le cnam

Software supply chain security for the cloud

Introduction

Jacopo Bufalino (CNAM)

Introduction

Focus and objective of this course

In this course we will go **through all the stages of software development** of cloud applications, from code to deployment, focusing on the security aspects of each stage.

Objective

The main objective is to understand how modern cloud-native systems are built and how to secure every step of their lifecycle.

This course is a mix of theoretical knowledge and practical exercises.

Topics covered

- Virtualization, containerization and cloud deployments (recap)
- Git, version control (recap)
- Docker and Kubernetes
- CI/CD pipelines
- The Software Supply Chain
- Security of cloud deployments
- Security of the Software Supply Chain
- Advanced topics on supply chain

Prerequisites

Basic knowledge of:

- ➜ Linux and its kernel
- ➜ Networking
- ➜ Security
- ➜ Git and version control

We will still cover the basics in this course

Learning Goals

By the end of the course you should be able to:

- ➜ Deploy and secure containerized applications in the cloud
- ➜ Create complex CI/CD pipelines with security in mind
- ➜ Understand and work with software supply chain concepts
- ➜ Understand the principles behind security scanning of code and infrastructure

Course organization

Practical info

The course consists of lectures and practical labs. We will have around one lab session per lesson. There will be a mix of tutorials and graded exercises. After that, there will be a final exam.

The labs will be available at: `https://cloudsec.jackops.dev`
Course slides and announcements will be posted in Moodle.

The deadline for each graded lab will be announced after the publishing date (typically one week).

About AI

Use of AI in this course is not recommended.

Course organization

Guest lecturer

Agathe Blaise will be teaching deployment and runtime security of cloud applications in April.

The cloud

Definitions

There are a lot of websites and applications running in the cloud. But what is ”the cloud” exactly?

- ➜ Software, infrastructure, and platforms running on-demand
- ➜ Managed instances and services
- ➜ Automatic provisioning, scaling, and replication
- ➜ Someone else’s computer

There are different types of cloud

- ➜ Public cloud: Services provided by third-party providers
- ➜ Private cloud: Self-hosted cloud infrastructure
- ➜ Hybrid cloud: Combination of public and private cloud

Why choosing the cloud?

- **Scalability:** Automatic resource scaling.
- **Cost efficiency:** Pay only for what you use.
- **Accessibility:** Access services and data from anywhere.
- **Maintenance:** Cloud providers handle maintenance.
- **Disaster recovery:** Built-in redundancy and backup options for data protection.
- **API-first:** Communication with the cloud is possible using APIs, which makes it easy to handle resources.

Startups and small companies benefit from cloud too

Maintenance and domain knowledge allow small dev teams to deploy and handle complex applications with ease.

Different computing paradigms

- **Infrastructure as a Service (IaaS):**
  - Provides virtualized computing resources over the internet.
  - Users manage operating systems, applications, and data.
- **Platform as a Service (PaaS):**
  - Provides a platform allowing customers to develop, run, and manage applications.
  - Users manage applications and data, while the provider manages the infrastructure.
- **Software as a Service (SaaS):**
  - Provides software applications over the internet on a subscription basis.
  - Users access the software via a web browser, with the provider managing the infrastructure and platform.

Examples of cloud computing paradigms

aws

netlify

Google Cloud

SAP

zoom

Microsoft Azure

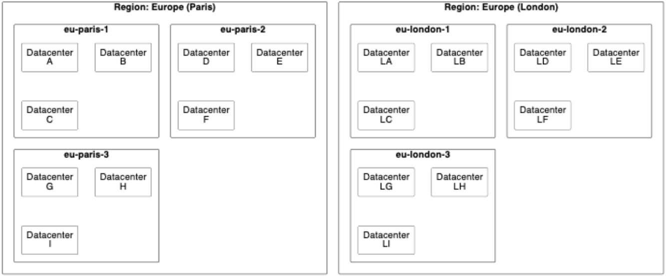
heroku

Spotify

Replication and Virtualization

Regions and Availability Zones

- **Regions:** Geographical areas that contain multiple Availability Zones. Each region is isolated to ensure data sovereignty and compliance. There are several regions in Europe.
- **Availability Zones (AZ):** Physically separate groups of data centers within a region.



