Table 1: The Secure Software Development Framework (SSDF) Version 1.1

Practices	Tasks	Notional Implementation Examples	References
Prepare the Organization (PO)			
Define Security Requirements for Software Development (PO.1): Ensure that security requirements for software development are known at all times so that they can be taken into account throughout the SDLC and duplication of effort can be minimized because the requirements information can be collected once and shared. This includes requirements from internal sources (e.g., the organization's policies, business objectives, and risk management strategy) and external sources (e.g., applicable laws and regulations).	**PO.1.1**: Identify and document all security requirements for the organization's software development infrastructures and processes, and maintain the requirements over time.	**Example 1**: Define policies for securing software development infrastructures and their components, including development endpoints, throughout the SDLC and maintaining that security. **Example 2**: Define policies for securing software development processes throughout the SDLC and maintaining that security, including for open-source and other third-party software components utilized by software being developed. **Example 3**: Review and update security requirements at least annually, or sooner if there are new requirements from internal or external sources, or a major security incident targeting software development infrastructure has occurred. **Example 4**: Educate affected individuals on impending changes to requirements.	**BSAFSS**: SM.3, DE.1, IA.1, IA.2 **BSIMM**: CP1.1, CP1.3, SR1.1, SR2.2, SE1.2, SE2.6 **EO14028**: 4e(ix) **IEC62443**: SM-7, SM-9 **NISTCSF**: ID.GV-3 **OWASPASVS**: 1.1.1 **OWASPMASVS**: 1.10 **OWASPSAMM**: PC1-A, PC1-B, PC2-A **PCISSLC**: 2.1, 2.2 **SCFPSSD**: Planning the Implementation and Deployment of Secure Development Practices **SP80053**: SA-1, SA-8, SA-15, SR-3 **SP800160**: 3.1.2, 3.2.1, 3.2.2, 3.3.1, 3.4.2, 3.4.3 **SP800161**: SA-1, SA-8, SA-15, SR-3 **SP800181**: T0414; K0003, K0039, K0044, K0157, K0168, K0177, K0211, K0260, K0261, K0262, K0524; S0010, S0357, S0368; A0033, A0123, A0151
	PO.1.2: Identify and document all security requirements for organization-developed software to meet, and maintain the requirements over time.	**Example 1**: Define policies that specify risk-based software architecture and design requirements, such as making code modular to facilitate code reuse and updates; isolating security components from other components during execution; avoiding undocumented commands and settings; and providing features that will aid software acquirers with the secure deployment, operation, and maintenance of the software. **Example 2**: Define policies that specify the security requirements for the organization's software, and verify compliance at key points in the SDLC (e.g., classes of software flaws verified by gates, responses to vulnerabilities discovered in released software). **Example 3**: Analyze the risk of applicable technology stacks (e.g., languages, environments, deployment models), and recommend or require the use of stacks that will reduce risk compared to others. **Example 4**: Define policies that specify what needs to be archived for each software release (e.g., code, package files, third-party libraries, documentation, data inventory) and how long it needs to be retained based on the SDLC model, software end-of-life, and other factors. **Example 5**: Ensure that policies cover the entire software life cycle, including notifying users of the impending end of software support and the date of software end-of-life. **Example 6**: Review all security requirements at least annually, or sooner if there are new requirements from internal or external sources, a major vulnerability is discovered in released software, or a major security incident targeting organization-developed software has occurred. **Example 7**: Establish and follow processes for handling requirement exception requests, including periodic reviews of all approved exceptions.	**BSAFSS**: SC.1-1, SC.2, PD.1-1, PD.1-2, PD.1-3, PD.2-2, SI, PA, CS, AA, LO, EE **BSIMM**: SM1.1, SM1.4, SM2.2, CP1.1, CP1.2, CP1.3, CP2.1, CP2.3, AM1.2, SFD1.1, SFD2.1, SFD3.2, SR1.1, SR1.3, SR2.2, SR3.3, SR3.4 **EO14028**: 4e(ix) **IEC62443**: SR-3, SR-4, SR-5, SD-4 **ISO27034**: 7.3.2 **MSSDL**: 2, 5 **NISTCSF**: ID.GV-3 **OWASPMASVS**: 1.12 **OWASPSAMM**: PC1-A, PC1-B, PC2-A, PC3-A, SR1-A, SR1-B, SR2-B, SA1-B, IR1-A **PCISSLC**: 2.1, 2.2, 2.3, 3.3 **SCFPSSD**: Establish Coding Standards and Conventions **SP80053**: SA-8, SA-8(3), SA-15, SR-3 **SP800160**: 3.1.2, 3.2.1, 3.3.1 **SP800161**: SA-8, SA-15, SR-3 **SP800181**: T0414; K0003, K0039, K0044, K0157, K0168, K0177, K0211, K0260, K0261, K0262, K0524; S0010, S0357, S0368; A0033, A0123, A0151
	PO.1.3: Communicate requirements to all third parties who will provide commercial software components to the organization for reuse by the organization's own software. [Formerly PW.3.1]	**Example 1**: Define a core set of security requirements for software components, and include it in acquisition documents, software contracts, and other agreements with third parties. **Example 2**: Define security-related criteria for selecting software; the criteria can include the third party's vulnerability disclosure program and product security incident response capabilities or the third party's adherence to organization-defined practices. **Example 3**: Require third parties to attest that their software complies with the organization's security requirements.	**BSAFSS**: SM.1, SM.2, SM.2-1, SM.2-4 **BSIMM**: CP2.4, CP3.2, SR2.5, SR3.2 **EO14028**: 4e(vi), 4e(ix) **IDASOAR**: 19, 21 **IEC62443**: SM-9, SM-10 **MSSDL**: 7 **NISTCSF**: ID.SC-3 **OWASPSAMM**: SR3-A

