6 Лучших инструментов управления конфигурациями в DevOps

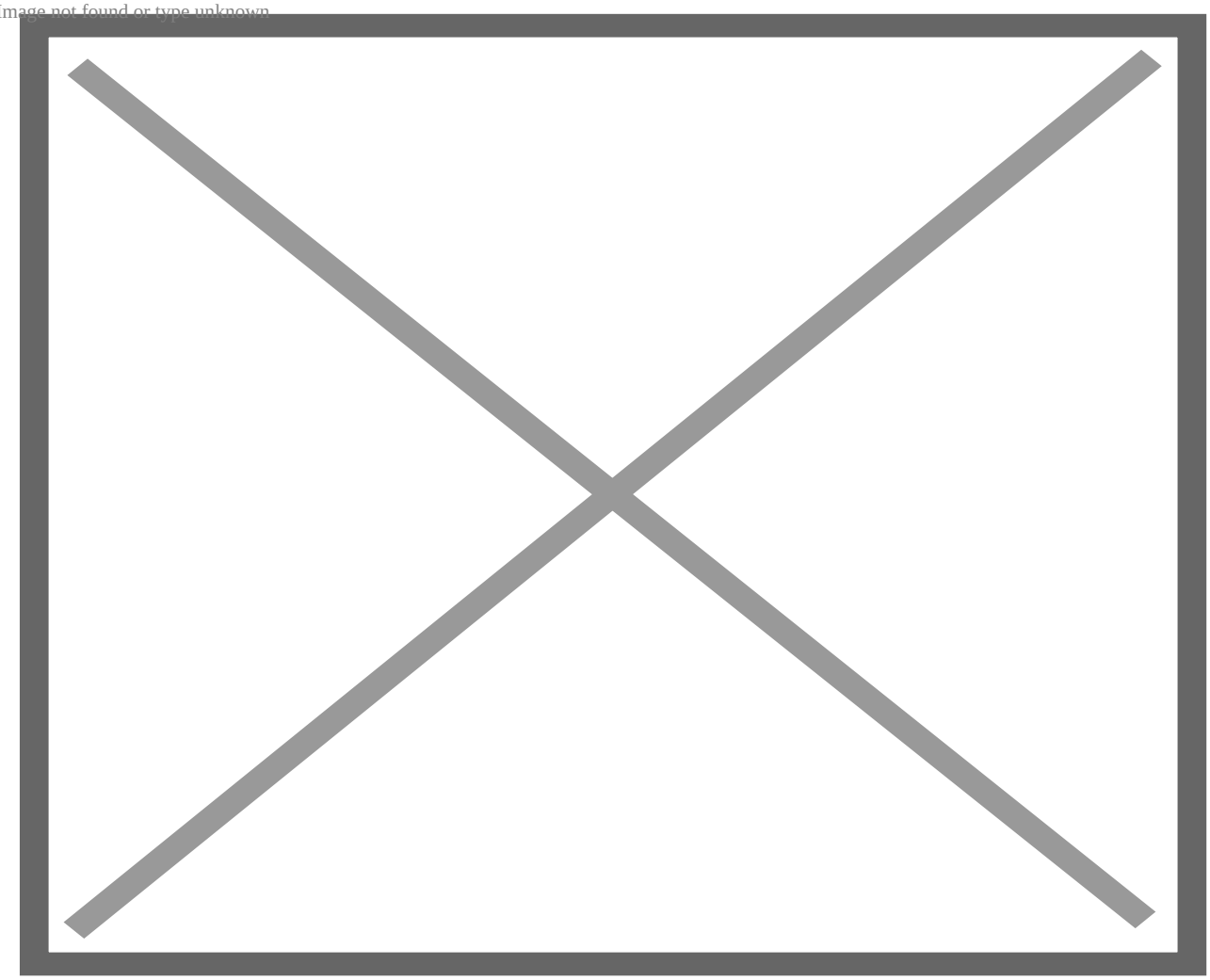
Описание

DevOps – это новое увлечение многих. Организации стремятся внедрить в свои экосистемы такие практики DevOps, как управление конфигурациями. При внедрении культуры DevOps программные инструменты имеют решающее значение независимо от того, работает ли организация с локальными или облачными сервисами. В этой статье вы узнаете о лучших инструментах, доступных для управления конфигурациями в DevOps. Но сначала давайте вкратце разберемся, что такое DevOps. Давайте начнем!

