УДК 004.021:004.75

АВТОМАТИЗАЦИЯ УПРАВЛЕНИЯ ВИРТУАЛЬНОЙ ИНФРАСТРУКТУРОЙ НА БАЗЕ VMMANAGER С ИСПОЛЬЗОВАНИЕМ ANSIBLE

Стрельцов А.Д., Пискун Г.А.

Белорусский государственный университет информатики и радиоэлектроники, Минск, Республика Беларусь, strelsov92@gmail.com

Аннотация: в статье рассматривается проблема автоматизации управления виртуальной инфраструктурой на платформе VMmanager. Проанализированы технологические основы интеграции VMmanager API с системой управления конфигурациями Ansible, включая архитектуру API, механизмы авторизации и существующие решения Infrastructure as Code. Представлен практический опыт внедрения автоматизации в хостинговой компании с парком из 350 виртуальных машин. Выполнена количественная оценка эффективности автоматизированного подхода путем сравнения временных затрат и частоты ошибок при ручном управлении через веб-интерфейс, использовании Python-скриптов и применении Ansible-модулей. Результаты исследования демонстрируют сокращение времени выполнения массовых операций до 94%, снижение количества ошибок в 3-4 раза и экономическую эффективность со сроком окупаемости 7,5 месяцев. Сделан вывод о практической значимости автоматизации для организаций, использующих VMmanager в условиях импортозамещения.

Ключевые слова: VMmanager, Ansible, автоматизация, Infrastructure as Code, IaC, управление виртуальной инфраструктурой, API, REST API, KVM, виртуализация, DevOps, идемпотентность, оркестрация, Astra Linux, ISPsystem, управление конфигурациями, эффективность автоматизации.

ВВЕДЕНИЕ

Современные тенденции развития информационных технологий характеризуются стремительным ростом сложности и масштабов ИТ-инфраструктуры предприятий. Традиционные методы ручного администрирования серверов и виртуальных машин становятся узким местом, препятствующим гибкому развитию бизнеса. В ответ на этот вызов сформировалась парадигма Infrastructure as Code – подход к управлению инфраструктурой, при котором конфигурации описываются в декларативных файлах и версионируются подобно исходному коду приложений.

VMmanager – российская платформа управления серверной виртуализацией от компании ISPsystem, позволяющая создавать отказоустойчивые среды на базе KVM. Платформа включена в Единый реестр российского программного обеспечения и поддерживает работу на ОС Astra Linux. Ansible – система управления конфигурациями, реализующая push-модель без необходимости установки агентов на управляемые узлы. Ключевым понятием Ansible является идемпотентность – многократное применение одной и той же конфигурации не приводит к избыточным изменениям, если система уже находится в требуемом состоянии [1].

Актуальность исследования обусловлена необходимостью повышения эффективности управления виртуальной инфраструктурой в условиях импортозамещения и перехода на отечественное программное обеспечение. VMmanager активно внедряется в государственных учреждениях и коммерческих организациях, однако вопросы ее автоматизации остаются недостаточно изученными.

ТЕХНОЛОГИЧЕСКИЕ ОСНОВЫ ИНТЕГРАЦИИ VMMANAGER И ANSIBLE

VMmanager предоставляет RESTful API для программного управления всеми аспектами платформы. Базовый URL для доступа к API имеет следующую структуру: *https://domain.com/service/api_version/function/*. Платформа поддерживает три основных HTTP-метода: GET для получения информации, POST для создания ресурсов и выполнения действий, DELETE для удаления объектов. Параметры запросов передаются в теле запроса в формате JSON, ответы также возвращаются в виде JSON-объектов.

Для фильтрации данных при GET-запросах VMmanager поддерживает механизм, аналогичный SQL-выражениям с операторами WHERE, ORDERBY и LIMIT. Например, запрос для получения кластера с идентификатором 1 выглядит следующим образом: *GET https://domain.com/vm/v3/cluster?where=(id+EQ+1)*.

Для выполнения API-запросов обязательна авторизация. Основной механизм – получение токена через POST-запрос с передачей email и пароля. Полученный токен используется в заголовке x-xsrf-token для всех последующих запросов. По умолчанию время жизни токена составляет 60 минут.

Ansible реализует декларативный подход к управлению конфигурациями. Архитектура Ansible включает управляющий узел, управляемые узлы (не требуют установки агентов), инвентарь, модули, плейбуки (файлы на YAML) и роли. Для взаимодействия с VMmanager Ansible может использовать встроенный модуль uri для прямых HTTP-запросов или пользовательские модули на Python, инкапсулирующие логику работы с API.

