# Quality Gate: повышение оценки защищенности для безопасной разработки ПО

настоящие возможности

росбанк 30 лет

# Спикер

# Илья Шмаков

Заместитель начальника отдела Информационной Безопасности управления рисками

- **Аналитик рисков** ИБ, **сертифицированный** DASA DevOps Product Owner, devops-инженер
- **Более** 8 лет работы в разработке ПО в части **DevSecOps**
- Работает **в сфере обеспечения ИБ** для продуктовой разработки в **BI, E-Commerce, Supply Chain, Cryptocurrency, GameDev**
- Проработал **двухэтапную защиту multisig** с учетом распределенного резерва согласно **эскроу-схемы** при проверке **горячего хранения** для Stable-коинов как **отдельного продукта**
- **Основатель и лидер** сообщества **FinDevSecOps** на базе **АФТ** и **MOEX** (при взаимодействии с **ЦБ**)
- **Ведет** курсы на платформе **InSeca.tech**
- **Руководит группой** офицеров информационной безопасности, развивает **DevSecOps** в банке

росбанк

# Quality Gate

## поставка продукта

**1**

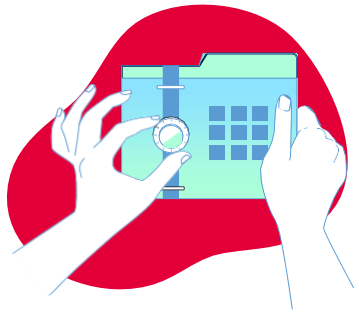
### Это критерии качества безопасности поставки

Цель в конечной безопасности продукта, - обучая команды паттернам безопасной разработки, вовлекая в вопросы ИБ и приравнивая задачи безопасности к дефектам кода

**2**

### Смысл в превентивном контроле CI/CD Pipeline

В конкретный момент времени, акцентируясь на Shift-Left подходе



**3**

### Критерии снижают риски

Внимание на имеющихся уязвимостях, не блокируя бизнес-задачи и сроки, учитывая особенности обеспечения безопасности продукта на стадии проработки бизнес-концепции

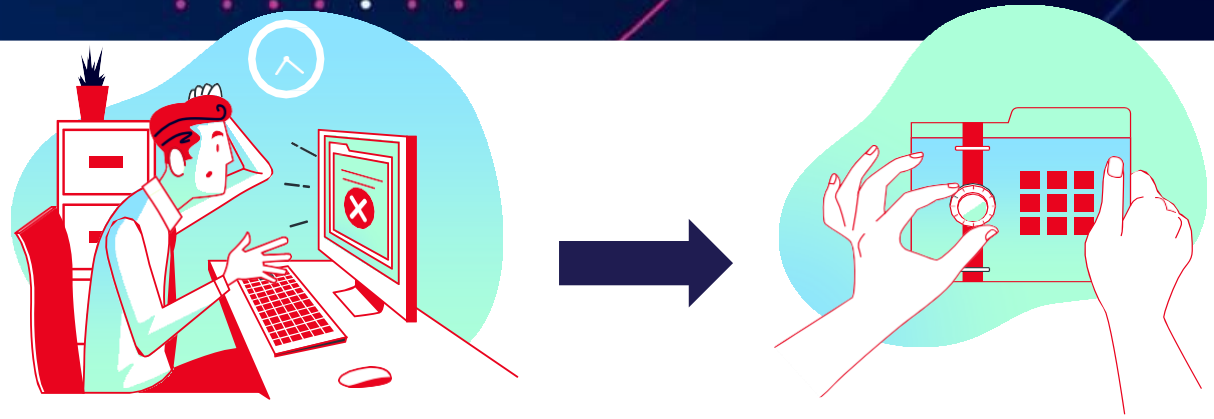
росбанк

# Что даёт команде?

1. Повышение уровня безопасности поставок и снижение рисков ИБ
2. Снижение Time to Market, осведомленность в ИБ
3. Укрепление Service Relationship Management
4. Стандартизация и оптимизация процессов
5. Позволяет ориентироваться на метрики ценности
6. Мотивация по Д. Пинку: цель, автономность, мастерство
7. Развитие практик: BSIMM, OWASP SAMM
8. Обучение & расширение Т-профиля компетенций до Ш-профиля
9. 


