

devon



# Certified DevSecOps Practitioner

Brochure

Version 1.2



## REVISION HISTORY

VERSION	DATE	REMARKS
1.0	19-Sep-20	Initial Version
1.2	2-Nov-20	Revised version (too much content)



## Contents

REVISION HISTORY.....	1
INTRODUCTION TO SYLLABUS.....	3
OVERVIEW.....	3
EXAMINABLE LEARNING OBJECTIVES .....	3
EXAMINATION DETAILS.....	4
MODULE 1: DEVSECOPS PRINCIPLES.....	5
MODULE 2: EXPLORING SECURITY CHECKPOINTS.....	5
MODULE 3: EMBEDDING SECURITY IN SOFTWARE DELIVERY .....	5
MODULE 4: EMBEDDING SECURITY IN SOFTWARE OPERATIONS .....	5
MODULE 5: COMPLIANCE AND GOVERNANCE IN DEVSECOPS.....	5
APPENDIX 1 - K- LEVELS EXPLANATION.....	6
ACKNOWLEDGEMENT .....	7



## INTRODUCTION TO SYLLABUS

### OVERVIEW

DevSecOps, Shift left, Security as Code are some of the interlacing paradigm shifts aiming to deliver secure software early and often. The DevSecOps practitioner course is carefully designed to help you understand the principles, practices, tools, and techniques that are essential for any organization to get started with and mature organically in DevSecOps. This 16 hour course is designed to be highly interactive, and hands on by DevSecOps veterans that bring their lessons working from the trenches.

### EXAMINABLE LEARNING OBJECTIVES

The Learning Objectives support the Business Outcomes and are used to create the examination for achieving the Certified DevOps Professional. In general, all parts of this syllabus are examinable at a K1 and K2 level. That is, the candidate will recognize, remember, and recall a term or concept. The specific learning objectives at K1, K2, and K3 levels are shown at the beginning of the pertinent chapter. The high level learning objectives of the DevSecOps Practitioner include

- Embedding Security in the pipelines by introducing SCA, SAST, DAST, Compliance as a Code, and various other tooling's and techniques that helps strengthen your organization's security posture
- Abilities to apply Practical Risk Management, Vulnerability Management concepts to real world security weaknesses in applications and infrastructure
- Explore the various DevSecOps reference models so to understand that DevSecOps is not a 'one type fits all' approach, and how to navigate through challenging implementations
- Understand the evolution behind traditional security practices, tooling, principles, and maturity models and how they compare to the practices, tooling, principles, and maturity models of DevSecOps
- Ability to envision and implement a DevSecOps program at scale and at enterprise by balancing the security and velocity
- Ability to apply a shift left approach in 'security' as code, build, and infrastructure
- Learn the essential skills to apply sensible security automation practices early on and in the various stages of the Software Development Lifecycle



## EXAMINATION DETAILS

Examination type	Multiple-choice questions
Number of questions	40
Pass mark	75%
Open book/notes	No
Electronic equipment/aides permitted	No
Time allotted for examination	90 Minutes



## MODULE 1: DEVSECOPS PRINCIPLES

- 1.1. (K1) Understanding security critical business factors
- 1.2. (K1) Separating security myths from reality
- 1.3. (K1) Understanding evolutionary DevSecOps
- 1.4. (K2) Exploring current security challenges
- 1.5. (K1) Exploring DevSecOps principles and movements

## MODULE 2: EXPLORING SECURITY CHECKPOINTS

- 2.1. (K1) Understanding the Adversary's perspective
- 2.2. (K2) Applied Risk Management and fast paced software delivery
- 2.3. (K1) Exploring SDL and security maturity models
- 2.4. (K1) Exploring public reference architectures
- 2.5. (K2) Understanding the DevSecOps checkpoints

## MODULE 3: EMBEDDING SECURITY IN SOFTWARE DELIVERY

- 3.1. (K2) Exploring the bleeding edge DevSecOps Toolset
- 3.2. (K2) Understand the Open Source Intelligence and Dependency Analysis
- 3.3. (K3) Integrating Static Application Security tools into CI/CD
- 3.4. (K3) Integrating Dynamic Application Security tools into CI/CD

