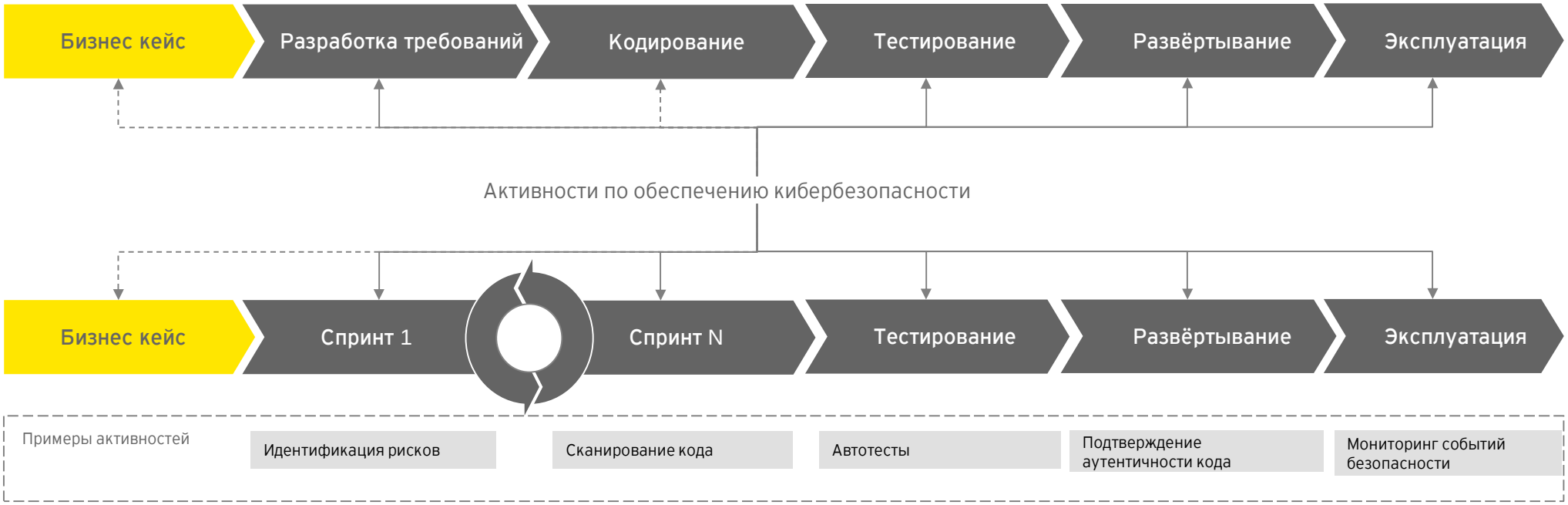
Как выбрать порядок внедрения практик DevSecOps?

The better the question. The better the answer.
The better the world works.

Building a better working world

Безопасная разработка включает в себя внедрение активностей во все этапы жизненного цикла разработки продуктов



Две наиболее популярные методологии включают в себя большой набор возможных активностей

1 BSIMM

- Модель оценки процессов безопасной разработки, основанная на изучении практик безопасной разработки в 128 компаниях из разных секторов экономики.
- Модель включает в себя 122 активностей по обеспечению безопасности ПО, которые объединены в 12 практик по 4 доменам.

GOVERNANCE	INTELLIGENCE	SSDL TOUCHPOINTS	DEPLOYMENT
1. Strategy & Metrics (SM)	4. Attack Models (AM)	7. Architecture Analysis (AA)	10. Penetration Testing (PT)
2. Compliance & Policy (CP)	5. Security Features & Design (SFD)	8. Code Review (CR)	11. Software Environment (SE)
3. Training (T)	6. Standards & Requirements (SR)	9. Security Testing (ST)	12. ConfigurationManagement & Vulnerability Management (CMVM)