росбанк

# Зачем в DevOps Quality Gate?



## Проблема

1. Интеграция принципов **Security** в построении **SDLC** с долей отказа
2. Зрелость продукта для бизнеса, - «костыльность» в решениях
3. Применение практик, не относящихся к **Agile-Lean**
4. Ориентация на контракт и срок
5. Принцип **legging** метрик
6. Присутствие стоп-факторов при поставках (бизнес, риски, непонимание ценности)

## Решение

1. Ориентация на **ценность бизнеса**, подготовка решений для нужд команд с упором на ИБ
2. Снижение числа метрик выживаемости (компоненты, заказчики, команды)
3. Ввод метрик производительности, обязательности QG
4. Приоритезация на ведущие метрики, нивелирование **legging**
5. **Контроль** pipeline с мерами ИБ

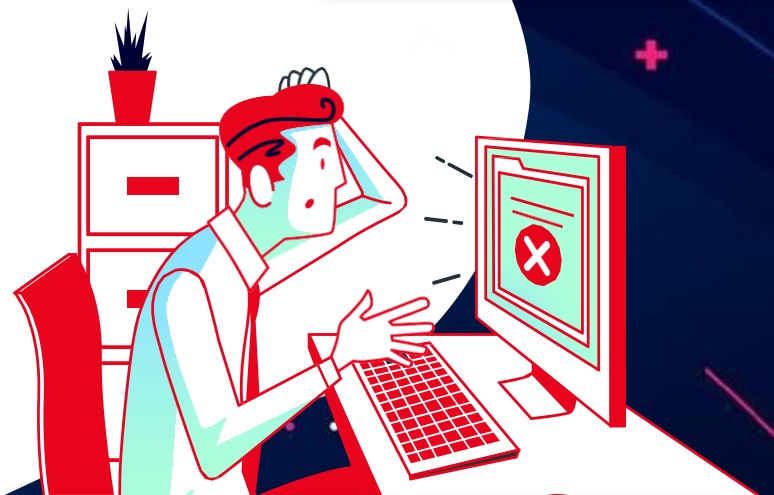
## Принципы

- **Равноценность** дефектов продукта и уязвимостей
- **Автоматизация** проверок ИБ
- Выполнение требований ИБ в более **короткие** сроки
- **Соответствие** требованиям законодательства
- Совершенствование ИБ продукта за счет **небольших** изменений
- Наполнение активностей понятным **собственным** смыслом
- Механизм **остановки** публикаций (допущения/ограничения)
- Риск- и бизнес-ориентированный подход при разработке
- Выявление «**исторических**» недостатков разработки
- Определение «**узких**» мест
- Развитие **BSIMM-практик** для команд с уклоном в их специфику

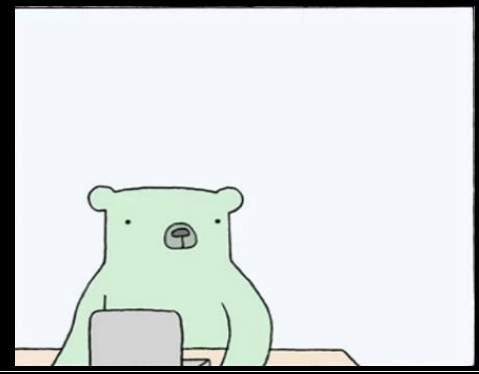
росбанк

## Сервис QG

- Учет дублирования уязвимостей и сверку наличия уже имеющихся задач,
- Feature Toggle (Red Button) с переключением для блокирование поставки в IRM,
- Учет ASA, SSF, критичности хранимых, обрабатываемых данных, компенсирующих мер, Risk Analysis, инфраструктуры сервисов, репозитории проектов, Contributor и флагов критичности ИС,
- Дефекты кода приравниваются к уязвимостям - STLC,
- Механизм допущений/ ограничений исходя из Risk Analysis (снижение до целевого уровня, либо повышение уровня риска),
- Группировка Merge/ Pull Reguest по Contributor.