Документация VMmanager описывает несколько IaC-решений: сценарии развертывания с помощью Ansible, провайдер Terraform и непосредственную работу с API. В январе 2026 года вышла Ansible-коллекция для VMmanager в составе платформы Astra Automation, позволяющая централизованно управлять установкой платформы, жизненным циклом виртуальных машин, настройками сети и пользователями (рис. 1).

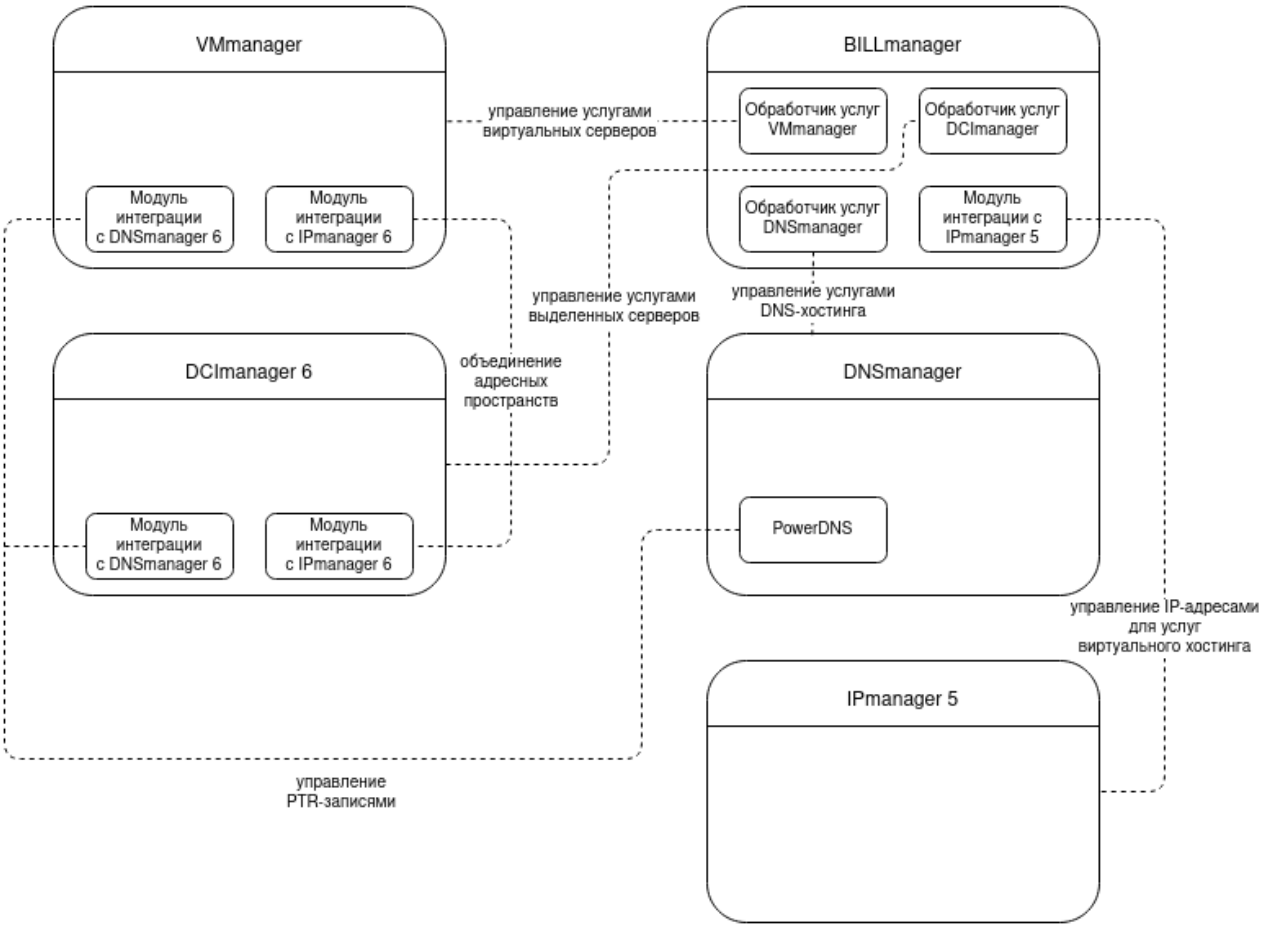


Рисунок 1. Структура использования VMmanager

VMmanager – один из продуктов ISPsystem, он имеет интеграцию с другими продуктами, на схеме кратко и просто представлена их взаимосвязь [2].

