

В работе рассмотрены теоретические аспекты обеспечения безопасности серверов, включая методы Linux Hardening, а также подходы к минимизации угроз информационной безопасности. Использование Ansible для автоматизации конфигурации межсетевых экранов позволяет централизованно управлять настройками, минимизируя человеческие ошибки и стандартизируя конфигурации на множестве серверов.

В процессе выполнения работы был разработан плейбук на языке YAML для автоматической настройки UFW. Плейбук позволяет включить межсетевой экран, задать политику по умолчанию (блокировка всего входящего трафика) и разрешить доступ только к необходимым портам. Тестирование проводилось в виртуальной среде, что позволило проверить работоспособность разработанных решений.

На основании выполненных исследований сделан вывод, что использование Ansible для автоматизации настройки межсетевых экранов не только упрощает управление безопасностью серверов, но и значительно повышает их устойчивость к кибератакам. Предлагаемые решения могут быть использованы в образовательных целях, а также в корпоративных и государственных организациях для повышения уровня информационной безопасности.

С. В. Васильев, Е. В. Рафалова
(ГГУ имени Ф. Скорины, Гомель)

ЧИСТАЯ АРХИТЕКТУРА В РАМКАХ РАЗРАБОТКИ ПРИЛОЖЕНИЙ

Архитектура программного обеспечения представляет собой фундаментальную организацию системы, выраженную через структуру её компонентов, связи между ними и принципы их взаимодействия. Она выполняет функцию каркаса, обеспечивая жесткое разделение ответственности, управление зависимостями и соблюдение ключевых принципов разработки. В современных приложениях, где растёт сложность бизнес-логики, критически важна продуманная архитектура, которая обеспечивает порядок, предсказуемость и воз-

возможность развития продукта без больших изменений кодовой базы. Архитектура помогает решать задачи масштабируемости, поддержки и адаптивности системы.

Чистая архитектура – это концепция, которая предлагает подход к проектированию приложений, ориентированный на создание гибких, долговечных и независимых от внешних факторов систем. Её ключевая идея заключается в разделении приложения на слои с чёткими границами, где бизнес-правила и логика изолированы от деталей реализации.

Чистая архитектура строится на принципе строгого разделения ответственности, при котором зависимости направлены от внешних слоев к внутренним через четко определенные интерфейсы. Это позволяет изменять один компонент без необходимости модификации других. Слои представляют собой логические уровни приложения, каждый из которых выполняет определенную роль и взаимодействует с другими слоями через четкие границы.

В слое Domain располагаются бизнес-правила и сущности, которые полностью независимы от любых внешних технологий. Этот слой представляет собой ядро приложения, где сосредоточена основная логика предметной области.

Слой Application реализует бизнес-логику и координирует взаимодействие между уровнями представления и домена. Здесь определяются сценарии использования приложения, которые связывают пользовательские запросы с бизнес-правилами.

Слой Infrastructure отвечает за поддержку структуры приложения и взаимодействие с внешними системами. Здесь реализуются технические аспекты, необходимые для работы приложения.

Слой Presentation отвечает за взаимодействие с пользователем и обработку его запросов. Этот слой включает в себя пользовательский интерфейс и механизмы, которые позволяют пользователю взаимодействовать с системой.

Таким образом, чистая архитектура обеспечивает четкое разделение ответственности между слоями, что упрощает разработку, тестирование и поддержку приложения, а также делает его более устойчивым к изменениям.