

КОМАНДНЫЕ СОРЕВНОВАНИЯ ФОРМАТА АТТАК/ДЕФЕНС КАК ИНСТРУМЕНТ ПРАКТИКО-ОРИЕНТИРОВАННОГО ОБУЧЕНИЯ ЗАЩИТЕ ИНФОРМАЦИИ

Солонович М.А., Балмаков Д.А., Супрун Ф.Н.

Учреждение образования «Белорусский государственный университет информатики и радиоэлектроники», г. Минск, Республика Беларусь, maxnest77max@icloud.com

Abstract. The paper considers team attack/defense CTF competitions as a practice-oriented form of training information security specialists. An architecture of a training ground is proposed: identical vulnerable web services for each team, an automated jury system that places fresh flags and verifies service availability, and an isolated VPN network. A scoring model combining attack, defense and service-level criteria is described. It is shown that competitions develop the skills of exploiting vulnerabilities, secure development and incident analysis while preserving the service functionality, thus improving the quality of practical training.

Современная подготовка специалистов в области информационной безопасности требует не только теоретических знаний, но и уверенных практических навыков выявления, эксплуатации и устранения уязвимостей. Традиционные лабораторные работы, как правило, фрагментарны: студент обрабатывает отдельный приём на заранее подготовленном стенде и не сталкивается с необходимостью защищать работающий сервис от активного противодействия. Одной из эффективных форм практико-ориентированного обучения, получивших широкое распространение в мировой практике [1, 2], являются соревнования Capture the Flag.

Выделяют два основных формата таких соревнований: jeopardy, в котором команды независимо решают набор изолированных задач, и attack/defense, в котором каждая команда одновременно атакует идентичные сервисы соперников и защищает собственные. Формат А/Д наиболее близко воспроизводит реальный цикл обеспечения безопасности информации: обнаружение уязвимости, её эксплуатацию, анализ инцидента, разработку и развёртывание исправления при обязательном сохранении работоспособности сервиса [2, 3].

Цель настоящей работы – описание методики проведения учебных соревнований формата attack/defense и её технической реализации на основе свободного программного обеспечения. Предлагаемая методика применима в учреждениях высшего образования при реализации практико-ориентированного обучения по дисциплинам направления «информационная безопасность» и не требует дорогостоящей инфраструктуры.

Учебный полигон включает три компонента (рисунок 1). Вулнбокс – сервер с заведомо уязвимым веб-сервисом; каждой команде выдаётся идентичная копия. Система жюри – центральный сервер, объединяющий чекер, модуль приёма флагов и скорбординг. Изолированная сеть – VPN-инфраструктура, разделяющая участников и ограничивающая допустимые направления подключений.

Игра состоит из раундов фиксированной длительности. В начале раунда чекер размещает в сервисе каждой команды уникальный флаг вида FLAG{случайная строка} и проверяет работоспособность сервиса от имени обычного пользователя. Команды эксплуатируют уязвимости чужих вулнбоксов, извлекают размещённые там флаги и

сдают их на сервер жюри, одновременно анализируя и исправляя собственный сервис.

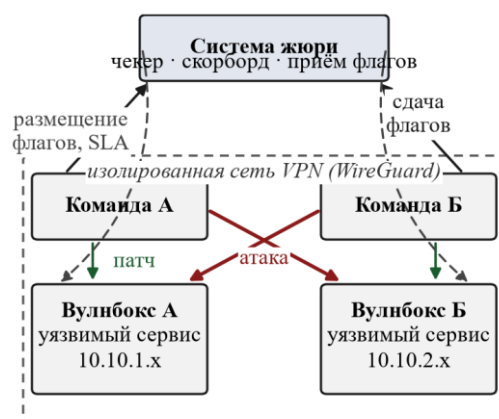


Рисунок 1 – Архитектура учебного полигона attack/defense

Модель начисления баллов объединяет три составляющие. Доступность: сервис, недоступный или некорректно работающий для чекера, приносит команде ноль баллов за раунд. Атака: команда получает баллы за каждый сданный флаг соперника, при этом стоимость флага может снижаться со временем, стимулируя оперативную разработку эксплоитов. Защита: баллы начисляются за флаги, оставшиеся неукраденными к концу раунда.

Особую методическую ценность имеет критерий SLA. Примитивная защита – удаление уязвимого модуля, отключение сервиса, блокировка всех подключений – немедленно наказывается потерей баллов доступности. Таким образом, соревнование воспроизводит ключевое производственное ограничение практики: исправление уязвимости не должно останавливать бизнес-функцию сервиса [2]. Умение вносить точечные исправления, сохраняя работоспособность, относится к числу наиболее востребованных навыков специалиста по защите информации.

Инфраструктура полигона разворачивается на одном сервере; для двух–четырёх команд достаточно машины с 4–8 ядрами и 8–16 ГБ оперативной памяти. Уязвимый сервис упаковывается в контейнер Docker, и для каждой команды запускается идентичный экземпляр. Сеть изолируется средствами VPN: жюри размещается в подсети 10.10.0.x, команды – в подсетях 10.10.1.x и 10.10.2.x; правила фильтрации разрешают командам доступ к чужим

вулнбоксам и серверу жюри, но запрещают прямые подключения к рабочим местам соперников и инфраструктуре полигона.

