

КОМПАНИЯ  
ПРАКТИВ

АКТИВ.  
CONSULTING

# Внедрение процесса безопасной разработки прикладного ПО



---

**Александр Моисеев,**

Ведущий консультант по информационной  
безопасности АКТИВ.CONSULTING

О чем сегодня пойдет речь

---



**AKTIV.**  
CONSULTING

## Блок I

Подходы к безопасной  
разработке

## Блок II

Проект внедрения  
безопасной разработки

# Блок I

Подходы к безопасной разработке

# Драйверы внедрения процесса БРПО

Разработчики ПО и ПАК		
	Разработчики прикладного ПО	Разработчики СЗИ
Бизнес-цели	<p><b>Увеличение объемов продаж за счет:</b></p> <ul style="list-style-type: none"><li>• новых конкурентных преимуществ</li><li>• новые клиентские сегменты</li></ul> <p><b>Сокращение стоимости разработки за счет:</b></p> <ul style="list-style-type: none"><li>• стандартизации процессов разработки</li><li>• повышения стабильности релизов</li></ul>	
Комплаенс	<p><b>ПО для ФО</b> ГОСТ Р ИСО/МЭК 15408</p> <p>ПО для Атомной энергетики <b>ГОСТ Р 60880</b></p> <p><b>ПО для КИИ</b> Приказ ФСТЭК РФ №239</p>	<p><b>СЗИ</b> Приказ ФСТЭК РФ №55, №76</p>

# Существующие подходы к безопасной разработке ПО

---

## 01 Национальные стандарты и НПА

- 239 приказ ФСТЭК России
- ГОСТ Р 56939 (+проекты новых стандартов)

- 
- 76 приказ ФСТЭК России (выписка)
  - \*Методика ВУ и НДВ (2020)

*\*ограниченного распространения*

## 02 Гармонизированные стандарты

- ГОСТ Р ИСО/МЭК 27034xx «Безопасность приложений»
- ГОСТ Р ИСО/МЭК 15408xx «Общие критерии»

## 03 Международные практики и стандарты

- Microsoft SDL
- Cisco SDL
- NIST
- и др.

