic Analysis Security Testing tools | Capability to correlate and normalize at least 4 SAST and 4 DAST tools with goal of having an agnostic ability to ingest and integrate any SAST or DAST tool. | Develop and evolve data ingesting framework to be tool agnostic |
| 7 | Develop a system that correlates and merges the results of open-source and commercial automated static and dynamic security scanning technologies; a hybrid analysis attack surface | Utilize DAST tools to reduce the amount of hidden attack surface to 35% of the current application attack surface | Utilize DAST scanning tools to reveal areas of the application attack surface that may be missed; reduce the amount of hidden attack surface to at least 15% of the current application attack surface |
|  | Develop next generation test cases that represent modern software to include complexity and size to measure static analysis tools in the area of precision (false-positives) and soundness (false-negatives). | Reduce false-positive rates by 33%, improve visibility into false-negatives by 48%. | *Reduce false-positives rates by 53%, improve visibility into false-negatives by 64%.* |
| 8 | Modernize open-source static analysis tools for better tool coverage and performance. | Reduce false-positive rates by 53%, improve visibility into false-negatives by 44% for each tool (maximum of 5 tools) | Reduce false-positives rates by 72%, improve visibility into false-negatives by 71%. |
| 9 | Provide automated red teaming and penetration testing capabilities to improve continuous monitoring activities. | Automate red-teaming and penetration testing which will help reduce the window of exposure by 25-30%. On-demand capability and automated remediation response will provide a proactive approach to reduce attacks to systems. | Automate red-teaming and penetration testing which will help reduce the window of exposure by 50%. On-demand capability and automated remediation response will provide a proactive approach to reduce attacks to systems. |
| 10 | Develop a research infrastructure that will help tool developers improve their analysis capabilities. | Develop a research infrastructure that hosts a corpus of 600+ diverse software packages in different programming languages, as well as host datasets/test suites from IARPA and NIST to provide an effective way to measure tool performance. | Develop a collaborative research environment that will assist tool developers in improving their tools. Similar to a lab environment, the SWAMP serves as a way to find new breakthroughs and advancements in the way software tools analyze complex software applications. |
| 11 | Expand the number of users using the SWAMP software assurance services | During the initial operating capability (IOC), 117 users were registered using the SWAMP | Increase the number of users in the SWAMP to over 1200 users |

Figure 4: Key Functions and Desired Performance Requirements

## Other Metrics

As previously discussed in Section 1.2, the SWA measures SWAMP testing statistics and the user base. As of Q1, FY17, there are registered SWAMP users from over 30 countries. There are also over 400 Git Hub users.

# Technology and Innovation

## 3.1 Core Technology

The SWAMP offers a no-cost, cloud-based, high throughput computing platform at mir-swamp.org that is capable of analyzing over 275 million lines of code each day. With this infrastructure, users can conduct a variety of tests on applications of any size in a timely manner. The SWAMP staff maintains and updates all tools and platforms available in the SWAMP, ensuring that users can focus on testing and not worry about the infrastructure necessary to support testing activities.

The SWAMP’s shared, continuous assurance facility simplifies integrating security into the classroom and software development life cycle by offering a marketplace of open-source and commercial software analysis tools to perform comprehensive security testing on your own applications. A library of public applications with known vulnerabilities is also provided for download, testing, and assessment. Research has shown that it is necessary to use more than one tool when testing a software package to ensure that secure coding practices were followed and that there are no vulnerabilities in the application. In the SWAMP, results from multiple testing tools are compiled into a single integrated viewer that presents identified weaknesses in a way that helps users prioritize and fix each error in the code. As weaknesses are addressed, software applications become more secure but should be reassessed to ensure that no new weaknesses were introduced during the remediation process or throughout the software development life cycle.

New science and technology generated from this program is listed below:

* New analysis techniques for static code analysis through the STAMP project
* Improved testing and evaluation techniques for static analysis tools
* Unified Threat Management (UTM) system. This provides the capability for security professional to monitor and manage a wide range of security related applications and infrastructure through a single management environment.

The key R&D center for this SWAMP is the Morgridge Institute of Research, where the testing tools and platforms are located. There are multiple projects and multiple contracts associated with this program, so there is no aggregate TRL level calculated for the SWA program as a whole. Projects may starts at different TRL levels, especially if they are leveraging previous work.

