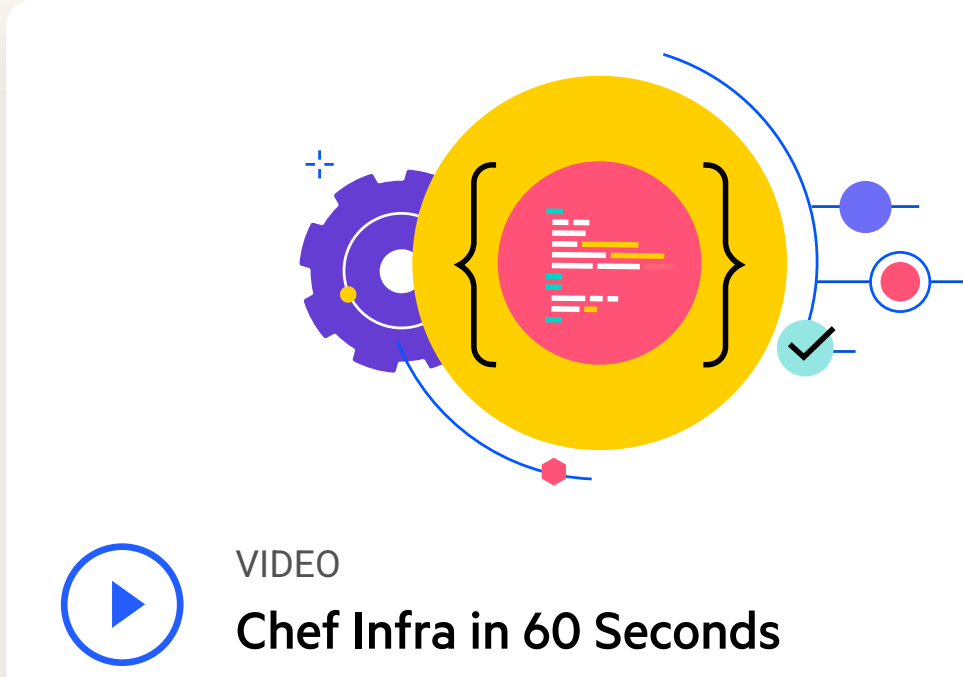


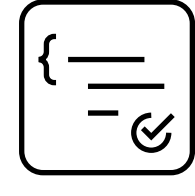
Modern Configuration Management: Policy as Code

When configuration is code, DevSecOps teams work more efficiently, enabling continuous automation across all IT processes.

FREE TRIAL

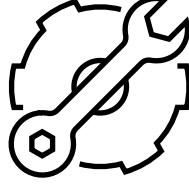


Configuration Management Automation Software



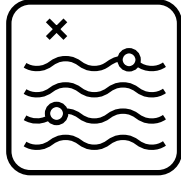
Confidence Through Code

Define configurations and policies as code that are testable, enforceable and can be delivered at scale as part of automated pipelines.



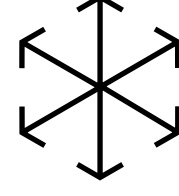
Drift Elimination

Ensure configurations only change if a system diverges from the desired defined state and automatically correct configuration drift, if needed.



Manage Disparate Systems as One

Manage Windows and Linux systems running on prem, ARM systems running in the cloud or Macs laptops running at the edge all the same way.



Extreme Flexibility

Use simple declarative definitions for common tasks or easily extend them to support the most unique environmental requirements.

Turn Configuration Management into a Competitive Advantage

Managing multiple virtual machines requires proper configuration—loading them with the right software and making sure that software can run. But how can you manage infrastructure when the number of machines you’re responsible for changes on a daily basis?

The only way to instead of disruptive churn is to implement a policy as code based automation solution that keeps environments consistent.

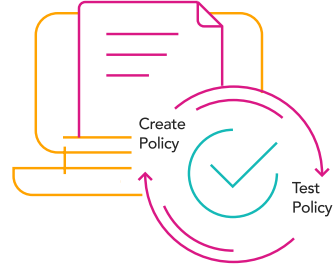
Chef enables DevSecOps teams to create pipelines that can cross both internal and external boundaries, standardizing environments and processes locally within the data center and up in the cloud.

As a result, you get a dynamic environment that’s stable no matter how complicated your configurations are. When your application deployment and infrastructure changes move at the same pace, your entire IT organization functions better.

