Activity

The activities below will help prepare you for the end-of-path quiz. Complete the questions for each milestone as you make your way along your learning journey.

Using secure coding practices may be more important than you think. Start your journey on the learning path, and answer the questions below.

1. (resource 1.a) How many types of pacemakers were hacked by a global research team, finding exploits that could allow wireless remote attackers to kill victims?

2. (resource 1.b) Which application-level package (or dependency) managers do you use? (Aka language-specific package managers) Examples of application-level package managers include:
    a. Apache Maven
    b. npm
    c. NuGet
    d. RubyGems
    e. PyPI

3. (resource 1.b) What is a universal package manager? Does your development organization use that today?

4. (resource 1.c) According to the article, what are three main benefits of shifting left?

5. (resource 1.d) What is a software bill of materials (SBOM)?

6. (resource 1.e) Where are your “parts” (OSS components) coming from?

7. As you navigate through Milestone 1 and reflect on your current coding practices, have you identified potential areas for improving security throughout your Software Development Life Cycle (SDLC)? List some of those areas for further review here:

How do you get started shifting left? Or how do you continue improving your secure coding practices? Continue on the learning path below.

1. (resource 2.a) When we think of Shifting Left in DevOps, what are some of the benefits?

2. (resource 2.b) In the eLearning course, you'll learn the following:

    - Define DevSecOps, an extension of DevOps, which includes "pushing quality closer to the source" and "shifting security left" in the SDLC.
    - Explain why DevSecOps is relevant for all modern software development organizations.
    - List three ways legacy security processes can produce unintended outcomes for software developers and application security teams.
    - Articulate three compelling reasons that adopting a DevSecOps process makes good business sense for any organization developing software.

3. (resource 2.c) Secure dev practices are _____ times more likely to proactively remove troublesome dependencies.

4. (resource 2.c) Exemplary projects (aka those that employ secure coding practices) are _____ times faster at updating dependencies.

5. (resource 2.c) For development teams that actively manage their software supply chains, the use of known vulnerable component releases were reduced by __ _____%.

6. (resource 2.d) As you reflect on the 4 tips for changing team culture related to Shifting Left:

   - Which do you feel might be easiest for your team to implement?

   - Which steps might be most difficult?

   - Why?

   - What are some ways to overcome those issues?

   - How will you celebrate success with your team?

In this final trail of your learning path, consider the insecure coding practices that may occur in your current processes. Reflect on where you're at currently, and how you can shift left.

1. (resource 3.a) Review the SSC Maturity Infographic. For each of the categories listed (Inventory, Suppliers, Consumption, etc), where does your team fit on the Zero to Hero scale?

2. (resource 3.a) As you consider the infographic, what are some proactive steps your team can take to help move toward the Hero side of the scale for each category?

3. (resource 3.b) Learn more about the OSS Index, and how it enables you to search millions of components to find any known, publicly disclosed vulnerabilities across a wide range of ecosystems.
    - In the Search for Components tab, key in *struts-core*, then click **Search**.
    - Click the component **pkg:maven/org.apache.struts/struts-core**.

    How many Critical Vulnerabilities were found?

    How many Severe Vulnerabilities were found?

What would you do next, regarding these issues?

4. (resource 3.b) Scan your projects for open source vulnerabilities, and build security into your development toolchain with native tools and integrations.
    - Click **Search your dependencies** tab.

   From the list provided, which scan tools do you use?

5. (resource 3.b) Identifying risk during development is critical, but what about the applications you've already built? Run the Nexus Vulnerability scanner on any existing or legacy applications to generate a complete Software Bill of Materials (SBOM)
    - Click **Try the Nexus Vulnerability Scanner for free.**
    - Scan the sample application provided, or your own application, by performing one of the following:
        i. Download the scanner https://www.sonatype.com/appscan
        ii. Scan your application online via file upload. https://www.sonatype.com/appscan-upload

6. Learn more about Vulnerability Scanner by reviewing the guide.
    - Read the Guide: http://cdn2.hubspot.net/hubfs/1958393/eBooks/AHC_Guide.pdf (For further details, review the FAQs: https://support.sonatype.com/hc/en-us/articles/213463928-Nexus-Vulnerability-Scanner-FAQ)

7. (resource 3.b) What does the scanner tell you about the scanned application?

8. (resource 3.c) Check out all of the free offerings available to support you at my.sonatype.com.
    - Collaborate with a community of users, and Sonatype experts.
    - Sign up to receive alerts each time Sonatype's Customer Education team publishes new content.