

Python для информационной безопасности

Научитесь находить слабые места в Python-сервисе и подтверждать исправления проверками.

120

академических часов

40 + 80

занятия + самоподготовка

8 + 1

проектов модулей + итог

Зачем проходить этот курс

- Свяжете модель угроз с конкретными изменениями кода и тестами.
- Разберёте безопасность API, файлов, секретов и зависимостей в одном проекте.
- Соберёте проверяемый набор мер защиты и отчёт для команды разработки.

Кому подойдёт

Python backend-разработчикам, AppSec- и DevSecOps-специалистам, инженерам ИБ с навыками программирования.

Что нужно на старте

Нужны уверенный Python, основы HTTP/API, SQL, аутентификации, Git и Docker. Практика проходит на изолированном учебном сервисе.

Онлайн • Живые занятия • Группа до 8 человек

Понятный путь к результату

40 ак. ч. с преподавателем + 80 ак. ч. самостоятельной работы. Один академический час - 45 минут. Практика и проекты входят в указанный объём.

Модуль	Занятия	Самост.
01. Модель угроз и учебный стенд	5	10
02. Security code review	5	10
03. Ввод, вывод и инъекции	6	12
04. Секреты и криптография	5	10
05. Аутентификация и доступ	5	10
06. Файлы, сериализация и логи	4	8
07. DevSecOps и цепочка поставок	6	12
08. Сборка и security review	4	8

Как устроена практика

На занятиях разбираются подход и ключевые решения. После каждого модуля вы завершаете мини-проект, проверяете его по критериям и оформляете результат. Итоговый проект объединяет эти работы; это один развиваемый кейс, а не дополнительная нагрузка сверх 120 часов.

Инструменты курса

Python 3 • FastAPI • Pydantic • pytest • Bandit • Semgrep • argon2 • cryptography • Docker • Trivy • CI

Ведущий преподаватель

Сергей Валерьевич Герасименко - технический директор и основатель IT-компаний «ТЕРАКОР» и «ТЕРАКОР Digital», ведущий преподаватель и практикующий эксперт.

Изучаете. Применяете. Проверяете.

МОДУЛЬ 01 / 5 ак. ч. занятий + 10 ак. ч. самостоятельно

Модель угроз и учебный стенд

- Поверхность атаки и активы сервиса
- Границы доверия и STRIDE
- OWASP как основа проверочного списка
- Подготовка изолированного FastAPI-стенда

ПРОЕКТ ПОСЛЕ МОДУЛЯ / Модель угроз API

Описать потоки данных, защищаемые активы и приоритетные сценарии злоупотребления.

Готово, когда: Для каждой значимой угрозы указаны компонент, последствие, мера защиты и способ проверки.

МОДУЛЬ 02 / 5 ак. ч. занятий + 10 ак. ч. самостоятельно

Security code review

- Процесс review и приоритизация риска
- Bandit, Semgrep и настройка правил
- Проверка ложных срабатываний
- Регрессионные тесты для исправлений

ПРОЕКТ ПОСЛЕ МОДУЛЯ / Отчёт аудита модуля

Проанализировать учебный код, подтвердить находки и исправить выбранные проблемы.

Готово, когда: У находки есть воспроизведение на стенде, исправление и тест; исключения обоснованы.

Изучаете. Применяете. Проверяете.

МОДУЛЬ 03 / 6 ак. ч. занятий + 12 ак. ч. самостоятельно

Ввод, вывод и инъекции

- Pydantic и явные ограничения входа
- Параметризованный SQL и границы доверия
- Безопасный вызов процессов и запрет eval для входных данных
- Контекстное кодирование вывода и негативные тесты

ПРОЕКТ ПОСЛЕ МОДУЛЯ / Безопасный импорт данных

Переработать endpoint импорта и проверить некорректные входные значения.

Готово, когда: Неподходящий ввод отклоняется; SQL параметризован; диагностический ответ не раскрывает секреты.

МОДУЛЬ 04 / 5 ак. ч. занятий + 10 ак. ч. самостоятельно

Секреты и криптография

- Argon2 для паролей и управление параметрами
- Шифрование готовыми библиотеками и жизненный цикл ключей
- HMAC для контроля целостности
- Секреты вне репозитория и маскирование журналов

ПРОЕКТ ПОСЛЕ МОДУЛЯ / Secure Export

Создать экспорт с проверкой целостности и конфигурацией ключа вне кода.

Готово, когда: Изменённый payload отклоняется; HMAC не назван шифрованием; ключ не попадает в файл и логи.

Изучаете. Применяете. Проверяете.

