POSITIVE TECHNOLOGIES

РЕАЛИЗАЦИЯ БЕЗОПАСНОЙ РАЗРАБОТКИ

нюансы технических и организационных аспектов на примере реального проекта

АЛЕКСЕЙ ЖУКОВ
Эксперт отдела систем защиты приложений

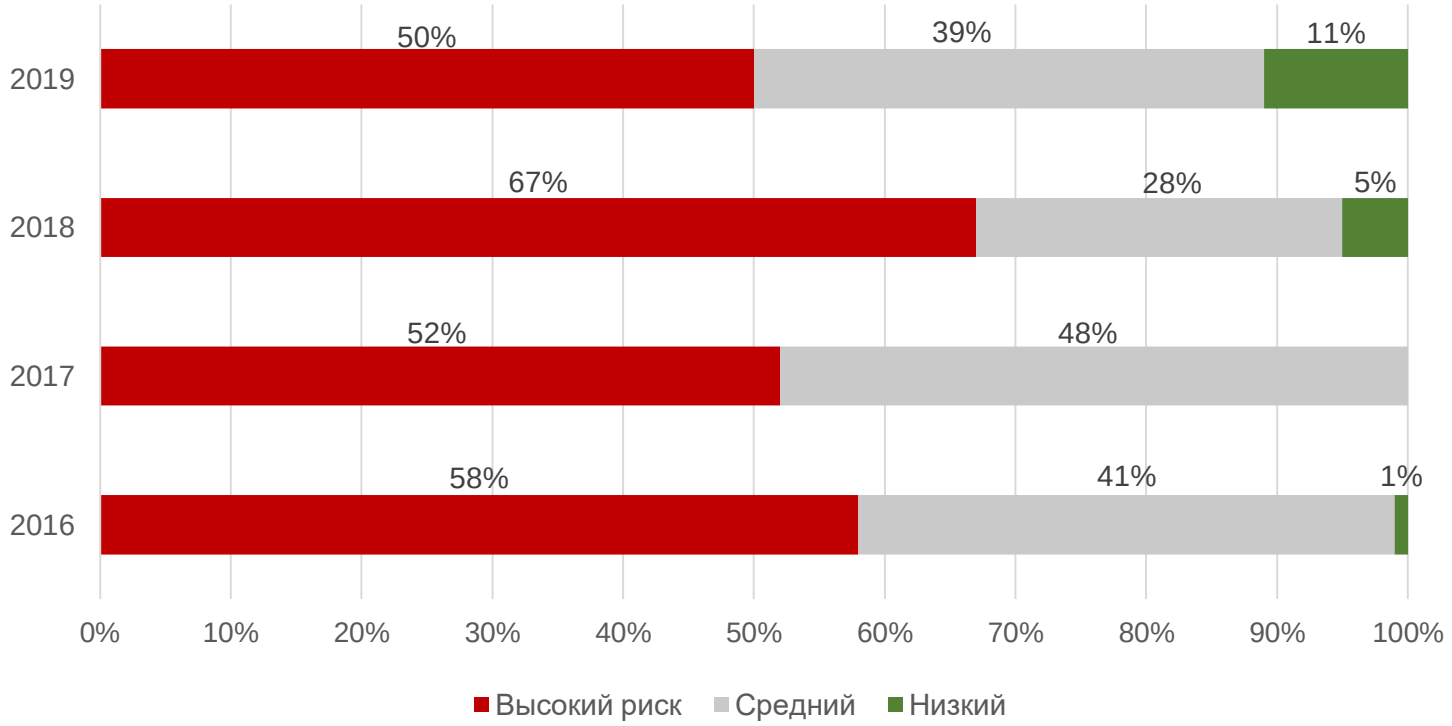
ptsecurity.com

Зачем: уязвимости

50% веб-приложений имеют критически опасные уязвимости

Каждая пятая атака направлена на веб-приложения

Четыре критически опасные уязвимости в среднем имеет каждое веб-приложение

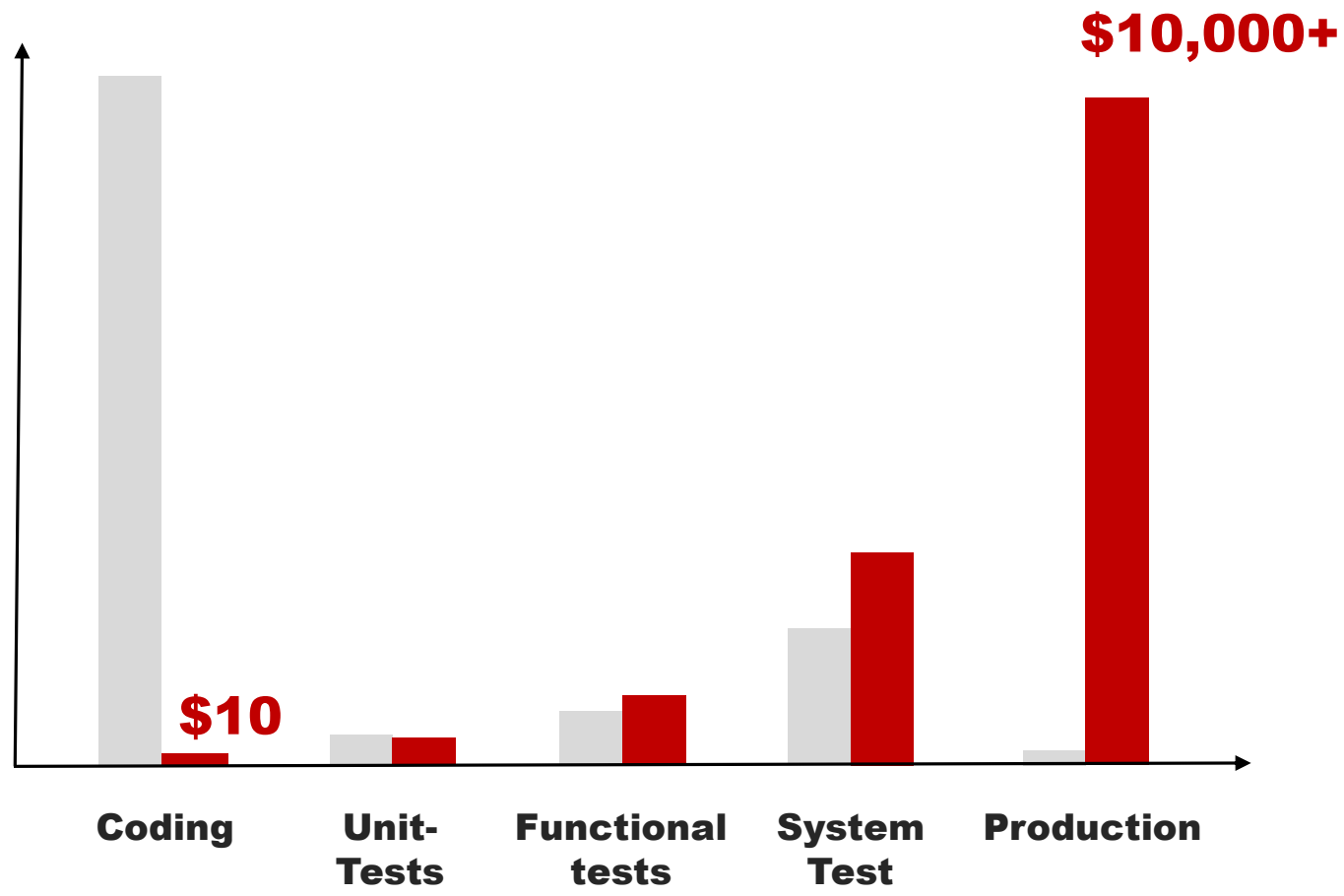


Уязвимости и угрозы веб-приложений в 2019 году, Positive Technologies

Зачем: экономика

Сколько стоит баг и зачем нужен shift-left

- Количество новых багов
- Стоимость исправления одного бага



Applied Software Measurement, Capers Jones

С чего начать

Методологии

MICROSOFT SDL

Набор практик и шагов в привязке к этапам цикла разработки ПО: анализ требований, дизайн, реализация.

Microsoft Security Development Lifecycle, Microsoft

OWASP SAMM

Оценка текущего уровня зрелости, формирование программы мероприятий, активности и практики для внедрения.