2 SAMM

- Модель оценки процессов безопасной разработки опубликованная OWASP (Open Web Application Security Project) и поддерживаемая профессиональным сообществом.
- Модель включает в себя 90 активностей по обеспечению безопасности ПО, которые объединены в 15 практик по 5 функциям.
- Любая практика состоит из 6 активностей, каждая из которых оценивается по четырёхбалльной шкале

Лидирующие ИТ-компании используют разный набор активностей при реализации процессов безопасной разработки

Компания 1
США
Онлайн продажа товаров и услуг

- “You build it, you run it”. Отсутствует разделение полномочий разработчиков и специалистов по эксплуатации приложений в промышленном окружении.
- Каждая команда разработчиков полностью поддерживает один отдельный сервис.
- Применение практик безопасного кодирования и использование проверенных компонентов контролируется на стадии инспекции кода.

Компания 2
США
Онлайн провайдер медиаконтента

- “From gates to guardrails”. Проверка требований безопасности осуществляется после развертывания приложения в промышленном окружении, а не на этапах разработки и тестирования.
- Разработкой и поддержкой разработанного ПО занимается инженер «полного цикла».
- No Ops. Вся инфраструктура реализована на основе облачных технологий и является для разработчиков по умолчанию безопасной.

Компания 3
США
Онлайн продажа товаров

- Акцент в обеспечении безопасности приложений смещён на оперативное устранение уязвимостей в промышленном окружении, а не на их выявление на разных этапах разработки.
- Приоритет отдаётся уменьшению time-to-market с целью максимально быстрого устранения возможных уязвимостей.
- Следование принципам MVP (minimum viable product) при поставке функциональности в промышленное окружение.
- Используются разные реализации DevOps pipeline для систем разной критичности (отличаются в том числе набором security gates).

Компания 4
США
Социальная сеть

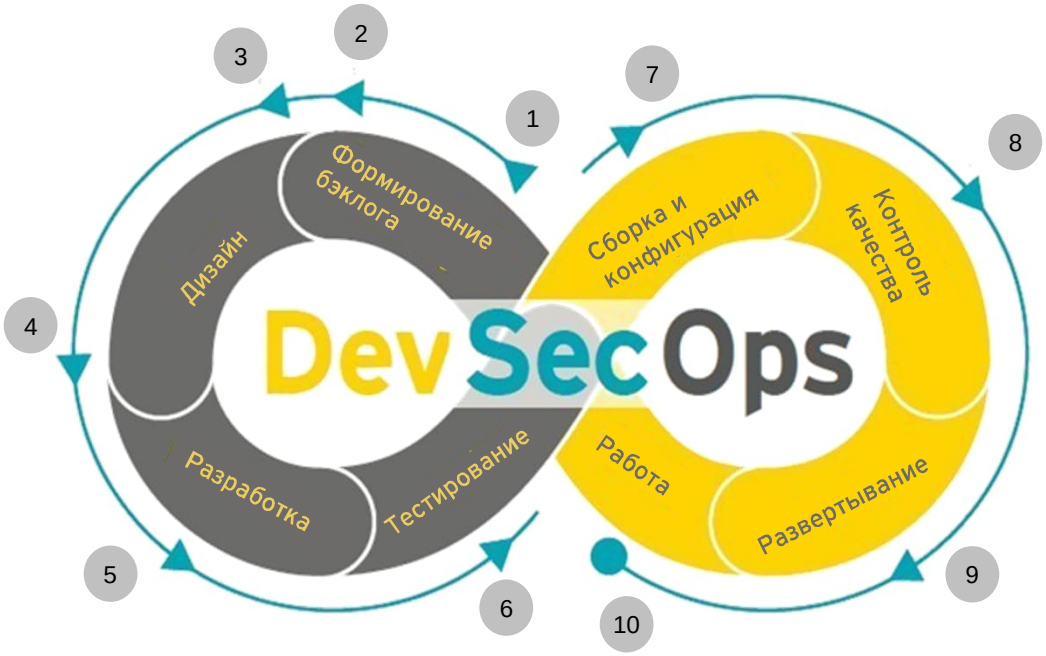
- Сканирование кода на безопасность осуществляется по инициативе разработчиков с помощью предоставляемого им сервиса.
- Результаты сканирования кода публикуются на общедоступном внутреннем дашборде, на котором разработчики могут акцептовать или отклонить выявленную уязвимость.
- Реализуется внешняя программа bug-bounty.