## Technology Scouting

Technology scouting for this program is performed through multiple methods:

* The PM stays current on emerging technology areas related to software assurance by attendance at technical conferences and workshops, often speaking on panels related to software security and by meeting with academic and industry researchers
* Outside technical studies by NSA and NIST were reviewed by the SWA PM and formed the basis of the SWA program because it identified gaps in existing software assurance tool coverage.
* SWA PM collaborates closely with other Federal R&D efforts, and has done so in the past (e.g. with NIST Software Assurance Metrics And Tool Evaluation (SAMATE) program and IARPAs Securely Taking On New Executable Software of Uncertain Provenance (STONESOUP)
* As mentioned in Section 1.4, releasing a BAA is in effect, performing a technology scouting exercise because the government will receive a range of proposed technical solutions to evaluate

As the SWA program conducts planning for its anticipated new start for Software Vulnerability Detection System (SVDS), it will engage, as needed with the Tech scouting office in RDP as well as continuing with its current practice of wide spread research and industry community engagement.

## 3.3 Technical Approach

### Trade Space Analysis

Preliminary non material solution analysis revealed the importance of attacking the software assurance challenge earlier in the software development process. Proper software coding best practices and education is an important foundation for this. Partially as a result of this analysis, the SWA program ensured that academic institutions were part of the outreach campaign for the SWAMP project. Numerous institutions have access to the SWAMP for instructional purposes to teach secure coding best practices.

### Preferred Alternative

Multiple proposals were received and reviewed for the major elements of the SWA program (e.g. SWAMP, ASTAM and STAMP). The proposals were evaluated per the criteria described in Section 4.1.2 below. Since this is a program with 4 different projects, there is no single preferred alternative or approach. CSD and the SWA program generally solicit technical solutions based on a technical topic area (common in research and development activities) vs. soliciting for solutions against prescribed requirements (more common in acquisition programs).

### Logistics Considerations

There are no significant logistical challenges associated with this project. Most project communication is done electronically or via email. In person reviews are scheduled annually, preferably around the primary CSD event, the annual R&D Showcase and Technical Workshop.

# Project Management

## 4.1 Acquisition Management

Streamlined Acquisition Plans (AP) will be submitted for each of the planned acquisitions and any subsequent acquisitions valued in excess of $150,000 and approved in accordance with the FAR and departmental policies and directives. Approved APs will become appendices to this AP.

### Program Acquisitions

The list of all SWA Program acquisitions are listed below in Figure 5.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Acquisition**  **Description** | **Acquisition Strategy**  **(Contract, BAA, Grant, IA etc.)** | **Location of Work To Be Performed** | **Period of Performance** | **Total Estimated Value (w/Options)** |
| Static Tool Analysis Modernization (STAMP) | BAA HSHQDC-16-R-B0002 | Kestrel Technology, LLC | 9/26/16 – 9/25/20 | $7.863,636 |
| Static Tool Analysis Modernization (STAMP) | BAA HSHQDC-16-R-B0002 | Grammatech | 9/19/16 – 9/18/20 | $8,000,031 |
| Application Security Threat Attack Modeling (ASTAM) | BAA HSHQDC-16-R-B0003 | Secure Decisions | 9/19/16 – 9/18/20 | $16,339,743 |
| Software Assurance Marketplace (SWAMP) | BAA 11-02 | Morgridge Institute for Research | 10/1/2012 – 9/30/2018 | $11,755,068 |
| Software Quality Assurance (SQA) | IA (AFRL) | Kestrel Technology, LLC | 11/5/2014 – 9/30/2016 | $1,050,181 |
| Software Quality Assurance (SQA) | IA (DOI) | JHU/APL | 3/25/2016 – 3/24/2017 | $491,143 |
| Software Quality Assurance (SQA) | IA (DOI) | Trustees of Indiana | 3/25/2016 – 3/24/2017 | $394,000 |
| Software Quality Assurance (SQA) | Contract | Univ California, Irvine | 9/23/2014 – 9/22/2017 | 499,999.74 |
| SBIR | IA (DOI) - SBIR | Applied Visions, Inc. | 3/15/2014 – 11/29/2016 | $1,034,352.27 |
| SBIR | IA (DOI) - SBIR | Denim Group, Ltd | 3/5/2015 – 11/30/2017 | $749,860.22 |
| SBIR | IA (DOI) - SBIR | RAM Laboratories, Inc | 9/28/2015 – 10/13/2017 | $749,993.51 |

Figure 5: Table of Acquisitions

### Acquisition Strategy

The majority of the SWA program work has been and will continue to be awarded through a Broad Agency Announcement (BAA). The BAA allows for each offeror to present their own, separate approach to solving various elements of a problem. This helps create technical competition through this “marketplace of ideas”. The Offerors draft their own statement of work and technical approach.