Что такое DevOps?

DevOps – это интеграция деятельности команд разработчиков (Dev) и операторов (Ops) с целью создания ценности для клиентов и обеспечения гибкости в разработке программного обеспечения.

Image not found or type unknown



DevOps ориентирован на то, как люди работают и сотрудничают, разделяя свои мыслительные процессы и приоритеты для ускорения разработки программного обеспечения. Основная идея DevOps, как культуры, заключается в оптимизации функций и эффективности команд, участвующих в разработке, независимо от используемых инструментов. Но как началось это единое развитие?

Раньше в жизненном цикле разработки ПО были разработчики, работа которых сводилась к написанию кода по заданию заказчика без настройки и поддержки среды для требуемого программного продукта. Операционная группа занималась производственной деятельностью и обслуживанием, испытывая на себе все кошмары, связанные с этапом производства.

Представьте себе управление программным продуктом, в разработке которого вы не участвовали! Тяжело, правда? На плечи команды Ops ложилось бремя устранения ошибок, управления инфраструктурными зависимостями и другие

проблемы, связанные с производственной средой ПО. Для устранения этого недостатка и появился DevOps, то есть объединение людей, задач и всех сквозных процессов, необходимых для передачи заказчику тщательно разработанного продукта.

Почему DevOps имеет значение?

Когда команды в любой среде разработки грамотно внедряют практики DevOps, такие как непрерывная интеграция и управление конфигурациями, компании получают следующие преимущества:

Сокращение циклов выпуска приложений

DevOps служит цели поддержания готовой к развертыванию кодовой базы, где в любой момент команда DevOps может запустить доступные версии программного обеспечения без сбоев в работе продукта. Конвейеры CI/CD, содержащие все средства автоматизации и тестирования, обеспечивают постепенную отправку стабильного программного продукта в производство, что позволяет разработчикам сократить цикл выпуска.

Видимость процессов разработки

Выявление программных дефектов, обнаружение угроз безопасности, инициирование откатов и даже реагирование на инциденты может быть затруднено, если среда разработки представляет собой “черный ящик”. Более короткие циклы выпуска и непрерывный мониторинг в DevOps обеспечивают большую видимость всех действий.

Что такое управление конфигурацией в DevOps?

Управление конфигурацией – это автоматизация значимых и повторяющихся действий в ИТ-среде. Управление конфигурацией позволяет решать задачи, масштабируемые на сотни и тысячи машин. К таким задачам можно отнести установку, модернизацию и обновление программного обеспечения, управление исправлениями, обеспечение безопасности, управление пользователями и многие другие. С появлением контейнерных технологий и других инфраструктурных усовершенствований системные администраторы сталкиваются с нелегкой задачей

обеспечения и конфигурирования ИТ-среды без средств автоматизации. К счастью, существуют средства управления конфигурацией, позволяющие создавать и оптимизировать среды времени выполнения. Средства управления конфигурацией в DevOps обеспечивают создание необходимой инфраструктуры с помощью скриптов/инфраструктуры как кода. Рассмотрим следующие широко используемые средства управления конфигурациями.

6 Лучших инструментов управления конфигурациями в DevOps

Ansible

Ansible позволяет автоматизировать конфигурирование инфраструктуры, развертывание приложений и предоставление облачных ресурсов, используя при этом сервисную модель Infrastructure as Code. Ansible – это полезный инструмент, который инженеры DevOps могут использовать для автоматизации инфраструктуры, приложений, сетей и контейнерных сред. Инженеры широко используют этот инструмент для автоматизации и настройки серверов.