Regions and Availability Zones

Importance of Regions and Availability Zones

- **High availability:** Ensures that applications remain available even if one zone fails.
- **Disaster recovery:** Provides a mechanism to recover from natural or technical disasters by replicating data across zones.
- **Low latency:** Improves performance by allowing users to connect to the nearest region.
- **Compliance:** Helps meet regulatory requirements by storing data in specific geographical locations (HIPAA, GDPR).

Typically it is enough to replicate across different AZs

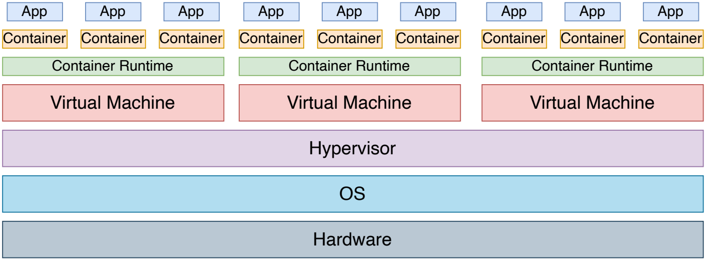
Mission-critical software can also be replicated across regions (more expensive).

Virtualization

Cloud applications and services run in virtualized environments

- ➜ Virtual Machines (VM)
  - Full OS virtualization
  - Individual kernel
- ➜ Containers:
  - Lightweight isolated processes
  - Same architecture as the host
  - Same kernel as the host
  - Easier to manage compared to VMs

Virtualization



Virtualization Stack

Cloud security - a prime

Cloud security

Security is a major concern in the cloud

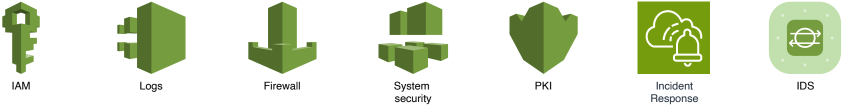
- 81% of organizations experienced a cloud-related security incident in 2023.
- 52% of organizations experienced data loss in the cloud due to misconfiguration in 2021.
- Cloud security incidents have resulted in an average financial loss of $263,000 per incident.

Cloud attack surface is large

- Many virtualization layers
  - Each layer can be targeted by attackers
- Interacting components
- Unclear responsibility

Cloud security - overview

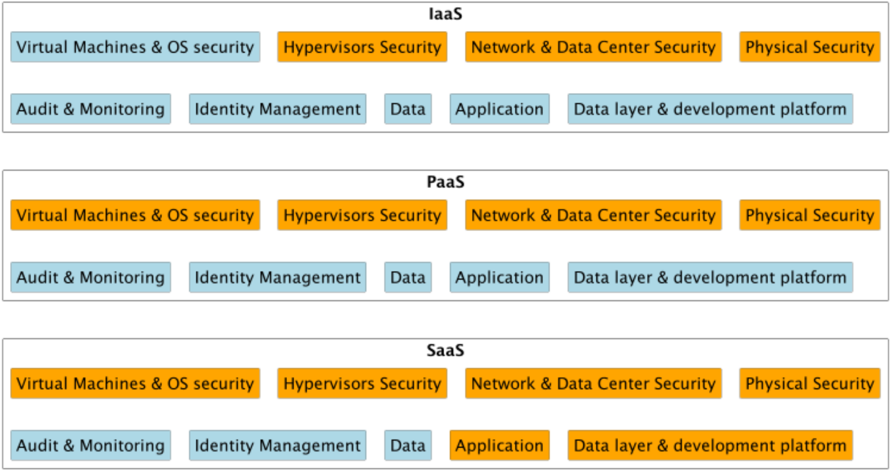
There are many types of security controls in the cloud.