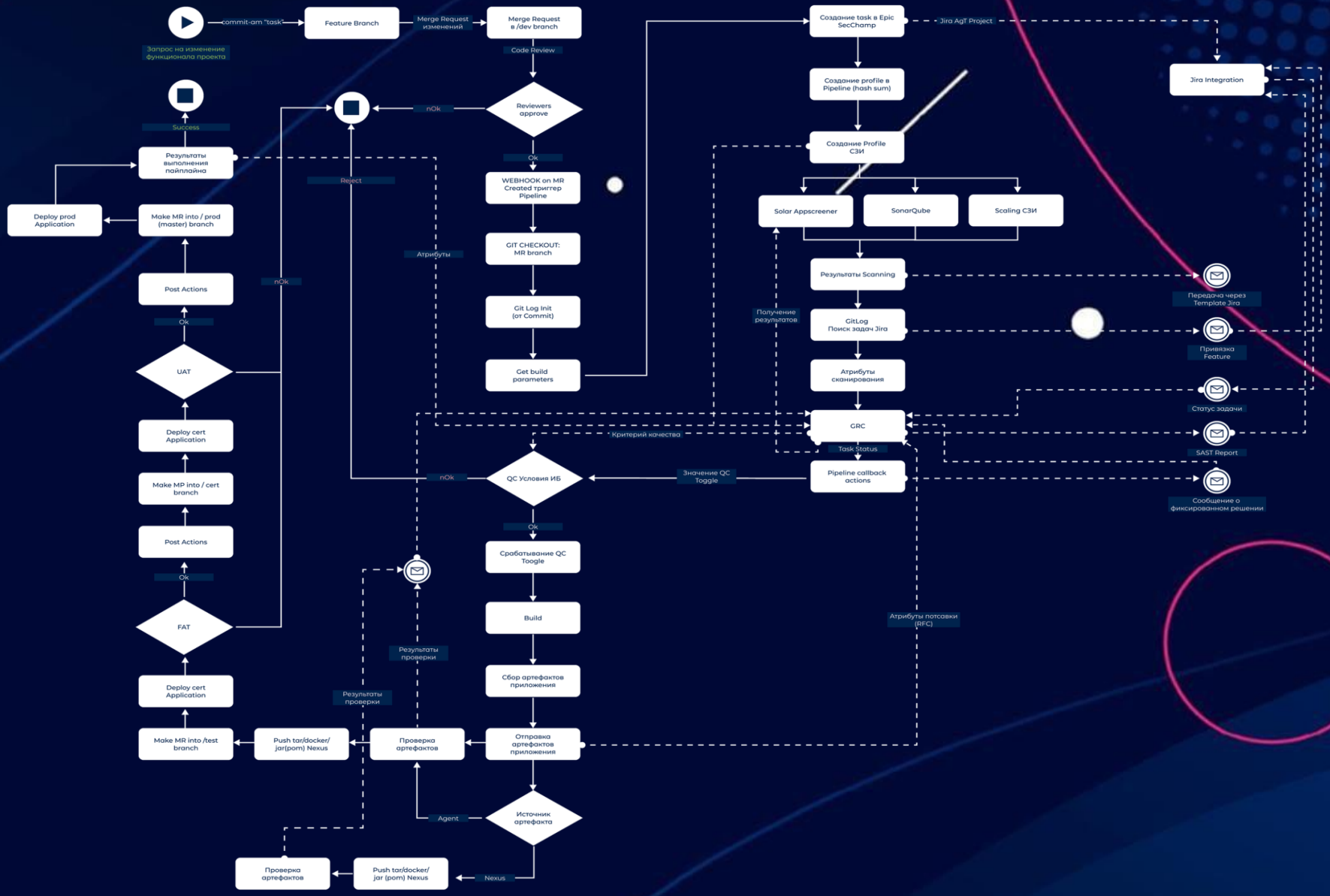


росбанк

| Этап DevSecOps | Описание |
|---|---|
| **Формирование требований** (Plan) | ▪ Регулярный анализ компетенций работников и технического долга команд разработки в части ИБ<br>▪ Обучение работников, для получения ими необходимых компетенций по ИБ<br>▪ Определение метрик для оценки эффективности процесса безопасной разработки<br>▪ Сбор и анализ требований к разрабатываемому ПО, а также анализ потенциальных рисков |
| **Проектирование** (Design) | ▪ Анализ поверхности атаки<br>▪ Моделирование угроз<br>▪ Проектирование ПО с учетом требований ИБ и выявленных в ходе анализа слабых мест |
| **Разработка** (Code) | ▪ Использование командами разработки стандартов безопасного программирования<br>▪ Анализ кода на наличие секретов<br>▪ Используются инструменты управления секретами, а также осуществляются проверки на соответствие требованиям ИБ, образов контейнеров, сборок кода<br>▪ Анализ кода для выявления:<br>- Скрытых уязвимостей<br>- Логических ошибок<br>- Ошибок построения архитектуры<br>- Дефектов реализации<br>- Неиспользуемых частей кода<br>- Закладок<br>- Общего уровня качества кода<br>- Отклонения от стандартов<br>- Компонентный анализ и контроль использования компонентов с открытым исходным кодом для контроля<br>- Известных уязвимостей в open-source<br>- Лицензионных ограничений<br>- Новых уязвимостей в используемых компонентах<br>  |