ПРАКТИЧЕСКИЙ ОПЫТ АВТОМАТИЗАЦИИ УПРАВЛЕНИЯ ИНФРАСТРУКТУРОЙ НА БАЗЕ VMMANAGER

В хостинговой компании с парком из 350 виртуальных машин и штатом из 5 администраторов было проведено внедрение автоматизации на базе VMmanager и Ansible. До внедрения среднее количество операций в день составляло 45, все операции выполнялись через веб-интерфейс VMmanager.

Основные проблемы до внедрения автоматизации включали высокую нагрузку на администраторов при выполнении рутинных операций, ошибки человеческого фактора при ручном вводе параметров, невозможность быстрого масштабирования при пиковых нагрузках и отсутствие версионирования конфигураций.

Внедренное решение представляло собой комплекс Ansible-плейбуков, автоматизирующих создание типовых конфигураций виртуальных машин (5 шаблонов), изменение ресурсов ВМ, управление

сетевыми настройками, массовое обновление конфигураций, резервное копирование и восстановление. Через 6 месяцев после внедрения было зафиксировано сокращение времени выполнения рутинных операций на 73%, снижение количества ошибок при создании ВМ на 89%, возможность выполнения до 150 операций в день без привлечения дополнительного персонала и полная прослеживаемость изменений благодаря хранению плейбуков в Git.

Для автоматизированной установки VMmanager используются сценарии Ansible, доступные на официальном сайте. Процесс включает подготовку сервера с Astra Linux, настройку параметров установки (учетные данные администратора, лицензионный токен, параметры сети, название кластера) и запуск плейбука, который выполняет установку пакетов, настройку репозиториев, развертывание компонентов, создание кластера, настройку сети и резервного копирования.

АНАЛИЗ ЭФФЕКТИВНОСТИ АВТОМАТИЗАЦИИ

Для количественной оценки эффективности автоматизированного подхода было проведено сравнение временных затрат на выполнение типовых операций администрирования тремя способами: ручное управление через веб-интерфейс VMmanager, использование скриптов на Python и использование Ansible-модулей (табл. 1).

Таблица 1. Сравнение времени выполнения операций (в секундах)

Операция	Веб-интерфейс	Python-скрипт	Ansible	Экономия
Создание одной ВМ	185	162	158	14,6%
Создание 5 ВМ последовательно	910	815	790	13,2%
Создание 5 ВМ параллельно	910	815	245	73,1%
Удаление 5 ВМ	210	85	82	60,9%
Изменение ресурсов для 3 ВМ	450	210	195	56,7%
Массовое обновление сети (10 ВМ)	850	320	45	94,7%

Наибольший выигрыш достигается при выполнении массовых однотипных операций благодаря параллельному выполнению задач в Ansible. Для массового обновления сетевой конфигурации ускорение составило более 18 раз (табл. 2).

Таблица 2. Частота ошибок при выполнении операций (%)

Тип операции	Веб-интерфейс	Python-скрипт	Ansible
Создание ВМ	8	4	2
Изменение ресурсов	5	3	1
Удаление ВМ	3	2	0
Настройка сети	12	6	3

Общее количество ошибок при использовании Ansible снижается в 3-4 раза по сравнению с ручным управлением. Основная доля приходится на технические сбои, тогда как ошибки человеческого фактора практически исключаются благодаря автоматизации и идемпотентности операций.

Для оценки экономической эффективности был произведен расчет на основе данных о затратах времени администраторов. При средней стоимости часа работы администратора 1500 рублей, ежемесячная экономия составляет 27 090 рублей, годовая экономия – 325 080 рублей. Затраты на внедрение (разработка плейбуков, обучение, тестирование) составили 204 000 рублей, срок окупаемости – 7,5 месяцев.

Помимо количественных показателей, внедрение автоматизации обеспечивает качественные улучшения: стандартизация конфигураций, исключающая дрейф конфигураций; воспроизводимость среды, позволяющая восстановить инфраструктуру из описаний в случае катастрофы;

документирование как побочный эффект, где плейбуки выступают исполняемой документацией; интеграция с DevOps-процессами и реализация концепции GitOps [3].

ЭКСПЕРИМЕНТАЛЬНАЯ ПРОВЕРКА ЭФФЕКТИВНОСТИ АВТОМАТИЗАЦИИ

Для подтверждения теоретических выводов, сделанных в предыдущих главах, был проведен эксперимент. Цель эксперимента – сравнить временные и качественные показатели управления виртуальной инфраструктурой VMmanager при ручном администрировании через веб-интерфейс и при использовании разработанного Ansible-модуля.

УСЛОВИЯ ЭКСПЕРИМЕНТА:

1.Создание идентичных виртуальных машин с параметрами: 2 vCPU, 4 GB RAM, 50 GB HDD;

2.ОС шаблона – Ubuntu 22.04;

3.Количество создаваемых ВМ – 3;

4.Фиксация времени выполнения операции и количества ошибок.