Компания 5
США
Разработка ПО и облачных сервисов

- Поддержка важности роли безопасности при разработке на уровне CEO.
- Автоматизирован процесс проверки следования практик безопасности на каждом этапе жизненного цикла.
- Security Engineering – отдельная структура, включающая в себя в том числе программистов-экспертов в кибербезопасности, помогает ставить процессы SDLC в новых командах.
- Гибкий баланс между зонами ответственности Security Engineer, разработчиками и другими участниками процесса, достигается за счет постоянной обратной связи.

Компоненты Sec интегрируются во все этапы выполнения процессов DevOps

1. Интеграция требований ИБ в требования к продукту
2. Идентификация и оценка рисков (evil user stories)
3. Анализ безопасности архитектуры
4. Контроль безопасности сторонних компонент
5. Статический анализ кода
6. Динамический анализ приложения
7. Применение стандартов безопасной конфигурации инфраструктуры/облака
8. Контроль безопасности при приемке
9. Непрерывный мониторинг
10. Тестирование на проникновение



Интеграция компонентов Sec в процесс DevOps осуществляется поэтапно с учетом уровня зрелости и культуры

1 Интеграция требований ИБ в требования к продукту
Разработка методики формирования требований безопасности

5 Статический анализ кода
Внедрение шлюза качества SAST и определение критериев его прохождения

6 Динамический анализ приложения
Внедрение шлюза качества DAST и определение критериев его прохождения

8 Контроль безопасности при приемке
Разработка методики контроля безопасности ПО

10 Тестирование на проникновение
Разработка регламента тестирования на проникновение и требований по устранению выявленных уязвимостей

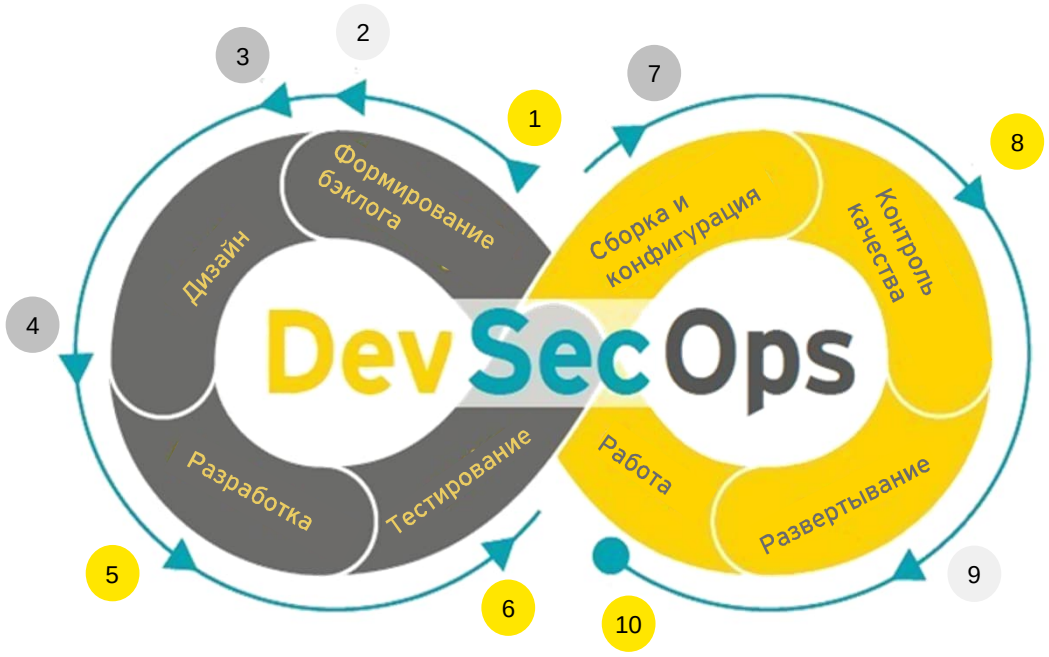
3 Анализ безопасности архитектуры
Разработка стандартных требований к безопасности архитектуры ПО