Each BAAs have their own evaluation criteria. In some cases, such as for the ASTAM and STAMP BAAs, special technical requirements for proposals were added. These included:

* Proposing technical and operational metrics that measure progress towards the final capability
* Proposing a go/no-go demonstration for the different Technical Topic Area (TTA)

Past performers for the SWA Program include:

* University of California, Irvine (UCI)
* Johns Hopkins University, Applied Physics Lab (JHU/APL)
* Trustees of Indiana
* Demin Group, LLC
* RAM Laboratories, Inc.
* Ball State University
* Applied Visions, Inc.
* Kestrel Technology, LLC
* Grammatech, Inc.

Performers are chosen primarily through a competitive solicitation process, generally a BAA. As outlined in Figure 5, this is how the majority of the SWA program funding was awarded (e.g. SQA, SWAMP, ASTAM and STAMP). The SWA program also utilizes the Small Business Innovation resear h (SBIR) program to leverage innovative ideas developed in the small business space. The proposal evaluation criteria are described in the relevant overarching BAA documents. For example, BAA 14-R-B0005, the evaluation criteria were as follows:

**Evaluation Criteria**. The evaluation criteria to be used for the peer/scientific review of both white papers and proposals received in response to each call issued under CSD BAA HSHQDC-14-R-B0005 is identified below in descending order of importance.

**Criterion I**: Responsiveness to Technical Topic and Potential Impact. Potential of the technology/solution for meeting the project goals provided in BAA call will be assessed. Further, an assessment will be made as to both the breadth of the responsiveness to the topic area goals the solution represents, and potential impact of successful implementation of the proposed solution.

**Criterion II**: Technical and Managerial Approach. Sound technical and managerial approach to the proposed work, including a demonstrated understanding of the critical technology or engineering challenges required for achieving the project goals of the BAA call will be assessed. An assessment will be made to determine not only the technical feasibility of the proposed development, but also of how the development will be managed including any relevant dependencies, to include milestones, to accomplish success. Furthermore, an assessment will be made to determine whether the task descriptions and associated technical elements are complete and in a logical sequence, with all proposed deliverables clearly defined such that a viable attempt to achieve project goals is likely as a result of award. The proposal identifies major technical risks and clearly defines feasible mitigation efforts.

**Criterion III**: Offeror’s Capabilities and Related Experience. The offeror’s prior experience in similar efforts will be assessed to determine if the offeror clearly demonstrates an ability to deliver products that meet the proposed technical performance within the proposed budget and schedule. In addition, the proposed team will be reviewed to determine whether the personnel have the expertise to perform the proposed work as well as the ability to manage the project cost and complete the project within the proposed schedule.

**Criterion IV**: Transition Approach. As technology adoption is a major DHS S&T CSD goal for R&D projects, a qualitative assessment will be made regarding how the proposed technology/solution will be transitioned to an operational user (e.g., commercialized or used by a DHS component). The assessment will determine the likelihood that the offeror will be able to deploy a technology and/or solution(s) that can be transitioned effectively to the user community either through commercialization of the technology, open source distribution, or through other means. Because intellectual property rights will impact technology transition, an assessment will be made of software/data to be included in the proposed technology for which the Government would not receive unlimited rights (as identified in the Assertion Table included with the proposal). The assessment will consider: 1) proposed use(s) of the software/data; 2) the explanation as to how the Government will be able to reach its technical goals (including transition) within the proprietary model offered; and 3) the nonproprietary alternatives in any area that might present transition difficulties or increased risk or cost to the Government under the proposed proprietary solution.

**Criterion V**: Cost.

(1) White Papers. The cost evaluation factor for white papers, when requested, is as follows: “Cost. White papers will be evaluated on the affordability of proposed technical work.”

(2) Proposals. The cost evaluation factor for proposals is as follows: “Cost Reasonableness and Cost Realism. The proposed costs are reasonable (i.e., reflect a sufficient understanding of the technical goals and objectives of the solicitation, and are consistent with the offeror’s technical/management approach (to include the proposed SOW)), and are based on realistic assumptions. The costs for the prime and subcontractors/consultants are substantiated by the details provided in the proposal (e.g., the type and number of labor hours proposed per task, the types and quantities of materials, equipment and fabrication costs, travel and any other applicable costs).”

In addition, each call may specific additional considerations and proposal requirements, for example, in BAA call HSHQDC-16-R-B0002, special submission technical requirements were included: Defining the target capabilities and associated technical and operational metrics, go/no go demonstrations and use of best practices in software design.

