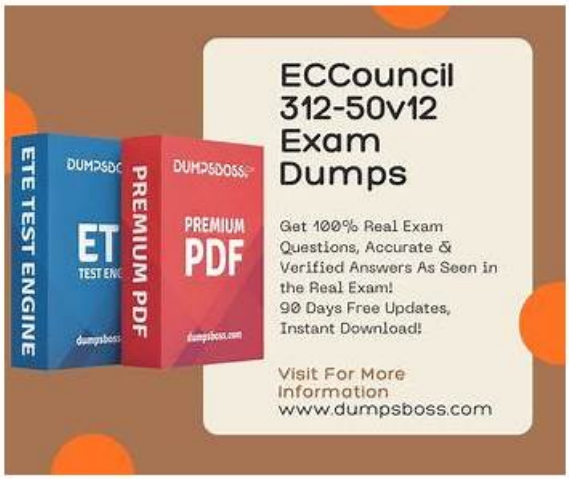
Exam ECCouncil 312-97 Dumps & 312-97 Minimum Pass Score



Our product backend port system is powerful, so it can be implemented even when a lot of people browse our website can still let users quickly choose the most suitable for his 312-97 learning materials, and quickly completed payment. It can be that the process is not delayed, so users can start their happy choice journey in time. Once the user finds the learning material that best suits them, only one click to add the 312-97 learning material to their shopping cart, and then go to the payment page to complete the payment, our staff will quickly process user orders online. In general, users can only wait about 5-10 minutes to receive our 312-97 learning material, and if there are any problems with the reception, users may contact our staff at any time. To sum up, our delivery efficiency is extremely high and time is precious, so once you receive our email, start your new learning journey.

ECCouncil 312-97 Exam Syllabus Topics:

Topic	Details
Topic 1	• DevSecOps Pipeline - Code Stage: This module discusses secure coding practices and security integration within the development process and IDE. Developers learn to write secure code using static code analysis tools and industry-standard secure coding guidelines.
Topic 2	• DevSecOps Pipeline - Build and Test Stage: This module explores integrating automated security testing into build and testing processes through CI pipelines. It covers SAST and DAST approaches to identify and address vulnerabilities early in development.
Topic 3	• Understanding DevOps Culture: This module introduces DevOps principles, covering cultural and technical foundations that emphasize collaboration between development and operations teams. It addresses automation, CI • CD practices, continuous improvement, and the essential communication patterns needed for faster, reliable software delivery.
Topic 4	• DevSecOps Pipeline - Release and Deploy Stage: This module explains maintaining security during release and deployment through secure techniques and infrastructure as code security. It covers container security tools, release management, and secure configuration practices for production transitions.
Topic 5	• DevSecOps Pipeline - Plan Stage: This module covers the planning phase, emphasizing security requirement identification and threat modeling. It highlights cross-functional collaboration between development, security, and operations teams to ensure alignment with security goals.

>> Exam ECCouncil 312-97 Dumps <<

100% Pass Quiz 2026 ECCouncil 312-97: Fantastic Exam EC-Council Certified DevSecOps Engineer (ECDE) Dumps

Taking practice tests is particularly helpful for those who have exam anxiety. Our practice tests are user-friendly and customizable. Windows support the desktop practice test software. Our web-based practice test is compatible with all browsers and operating systems. The web-based EC-Council Certified DevSecOps Engineer (ECDE) (312-97) practice test is similar to the desktop-based exam and can be taken on any browser without needing to download separate software.

ECCouncil EC-Council Certified DevSecOps Engineer (ECDE) Sample Questions (Q26-Q31):

NEW QUESTION # 26
(Gabriel Bateman has been working as a DevSecOps engineer in an IT company that develops virtual classroom software for online teaching. He would like to clone the BDD security framework on his local machine using the following URL,https://github.com/continuumsecurity/bdd-security.git. Which of the following command should Gabriel use to clone the BBD security framework?)

- A. git clonehttps://github.com/continuumsecurity/bdd-security.git.
- B. github clonehttps://github.com/continuumsecurity/bdd-security.git.
- C. github clonehttps://github.com/continumsecurity/bdd-security.git.
- D. git clonehttps://github.com/continumsecurity/bdd-security.git.

Answer: A

Explanation:
To clone a repository from GitHub, the correct command is git clone followed by the accurate repository URL. The organization name continuumsecurity and repository name bdd-security must be spelled correctly for the command to succeed. Options using github clone are invalid because github is not a standard Git command-line utility. Options with misspelled organization names will result in errors. Cloning security testing frameworks during the Code stage enables DevSecOps engineers to evaluate, customize, and integrate security automation tools into development workflows, supporting secure application development and testing practices.

NEW QUESTION # 27
(Joyce Vincent has been working as a senior DevSecOps engineer at MazeSoft Solution Pvt. Ltd. She would like to integrate Trend Micro Cloud One RASP tool with Microsoft Azure to secure container-based application by inspecting the traffic, detecting vulnerabilities, and preventing threats. In Microsoft Azure PowerShell, Joyce created the Azure container instance in a resource group (ACI) (named "aci-test-closh") and loaded the container image to it. She then reviewed the deployment of the container instance. Which of the following commands should Joyce use to get the logging information from the container?.)