| **Сборка** (Build) | ▪ Сбор артефактов<br>▪ Загрузка сборок в хранилище Nexus и их анализ<br>▪ Получение hash sum и слепка profile конфигурации образа для сверки достоверности тестируемой конфигурации при Continuous Delivery (в рамках QG) |
| **Тестирование** (Test) | ▪ Динамический анализ кода для выявления<br>▪ Недекларированного поведения приложения<br>▪ Утечек памяти<br>▪ Перерасхода ресурсов<br>▪ Ошибок аутентификации<br>▪ Фаззинг-тестирование<br>▪ Функциональное тестирование ИБ на соответствие требованиям, заложенным на этапе проектирования<br>▪ Подпись исполняемого кода для контроля изменений |
| **Выпуск и развертывание** (Release, Deploy) | ▪ Проверка целостности компонентов ПО<br>▪ Проверка подписи всех исполняемых/бинарных файлы, включенных в релиз<br>▪ Проверка компонентов и образов на соответствие требованиям ИБ<br>▪ Тестирование на проникновение<br>▪ Проверка соответствия ПО требованиям регуляторов в части ИБ<br>▪ Подтверждение, что все дефекты с высоким уровнем критичности, включая уязвимости, связанные с ресурсами ИТ-инфраструктуры, устранены |
| **Мониторинг и реагирование** (Operate, Monitor) | ▪ Управление конфигурациями и изменениями, которые влияют на уровень безопасности ПО<br>▪ Мониторинг угроз и реагирование на инциденты ИБ<br>▪ Регулярные проверки на соответствие требованиям ИБ<br>▪ Формирование перечня задач по доработке ПО при следующей итерации изменений |