Not only security, compliance is also important

Shared Responsibility Model

Cloud Provider and user share the responsibility for security.
Depends on the computing paradigms (Orange is cloud-responsibility)



Software Supply Chain

How Code is Built: Past vs. Present

Traditional Approach

- Code written and tested locally without any external dependency
- Manual deployment to servers
- Version control systems (e.g., Git) used to sync work

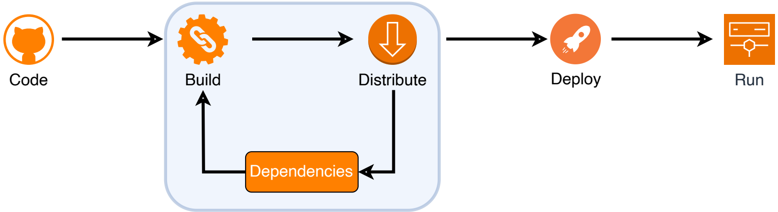
Modern Approach

- Developers write code and tests. They use many dependencies
- DevOps pepare the app to run on the cloud
- Applications are tested and deployed automatically
- Collaboration through shared platforms and cloud environments

Software Supply Chain (SSC)

Set of processes, tools and actors that contribute in transforming source code into a running artifact

➜ Each link of the chain is a code transformation

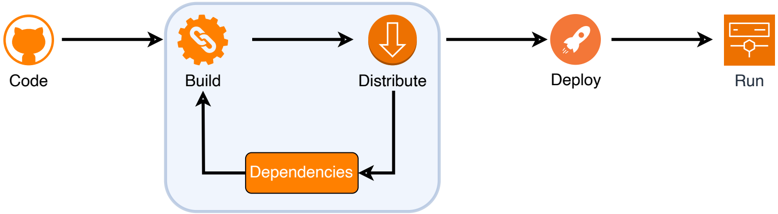


Software Supply Chain: Overview

Software Supply Chain (SSC)

How SSC and cloud are linked

- ➜ Final artifacts are usually deployed to the cloud
- ➜ Many steps of the SSC run on the cloud
- ➜ Code is also hosted on the cloud



Software Supply Chain: Overview

Supply Chain Security

One link away from disaster

The interconnection of the SSC steps, especially in cloud environments, makes is possible to generate sophisticated attacks that target the weakest link in the chain.

Supply Chain Security: well known attacks

SolarWinds attack (2020)

Attackers compromised the build system of SolarWinds, a software company. They injected malicious code during compilation so it was automatically **digitally signed and distributed as a legitimate update**. The company then unknowingly distributed backdoored software to customers.

Log4Shell (2021)

A vulnerability in the Log4j Java library allowed attackes to remote-execute commands to vulnerable servers.

Shai-Hulud (2025)

Small but highly used libraries were compromised and injected with malicious code which propagated across all of the software using that library.

Why Cloud Environments Amplify Supply Chain Risks

- **Increased Complexity:** Cloud-native applications often rely on numerous third-party services, libraries, and microservices, increasing the number of potential attack vectors.
- **Dynamic and Distributed Nature:** Cloud environments are highly dynamic, with frequent updates, scaling, and deployment cycles, making it harder to track and secure all components.
- **Automated Toolchains:** CI/CD pipelines in the cloud often automate many steps in the supply chain, which can be exploited if any part of the pipeline is compromised.
- **Lack of Visibility:** It is often difficult to gain full visibility into all dependencies and components used in cloud applications, especially when using proprietary or open-source libraries.

Working with Software Supply Chain in this course

Our Example: CICDiaries

We will use a fictional application called **CICDiaries** to illustrate the concepts and practices discussed in this course. It will provide hands-on experience with CI/CD pipelines and security practices in the software supply chain. You will receive credentials later on in the course.

Tips

I will try to give you early access to the platform and I strongly encourage you to start learning about git. We will have a theoretical class that will be followed by a practical session.

Complete the Supply chain Labs on time as the outcome of the previous lab will be used for the next ones.