WHAT'S CNAB? THE CLOUD NATIVE APPLICATION BUNDLE EXPLAINED

Cloud-native applications promise performance on the cloud by taking full advantage of cloud-native features integrated with an application programming interface (API) to your containerized applications.

However, working with the cloud is inherently complicated—and running containerized applications has its limitations: high deployment complexity and vendor lock-in due to limited API support. Despite the proliferation of open-source container orchestration solutions such as Kubernetes, the process of managing multiple containers supporting a large application in a multi-cloud environment is challenging.

In order to address these challenges, Cloud Native Application Bundle debuted in 2018 as a cloud-agnostic package format specification for multi-component containerized applications. How does a CNAB achieve that? What are the main elements of a CNAB system? Who should use CNAB?

Let's discuss these questions in detail.

The Cloud Native Application Bundle (CNAB) solution

Consider the case of distributed container-based applications that use several API integrations.

Cloud Native Application Bundle

The containers operating as executable units of the software application package the code, libraries, and dependencies in a standardized way to run on traditional data centers or in cloud environments. Additional services including virtual machines (VMs) and Function as a Service (FaaS) are also packaged, along with load balancers and other REST API connected services.

Before a package is deployed for a specific cloud environment, it must include all components necessary to enable full integration and support with the host infrastructure.

The problem arises with the limited integration support and manual configuration efforts required to run the app in dynamic cloud environments or migrating between cloud vendors. Additionally, you have a range of manual tasks associated with configuration management and container package installation, including:

- Managing and verifying package content credentials
- Managing offline and versioned installations
- Testing, signing, and verifying installer details
- Managing audit trails and regular reporting for regulatory and organizational policy compliance
- Uninstalling and deprovisioning of resources

Components of CNAB

The Cloud Native Application Bundle format provides a standardized description and guideline for packaging apps to run on multi-cloud environments, edge computing, or other IoT applications and services. CNAB contains the following *bundle* items:

- **The Bundle description** such as version, tags, keywords, and other asset descriptions. These descriptions include the schema version, top-level package information, list of credentials and schema definitions.
- **Details of the installer program**, called the 'invocation image', that helps locate and execute the referenced items. The map of images and information on invocation images is included.
- **List of parameters required by the bundle**, including standardized and user-defined configuration items. Custom actions may be referenced as required.
- **The outputs, path, and environment variables** necessary to execute the bundle.

The CNAB packages the bundle and runtime filesystems that can be processed to retrieve the referenced content from existing repositories. Cryptographic verification is adopted to ensure that the sourced repositories are trusted and referenced correctly.

The CNAB document also contains details on how the invoked file images are installed, including the system layout and schema.

Who is CNAB for?

CNAB is an infrastructure tool that can be used by developers, Ops, and even B2B users in a bundle marketplace—or customers directly could use it, too. Consider the CNAB as a unified immutable install of an app container system that can be reused for every installation process in a few simple clicks.

CNAB has many use cases, particularly in a DevOps environment:

Installation automation

A developer building an application may also write a detailed installation process guide as a readme.txt file. With the readme file available, the ops personnel handling the infrastructure provisioning process will want to automate the installation process using a command-line tool. With the CNAB bundles, users don't have to rewrite the bash script files but instead automate the installation process as part of the CI/CD pipeline.

Deeper automation

Developers can further extend the standardized automation process by transferring the snapshot of a complex but immutable setup to automation systems, use it for digital signing and verification, and continuous deployment. The next set of users including Ops personnel or customers then don't have to learn and go through the entire process of installing the app as long as they have appropriate credentials.

(Explore the role of automation in DevOps.)

It's important to understand that CNAB bundles, installation containers, and executable units are the Open Container Initiative (OCI) images by default executed using tools such as Porter. These images can be published to distribution registries including private local registries. The result is that the CNAB now contains not only a single-service component Docker image but the entire application and deployment stack.

This makes immutability and the distribution of containerized stacks convenient for all users in the DevOps teams. In many cases, the process will be a few clicks as long as all standardized specifications are adopted as part of the CNAB best practices.

Related reading

- BMC DevOps Blog
- Cloud Native Security: A Beginner's Guide
- What Is Cloud Native DevOps?
- Introduction To Docker: A Beginner's Guide
- What is Kubernetes (K8s)? A Kubernetes Basics Tutorial
- Introduction to Immutable Infrastructure

Brought to you by:

Microsoft Azure
docker
Pivotal
bitnami
HashiCorp
codefresh