МОДУЛЬ 05 / 5 ак. ч. занятий + 10 ак. ч. самостоятельно

Аутентификация и доступ

- Проверка токенов, срока действия и допустимого алгоритма
- Роли и объектная авторизация
- Cookies, CSRF и выбор модели сессии
- Ограничение частоты и аудит входа

ПРОЕКТ ПОСЛЕ МОДУЛЯ / Модуль доступа

Реализовать вход и API с ролями и проверкой владельца объекта.

Готово, когда: Чужие объекты недоступны; просроченный токен отклоняется; негативные тесты автоматизированы.

МОДУЛЬ 06 / 4 ак. ч. занятий + 8 ак. ч. самостоятельно

Файлы, сериализация и логи

- Отказ от недоверенного pickle и небезопасного YAML
- Пути, ограничения каталогов и временные файлы
- Безопасная сериализация и схема данных
- Редактирование чувствительных полей в логах

ПРОЕКТ ПОСЛЕ МОДУЛЯ / Безопасное файловое хранилище

Обработать загрузку и чтение файлов только внутри разрешённого каталога.

Готово, когда: Выход за каталог блокируется; размер ограничен; ошибки не раскрывают внутренние пути и секреты.

Изучаете. Применяете. Проверяете.

МОДУЛЬ 07 / 6 ак. ч. занятий + 12 ак. ч. самостоятельно

DevSecOps и цепочка поставок

- CI: тесты, SAST и правила блокировки
- Сканирование зависимостей и SBOM
- Контейнер non-root и минимальные права
- Triage находок и срок действия исключений

ПРОЕКТ ПОСЛЕ МОДУЛЯ / Контролируемый pipeline

Собрать CI для тестов, анализа кода и сканирования контейнера.

Готово, когда: Контрольная находка блокирует pipeline; исключение имеет причину и срок; отчёты доступны.

МОДУЛЬ 08 / 4 ак. ч. занятий + 8 ак. ч. самостоятельно

Сборка и security review

- Интеграция мер защиты
- Проверка модели угроз по готовому API
- Security README и остаточные риски
- Защита проекта с демонстрацией тестов

ПРОЕКТ ПОСЛЕ МОДУЛЯ / SecGuard API

Собрать учебный сервис и показать цепочку «угроза - исправление - проверка».

Готово, когда: Негативные тесты проходят; все значимые находки разобраны; ограничения проекта описаны.

Ваш главный кейс в портфолио

SecGuard API

Учебный API с ролевым доступом, безопасной обработкой данных и аудитом событий. Проект сопровождается моделью угроз, негативными тестами и CI-проверками; остаточные риски описаны явно.

Критерии готовности

- Угрозы связаны с мерами защиты и соответствующими тестами.
- Проверены отсутствие доступа к чужим объектам и отказ при неверных полномочиях.
- Секреты вынесены из кода; чувствительные значения скрыты в логах.
- SAST, проверка зависимостей и образа входят в CI; исключения обоснованы.
- Есть отчёт до/после исправлений и список ограничений учебного решения.

Что останется после обучения

Репозиторий с исходным кодом, учебные данные или конфигурация стенда, автоматические проверки, инструкция запуска и описание решений. Это материал для демонстрации навыков на работе и техническом собеседовании.

Как представить результат

Покажите задачу, запустите основной сценарий, продемонстрируйте обработку ошибки и объясните один архитектурный выбор. Отдельно зафиксируйте, что можно улучшить следующим шагом.

Итоговый кейс собирается из проектов модулей. Его завершение и подготовка защиты включены в часы последнего модуля.

Что вы сможете делать

- Описывать границы доверия и приоритизировать угрозы.
- Проводить security code review и проверять находки анализаторов.
- Реализовывать валидацию, разграничение доступа и безопасное хранение секретов.
- Встраивать проверки кода, зависимостей и контейнеров в CI.

Практическое обучение в ТЕРАКОР

Живой онлайн-формат позволяет задавать вопросы и разбирать решения. Программа сочетает занятия с преподавателем и самостоятельную практику. После успешного прохождения - сертификат ТЕРАКОР и оформленный проект.

Перед записью

Обсудите исходный уровень, рабочую задачу и время для практики. Для уверенного результата важно проходить модули последовательно и завершать проекты. Курс развивает конкретные навыки; профессиональный уровень зависит также от предыдущего опыта и объёма самостоятельной работы.

Начните с консультации по программе

Подберём стартовый уровень и уточним формат обучения.

Открыть страницу курса → | info@terakor.ru

+7 (960) 352-38-05 • terakor.ru

Сроки старта, стоимость и состав выбранного тарифа уточняются при записи.

Программа оптимизирована на основе страницы курса ТЕРАКОР, изученной 29.09.2026. Это предлагаемая редакция для обновления курса и его описания.

Источник: terakor.ru/catalog/python/python-infosec/