Secure Software Development Lifecycle Practices Professional Certification Program

Submission Guidelines:

- Submit a detailed report (**maximum** 3-**5 pages**) of the approach followed/preferred by you in solving the case study assigned to you.
- In the report, please justify your approach and provide relevant diagrams, code snippets, etc., wherever applicable.
- Give proper headings/sub-headings in your report.
- Submit the case study via email to **ssdlcp@cdac.in** in a ZIP file named as <<YourCaseStudyCode_YOURNAME_DATE>>

Note: Do not use the screenshots taken during the training program.

Evaluation criteria depend on:

- Justification given for choosing a particular technique/tool/library
- Application of security controls and best practices.
- Structured responses, proper formatting, and clarity in explanations.

Secure Software Development Lifecycle Practices Professional Certification Program

Case Study SSDLCPCS10: Containerizing the CMS Application

Scenario

A development team have to containerize a CMS application composed of a Node.js server, a Python server, and a PostgreSQL database. The focus is to implement Docker Secrets to securely manage sensitive data, such as database credentials during the container runtime. The application is to be deployed using Docker Compose, ensuring service dependencies and health checks are properly configured.

Security Challenges

- While containerizing the services ensure authentication credentials or other sensitive information or config parameters are not exposed to the external world.
- Managing service dependencies so that containers relying on the database only start when the database container is healthy and ready.
- Building secure and minimal Docker images for the Node.js and Python servers.

Expected Outcome:

Describe the techniques, tools and code snippets that you would like to follow in addressing the above challenges in the following SDLC Phases.

(a) Deployment
(b) Testing

Submit a detailed report on the above problem statement by comparing available alternative options and justifying your approach. Explain the reason behind how the techniques described in your approach prevent the attack scenario or address above mentioned security challenges.