В качестве системы жюри используется свободная платформа ForcAD, реализующая планировщик чекеров, приём флагов и скорбординг; альтернативами являются checksystem объединения Hackerdom и CTFd с плагинами формата A/D. Под конкретный сервис разрабатывается собственный чекер: регистрация пользователя, создание записи с флагом, её чтение и проверка доступности интерфейса. Результаты проверок фиксируются автоматически, что обеспечивает объективность оценивания и прозрачность скорборда для участников.

Уязвимый сервис представляет собой учебное веб-приложение, в которое заложены три-четыре документированные уязвимости, соответствующие наиболее распространённым классам рисков [4]. Каждая закладка допускает локальное исправление без остановки сервиса; перечень уязвимостей и отрабатываемых способов их устранения приведён в таблице 1.

Таблица 1 – Заложенные уязвимости и отрабатываемые способы их устранения

Уязвимость	Эксплуатация (атака)	Устранение (защита)
SQL-инъекция в форме поиска	Извлечение флагов из базы данных с помощью логики булевых и union-инъекций	Параметризованные запросы, минимальные права учётной записи БД
IDOR при доступе к заметкам	Чтение чужих записей перебором идентификаторов	Проверка принадлежности объекта пользователю на сервере
Инъекция команд ОС в утилите проверки хоста	Выполнение произвольных команд, чтение файла с флагом	Валидация ввода, запрет метасимволов, отказ от вызова shell
Хранимая XSS в комментариях	Кража cookie учётной записи администратора у бота-«пользователя»	Экранирование вывода, политика безопасности содержимого

Существенным эксплуатационным преимуществом контейнерного подхода является возможность быстрого восстановления: команда организаторов возвращает сломанный вулнбокс к исходному состоянию одной командой `docker compose restart`. Это снижает требования к сопровождению полигона и позволяет использовать его регулярно, в том числе в рамках лабораторного практикума.

Оптимальная организационная схема: команды по три-пять человек, продолжительность игры два-четыре часа либо серия раундов, распределённых по практическим занятиям. Игра предваряется инструктажем (правила, выдача доступов, границы допустимых действий) и завершается разбором: организатор демонстрирует обнаруженные уязвимости

и эталонные исправления, команды анализируют собственные журналы и ошибки.

Внутри команды возникает естественное разделение ролей: анализ исходного кода, разработка эксплойтов, мониторинг журналов и обнаружение атак, разработка и развёртывание исправлений. Участники поочерёдно выполняют функции атакующего и защитника, что обеспечивает освоение обеих сторон дисциплины. Соревновательная форма повышает вовлечённость, а автоматический скорбординг даёт объективную обратную связь в реальном времени.

Соревнование формирует комплекс практических умений: применение типовых уязвимостей и инструментальных средств их эксплуатации; чтение и анализ чужого кода; безопасную разработку (параметризованные запросы, контроль доступа, валидацию ввода, экранирование вывода); обнаружение инцидентов по журналам; командную коммуникацию в условиях ограниченного времени. Проведение в изолированной сети на собственном полигоне решает и этическую задачу: навыки легального тестирования отрабатываются без риска для реальных информационных систем.

Предложена методика проведения учебных соревнований по информационной безопасности формата *attack/defense*: описаны архитектура полигона на основе контейнеров Docker и VPN, модель оценивания с критерием доступности и пример уязвимого сервиса с документированными закладками. Формат воспроизводит полный цикл работы специалиста по защите информации – от эксплуатации уязвимости до развёртывания исправления с сохранением работоспособности сервиса – и может быть развёрнут на базе свободного программного обеспечения силами одной кафедры.

Направления дальнейшей работы: интеграция соревнований в учебные планы с пересчётом результатов скорборда в баллы текущей аттестации; автоматизированный сбор учебной аналитики как инструмент объективного контроля сформированности компетенций; масштабирование полигона на большее число команд и подключение дополнительных уязвимых сервисов.

Литература

1. Werther J., Zhivich M., Lekakis T., Zeldovich N. Experiences Teaching Security Using CTF Competitions // ;login: – 2011. – Vol. 36, № 3.
2. Gondree M., Peterson Z. N. J. Valuing Security by Getting {D0x3d}!: Experiences in Cyber Security Education Using Attack and Defend Games // ;login: – 2013. – Vol. 38, № 4.
3. Vykopal J., Barták M. On the Design of Security Games: From Frustrating to Engaging Learning // Proceedings of the 2016 USENIX Cyber Security Experimentation and Test Workshop (CSET '16). – Berkeley: USENIX Association, 2016.
4. OWASP Top 10:2021 [Электронный ресурс] // OWASP Foundation. – Режим доступа: <https://owasp.org/Top10/>. – Дата доступа: 07.09.2026.