росбанк

# Схема интеграции QG




росбанк

| Направление | Разделение | Тип | Класс | Вендорское ПО | Open Source |
|---|---|---|---|---|---|
| **Инструменты DevSecOps** | **AppSec** | **SAST** | **Статический анализатор** | Solar AppScreener (SAST), PVS Studio, Application, Inspector, GitLab SAST, CheckMarx, Coverity, Klockwork, Fortify, Sonar Qube | SVACE, Bandit, Terrascan, CodeQL, Chechov, Weggli, Joern, MATE, Semgrep, tfsec, kics, bearer, njsscan, clj-holmes, Dagda, Blackduck, Osquery, PMD, Retire.js |
| | | **Taint Analysis** | **Анализ поверхности атаки** | PVSStudio, ИСП РАН Natch, Qodana | pyre-check, psalm, Triton, find-sec-bugs, pyt, airbus-seclab (bincat), polytracker, FlowDroid, Ponce |
| | | **IAST** | **Интерактивный анализатор** | AppInspector, Synopsys Seeker, Veracode Interactive Analysis, Microfocus Fortify WebInspect, Checkmarx CxIAST, Contrast AssessHdiv Detection (IAST), CAST, Codename SCNR (Arachni) | DongTai, Contrast |
| | | **SCA** | **Анализатор** | Solar AppScreener (SCA), Code Scoring, Сканер уязвимостей Ya Cloud, AppSecTrack, Spectral, Snyk, Azul, qwiet, Black Duck, Debricked, Mend SCA | Dependancy Track, Dependency Check, Grype, Speed, OpenSCAP |
| | | | **SBOM** | CycloneDX (CDXGen), Syft, FOSSA, Spectral, Codenotary, Cybeats SBOM Studio, endorlabs, rezilion | Trivy, Jit, spdx-sbom-generator |
| | | **BCA** | **Бинарный анализатор** | BinaryNinja, IDAPRO, Procyon, Gidra (Hydra) | Orbit, OlydDBG, JADX, CFR, JDCore, Fernflower, ILSpy, DNSpy, Dotpeek, BinSide |
| | | **DAST (вкл. фаззинг, network protocol)** | **Динамический анализатор** | Solar AppScreener (DAST), Application Inspector, PT Black Box, OWASP ZAP, BurpSuite, ИСП РАН Crusher, Frida, HCL Security AppScan, Synopsys Managed DAST, Veracode Dynamic Analysis, Burp Suite, InsightAppSec, Invicti, Indusface WAS, Acunetix | Cifuzz, sec-helpers, ZAP, GoLismero, Nikto, W3AF, Zed Attack Proxy, Wapiti, OpenVAS, Nuclei, Boofuzz, Fuzzer, JBroFuzz, Jscrambler, Sydr/Sydr-fuzz |
| | | **MAST** | **Анализатор mobapp** | Стингрей, Objection, App-Ray, Codified Security, StaCoAn, Quixxi, ImmuniWeb, ostorlab | Cycript, SandDroid, QARK, dexcalibur, RMS, MitmProxy |
| | | **RASP** | **Анализатор при runtime** | Fortify, Sqreen, Jscrambler, Imperva, Signal Sciences, Appsealing, Appdome, Liapp, Falco | OpenRASP, hdiv, Baidu’s OpenRASP |
| | | **API** | **Анализатор API/ WAF** | APIsec, Akto, appsentinels, brightsec, stackhawk, fastly, PT AF, Fortify Webinspect | 42Crunch, LAST (latio), API Security Empire, UsageFinder, Astra, WSO2, SSL Kill Switch 2, Probely |
| | **Vulnerability management (на этапе разработким, включая TMS, STLC, codecoverage, MOBsf, Third-party management)** | | | DefectDojo, AppSec.Hub, Qualys, Tenable.sc, Tripwire, IP360, Faraday, ThreatMapper, anchore, devprom, ALM, Nox, GFI LanGuard, Hardening, Profiscope CodeScoring | Reliability, ThreatPlaybook, osv, Allure, TestOps, start-req, gcov, bullseye, llvm-cov, JaCoCo, BDD Automated Security Tests, Checkstyle, Chef, Codacy, SourceClear |
| | **ASTO (включая разные решения)** | | | InfluxDB, spectral, synopsys, K2 Security Platform, Alert Logic, Pulumi, Puppet, Signal Sciences, Tenable | betterscan-ce, sechub, Clouddefense, soos, Find Security Bugs, Phabricator, Sysdig |
| | **Container Security (включая VM, proxy, VPN)** | | | Luntry, PT Container Security, Kaspersky Container Security, Bastion, cloudpods, Snyk Container, Sonatype Nexus Lifecycle, JFROG, Harbor, Quay, CloudSec, Сфера, Aqua Security, Check Point CloudGuard, Cloudflare, GitLab Container Scanning, Prisma Cloud | js-x-ray, cilium, Calico, Clair, Anchore, OpenSCAP, Falco, gitleaks, openvpn |
| **Инфраструктура** | **Secrets Management (Хранение образов, docker registry, Artifact Signature)** | | | Hashicorp Vault, BitWarden, Consul, Senhasegura, StrongHold, CoSign, OPA Gatekeeper, Spectral Secret Scanner, AWS Secrets Manager, Cloud KMS, Azure Key Vault, Confidant, Keywhiz, Knox, Strongbox, CyberArk Conjur, Yandex Lockbox | Spring Cloud Config Server AppArmor, detect-secrets, Gerrit, git-all-secrets, git-secrets |