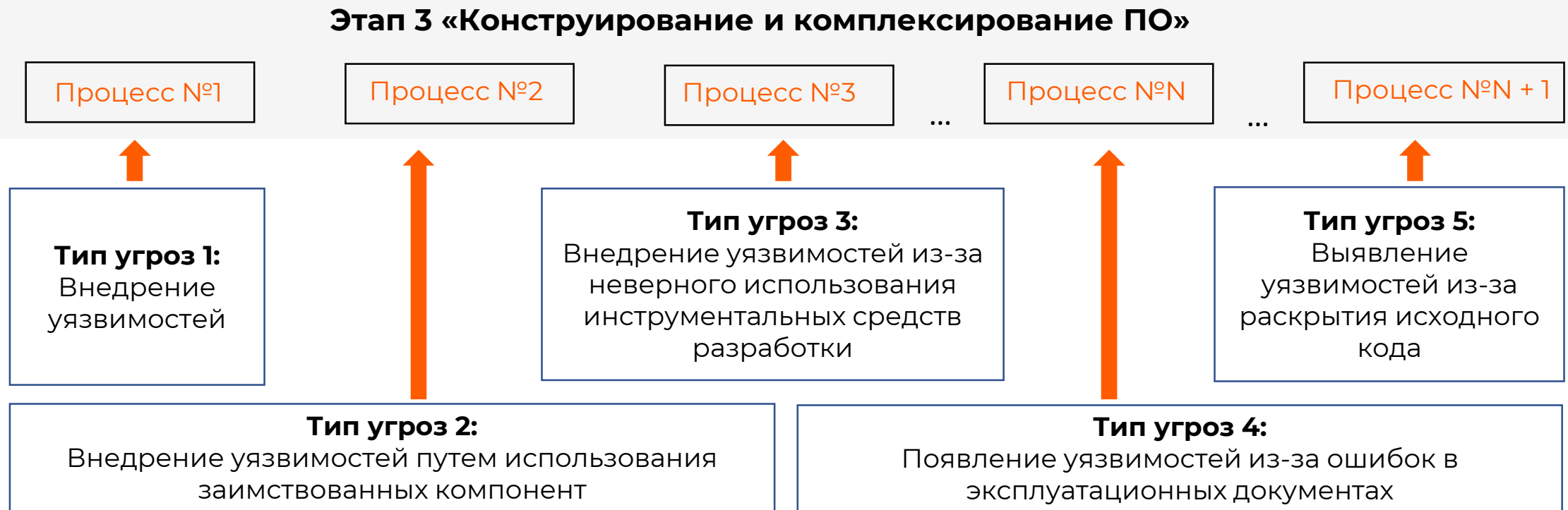
# Меры безопасности по ГОСТ Р 56939

Основные Процессы	<ol style="list-style-type: none"><li>1. Определение требований по безопасности</li><li>2. Анализ и моделирование угроз*</li><li>3. Уточнение проекта архитектуры ПО</li><li>4. Использование идентифицированных инструментальных средств разработки</li><li>5. Создание ПО на основе уточненного проекта архитектуры</li><li>6. Порядок оформления исходного кода</li><li>7. Статический анализ</li><li>8. Экспертиза исходного кода</li><li>9. Функциональное тестирование</li><li>10. Тестирование на проникновение</li><li>11. Динамический анализ</li><li>12. Фаззинг</li><li>13. Обеспечение целостности ПО в процессе передачи пользователю</li><li>14. Поставка пользователю эксплуатационных документов</li><li>15. Процедуры отслеживания и исправления ошибок и уязвимостей ПО в ходе ЖЦ</li><li>16. Систематический поиск уязвимости</li></ol>
Обеспечивающие процессы	<ol style="list-style-type: none"><li>17. Процедуры уникальной маркировки версий ПО</li><li>18. Использование системы управления конфигурацией ПО</li><li>19. Защита от НСД к элементам конфигурации</li><li>20. Резервное копирование</li><li>21. Регистрация событий, связанных изменения элементов конфигурации</li><li>22. Периодическое обучение сотрудников и анализ программы обучения</li></ol>

\*меры соответствуют требованиям 239 приказа ФСТЭК России

# Мера 2. Анализ и моделирование угроз

Пример рассмотрения типов угроз  
(на примере этапа 3):



Рекомендуем: Перечень угроз ПО в соответствии с ГОСТ Р 58412. Методика моделирования угроз ФСТЭК (от 2021)

# Мера 7. Статический анализ кода

- Выбор инструментов SAST
- Классификация ошибок
- Классификация применяемых методов
- Сопоставление типов предупреждений анализатора списку критических ошибок

Пример выявленной анализатором проблемы утечки из-за незакрытого потока:

```
Information about the snapshot | Source code | Show table | Confirmed | Unspecified |  
/.build/app/src/main/java/.../VersionController.java  
27 private String readVersionFromFile() {  
    Confirmed Unspecified Fix required HANDLE_LEAK The handle 'variablesStream' was created at VersionController.java:28 by calling  
    function 'java.lang.ClassLoader.getResourceAsStream(String)InputStream' and lost at VersionController.java:35.  
28     InputStream variablesStream = this.getClass().getClassLoader().getResourceAsStream("git.properties");  
19     private final String version = readVersionFromFile();  
20 }
```

**Проблема!** Не все ЯП поддерживаются инструментами статического анализа



# Мера 11. Динамический анализ кода

Пример фрагмента отчета о покрытии модульных тестов с нулевым покрытием:

- Определение функций, принимающих на вход внешние данные
- Инструментирование кода санитайзерами / отладочными аллокаторами
- Проведение модульного и регрессионного тестирования
- Сбор покрытия
- Анализ результатов
- ГОСТы по тестированию: 56920, 56921, 56922