OWASP SAMM, OWASP

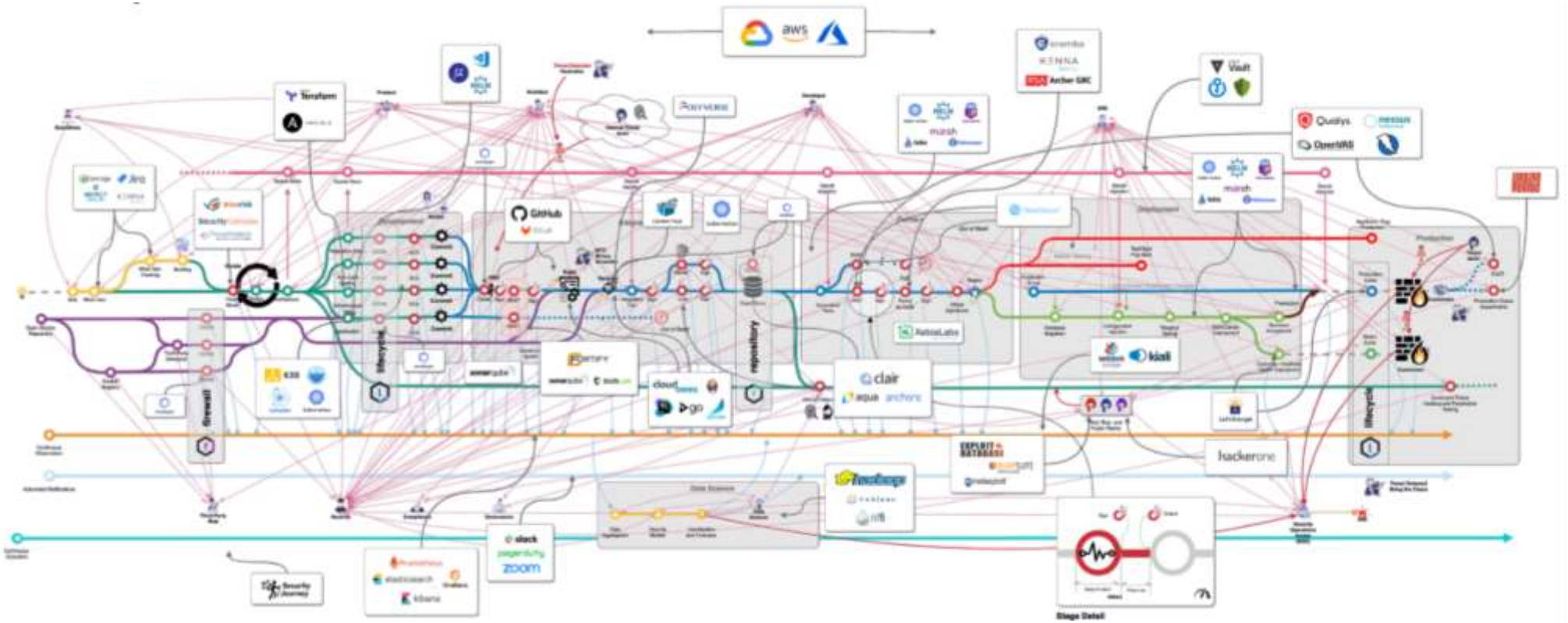
BSIMM

Возможность постепенного построения в соответствии с выбранными приоритетами. Отчеты BSIMM как примеры приоритетов по сегментам рынка

Building Security In Maturity Model, BSIMM

Картина в целом

- **DevSecOps Reference Architectures 2019**
  40+ готовых архитектурных решений
- **DevSecOps Reference Architecture**



DevSecOps Reference Architectures 2019, Sonatype

DevSecOps Reference Architecture, Sonatype

Методология

описывает концепцию, шаги и этапы,
но **не затрагивает** практические вопросы

Как получить доступ к коду?

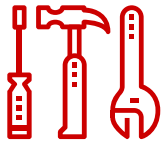
Как «продвигать» безопасность в команде разработки?

Как работать со сторонними библиотеками (3rd party code)?

Как встроиться в существующие процессы разработки?

Как работать с найденными уязвимостями?

Наш клиент



ПРОФИЛЬ

- IT-компания
- Заказная разработка
- 1000х сотрудников
- 20+ лет на рынке

РАЗРАБОТКА

- 10х команд
- Независимы друг от друга: могут выбирать CVS, CI, stack

БЕЗОПАСНОСТЬ

- Ручные проверки
- Периодические тесты на проникновение
- Отсутствие системности, процесса
- Security Champions

Анализ кода

ДОСТУП К КОДУ В ГЕТЕРОГЕННОЙ СРЕДЕ

Организационные сложности получения доступа к десяткам репозиториев (спасаем IT-отдел от кучи заявок)

РЕШЕНИЕ:
интеграция с CI/CD-системой

АНАЛИЗ СТОРОННЕГО КОДА

Сторонние библиотеки, модули и фреймворки — нужен способ их безопасного получения в том числе в среде без доступа в интернет. Части кода может не быть: файлы с исходниками могут генерироваться динамически (то есть их нет в репозитории)

