А.Е. Козлов, В.О. Нестор, И.Р. Зулькарнеев

Тюменский государственный университет, г. Тюмень

УДК 004.056

ПЛАТФОРМА ДЛЯ ПРОВЕДЕНИЯ СОРЕВНОВАНИЙ ПО ИНФОРМАЦИОННОЙ БЕЗОПАСНОСТИ

Аннотация. В данной статье рассматривается процесс разработки жюрейной системы для соревнований по информационной безопасности в формате CTF-Jeopardy. Определены функциональные требования к данной системе. Предложена система реализации данной платформы в три этапа. Описаны результаты реализации базового функционала (первого этапа).

Ключевые слова: информационная безопасность, CTF, jeopardy, соревнования.

Тюменский государственный университет уже два года проводит соревнования по информационной безопасности в формате CTF, для студентов и школьников, онлайн - «TyumenCTF» и очно - «UralCTF». После проведения соревнований в 2017 году перед командой организаторов встал вопрос реализации собственной платформы для проведения соревнований CTF в формате Task-based.

Task-based или Jeopardy – вид CTF соревнований, где командам предоставляется набор заданий в различных категориях, решая которые они получают скрытый ответ (далее – флаг). Заданием может быть любой файл, онлайн сервис или объект для поиска в открытых источниках информации. Соревнования, формата task-based, просты в организации (простая игровая инфраструктура) и регулировке уровня сложности. Именно поэтому данный формат удобен как для начинающих, так и опытных участников[1].

Для организации соревнований данного формата требуется специальная журейная платформа (далее – система), позволяющая

администрировать их и отслеживать текущую ситуацию во время проведения.

Разработку данной платформы целесообразно поделить на этапы с целью проверки реализованного функционала, нагрузочного тестирования и апробации в реальных условиях. После внесения изменений и их проверки необходимо будет переходить на следующий этап. Всего определим три этапа.

Первым этапом является разработка базового функционала, определяющему ядро будущей системы. Этому посвящена данная статья.

Следующим этапом станет добавление дополнительного функционала, позволяющим осуществлять более гибкое и удобное управление системой, реализация дополнительных мер безопасности и отказоустойчивости, а также оповещение организаторов о сбоях системы и нарушениях правил командами.

Завершающим этапом должно стать добавление функционала отображения статистических данных, позволяющих в режиме реального времени получать подробную информацию в удобном представлении как для участников, так и для организаторов.

При реализации первого этапа к системе должны быть предъявлены следующие требования, описывающие минимально необходимый функционал для ее работы:

- регистрация команд участников соревнований;
- загрузка заданий в систему и их редактирование;
- настройка автоматизированной выдачи заданий;
- проверка присылаемых флагов;
- логирование действий пользователей, администраторов, принимаемых флагов;
- отображение текущего рейтинга.

Функциональная структура системы и взаимодействие основных элементов представлены на Рисунке 1.

Система регистрации необходима для формирования списка участников, распределения по различным рейтингам, взаимодействия между участниками и организаторами, а также обратной связи. Необходимый минимальный набор данных, вводимых при регистрации определяется условиями проведения соревнований.

В связи с тем, что организаторам соревнований для выдачи сертификатов

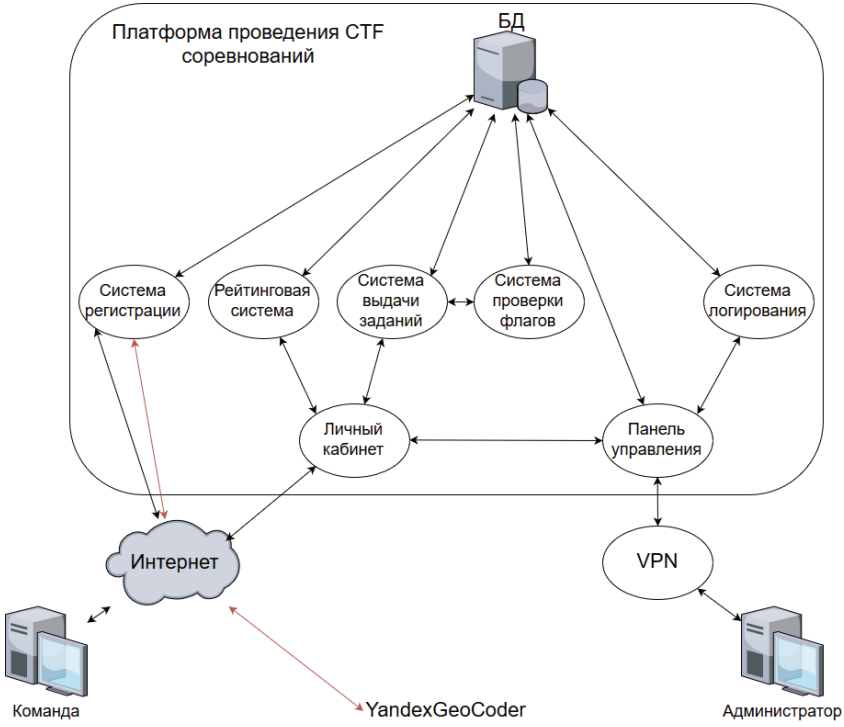

Рис. 1. Функциональная структура системы.

и дипломов, а также для подтверждения принадлежности участников к конкретному учебному заведению в определенном городе, необходимы персональные данные участников, на странице регистрации расположена специальная форма о согласии на обработку этих персональных данных и разрешения на присвоения им статуса “общедоступные” с целью реализации

требований законодательства РФ [2]. Без отметки в данной форме регистрация невозможна.

При загрузке заданий через панель управления необходимо определить формат загрузки и параметры. Для каждого задания определены следующие обязательные атрибуты, которые приведены в Таблице 1.