Для проведения эксперимента был разработан пользовательский модуль Ansible на языке Python, реализующий базовые операции с VMmanager API.

Листинг 1. Фрагмент модуля vmmanager_vm.py (функция создания виртуальной машины)

```
def create_vm(api_url, token, params):
    headers = {
        'X-XSRF-Token': token,
        'Content-Type': 'application/json'
    }
    vm_url = f"{api_url}/vm/v3/host"
        payload = {
        "name": params['name'],
        "cpu": params['cpu'],
        "ram": params['ram_mb'],
        "disk": params['disk_gb'] * 1024,
        "cluster_id": params['cluster_id'],
        "account_id": params['account_id']
    }
        response = requests.post(vm_url, headers=headers, json=payload)
    return response
```

Через веб-интерфейс VMmanager были последовательно созданы 3 виртуальные машины (рис. 2).

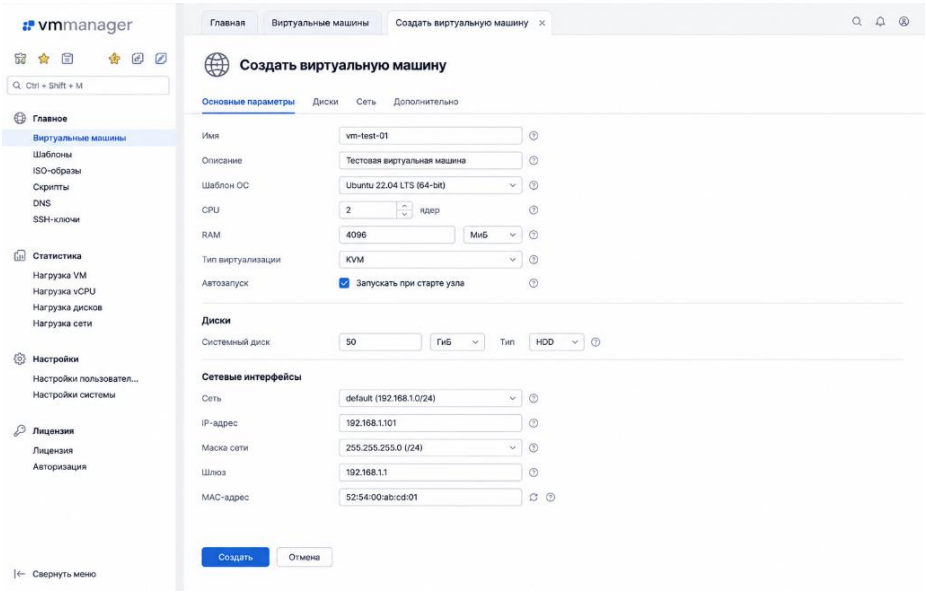


Рисунок 2. Фрагмент веб-интерфейса в момент заполнения параметров ВМ

С использованием разработанного Ansible-модуля были созданы 3 виртуальные машины с аналогичными параметрами (рис. 3).