### Potential Sources

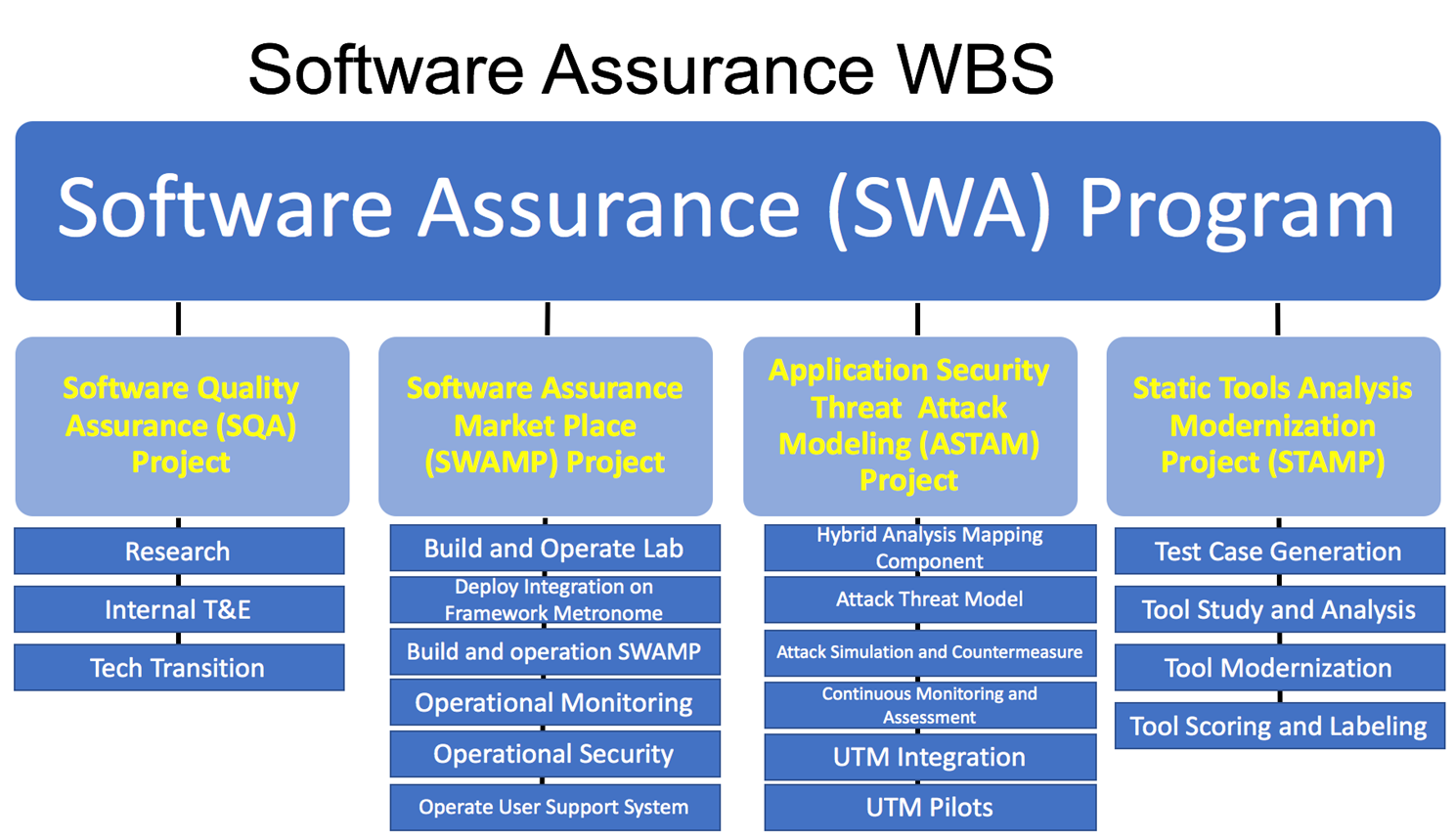
There are numerous organizations that could be used as transition candidates. Tow obvious examples within DHS are: the DHS S&T OCIO and the Domestic Nuclear Detection Office (DNDO), both of which have piloted and deployed Code Dx in their environment. Based on the above market facts, identify sources that could reasonably be expected to be involved in the transition. Analyze whether the sources have the ability to produce and manufacture the product(s). Attach a copy of the analysis (e.g., the Market Research Report required by the HSAM), if applicable.

## Scope Management

The scope of the project is defined in Section 4.2.1, Work Breakdown Structure, and the processes used to control the scope are defined in Section 4.2.2, Scope Control Plan.