# Подход к управлению рисками

## в процессах Quality Gate

1. **Анализ взаимосвязи безопасности данных с бизнес-ценностью продукта** и устранением уязвимостей

2. **Разработка корректируется на ценности продукта** конечному пользователю исходя из повышение безопасности, вследствии устранения уязвимостей. Влияют условия принятия рисков ИБ и возможность их минимизации

3. **Разработка с учетом выполнения мер по минимизации значимых рисков** перед выпуском в промышленную среду (PRE-условия) или после с последующим контролем (POST-условия)

4. **Quality Gate контролирует технологически конфигурационные файлы**

## Повышение эффективности управления рисками и уязвимостями

росбанк

# Сервис Quality Gate 2.0 : **CMIB_FLEXCUBE**

Created Date: 29/11/2022 00:10 Last Updated: 21/12/2023 14:04

0 of 0 Completed

## ▼ QUALITY GATE - RED BUTTON

**Quality Gate Toggle (переключение):**

**Auto Decision (Flags):** Rejected

**Время релиза поставки:** 21/12/2023 14:02

**Идентификатор Pipeline:** 61271995-866d-4629-9912-9b73af8644ef

**QG Flags:**

- Флаг 1: Оценка безопасности Solar не пройден
- Флаг 2: Риск фактор увеличился
- Флаг 3: Крит фактор увеличился
- Флаг 5: Плотность критических и высоких уязвимостей увеличилась
- Флаг 6: Количество критических уязвимостей не увеличилось
- Флаг 7: Количество критических и высоких уязвимостей увеличилось
- Итого Флаги для Minor поставки: F1:[0] F2:[0] F3:[0] F5:[0] F6:[1] F7:[0] F8:[1]
- Оценка безопасности GRC:[0.05] из 1
- Quality Gate GRC не пройден

## ▼ PROJECT

**Группа проектов**

| Группа проектов | Team | uuid |
|---|---|---|
| CMIB | CMIB | 16f3266a-99c8-4224-a0f4-0d836396aed1 |

**Наименование Проекта:** CMIB_FLEXCUBE

**Комментарий команды:** Пилот (Демо)

**Git Init:** ssh://git@bitbucket.gts.rus.socgen:7999/flex/core.git

**Проект есть в GitLab:** No

**Репозиторий:** Https://bitbucket.gts.rus.socgen:7999/flex/core.git

**Критичность ИТ-системы:** Высокая

**ИТ Системы**

| Application Name | Application Manager | Application Owner | Application Type | Business Domain | PCI DSS |
|---|---|---|---|---|---|
| Оракл ФлексКуб | VDOVENKO, Yuriy | KOKORINA, Mariya | | | No |

**Agile Team**

| Business Unit | Major Applications | IS Officer | Security Champion (Group) | Percent Assesment Score | Risk analysis (SSF 2) (Calculated) | Jira link |
|---|---|---|---|---|---|---|
| CMIB | | GONCHAROV, Sergey | VDOVENKO, Yuriy | 0 | | |