4 Контроль безопасности сторонних компонент
Внедрение шлюза качества SCA и определение критериев его прохождения

7 Применение стандартов безопасной конфигурации инфраструктуры/облака
Разработка стандартных требований к конфигурации среды

2 Идентификация и оценка рисков (evil user stories)
Разработка методики управления рисками информационной безопасности ПО

9 Непрерывный мониторинг
Внедрение процессов мониторинга и реагирования на инциденты ПО



Этап запуска Этап развития Этап совершенствования

Подход EY был успешно использован для улучшения процесса Secured Agile & DevOps в крупной организации

Область: Российская компания, ведущая самостоятельную активную разработку и развитие ПО

Задача: Оценка процесса Secured Agile & DevOps и разработка дорожной карты улучшений на 2 года

Этапы проекта	Описание этапа	Результаты этапа
Оценка	Оценка текущего процесса разработки ПО с использованием фреймворков BSIMM и SAMM по 12 параметрам	Определён базовый уровень для дальнейшего улучшения
Идентификация рисков	Выявление рисков текущего состояния и ключевых моментов для улучшения	Определена целевая модель процесса на основе наиболее критичных рисков текущего состояния.
Анализ несоответствий	Определение действий, необходимых для устранения пробелов в проблемных областях бизнеса Клиента.	Ряд инициатив и мероприятий разработан на основе несоответствий между двумя состояниями процесса.
Рекомендации	Анализ лучшего и худшего международного опыта, адаптация сценария реализации инициатив с учётом особенностей Клиента	Точные сценарии реализации определены и подробно описаны
Дорожная карта	Приоритизация основных активностей, установление взаимосвязей и планирование реализации	Разработана дорожная карта реализации инициатив, у Клиента есть полное понимание действий, которые необходимо выполнить для улучшения процесса.

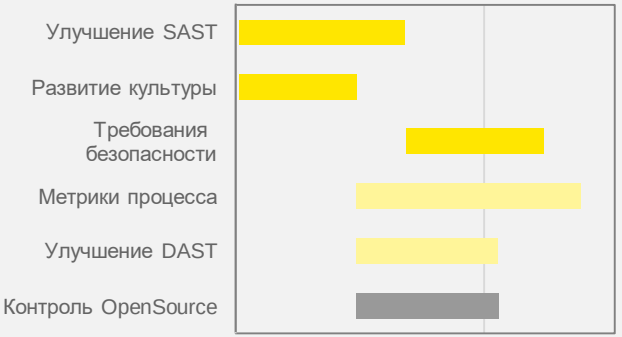
Пример результатов проекта по совершенствованию практик безопасной разработки

Оценка → Идентификация рисков → Анализ несоответствий → Рекомендации → Дорожная карта

Анализ несоответствий между текущим и целевым состояниями, разработка ряда активностей:

- Настройка инструментов автоматического тестирования безопасности (* AST)
- Внедрение риск-ориентированного подхода в процедурах тестирования безопасности
- Улучшение культуры обеспечения безопасности ПО
- Интеграция средств управления решениями с открытым исходным кодом
- Установка показателей качества процесса Agile & DevOps

Глобальные инициативы планируется реализовать в течение нескольких лет:

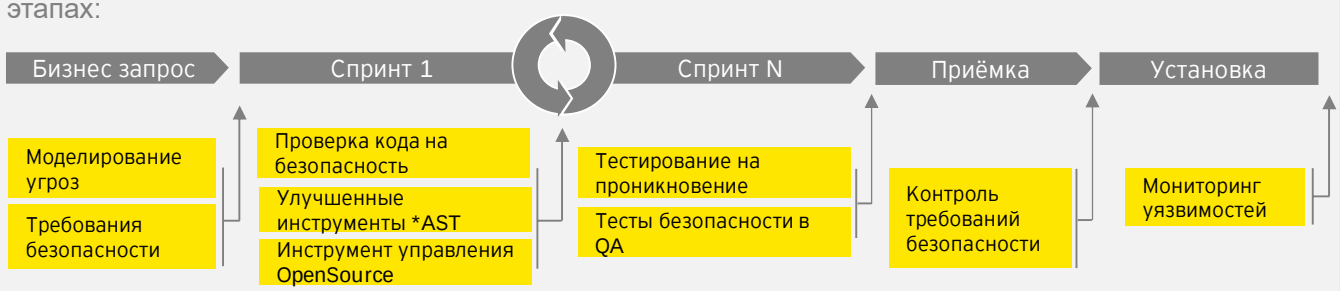


Сильные и слабые стороны процесса Secured Agile&DevOps определяются в соответствии с уровнями BSIMM и производится оценка рисков

Низкий уровень → Идентификация и уменьшение рисков

Высокий уровень → Действий не требуется

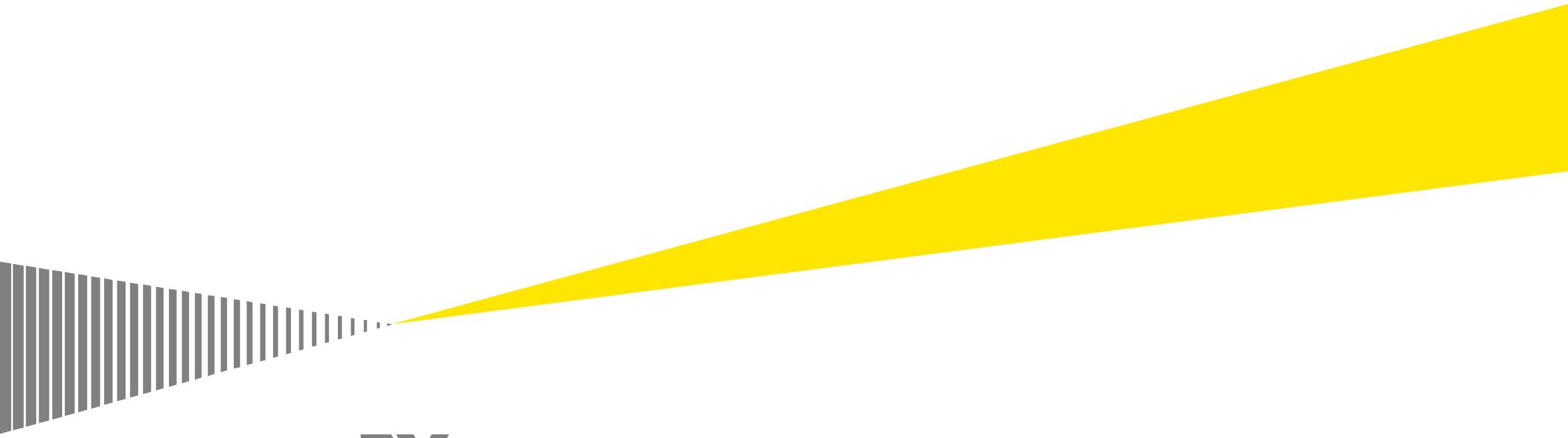
Общий процесс изменяется путем добавления действий по обеспечению безопасности на разных этапах:



Спасибо за внимание

Сергей Машошин
Менеджер

Тел: +7 (495) 755-9700
Sergei.Mashoshin@ru.ey.com



EY
Совершенствуя бизнес,
улучшаем мир