МИНОБРНАУКИ РОССИИ
Федеральное государственное бюджетное образовательное учреждение
высшего образования

«САРАТОВСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ ИМЕНИ Н.Г. ЧЕРНЫШЕВСКОГО»

Кафедра математической теории упругости и биомеханики

Создание интегрированного инструмента для решения задач системного администрирования

АВТОРЕФЕРАТ БАКАЛАВРСКОЙ РАБОТЫ

студента 4 курса 442 группы
направления 09.03.03 – Прикладная информатика

механико-математического факультета

Британа Дмитрия Юрьевича

Научный руководитель
доцент, к.ф. –м.н. А.М. Донник

Зав. кафедрой
зав. кафедрой, д.ф. – м.н., профессор Л.Ю. Коссович

Саратов 2025

Введение. Современное системное администрирование — это ключевая область в поддержке информационной инфраструктуры предприятий. В крупных организациях оно охватывает десятки и сотни рабочих станций, серверов и сервисов, требующих постоянного контроля, обновления, резервного копирования и обеспечения безопасности. Даже в условиях развития автоматизированных решений и средств удалённого управления, системный администратор остаётся основным звеном, отвечающим за стабильную и безопасную работу IT-среды.

Однако подход к системному администрированию значительно варьируется в зависимости от масштаба организации. Например, в крупном предприятии вычислительная сеть может включать сотни компьютеров, что требует чёткой структуризации и постоянного наблюдения. Теоретически, на каждую рабочую станцию можно назначить собственного администратора, но на практике такая модель нецелесообразна: большая часть задач носит пиковый характер, и постоянная загрузка специалистов невозможна. Это приводит к перерасходу бюджета и снижению эффективности.

Рациональной стратегией становится делегирование задач ограниченному числу специалистов с применением интегрированных инструментов удалённого управления. Это позволяет централизованно контролировать инфраструктуру, оперативно решать инциденты и масштабировать управление при минимальных ресурсах.

Несмотря на то, что на рынке уже существует множество инструментов системного администрирования, они, как правило, покрывают лишь отдельные группы задач. В крупных корпоративных решениях задачи решаются централизованно (например, с помощью SCCM или GPO), но в малых и средних предприятиях их приходится решать разными способами, используя

несколько специализированных утилит. Это увеличивает нагрузку, требует поддержки сложных интеграций и повышает вероятность ошибок.

Актуальность темы обусловлена необходимостью создания инструмента, способного охватывать широкий круг задач: от мониторинга и диагностики до управления пользователями и обновления программного обеспечения. Существующие решения, как правило, либо избыточны по функциональности и требуют сложной инфраструктуры, либо ограничены и не позволяют централизованно управлять системой.

Цель настоящей работы: создание универсального и лёгкого в использовании инструмента системного администрирования с возможностью масштабирования, интеграции с доменной структурой и удалённого управления рабочими станциями.

Для достижения цели поставлены следующие **задачи:**

- провести анализ задач, с которыми сталкиваются системные администраторы;
- выполнить обзор применяемых инструментов и их ограничения;
- разработать и реализовать инструмент, обеспечивающий централизованное и безопасное выполнение задач.

Структура и объем работы. Бакалаврская работа включает в себя введение, три раздела основной части, заключение, приложения и содержит 49 страниц. Список использованных источников включает 20 наименований.

Раздел 1. Анализ задач системного администрирования.

Раздел 2. Анализ существующих решений.

Раздел 3. Реализация инструмента.

Основное содержание работы. Работа посвящена разработке инструмента системного администрирования, отличающегося отсутствием требований к инфраструктуре и возможностями интеграции.

Во введении представлены основные сведения о теме бакалаврской работы, обоснована её актуальность, сформулированы цель и задачи исследования.

В первом разделе рассматриваются ключевые задачи системного администратора, охватывающие следующие сферы:

- управление пользователями и правами доступа;
- обслуживание рабочих станций и серверов;
- обеспечение информационной безопасности;
- организация резервного копирования и восстановления данных;
- мониторинг состояния ИТ-инфраструктуры и диагностика проблем;
- централизованное развёртывание и обновление программного обеспечения.

Каждая из этих сфер связана с обеспечением бесперебойной работы ИТ-систем, повышением устойчивости корпоративной инфраструктуры и снижением рисков, связанных с человеческим фактором. Так, управление пользователями обеспечивает соблюдение политики безопасности, обслуживание станций поддерживает работоспособность оборудования, а мониторинг позволяет выявлять аномалии до возникновения критических сбоев. Все задачи рассмотрены с точки зрения повседневной практики администратора, включая типовые проблемы, с которыми он сталкивается, и способы их решения. Также выделены общие требования к инструментам: простота использования, масштабируемость, наличие обратной связи и возможность централизованного управления.