РЕШЕНИЕ:
встраивание после этапа сборки приложения

БЕЗБОЛЕЗНЕННОЕ ВСТРАИВАНИЕ В ДЕЙСТВУЮЩИЕ ПРОЦЕССЫ

Подход «мы будем жить теперь по-новому» для сложившихся команд разработки вызовет отторжение.
Нужен способ встроиться в существующие процессы

РЕШЕНИЕ:
не влиять на процесс сборки

Работа с результатами

БОЛЬШОЕ КОЛИЧЕСТВО НАЙДЕННЫХ УЯЗВИМОСТЕЙ

сложно обработать вручную

РЕШЕНИЕ:
инкрементальный анализ

ЛОЖНЫЕ СРАБАТЫВАНИЯ

неизбежно присутствуют в отчетах любого анализатора кода

РЕШЕНИЕ:
сохранение результатов предыдущих сканирований

Влияние на процесс сборки

ТАБЛИЦЫ, СПИСКИ, ГРАФИКИ, ДИАГРАММЫ

нужны людям для формирования отчетов, анализа и принятия решений, но бесполезны для CI-системы. CI-системе необходимо однозначное бинарное решение о целесообразности продолжения сборочного конвейера

РЕШЕНИЕ:
формирование правил, позволяющих автоматически прерывать сборку при неудовлетворительном результате анализа.

Инструкции и обучение

Вводная встреча

- Общая информация об архитектуре системы, целях её создания
- Инструкции по подключению

Обучение персонала

- что такое уязвимости, как они «выглядят», как их устранять
- навыки работы с продуктом
- демонстрация работы всей цепочки на примере проекта конкретной команды

Результаты

01

Анализатор кода встроен в CI-систему

02

Проведен пилотный проект

Были выбраны 20 проектов (10 команд разработки)

03

Анализ кода встроен в цикл сборки

Сборки пропускаются (блокируются) по вердикту анализатора кода

04

Создан курс обучения

- Архитектура системы, цели её создания
- Проблематика уязвимостей
- Работа с продуктом
- Инструкции по подключению и работе с системой

05

Масштабирование

Чек-лист

01

ПОДДЕРЖКА РУКОВОДСТВА

- Убедитесь, что руководство понимает важность внедрения SSDLC.
- Зафиксируйте цели внедрения.

02

SECURITY CHAMPION

- Выделите участника команды разработки, заинтересованного в безопасности продукта (позволяет решить технические, организационные и другие проблемы).

03

ПИЛОТИРОВАНИЕ РЕШЕНИЙ

- Изучите рынок сканеров приложений, выберите наиболее подходящий для ваших целей.
- Не стремитесь покрыть все проекты сразу — выделите пилотную зону.

04

ПОЛНОТА АНАЛИЗИРУЕМОГО КОДА

- Сделайте так, чтобы весь ваш код, включая сторонние компоненты, был доступен для инструментального анализа.
- Оптимальный вариант — интеграция с CI/CD системой.

05

СОКРАЩЕНИЕ ЧИСЛА ЛОЖНЫХ СРАБАТЫВАНИЙ

- Проанализируйте результаты работы анализатора и отрегулируйте настройки сканирования, чтобы добиться приемлемого числа ложных срабатываний.

06

SECURITY GATE

- Определите критерии нарушения безопасности, которые будут вызывать автоматическую остановку сборки.

07

ОБУЧЕНИЕ

- Система: архитектура и цели создания, инструкции по подключению.
- Предметная область: уязвимости, условия их эксплуатации, примеры, последствия.
- Работа с продуктом, включая демонстрацию работы всей цепочки сборки.

Полезные материалы

ПРО АКТУАЛЬНЫЕ УЯЗВИМОСТИ:

Уязвимости и угрозы веб-приложений в 2019 году, Positive Technologies

ПРИМЕРЫ АРХИТЕКТУР SSDLC:

DevSecOps Reference Architectures 2019, Sonatype
DevSecOps Reference Architecture, Sonatype

О МЕТОДОЛОГИЯХ SSDLC:

Microsoft Security Development Lifecycle, Microsoft
OWASP SAMM, OWASP
Building Security In Maturity Model, BSIMM

КЛАССИКА:

Applied Software Measurement: Global Analysis of Productivity and Quality, Capers Jones
Software Engineering Economics, Barry W. Boehm


POSITIVE
TECHNOLOGIES
СПАСИБО!
@BigAppSec
ptsecurity.com