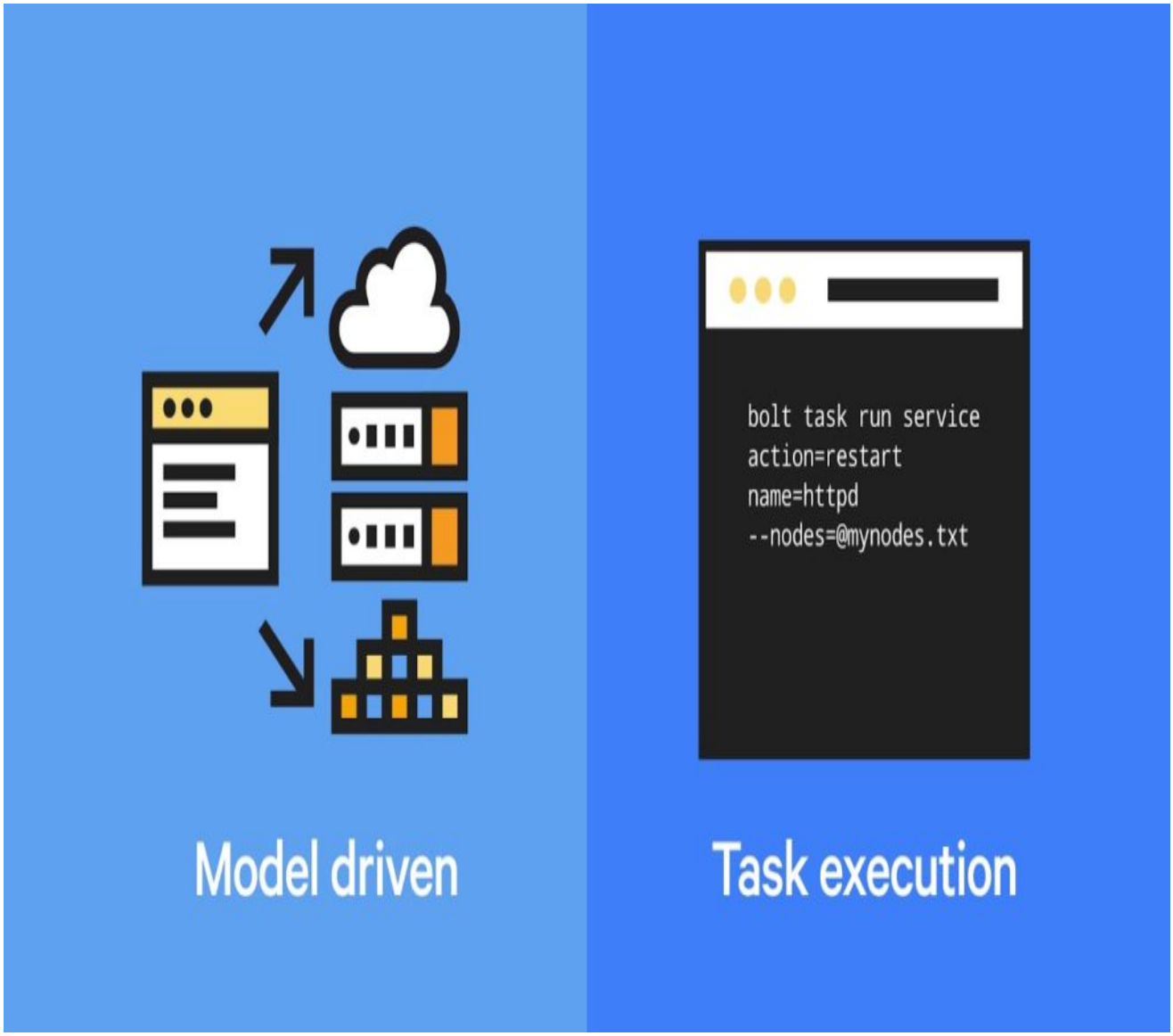
Этот инструмент позволяет сократить количество повторяющихся задач по администрированию инфраструктуры с помощью определенных плейбуков. В данном случае playbook – это простой файл сценария YAML, в котором подробно описаны действия, выполняемые механизмом автоматизации Ansible. Благодаря автоматизации Ansible операционная команда может создавать группы машин, на которые будут действовать определенные задачи, и контролировать работу машин в производственных средах. По имеющимся данным, Ansible используют такие известные компании, как Udemy, Alibaba Travels, Tokopedia и многие другие.

Особенности:

- Ansible Tower, платформа в составе Ansible, представляет собой панель визуализации всей ИТ-среды.
- С помощью ролевого управления доступом (RBAC) в области Ansible можно создавать пользователей, а также управлять разрешениями для сред.
- Ansible поддерживает конфигурации как локальных, так и мультиоблачных инфраструктур.