### Work Breakdown Structure

Work breakdown Structures can be found within individual activity levels within each SWA project. The high level WBS for the program is depicted below:



### Scope Control Plan

The scope is managed by the SWA PM. For each project, the SWA PM and SETA continuously monitor all status reports and task execution (see section 4.5.1) of the applicable project effort to ensure performance remains in scope. Products and key activities are identified in the WBS and a baseline is established at the conclusion of the Begin Project step; the baseline for external products is established at the conclusion of the Finalize Plan step. Baselines cannot be changed without approval from the review authority. Key functions and desired performance requirements as well as technical requirements will be tracked according to the plan articulated in Section 2.1. The requirements baseline is established at Decision Review II, Approve Strategy and cannot be changed without approval from the review authority and the customer.

## 4.3 Schedule Management

The major schedule milestones are listed in Section 4.3.1, Milestones/Deliverables, and the processes used to control the schedule are defined in Section 4.3.2, Schedule Control Plan.

### Milestones/Deliverables

The major schedule milestones are listed below.

| **Major Project Milestones/Deliverables by Phase** | **Scheduled Date** |
| --- | --- |
| **Analysis Phase** | |
| IPT requirements validated by SSAD Sub IPT – Complete | 10/15/2016 |
| New start analysis | 6/1/17 |
| **Development Phase** | |
| STAMP Test Case Development | 01/2017 |
| Working prototypes for Hybrid Analysis Mapping (HAM), Application Threat Modeling (ATM), Attack Simulation and Countermeasures Modeling (ACSM), Continuous Monitoring and Assessment (CMA) | 02/2017 |
| Tool study report that provides detailed analysis of strengths and  weakness in tools, V1 | 06/2017 |
| Technical report on overlapping tool coverage and complimentary tool  Coverage | 09/2017 |
| Go/No Go Demos for Hybrid Analysis Mapping (HAM), Application Threat Modeling (ATM), Attack Simulation and Countermeasures Modeling (ACSM), Continuous Monitoring and Assessment (CMA) | 06/2018 |
| Universal Threat Model (UTM) Working Prototype | 09/2018 |
| Scoring and Benchmarking Tool Version 1 | 09/2018 |
| STAMP Operational Demonstration | 12/2019 |
| **Transition Phase** | |
| SWAMP | 10/2017 |
| UTM Operational Pilot 1 | 01/2019 |
| UTM Operational Pilot 2 | 06/2019 |

Figure 6: Major Project Milestones and Deliverables by Phase

### Schedule Control Plan

The program schedule will be reviewed monthly with the project team and the performers to ensure all understand the current status of each task and to assess impacts of any delayed tasks. If milestones are at risk based on task slippage, corrective action will be taken by the responsible party to both a) get the tasks back on track and b) assess the root cause of the slippage in order to prevent future slippage for the same reason. All activities and progress will be tracked and monitored using various program/project scheduling tools. Each project (e.g. ASTAM, STAMP, SQA and SWAMP) has a detailed project schedule associated with each performer for that work.

During the monthly schedule reviews, the master schedule will be reviewed and performer deliverables and milestones due over the next 3 months will be examined for actual progress.

## Cost Management

The funding profile for this program is highlighted in Section 1.5.

### Cost Estimate

The project cost estimates for each individual project is derived from the labor, equipment, travel, and any other directed costs associated with the effort required to make it a successful research and development effort. The estimated costs are entered into the independent government cost estimate (IGCE) and negotiated with the performers for the actual cost prior to contract award. Post award, the PM and SETA team maintain the spend plan for the overall SWA program. They also monitor, on a monthly basis, or more frequently as needed, the cost of each performer’s project.

### Cost Baseline

The cost baseline varies for each individual budget level activity and contract across the SWA program. The PM establishes each as part of the contract/IA award in conjunction with the Contracting Officer (CO). These are detailed and captured in each individual financial tracking tool per contract.

### Cost Control Plan

Cost reporting will be done on a monthly basis for each performer supporting the project. Monthly costs will be compared against the expected costs established as part of the baseline. If there is a monthly cost variance greater than 5%, further examination will be conducted by the PM team to determine the root cause(s) of the variance, any impacts to deliverables and any corrective actions required. We will also examine schedule progress as part of the cost review to maintain a comprehensive picture of the project.