В ходе анализа установлено, что большинство административных задач выполняются вручную или с применением узкоспециализированных средств. Это приводит к фрагментации процессов, увеличению вероятности ошибок, затрате избыточного времени и снижению общей эффективности управления инфраструктурой.

Во втором разделе рассматриваются существующие программные решения, их механизм работы, целевое назначение, архитектурные особенности и применимость в условиях различного масштаба инфраструктуры. В частности, анализируются Active Directory (AD) и Group Policy (GPO) как инструменты централизованного управления пользователями, PowerShell и PsExec - как средства скриптового администрирования, SCCM — как промышленный комплексный инструмент управления ИТ-инфраструктурой, Ansible - как декларативный инструмент автоматизации, Prometheus — как средство мониторинга, и TSD — как утилита для развёртывания ПО в небольших сетях.

Проведено сравнение по следующим критериям: универсальность (охват задач), масштабируемость (возможность применения в сетях разного размера), сложность внедрения (необходимые ресурсы и квалификация), функциональность (глубина и гибкость выполняемых операций). Выводы анализа:

- AD и GPO незаменимы для доменной инфраструктуры, но бесполезны вне её;
- SCCM предоставляет широкий функционал, но требует значительных ресурсов, а также недоступен на территории России;
- PowerShell универсален, но не централизован и требует ручного написания скриптов;
- Prometheus требует сложной настройки и не предназначен для выполнения административных задач;
- TSD удобен, но не поддерживает мониторинг и журналирование.

На основании анализа инструментов выявлена их высокая эффективность в рамках своих узких областей. Однако также были выявлены их ограничения, которые делают невозможным их универсальное применение.

Глубокая экспертиза существующих платформ позволила выявить четыре ключевых ограничения, препятствующих их широкому применению в малых и средних организациях:

- проблемы обратной связи и прозрачности**.** Многие инструменты (особенно GPO, TSD и PsExec) работают по принципу "запустил и забыл", не предоставляя администратору информации о фактическом выполнении задач. Это превращает процесс управления в "стрельбу вслепую" и значительно усложняет диагностику проблем;

- чрезмерная сложность внедрения**.** Решения корпоративного уровня (SCCM, Ansible, Prometheus) требуют развертывания сложной инфраструктуры, выделенных серверов и значительных временных затрат на настройку. Для организаций с ограниченными ресурсами такие требования часто оказываются неподъемными;

- жесткая привязка к экосистемам**.** Большинство коммерческих продуктов создают vendor lock-in, делая организацию заложником конкретного поставщика. Это ограничивает гибкость управления и создает дополнительные риски;

- несоответствие реальным рабочим процессам**.** Существующие инструменты часто навязывают администратору искусственные workflow, не соответствующие его повседневным задачам, что приводит к снижению эффективности работы.

Концепция нового подхода - на основании выявленных проблем сформулированы требования к принципиально новому решению, которое должно сочетать:

- универсальность функционала**.** Объединение ключевых административных задач (управление пользователями, мониторинг, развертывание ПО, диагностика) в едином интерфейсе с сохранением глубины функциональности специализированных инструментов;

- гибкая архитектура**.** Применение легковесной клиент-серверной модели, позволяющей работать как с агентами, так и через стандартные протоколы (WinRM, WMI). Особое внимание уделяется минимальным требованиям к инфраструктуре;

- естественная интеграция. Максимальное использование встроенных возможностей ОС (особенно PowerShell для Windows-сред) без привязки к конкретным коммерческим продуктам;

- прозрачность управления. Сквозное логирование всех операций, детальная обратная связь о выполнении задач и встроенные механизмы аудита.

Обоснована необходимость создания нового инструмента, ориентированного на малые и средние организации, не обладающие ресурсоёмкой инфраструктурой. Потребность в таком решении формируется из практики: у администраторов отсутствуют удобные средства, сочетающие управление, мониторинг и развёртывание ПО. Требования к инструменту сформированы на основе анализа ограничений существующих решений, типовых административных задач, а также интервью с администраторами: безагентная архитектура (минимизация нагрузки), использование встроенных механизмов Windows (доступность и безопасность), централизованное управление, масштабируемость, лёгкость настройки и расширяемость.

Третий раздел посвящен проектированию и реализации инструмента, в основе которого лежит модель "клиент-сервер". Выбор этой модели обусловлен необходимостью централизованного управления с возможностью распределённого исполнения задач на удалённых машинах. Это позволяет обеспечить как масштабируемость (добавление новых клиентов без изменения архитектуры), так и удобство администрирования.