## MODULE 4: EMBEDDING SECURITY IN SOFTWARE OPERATIONS

- 4.1. (K1) Exploring the lurking dangers of supply chain vulnerabilities
- 4.2. (K2) Securing the infrastructure as code baselines
- 4.3. (K2) Exploring the state of the art Identity and Access Management
- 4.4. (K3) Building perimeter security and network defences
- 4.5. (K3) Understanding and Implementing container security

## MODULE 5: COMPLIANCE AND GOVERNANCE IN DEVSECOPS

- 5.1. Understanding Compliance and how it compliments DevSecOps
- 5.2. Exploring Security Tooling to achieve compliance
- 5.3. Achieving Compliance as Code
- 5.4. Building a DevSecOps Program



## APPENDIX 1 - K- LEVELS EXPLANATION

Source: "Writing Learning Objectives" prepared by Raoul A. Arreola, Ph.D., The University of Tennessee, Memphis.

### What is a Learning Objective?

A learning objective is a statement of what students will be able to do when they have completed instruction. A learning objective has three major components:

1. A description of what the student will be able to do
2. The conditions under which the student will perform the task
3. The criteria for evaluating student performance

### Cognitive Learning

*K1. Basic Knowledge:*

**To recall and memorize** - Assess by direct questions. The object is to test the students' ability to recall facts, to identify and repeat the information provided.

*Recall, identify, recognize, acquire, distinguish*

*K2. Comprehension:*

**To translate from one form to another** - Assess by having students'

- restate material in their own words,
- reorder or extrapolate ideas, predict or estimate.

Assessments must provide evidence that the students have some understanding or comprehension of what they are saying.

*Translate, extrapolate, convert, interpret, abstract, transform*

*K3. Application:*

**To apply or use information in a new situation** - Assess by presenting students with a unique situation (i.e., one not identical to that used during instruction) and have them apply their knowledge to solve the problem or execute the proper procedure.

*Apply, sequence, carry out, solve, prepare, operate, generalize, plan, repair, explain, recognize*

*K4. Analysis:*

**To examine a concept and break it down into its parts** - Assess by presenting students with a unique situation of the same type but not identical to that used during instruction, and have them analyze the situation and describe the appropriate procedure or solution to the problem.

*Analyze, estimate, compare, observe, detect, classify, discover, discriminate, identify, explore, distinguish, catalog, investigate, breakdown, order, determine*



*K5. Synthesis:*

**To put information together in a unique or novel way to solve a problem** – Assess by presenting students with a unique situation NOT of the same type used during instruction, and have them solve a problem by selecting and using appropriate information.

*Write, plan, integrate, formulate, propose, specify, produce, organize, theorize, design, build*

*K6. Evaluation:*

**To make quantitative or qualitative judgments using standards of the appraisal** - Assess by presenting the students with a situation which includes both a problem and a solution to the problem and have them justify or critique the solution.

*Evaluate, verify, assess, test, judge, rank, measure, appraise, select, check*

## ACKNOWLEDGEMENT

A diverse team created this syllabus. We want to thank the following people (in no particular order) for their contributions in writing and reviewing this document:

*DEVON*

Marudhamaran Gunasekaran, and Mohammed Siyam, both Chief Consultants from DevOn

*ISQI*

Erika Paasche, Corinna Flemming – Vogt, and Tuyen Tran



At DevOn, we believe that organizations need to radically improve their software development to stay ahead of the competition. We do not make step-by-step improvements, on the contrary, we implement innovative solutions that radically change software development for the better. We help organizations solve their impediments and move beyond traditional ways of work.

Moreover, with over fifteen years of experience with Distributed Agile Software Development, DevOn is a trusted partner of various organizations. Since 2004 we have been combining best practices in the field of Lean and Agile Software Development with our experience from Agile transformations. Our Agile Software Development Lab in India consists of self managing teams that deliver new functionalities every sprint, which are ready for production.

This document contains information that may be privileged or confidential and is the property of DevOn