## 4.5 Status Reporting

Each individual activity within the SWA program submits a Monthly Status Report (MSR) as part of their contractually required deliverables. MSRs are submitted to the PM, Contracting Officer and CSD Leadership via the Division reports mailbox: [SandT-Cyber-Reports@HQ.DHS.GOV](mailto:SandT-Cyber-Reports@HQ.DHS.GOV). Depending on the contract and nature of the research, some activities also require mid-term or quarterly technical reports. Each MSR contains the following sections:

* Status of effort for the reporting period
* Status of effort for the next reporting period
* Financial status, to include any planned and actual expenditures
* Invoices

• Cost, schedule and performance variance

Upon receipt, the PM team reviews the MSR and performs the following actions:

* Updates the SWA Program Financial Tracking Tool
* Updates SWA Program Deliverables list and identifies any missing/overdue deliverables
* Prepares a summary of discrepancies in an exceptions report for PM review
* Transmits applicable exceptions report elements to the appropriate project performer

The final report will cover all activities conducted during the period of performance and compare the delivered outcomes against the planned deliverables in the statement of work.

## Cost/Schedule/Performance Trade-Offs

# Risk Management

## 5.1 Risk Management Plan

The SWA program maintains a separate Risk Management Plan (RMP). This plan describes how generic and program-specific risks are identified, analyzed and prioritized. The RMP is a separate document which outlines procedures for tracking risks, evaluating changes in the level of individual risks (e.g. when a risk goes from High to Medium), and responses to those changes.

The risk management plan will describe how to address the following questions for each risk:

1. What might happen?
2. How likely is it that it will happen (and when, if this can be addressed)?
3. If it does happen, what are the consequences (and when, if this can be addressed)?
4. What could be done about the given risk (and when, if this can be addressed)?
5. What should be done about the given risk (and when, if this can be addressed)?
6. What is going to be done about the given risk (and when, if this can be addressed)?
7. How well is the chosen course of action (of mitigating the risk) working?
8. Has anything changed that requires altering existing risk management measures (and when, if this can be addressed)?
9. Are there current trends and/or potential future development that could require altering existing risk management measures (and when, if this can be addressed)?
10. What information about the given risk needs to be communicated (and when, if this can be addressed)?
11. Who needs to know about the given risk (and when, if this can be addressed)?
12. How can necessary risk information be most effectively communicated (and when, if this can be addressed)?

## 5.2 Risk Identification and Assessment

The risks associated with this project are described in Sections 5.2.1 – 5.2.3 and are also recorded on the risk register:

### Technical Risks

The key technical risks for the SWA program are listed at a high level below and further detailed in the risk register.

* Developing and evolving realistic test cases. Current test cases and data sets are synthetic and don’t represent real world programs. Developing next generation set of test cases with complexity and models modern software, across various problems language is needed to measure where gap exists in tools.
* ASTAM component integration. Automating and integrating the different technology areas in ASTAM to create a Unified Threat Management System is a technical risk. Each component has context that helps improve visibility into potential weaknesses that expose vulnerabilities.
* Support DevOps and continuous integration/delivery (CI/CD) for tool adoption. Many SQA tools (open-source and commercial) struggle to integrate into a (Why?) DevOps environment to achieve security at-speed. This results in software developers not using these tools early in the software development process not realizing the full benefits of using these tools as part of continuous assurance activities.
* Legacy open-source SQA tools are too mediocre to modernize. Risk is that the tools will be technically obsolete or not able to be brought up to a technically useful level.

### Programmatic Risks

#### Cost Risks

* If current or out-year budgets are unexpectedly reduced, then the program’s goals would not be fully achievable.
* Passing of a full budget for DHS at the start of the Fiscal Year rarely occurs.

#### Scope/Performance Risks

* See Risk register Section 5.3

#### Schedule Risks

* See Risk register, Section 5.3.

### Operational Risks

There are too many potential risk instances involving software since the vast majority of critical infrastructures and operational mission focused systems are powered by software.

## 5.3 Risk Register

The Risk Register for the SWA program is outlined in Figure 9 below. Note this Risk Register is maintained as an independent document and updated frequently, in accordance with the SWA Risk Management Plan.

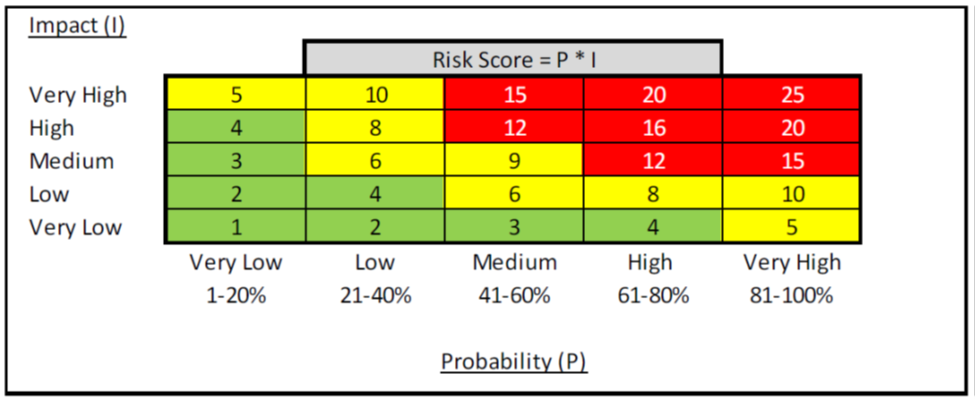
The SWA Program uses the Probability and Impact/Consequence model outlined in Figure 7 below to assign risk scores to each identified risk in the program.