Puppet

Puppet – еще одна платформа с открытым исходным кодом, подходящая для создания отказоустойчивой инфраструктуры. Инженеры DevOps могут использовать Puppet для конфигурирования, развертывания, запуска серверов и автоматизации развертывания приложений на сконфигурированных серверах.



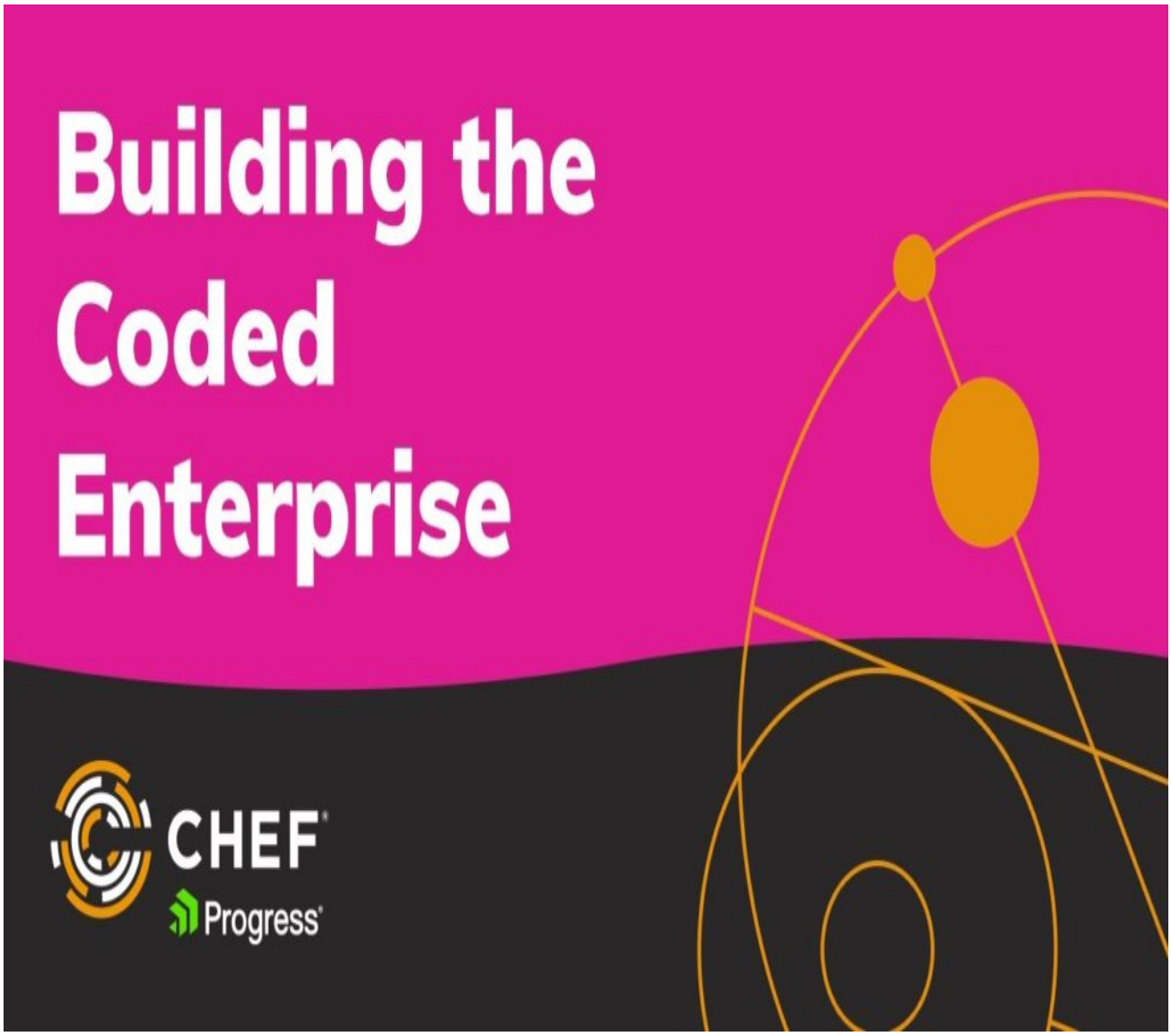
С помощью Puppet можно устранять операционные риски и риски безопасности в ИТ-среде, обеспечивая постоянное соответствие требованиям. В Puppet реализованы функции автоматизации оконной инфраструктуры, управления исправлениями и управляемыми операциями с приложениями. Тысячи компаний, включая Google, Cisco и Splunk, используют Puppet для управления конфигурациями.