Таблица 1. Обязательные атрибуты задания

Название атрибута	**Формат**
Наименование задания	Любая строка длиной до 255 символов
Описание	Текст в формате разметки markdown
Прикрепленные файлы (необязательный атрибут)	Файл любого формата (exe, jpg, txt,...)
Разбор задания	Файл формата .pdf
Флаг	Набор символов заданного формата
Стоимость задания	Число, кратное 100
Подсказка (необязательный атрибут)	Текст в формате разметки markdown. Стоимость: отрицательное число, кратное 50
Категория	Название из списка: Reverse, Web, Stegano, OSINT, Crypto, Pwn, Joy, Misc, Admin, Forensic, PPC

Система выдачи заданий может функционировать в следующих режимах:

- автоматически сразу после начала соревнований;

- автоматически при выполнении условия, заданного организаторами;
- вручную.

Основным функционалом жюрейной платформы является система проверки флагов от участников. К данной системе предъявляются следующие требования:

- соотнесение/проверка флага полученного от команды с указанным для данного задания в БД;
- проверка корректности введенного флага;
- логирования всех присланных флагов;
- система должна обрабатывать количество подключений, сопоставимое с количеством участников.

Система логирования агрегирует данные об отправке флагов командами, а также стандартные логи от Flask, Nginx и docker.

Система рейтинга формирует отчет о командах по следующим правилам: в начале рейтинга оказывается команда, набравшая наибольшее количество очков, если две команды набрали одинаковое количество очков, то выше оказывается та, которая решила последнее задание раньше.

Минимальные требования к техническим характеристикам оборудования для такой системы зависят от планируемого количества участников. При проведении TyumenCTF расчет параметров исходил из планируемого числа участника порядка 500 человек. Система осуществляла свою работу, используя ресурсы сервера, имеющего 4ГБ оперативной памяти, 4-х ядерный процессор Intel(R) Xeon(R) CPU E5620 @ 2.40GHz, 15ГБ свбодного дискового пространства. На данном техническом обеспечении, под руководством ОС Debian 9.4 были развернуты docker-контейнеры из-за легкости в настройке, удобства управления, быстроты восстановления и для повышения отказоустойчивости. [3] В разных контейнерах находились сервисы PostgreSQL и веб-сервер Nginx, с обработкой пользовательских запросов с помощью фреймворка Flask. Веб-

сервер Nginx работал в связке с uWSGI с целью обеспечения увеличения производительности и уменьшения количества ошибок. [4] Архитектура системы представлена на Рисунке 2.

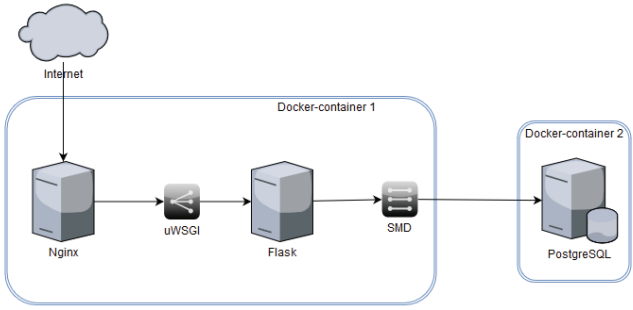

Рис. 2. Архитектура системы.

При обращении пользователя по адресу 2018.tyumenctf.ru Nginx обрабатывает запросы вида 2018.tyumenctf.ru/static/*, для снижения нагрузки на остальную инфраструктуру сервера. Данные запросы не нуждаются в сложной обработке, ответом на них всегда является файл (CSS, JS, шрифты, логотипы команд, лейблы и прочие изображения). Все иные запросы перенаправляются на uWSGI, который в свою очередь перенаправляет их приложению на Python (Flask). Flask выполняет обработку пользовательских запросов и реализует генерацию ответа. Для выполнения данных запросов необходимо взаимодействие с базой данных. Оно осуществляется с помощью драйвера SMD, специально разработанного для данной системы. Драйвер выполняет вызов хранимых процедур базы данных и обработку возвращаемого ими результата. Все хранимые процедуры строго описаны и документированы.

Проведя апробацию разработанной системы во время открытых онлайн соревнований в области информационной безопасности «TyumenCTF», были сделаны следующие выводы:

- функционал панели администратора нуждается в доработке;
- необходимо ввести систему уникальных флагов для каждой команды;

- необходимо внедрение системы оповещений организаторов о различных инцидентах;
- необходимо повышение отказоустойчивости и стабильности системы;
- необходимо ввести систему расчета технических характеристик в зависимости от количества участников и функционала системы.

После внесения данных изменений необходима реализация следующих этапов, которые потребуют апробации на Межрегиональных открытых соревнованиях в области информационной безопасности «UralCTF».

СПИСОК ЛИТЕРАТУРЫ

1. Горбунов К.С., Семакин А.Е., Зулькарнеев И.Р. Организация внутривузовских соревнований по информационной безопасности в формате CTF // Безопасность информационного пространства: Материалы XV Всероссийской научно-практической конференции студентов, аспирантов и молодых ученых. Курган: РИЦ Курганского государственного университета, 2016. - С. 11-14.
2. Федеральный закон от 27.07.2006 № 152-ФЗ «О персональных данных».
3. Литвиненко А.А. Использование программных средств обеспечение безопасности Linux-сервера в рамках соревнований CTF // Вопросы кибербезопасности, 2013. № 3. С. 33-35.
4. Ludovic Gasc. Benchmark Python Web production stack: Nginx with uWSGI, Meinheld and API-Hour. 2015. URL: http://blog.gmludo.eu/2015/02/macro-benchmark-with-django-flask-and-asyncio.html (дата обращения 12.04.2018)