**Risk Probability Scale**:

|  |  |
| --- | --- |
| **Probability Description** | **Probability Range** |
| Very High (Extremely likely) | 81% – 99% |
| High (Probable) | 61% – 80% |
| Medium (Possible) | 41% – 60% |
| Low (Unlikely) | 21% – 40% |
| Very Low (Highly improbable) | 1% – 20% |

**Consequence Probability Scale**:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Project Objective** | **Very Low 1** | **Low 2** | **Medium 3** | **High 4** | **Very High 5** |
| Cost | Insignificant increase | Increase  < 2% of  budget baseline | Increase 2–5% of  budget baseline | Increase 6–10% of  budget baseline | Increase  > 10% of budget baseline |
| Schedule | Insignificant slippage | Slippage < 2% of project baseline schedule | Slippage 2–5% of project baseline schedule | Slippage 6–10% of project baseline schedule | Slippage > 10% of project baseline schedule  — OR —  Slippage past a milestone mandated by Congress |
| Scope | Scope decrease barely noticeable | Minor areas of scope affected | Major areas of scope affected | Scope reduction unacceptable to sponsor | Project outcome is effectively useless |
| Quality | Quality degradation barely noticeable | Quality degradation noticeable, but does not fail acceptance criteria | Quality reduction requires sponsor approval | Quality reduction unacceptable to sponsor | Project outcome is effectively useless |



**Figure 7: Risk Scoring Assessment Matrix**

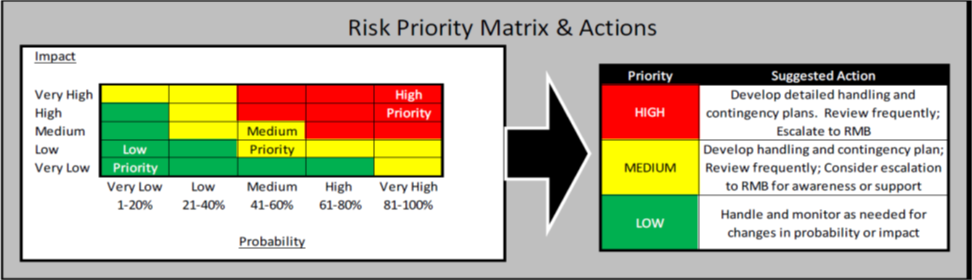


Figure 8: Risk Actions Based on Scoring

Risk Register (as of January 2017). The risk register included in the PMP is a snapshot. An up to date risk register will be maintained and updated more frequently than the PMP.