Особенности:

- Высокая расширяемость, поддержка нескольких инструментов разработчика и API.
- В состав Puppet входит Bolt – мощный оркестровщик задач для автоматизации ручных операций.
- Puppet хорошо интегрируется с Kubernetes и Docker.

Chef

Chef как инструмент DevOps позволяет выполнять задачи управления конфигурацией серверов и других вычислительных ресурсов. Подход Chef к управлению инфраструктурой использует агенты, такие как Chef Infra, для автоматизации конфигурирования инфраструктуры. Использование Chef в процессах автоматизации очень просто. Всего несколько щелчков мышью – и несколько узлов могут быть запущены в работу.



Команды DevOps определяют “рецепты” для управления конфигурацией. Рецепты содержат описание ресурсов и программных пакетов, необходимых для конфигурирования серверов. В качестве основных компонентов для конфигурирования и автоматизации Chef использует книги рецептов, серверы Chef и узлы. Такие ведущие компании, как Facebook, Slack и Spotify, используют Chef в своих экосистемах.

Особенности:

- Chef – это платформа автоматизации на основе агентов.
- Chef работает с инфраструктурой как код.
- Поддерживает все операционные системы и интегрируется с любыми облачными технологиями.

- В Chef реализована аналитика Chef для мониторинга изменений, происходящих на сервере Chef.

Saltstack

Saltstack или просто salt – это масштабируемый инструмент управления конфигурациями и оркестровки. Команды DevOps используют salt для управления ИТ-средами, такими как центры обработки данных, посредством событийной оркестровки и удаленного выполнения конфигураций.

Система управления конфигурацией Salt использует состояния и конфигурационные файлы для отображения процесса инициализации и развертывания ИТ-инфраструктуры. Конфигурационные файлы описывают

устанавливаемые пакеты инфраструктуры, запускаемые и останавливаемые службы, пользователей и процессы создания пользователей, а также многие другие задачи, необходимые для обеспечения ИТ-среды.

Возможности:

- Платформа Salt Cloud для инициализации систем в облаке.
- Поддерживает как агентное, так и безагентное управление узлами.
- Поддерживаются операционные системы *NIX и Windows.