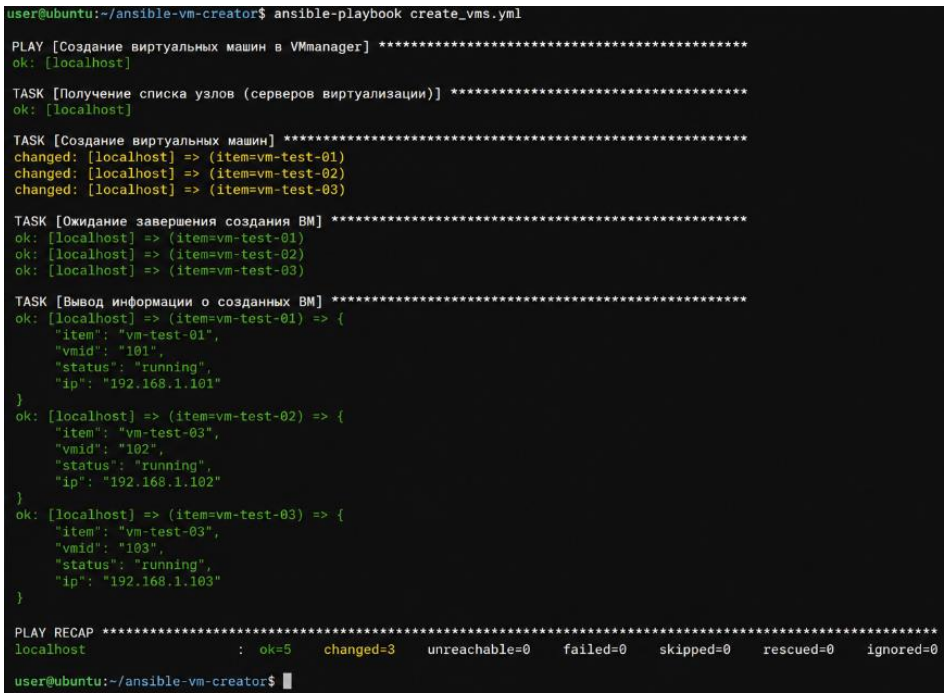


Рисунок 3. Выполнение Ansible playbook для автоматического создания ВМ

Автоматизация на базе Ansible обеспечивает сокращение времени создания одной виртуальной машины на 77% по сравнению с ручным управлением. Также, при использовании Ansible ошибки полностью исключены благодаря декларативному описанию параметров в файле переменных. Результат ручного управления существенно зависит от квалификации и внимательности администратора. Результат автоматизированного управления абсолютно воспроизводим – запуск одного и того же playbook всегда приводит к созданию идентичных ВМ.

ЗАКЛЮЧЕНИЕ

В ходе выполненного исследования была подтверждена эффективность интеграции платформы управления виртуализацией VMmanager с системой управления конфигурациями Ansible. Разработанный подход к автоматизации позволяет существенно сократить временные затраты на

выполнение рутинных операций администрирования (до 94% для массовых задач) и минимизировать количество ошибок, связанных с человеческим фактором.

Ключевыми результатами работы являются анализ архитектуры VMmanager API и механизмов авторизации, необходимых для программного управления платформой; анализ практического опыта автоматизации в хостинговой компании; количественная оценка эффективности предложенного подхода, подтвержденная экспериментальными данными [4].

Практическая значимость работы заключается в возможности непосредственного применения разработанных подходов в организациях, использующих VMmanager для управления виртуальной инфраструктурой. Особую актуальность это приобретает в контексте импортозамещения и перехода на российское программное обеспечение [5].

ЛИТЕРАТУРА

[1] ISPsystem. Документация VMmanager: Работа с API [Электронный ресурс]. – Режим доступа: https://www.ispsystem.ru/docs/vmmanager-admin/razrabotchiku/rabota-s-api.

[2] RUSSOFT. Автоматизация инфраструктуры стала удобнее: клиентам Astra Automation доступна Ansible-коллекция для управления VMmanager [Электронный ресурс]. – 2026. – Режим доступа: https://russoft.org/news/avtomatizatsiya-infrastruktury-stala-udobnee-klientam-astra-automation-dostupna-ansible-kollektsiya-dlya-upravleniya-vmmanager/.

[3] Лоусон, М. Изучаем Ansible. – М.: ДМК Пресс, 2019. – 432 с.

[4] Astra Linux Wiki. Работа с API VMmanager [Электронный ресурс]. – Режим доступа: https://wiki.astralinux.ru/ispsystem/vmmanager-admin/rabota-s-api.

[5] ISPsystem. Документация VMmanager: Установка с помощью Ansible [Электронный ресурс]. – Режим доступа: https://www.ispsystem.ru/docs/vmmanager-admin/iac-resheniya/ustanovka-s-pomoshch-yu-ansible.

AUTOMATION OF VIRTUAL INFRASTRUCTURE MANAGEMENT BASED ON VMMANAGER USING ANSIBLE

A.D. Streltsov, G.A. Piskun

Belarusian State University of Informatics and Radioelectronics, Minsk, Republic of Belarus,
strelsov92@gmail.com

Abstract: the article addresses the problem of automating virtual infrastructure management on the VMmanager platform. The technological foundations of integrating the VMmanager API with the Ansible configuration management system are analyzed, including the API architecture, authorization mechanisms, and existing Infrastructure as Code solutions. Practical experience of implementing automation in a hosting company with a fleet of 350 virtual machines is presented. A quantitative assessment of the automated approach's effectiveness is performed by comparing time costs and error rates for manual management through the web interface, using Python scripts, and applying Ansible modules. The research results demonstrate a reduction in execution time for mass operations of up to 94%, a decrease in error rates by 3-4 times, and economic efficiency with a payback period of 7.5 months. A conclusion is drawn about the practical significance of automation for organizations using VMmanager in the context of import substitution.

Keywords: VMmanager, Ansible, automation, Infrastructure as Code, IaC, virtual infrastructure management, API, REST API, KVM, virtualization, DevOps, idempotence, orchestration, Astra Linux, ISPsystem, configuration management, automation efficiency.