| **Risk ID** | **Risk Name** | **Risk Description** | **Risk Category** | **Risk Score** | **Response Strategy** | **Action**  **Code** |
| --- | --- | --- | --- | --- | --- | --- |
| 1 | Software Assurance Tech Transition Risk | If organizations are not willing to pilot CSD’s software assurance solutions in their operational environments or don’t have resources to dedicate, then the overall acceptance of the solutions will not be fully realized thus potentially reducing investment benefits. | Programmatic | 9 (I: Medium, P: Medium) | Engage and involve potential transition partners early, as part of proposal review, solicitation approval, review of reports and findings to ensure proper fit for technology and that proper resources are allocated appropriately and in a timely fashion. Pursue written pilot agreements early as possible. | PM |
| 2 | SSAD Sub-IPT Participation | If SSAD Sub-IPT participation is limited or requirements are not good, then transition within HSE would be limited, technology usage would not be forward-leaning or innovative. | Programmatic | 4 (I: Medium, P: Medium) | Accept. Encourage participation in SSAD sub IPT but also maintain awareness of broader SWA trends in the marketplace. | PM |
| 3 | Outsourcing of Federal Software Development | Most of the federal government outsource their software development, or procure/acquire software systems commercially through integrators. | Programmatic | 8 (I: Low , P: High | Most of software is comprised of open-source software. Engage open-source communities, and developers, work with performers to engage and evangelize in those communities of interest. | PM |
| 4 | Inability to grow SWAMP community of users | Attracting users to the SWAMP with a growing concern for privacy as it relates to private/public issues. SWAMP leverages a collection of open-source SQA tools, which are known to be mediocre and underperform. “Wont realize true value of the SWAMP”. | Programmatic | 6 (I: Medium , P: Low | Open-source SWAMP (SWAMP in the box), implement role-based and identity controls for privacy, engage commercial tool vendors for great appeal. In addition, STAMP will modernize open-source SQA tools which will consistently grow the SWAMP community. | PM |
| 5 | Developing and evolving realistic test cases | Current test cases and data sets are synthetic and don’t represent real world programs. Developing next generation set of test cases with complexity and models modern software, across various problems language is needed to measure where gap exists in tools (as we move toward certifying software through a CyberUL). | Technical | 8 (I: High , P: Low | The SWA program has two efforts in STAMP to develop test-cases to improve feasibility and realism. Leverage the work from other federal funded programs that provide a way forward. The PM has requested that performers engage NIST/NSA and other research community for input. | PM |
| 6 | ASTAM component integration | Automating and integrating the different technology areas in ASTAM to create a Unified Threat Management System. Each component has context that helps improve visibility into potential weaknesses that expose vulnerabilities. | Technical | 12 (I: High, P: Medium) | Develop and standardize system engineering/architecture, improve collaboration through information sharing, develop common framework for information sharing across technology areas, and encourage collaboration in developing of test plans and designs to ensure interoperability. | PM |
| 7 | Support DevOps and continuous integration/deliver (CI/CD) for tool adoption | Many SQA tools (open-source and commercial) struggle to integrate into a DevOps environment to achieve security at-speed. This results in software developers not using these tools early in the software development process not realizing the full benefits of using these tools as part of continuous assurance activities. | Technical | 9 (I: Medium, P: Medium) | Modernizing and creating better performing tools increases the adoption rate of these tools early in the software development process. Developing capabilities to integrate/interop with CI/CD environments and technologies. Performers have started engaging software development communities to get feedback and better understand their pain points to improve the likelihood of tech transition. (This would be part of an updates to the risk register) | PM |
| 8 | Legacy open-source SQA tools are of such low fidelity and quality that they will be too costly to modernize | Many of the open-source SQA tools have been neglected and may be too outdated to modernize. The currently community is not active and could potentially impact support after tech transition. | Technical | 12 (I: Medium, P: High) | Develop a candidate list of tool, across various programming languages, conduct market research to determine most popular used open-source SQA tools, engage open-source community to determine the level of support for code-base to maintain and sustain tools. In addition, conduct a tool study on candidate tools pre/post test case generation for gap/improvement analysis. | PM |
| 9 | Budget Reduction | If current or out-year budgets are unexpectedly reduced, then the program’s goals would not be fully achievable. | Cost | 10 (I: Low, P: Very High) | Work with stakeholders to reassess priorities and re asses/re-baseline projects as needed |  |
| 10 | Continuing resolutions (CR) | Passing of a full budget for DHS at the start of the Fiscal Year does not usually happen | Cost | 10 (I: Low, P: Very High) | Identify and prioritize SWA activities that need funding. Work with CSD Finance team to ensure SWA projects are funded in the appropriate CR increment. CSD is well versed in executing CR funding “what if” scenarios. | PM |
| 11 | Delays in pilot completion | If customer pilots are delayed or prolonged due to resource constraints, then overall project schedule and product strategy could be impacted. | Schedule | 9 (I: Medium, P: Medium) | The PM will work with potential transition partners and performers to align resources to avoid slippage in schedule. Feedback and exchange between performer and transition partners is critical to identify enhancements and functionality gaps. The PM has modified pilot agreements to stipulate specific milestones, success factors, defined duration for pilot (ranging from 35-60 days), and a process to request additional time. This allows for proper alignment with project goals. | PM |

**Figure 9: SWA Risk Register**

1. Over, James W. “Team Software Process for Secure Systems Development” https://resources.sei.cmu.edu/asset\_files/Presentation/2002\_017\_001\_24393.pdf [↑](#footnote-ref-2)
2. Woody, Carol., Ellison, Robert., & Nichols, William. (2014). Predicting Software Assurance Using Quality and Reliability Measures (CMU/SEI-2014-TN-026). Retrieved February 20, 2017, from the Software Engineering Institute, Carnegie Mellon University website: http://resources.sei.cmu.edu/library/asset-view.cfm?AssetID=428589 [↑](#footnote-ref-3)
3. National Science and Technology Council, Federal Cybersecurity Research and Development Strategic Plan, February 2016 [↑](#footnote-ref-4)