Серверная часть реализована на Python с использованием фреймворков Uvicorn и FastAPI. Uvicorn выбран за простоту и надёжность, а так же за возможность интеграции с FastAPI, который в свою очередь выбран за высокую производительность и встроенную поддержку REST API. Такая комбинация

позволяет быстро реализовывать и расширять функциональность, а также легко интегрировать интерфейс с внешними системами.

В качестве базы данных выбрана SQLite — лёгкая, встраиваемая СУБД, не требующая отдельного сервера. Этот выбор сделан в пользу простоты развёртывания и отсутствия дополнительных зависимостей. Для более нагруженных сценариев возможна миграция на PostgreSQL без изменения логики приложения.

Клиентская часть — это PowerShell-агент, выполняемый через Task Scheduler. Выбор PowerShell обусловлен его нативной интеграцией в Windows, мощным API, широкими возможностями по диагностике и управлению, а также отсутствием необходимости устанавливать дополнительные компоненты. Использование Task Scheduler позволяет реализовать периодическое выполнение без создания фоновых служб.

В качестве альтернатив могли быть выбраны:

- агенты на базе Windows Service, но это усложнило бы установку и сопровождение;

- использование сторонних решений (например, SaltStack), что потребовало бы дополнительных зависимостей и инфраструктурной поддержки.

Функциональность системы включает:

- удалённое выполнение команд — позволяет запускать любые скрипты и команды без входа на машину;

- управление пользователями и группами — централизованное создание, удаление и назначение прав;

- аудит и сбор диагностической информации — системные логи, состояние служб, загрузка ресурсов;

- контроль параметров безопасности — проверка состояния антивируса, политики паролей, брандмауэра;

- запуск сценариев резервного копирования — автоматизация и контроль процедур бэкапа;
- мониторинг ресурсов и журналирование событий — создание отчётов и база для анализа;
- развёртывание программного обеспечения — запуск установщиков и проверка версий ПО.

Выбор этих функций обусловлен их высокой частотностью в повседневной работе администратора, их критичностью для устойчивости системы и сложностью при выполнении вручную. Централизация и автоматизация этих задач повышают надёжность и сокращают время реакции.

Вся информация передаётся по защищённому каналу (HTTPS) с использованием встроенных средств Python. Это гарантирует безопасность обмена данными. В системе реализовано логирование всех операций и централизованное хранение результатов, что необходимо для аудита и последующего анализа.

Пример практического сценария: установка тонкого клиента 1С. Администратор формирует задание в веб-интерфейсе, указывает путь к установщику и параметры, после чего агент на клиентской машине выполняет установку и возвращает результат в виде отчёта.

В данной главе было подробно рассмотрено создание и обоснование архитектуры интегрированного программного инструмента для системного администрирования. Разработка охватывает ключевые задачи, обозначенные в первой главе, такие как управление пользователями, обслуживание оборудования, обеспечение безопасности, мониторинг и развертывание программного обеспечения.

Предложенная архитектура демонстрирует, что платформа, основанная на PowerShell, REST API и легкой клиент-серверной модели, способна эффективно решать задачи, которые не под силу существующим универсальным решениям без значительных затрат ресурсов и усилий. В

отличие от крупных корпоративных решений, таких как SCCM, GPO, Prometheus и Ansible, разработанный инструмент предлагает преимущества в простоте, прозрачности и адаптивности, а также требует минимальных инфраструктурных ресурсов.

Также были проанализированы возможности дальнейшей интеграции системы с внешними компонентами, такими как аналитические платформы, системы безопасности, базы данных и внешние интерфейсы. Это делает приложение перспективной платформой, способной масштабироваться по мере роста инфраструктуры и увеличения требований к управлению.

Заключение. Разработанный программный инструмент решает задачу комплексного подхода к системному администрированию в условиях ограниченной ИТ-инфраструктуры. В отличие от существующих решений, он обеспечивает объединение основных функций в одном интерфейсе при минимальных требованиях к ресурсам.

Практическая значимость заключается в возможности применения разработанного решения в организациях с ограниченным ИТ-бюджетом. Система может быть быстро развёрнута, расширена и адаптирована к конкретным условиям эксплуатации.

Перспективы развития включают интеграцию с внешними платформами (например, Telegram, Zabbix, ELK), расширение набора встроенных скриптов, реализацию графического интерфейса для администратора, добавление модуля визуализации метрик и аналитики. Также возможна миграция базы данных на более производительные СУБД и внедрение системы событий с подпиской и уведомлениями в реальном времени.

Таким образом, в рамках ВКР достигнута поставленная цель – создан универсальный инструмент системного администрирования, способный эффективно функционировать в среде с ограниченными ресурсами и при этом обеспечивать централизованное управление ИТ-инфраструктурой.