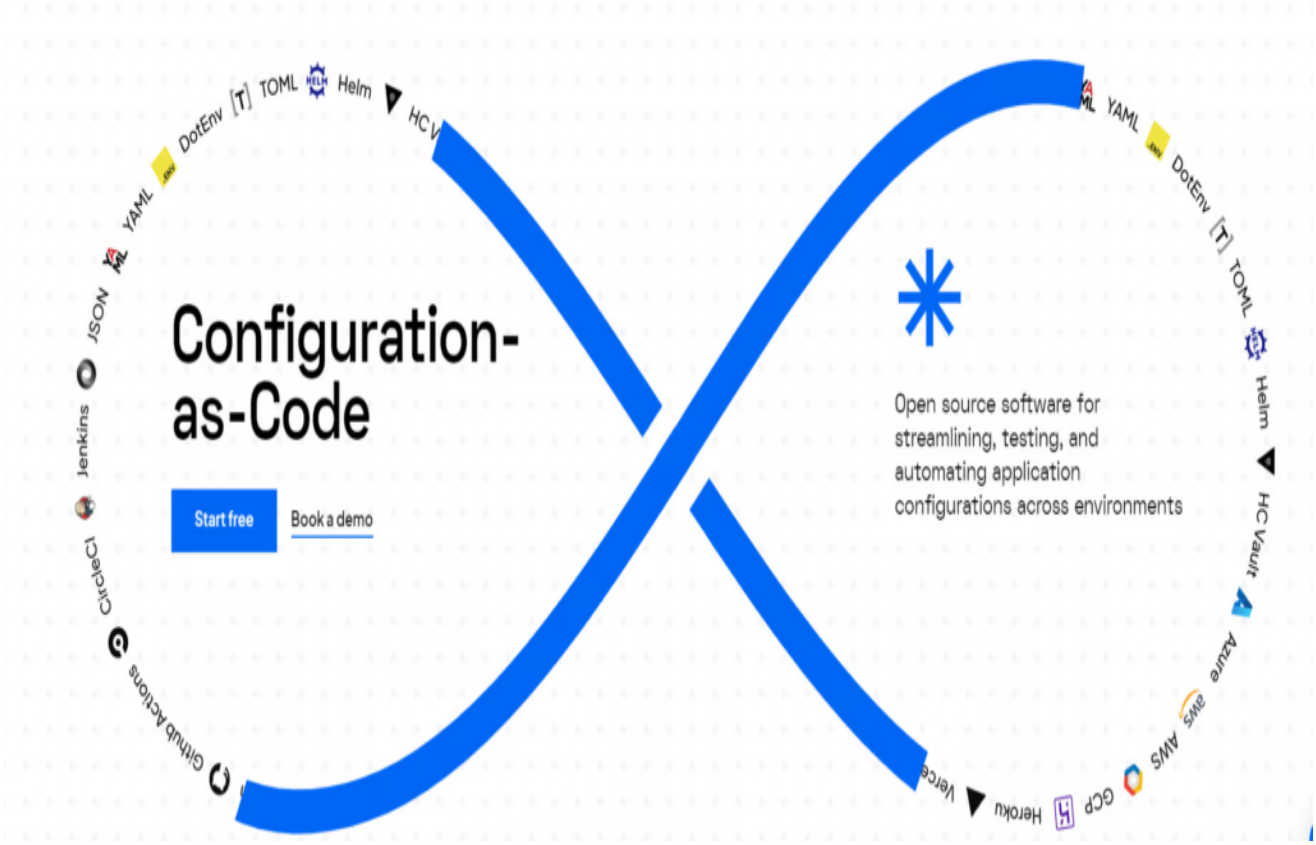
CFEngine

CFEngine – это высокомасштабируемая платформа для автоматизированного управления ИТ-инфраструктурой. С помощью CFEngine команды могут выполнять инициализацию физических и виртуальных инфраструктур, управление исправлениями, контроль доступа, управление пользователями и обеспечение безопасности систем, находясь в одном месте. С помощью CFEngine автономные агенты постоянно работают над непрерывным мониторингом, ремонтом, обновлением и восстановлением ИТ-инфраструктуры. Непрерывные проверки систем и автоматическое устранение неполадок в CFEngine гарантируют надежность и согласованность всей инфраструктуры.

Особенности:

- Высокая гибкость благодаря схеме конфигурирования ‘write once use repeatedly’.
- Имеет CFEngine Enterprise Mission Portal – центральную платформу приборной панели для наглядности и мониторинга ИТ-систем в реальном времени.
- Использование легких агентов автоматизации в рамках платформы WebScale для конфигурирования и управления несколькими узлами.

Configu



Как известно, управление конфигурациями – это автоматизация значимых и повторяющихся действий в ИТ-среде. И Configu как раз подходит под это определение. Configu – это инструмент управления конфигурациями, который облегчит работу всех разработчиков и менеджеров. Разработчики имеют широкие возможности для расширения сотрудничества и повышения производительности с помощью Configu. Кроме того, менеджеры сообщества считают его очень надежным и безопасным.

Особенности:

- Автоматизированное управление конфигурациями
- Расширение совместной работы для разработчиков
- Надежность и безопасность для менеджеров
- Формат конфигурации с открытым исходным кодом

Configu позволяет создавать конфигурации в формате с открытым исходным кодом. Более того, она позволяет автоматизировать процесс с помощью расширенных возможностей управления, таких как тестирование, визуализация, отслеживание и безопасность. Хотелось бы также отметить важный фактор, связанный с

развертыванием Configu: он позволяет оптимизировать процессы с помощью всего одной команды. Так что попробуйте!

Заключение

Лучший способ найти лучшие инструменты для своих нужд – это попробовать их. То, что работает для других, может не подойти вам, поэтому попробуйте их, чтобы увидеть, как они работают, как помогают вашей организации поддерживать последовательность и безопасность конфигурации.

Дата Создания
03.08.2023