Practices	Tasks	Notional Implementation Examples	References
		Example 4: Require third parties to provide provenance[5] data and integrity verification mechanisms for all components of their software. **Example 5**: Establish and follow processes to address risk when there are security requirements that third-party software components to be acquired do not meet; this should include periodic reviews of all approved exceptions to requirements.	**SCAGILE**: Tasks Requiring the Help of Security Experts 8 **SCFPSSD**: Manage Security Risk Inherent in the Use of Third-Party Components **SCSIC**: Vendor Sourcing Integrity Controls **SP80053**: SA-4, SA-9, SA-10, SA-10(1), SA-15, SR-3, SR-4, SR-5 **SP800160**: 3.1.1, 3.1.2 **SP800161**: SA-4, SA-9, SA-9(1), SA-9(3), SA-10, SA-10(1), SA-15, SR-3, SR-4, SR-5 **SP800181**: T0203, T0415; K0039; S0374; A0056, A0161
Implement Roles and Responsibilities (PO.2): Ensure that everyone inside and outside of the organization involved in the SDLC is prepared to perform their SDLC-related roles and responsibilities throughout the SDLC.	**PO.2.1**: Create new roles and alter responsibilities for existing roles as needed to encompass all parts of the SDLC. Periodically review and maintain the defined roles and responsibilities, updating them as needed.	**Example 1**: Define SDLC-related roles and responsibilities for all members of the software development team. **Example 2**: Integrate the security roles into the software development team. **Example 3**: Define roles and responsibilities for cybersecurity staff, security champions, project managers and leads, senior management, software developers, software testers, software assurance leads and staff, product owners, operations and platform engineers, and others involved in the SDLC. **Example 4**: Conduct an annual review of all roles and responsibilities. **Example 5**: Educate affected individuals on impending changes to roles and responsibilities, and confirm that the individuals understand the changes and agree to follow them. **Example 6**: Implement and use tools and processes to promote communication and engagement among individuals with SDLC-related roles and responsibilities, such as creating messaging channels for team discussions. **Example 7**: Designate a group of individuals or a team as the code owner for each project.	**BSAFSS**: PD.2-1, PD.2-2 **BSIMM**: SM1.1, SM2.3, SM2.7, CR1.7 **EO14028**: 4e(ix) **IEC62443**: SM-2, SM-13 **NISTCSF**: ID.AM-6, ID.GV-2 **PCISSLC**: 1.2 **SCSIC**: Vendor Software Development Integrity Controls **SP80053**: SA-3 **SP800160**: 3.2.1, 3.2.4, 3.3.1 **SP800161**: SA-3 **SP800181**: K0233
	PO.2.2: Provide role-based training for all personnel with responsibilities that contribute to secure development. Periodically review personnel proficiency and role-based training, and update the training as needed.	**Example 1**: Document the desired outcomes of training for each role. **Example 2**: Define the type of training or curriculum required to achieve the desired outcome for each role. **Example 3**: Create a training plan for each role. **Example 4**: Acquire or create training for each role; acquired training may need to be customized for the organization. **Example 5**: Measure outcome performance to identify areas where changes to training may be beneficial.	**BSAFSS**: PD.2-2 **BSIMM**: T1.1, T1.7, T1.8, T2.5, T2.8, T2.9, T3.1, T3.2, T3.4 **EO14028**: 4e(ix) **IEC62443**: SM-4 **MSSDL**: 1 **NISTCSF**: PR.AT **OWASPSAMM**: EG1-A, EG2-A **PCISSLC**: 1.3 **SCAGILE**: Operational Security Tasks 14, 15; Tasks Requiring the Help of Security Experts 1 **SCFPSSD**: Planning the Implementation and Deployment of Secure Development Practices **SCSIC**: Vendor Software Development Integrity Controls **SP80053**: SA-8 **SP800160**: 3.2.4, 3.2.6 **SP800161**: SA-8 **SP800181**: OV-TEA-001, OV-TEA-002; T0030, T0073, T0320; K0204, K0208, K0220, K0226, K0243, K0245, K0252; S0100, S0101; A0004, A0057

[5] *Provenance* is "the chronology of the origin, development, ownership, location, and changes to a system or system component and associated data. It may also include personnel and processes used to interact with or make modifications to the system, component, or associated data" [SP80053].