## webapp

Element	Missed Instructions	Cov.	Missed Branches	Cov.
# [REDACTED]		0 %		n/a
# <a href="#">ice.admin.settings.counterpart</a>		0 %		n/a
# <a href="#">ice.common.logging.event.mq</a>		0 %		0 %
# [REDACTED]		0 %		n/a
# [REDACTED]		0 %		n/a
# <a href="#">ice.admin.role.authority</a>		0 %		n/a
# <a href="#">figuration.websocket</a>		0 %		0 %
# <a href="#">ice.verifier.check.scenario.instance.counterpart</a>		0 %		n/a
# <a href="#">ain.services.publisher.mq</a>		0 %		0 %
# <a href="#">ice.admin</a>		0 %		0 %
# [REDACTED]		0 %		n/a
# <a href="#">ice.common.settings.mq.ldap</a>		0 %		0 %
# <a href="#">ice.common.settings.mq.general</a>		0 %		0 %
# <a href="#">ice.verifier.check.scenario.instance.transfer</a>		0 %		0 %
# <a href="#">ice.verifier.check.routine</a>		0 %		0 %
# <a href="#">ice.verifier.check.stage.instance</a>		0 %		0 %
# [REDACTED]		0 %		n/a
# [REDACTED]		0 %		0 %
# [REDACTED]		0 %		0 %
# <a href="#">ice.verifier.check.report</a>		0 %		0 %
# <a href="#">figuration.db</a>		0 %		n/a
# [REDACTED]		0 %		0 %

# Важное по первому блоку

- 1 Драйверы внедрения БРПО:
  - Регуляторные требования к заказчикам вашего ПО
  - Внутренняя бизнес-потребность
- 2 Новые клиентские сегменты, конкурентное преимущество
- 3 Сокращение стоимости разработки

# Блок II

## Проект внедрения процесса БРПО

# Пример проекта по внедрению БРПО

---

## 01 этап Обоснование

- Определение целей проекта
- Определение задач проекта
- Определение выгод от реализации проекта

### Результат этапа:

ТЭО и принятое на его основе решение руководства

## 02 этап Аудит

- Обследование текущих бизнес-процессов разработки
- Формирование целевого состояния процессов
- GAP-анализ
- Разработка организационных и технических мер

### Результат этапа:

Подготовка перечня организационных и технических мер

# Пример проекта по внедрению БРПО

## 03 этап Разработка дорожной карты

- Ранжирование организационных и технических мер
- Разработка «Плана перехода...»
- Разработка «Пояснительной записки...»

### Результат этапа:

подготовка  
«Дорожной карты...»

## 04 этап Внедрение процессов

Внедрение процессов и мер, которые включают:

- Организационные меры (ОРД, ЛНА, обучение)
- Технические меры (SAST, DAST, Fuzzing, пентест, code review, КЦ)
- Обеспечивающие меры (менеджмент БРПО и политики лицензирования)

### Результат этапа:

функционирующие  
процессы БРПО

## 05 этап Контроль

Проверка полноты и достаточности внедренных мероприятий

### Результат этапа:

свидетельства  
функционирования БРПО

# Важное по второму блоку

**01** Необходимо выбрать единую точку ответственности (т.е. руководителя проекта)

**02** Собрать команду проекта: ИТ, ИБ, Бизнес

**03** Определить этапы проекта и промежуточные результаты

**04** Оценить и запланировать ресурсы (время, специалистов, финансы)

# В качестве резюме

- 01 С высокой вероятностью вашим клиентам потребуется предоставление свидетельств БРПО, иначе возможен отказ от продукта
- 02 Основой для внедрения БРПО можно использовать отечественную регуляторную базу – серию ГОСТ Р 56939\*
- 03 Оптимальное разделение ролей: техническое лидерство должно быть за разработкой, организационное лидерство за ИБ

\*Требования ГОСТа совпадают с 239 Приказом ФСТЭК России, это будет важно для заказчиков, которые будут использовать ваше ПО



**Александр Моисеев**

Ведущий консультант  
по информационной безопасности

[moiseev@aktiv.consulting](mailto:moiseev@aktiv.consulting)