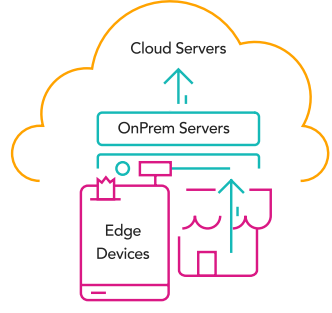
In fact, environmental configurations are foundational to application and business success. A DevSecOps team that turns configuration into code can leverage the same tools and processes you use on your applications to efficiently and successfully prepare environments to run applications.

Create A Workflow to Release Configurations

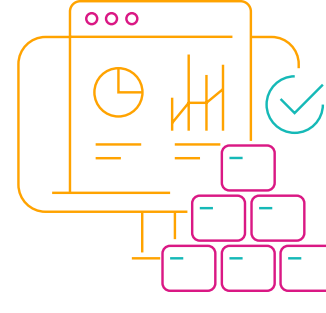
Using [Chef Infrastructure Management](#) automation DevOps teams ensure configurations are applied consistently in every environment.



Create and Test Policies: Reduce risks by iterating on policy changes before pushing them to production



System State Enforcement: Enforce policy by converging the system to the state declared by the various resources.



Data Aggregation and Validation: View and validate intended and actual state across all systems.

Code doesn’t get released without going through a workflow that culminates in testing and approval, and turning configuration into code lets you manage workflow the same way for the infrastructure your applications depend on. Configuration change testing becomes parallel to application change testing. Test-driven development identifies defects through the unit test process; test-driven infrastructure lets you test the changes you make to configurations before unleashing them on production. Your DevOps team delivers environments the same way your developers deliver applications.

Why Customers Choose Chef for Configuration Management Software

Getting started with Chef is straightforward and user-friendly. Chef uses a language where users define configurations once and then apply them easily across a mix of Linux, Mac, and Windows systems, regardless of the operating system version or architecture. The Chef Infra language comes with a variety of pre-built resources, helpers, and cookbooks, both from Chef and the Chef Community.

- ✓ **Resources:** These are tools for configuring various components like packages, files, directories, or firewalls. The Chef Infra Client provides over 150 resources for common automation tasks, such as managing users, files, kernel modules, and Windows tasks.
- ✓ **Helpers:** Helpers help users make configuration decisions based on factors like operating systems, cloud environments, virtualization hypervisors, and more.
- ✓ **Chef Developer Community:** The Chef community has created numerous freely available configuration templates (cookbooks) for Chef Infra. These can be used as they are or as a starting point for an organization's specific configurations.
- ✓ **Agile Test-Driven Development Practices:** Chef has been a pioneer in test-driven infrastructure, contributing to the development of DevOps principles and test-driven practices. These practices are now widely used by agile-based delivery teams for testing and deploying systems across diverse IT environments, from smart devices on the edge to on-premises servers and cloud-based workloads.
- ✓ **Immutable Deployments:** In DevOps deployments, it's crucial to ensure consistency across development, test, and production environments. Chef achieves this with Policyfiles, ensuring that the Chef Cookbooks running in production are the exact versions that were tested in development. Policies are immutable, meaning they cannot be changed once bundled. This eliminates concerns about configurations changing during runtime, providing stability and reliability.

Ready to Get Started?

FREE TRIAL

COMPANY

[Blog](#)
[About Progress](#)
[Leadership](#)
[Customer Stories](#)
[Press Coverage](#)
[Press Releases](#)
[Events](#)
[Careers](#)
[Contact Us](#)

USING CHEF

[LearnChef](#)
[Docs](#)
[Webinars](#)
[Forums](#)
[Training](#)
[Security](#)
[Contributing to Chef](#)
[Code of Conduct](#)

LEGAL

[Terms & Conditions](#)
[Privacy Policy](#)
[Master License and Services Agreement](#)
[Trademark Policy](#)
[3rd Party Software Licenses](#)
[Cookie Policy](#)

CONNECT WITH US

CONTACT US



Chef is part of the Progress product portfolio. Progress is the leading provider of application development and digital experience technologies.

[About Us](#) [Awards](#) [Press Releases](#) [Media Coverage](#) [Careers](#) [Offices](#)

Copyright © 2025 Progress Software Corporation and/or its subsidiaries or affiliates. All Rights Reserved.

Progress and certain product names used herein are trademarks or registered trademarks of Progress Software Corporation and/or one of its subsidiaries or affiliates in the U.S. and/or other countries. See [Trademarks](#) for appropriate markings.

[Security Center](#) [License Agreement](#)
[Do Not Sell or Share My Personal Information](#)
Powered by [Progress Sitefinity](#)