

1. Record Nr.	UNINA9910254563903321
Autore	Pathania Nikhil
Titolo	Pro Continuous Delivery [[electronic resource]] : With Jenkins 2.0 // by Nikhil Pathania
Pubbl/distr/stampa	Berkeley, CA : , : Apress : , : Imprint : Apress, , 2017
ISBN	1-4842-2913-4
Edizione	[1st ed. 2017.]
Descrizione fisica	1 online resource (288 pages) : illustrations, tables
Disciplina	005.11
Soggetti	Open source software Computer programming Computer engineering Computer software—Reusability Computers Open Source Computer Engineering Performance and Reliability Models and Principles
Lingua di pubblicazione	Inglese
Formato	Materiale a stampa
Livello bibliografico	Monografia
Nota di bibliografia	Includes bibliographical references at the end of each chapters and index.
Nota di contenuto	Chapter 1: Elements of Continuous Delivery -- Chapter 2: HA Jenkins Setup Using Pacemaker, Corosync, and DRBD -- Chapter 3: HA Jenkins Setup Using CoreOS, Docker, and GlusterFS -- Chapter 4: Setting Up Jenkins on Docker and Cloud -- Chapter 5: Pipeline as a Code -- Chapter 6: Using Containers for Distributed Builds -- Chapter 7: Pre-Tested Commits Using Jenkins -- Chapter 8: Continuous Delivery Using Jenkins Pipeline.
Sommario/riassunto	Follow this step-by-step guide for creating a continuous delivery pipeline using all of the new features in Jenkins 2.0 such as Pipeline as a Code, multi-branch pipeline, and more. You will learn three crucial elements for achieving a faster software delivery pipeline: a fungible build/test environment, manageable and reproducible pipelines, and a scalable build/test infrastructure. Pro Continuous Delivery demonstrates how to create a highly available, active/passive Jenkins

server using some niche technologies. What You'll Learn: Create a highly available, active/passive Jenkins server using CoreOS and Docker, and using Pacemaker and Corosync Use a Jenkins multi-branch pipeline to automatically perform continuous integration whenever there is a new branch in your source control system Describe your continuous delivery pipeline with Jenkinsfile Host Jenkins server on a cloud solution Run Jenkins inside a container using Docker Discover how the distributed nature of Git and the “merge before build” feature of Jenkins can be used to implement gated check-in Implement a scalable build farm using Docker and Kubernetes.