- A. az container logs -resource-group ACI -name aci-test-closh.
- B. az container logs --resource-group ACI --name aci-test-closh.
- C. azure container logs -resource-group ACI -name aci-test-closh.
- D. azure container logs --resource-group ACI --name aci-test-closh.

Answer: B

Explanation:
Azure Container Instances (ACI) exposes container logs via the Azure CLI using the az container logs command. To retrieve logs, you must provide the resource group and the container group name using the long- form parameters --resource-group and --name. Option A matches the correct CLI structure and parameter format: az container logs --resource-group ACI --name aci-test-closh. Options B and D incorrectly use single- dash forms (-resource-group and -name), which are not valid for these long option names. Options C and D incorrectly use azure instead of az; the Azure CLI command group is invoked with az, not azure. Getting logs after deployment review is a critical Operate and Monitor activity: it helps confirm the container started correctly, diagnose runtime errors, and validate that runtime protection (such as a RASP/micro-agent) is functioning. This visibility supports faster incident response and

helps ensure the containerized workload remains secure and stable in its runtime environment.
==========

NEW QUESTION # 28
(Dustin Hoffman is a DevSecOps engineer at SantSol Pvt. Ltd. His organization develops software products and web applications related to mobile apps. Using Gauntlt, Dustin would like to facilitate testing and communication between teams and create actionable tests that can be hooked in testing and deployment process. Which of the following commands should Dustin use to install Gauntlt?.)

- A. $ gems install Gauntlt.
- B. $ gem install gauntlt.
- C. $ gems install gauntlt.
- D. $ gem install Gauntlt.

Answer: B

Explanation:
Gauntlt is a security testing framework written in Ruby and distributed as a Ruby gem. The correct way to install a Ruby gem is using the gem install command followed by the lowercase gem name. RubyGems are case-sensitive and standardized to lowercase naming conventions, which makes gem install gauntlt the correct command. The gems command does not exist in Ruby's package management ecosystem, and using uppercase names such as Gauntlt can lead to installation failures. Installing Gauntlt allows DevSecOps teams to write human-readable security tests and integrate them into CI/CD pipelines, enabling automated and collaborative security validation during the Build and Test stage.
==========

NEW QUESTION # 29
(Andrew Gerrard has recently joined an IT company that develops software products and applications as a DevSecOps engineer. His team leader asked him to download a jar application from the organization GitHub repository and run the BDD security framework. Andrew successfully downloaded the jar application from the repository and executed the jar application; then, he cloned the BDD security framework. Which of the following commands should Andrew use to execute the authentication feature?.)

- A. ./gradlew -Dcucumber.options="-tags @authentication -tags ~@skip".
- B. /gradlew -Dcucumber.options="-tags @authentication -tags @skip".
- C. /gradlev -Dcucumber.options="-tags @authentication -tags @skip".
- D. ./gradlev -Dcucumber.options="-tags @authentication -tags ~@skip".

Answer: A

Explanation:
The BDD Security framework is executed through Gradle wrapper commands, and the correct wrapper script on Unix-like systems is ./gradlew (dot-slash indicates "run the wrapper from the current directory"). Options using /gradlew or /gradlev imply an absolute path at filesystem root and are typically incorrect for a cloned project. Also, the wrapper name isgradlew, notgradlev. For executing only the authentication feature (or scenarios tagged for authentication), Cucumber tag expressions are used through the -Dcucumber.options system property. The command must include --tags @authentication to select authentication-tagged scenarios. To skip scenarios tagged "skip," the exclusion operator is used as --tags ~@skip (meaning "exclude @skip").
Options A and B incorrectly include --tags @skip which wouldincludeskipped tests rather than exclude them.
Therefore, ./gradlew -Dcucumber.options="--tags @authentication --tags ~@skip" is the correct choice to run authentication scenarios while excluding anything marked to skip.
==========

NEW QUESTION # 30
(Jason Wylie has been working as a DevSecOps engineer in an IT company located in Sacramento, California. He would like to use Jenkins for CI and Azure Pipelines for CD to deploy a Spring Boot app to an Azure Container Service (AKS) Kubernetes cluster. He created a namespace for deploying the Jenkins in AKS, and then deployed the Jenkins app to the Pod. Which of the following commands should Jason run to see the pods that have been spun up and running?)

- A. kubectl get pods -k Jenkins.
- B. kubectl get pods -s jenkins.

- C. kubectl get pods -p jenkins.
- D. kubectl get pods -n jenkins.

Answer: D

Explanation:
Kubernetes uses namespaces to logically isolate resources such as pods, services, and deployments. When an application like Jenkins is deployed into a specific namespace, the correct way to view the pods running in that namespace is by using the -n (or --namespace) flag with the kubectl get pods command. The command kubectl get pods -n jenkins instructs Kubernetes to list all pods in the "jenkins" namespace. The other options use invalid or unrelated flags that are not supported for namespace selection. Verifying pod status during the Release and Deploy stage is essential to ensure that applications have been deployed successfully and are running as expected before exposing services or proceeding to monitoring. This step supports deployment validation and operational readiness in Kubernetes-based DevSecOps environments.

NEW QUESTION # 31
.....