**Исключения ИБ**

| Exception ID | Application | Business Unit | Заявитель | Офицер ИБ | Название исключения | Критичность исключения | Статус Исключения | Установленный срок действия исключения |
|---|---|---|---|---|---|---|---|---|
| No Records Found | | | | | | | | |

## ▼ QUALITY GATE - RED BUTTON

**Quality Gate Toggle (переключение):**

**Auto Decision (Flags):** Approved

**Время релиза поставки:** 21/12/2023 14:10

**Идентификатор Pipeline:** cf985faf-e86a-4262-8d62-291afc03d2f0

**QG Flags:**

- Отсутствует -10 скан
- Десять поставок еще не прошло, F8 не активен
- Флаг 1: Оценка безопасности Solar не пройден
- Флаг 2: Риск фактор не увеличился
- Флаг 3: Крит фактор увеличился
- Флаг 5: Плотность критических и высоких уязвимостей не увеличилась
- Флаг 6: Количество критических уязвимостей не увеличилось
- Флаг 7: Количество критических и высоких уязвимостей не увеличилось
- Итого Флаги для Minor поставки: F1:[0] F2:[1] F3:[0] F5:[1] F6:[1] F7:[1] F8:[1]
- Оценка безопасности GRC:[0.6] из 1
- Quality Gate GRC пройден

▼ SOLAR APPSCREENER LAST SCAN

**SAST Last Score:** 1.9

**Дата сканирования:**

**Идентификатор SAST (Solar Appscreener) :** a31d2629-ae4c-4588-b467-4d21cc6dbae9

**Используемые языки:** PL/SQL

**Сканирования Solar Appscreener**

View Less

| Scan ID | Дата сканирования | Номер | Тип поставки | Языки программирования | Общая плотность | Плотность критических и высоких | Риск-фактор | Крит-фактор | Динамика уязвимостей в кодовой базе | Количество строк кода | Критических уязвимостей | Высоких уязвимостей | Средних уязвимостей | Низких уязвимостей | Количество уязвимостей | Оценка безопасности | QG |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 7c4f55e9-e0af-4633-a140-2e2961f19de6 | 08/12/2023 16:45 | 10 | Минорная | JavaScript PL/SQL T-SQL | 372.56 | 620.93 | 60.0 % | 40.00 % | 38 | 9314 | 10 | 5 | 9 | 1 | 25 | 1.9 | |
| 8aeec46a-890e-424b-9c71-f5901a1ef04b | 08/12/2023 16:34 | 9 | Минорная | JavaScript PL/SQL T-SQL | 386.50 | 662.57 | 58.3 % | 41.67 % | 38 | 9276 | 10 | 4 | 9 | 1 | 24 | 1.9 | |
| 802453a7-32a5-4185-b005-665f7f3c1959 | 08/12/2023 16:30 | 8 | Минорная | JavaScript PL/SQL T-SQL | 347.78 | 552.35 | 63.0 % | 37.04 % | 75 | 9390 | 10 | 7 | 9 | 1 | 27 | 1.9 | |
| bc7af2e3-3657-474b-998b-ed8b58f8bcc7 | 08/12/2023 16:25 | 7 | Минорная | JavaScript PL/SQL T-SQL | 358.27 | 582.19 | 61.5 % | 42.31 % | 1 | 9315 | 11 | 5 | 9 | 1 | 26 | 1.8 | |
| d1f661b7-9a5e-4133-ab97-307fe35e2e1f | 08/12/2023 16:17 | 6 | Минорная | JavaScript PL/SQL T-SQL | 423.23 | 775.92 | 54.5 % | 31.82 % | 1 | 9311 | 7 | 5 | 9 | 1 | 22 | 2.2 | |
| e4fab8b2-3d7a-43bf-9f92-52f4598dfe7f | 08/12/2023 16:12 | 6 | Минорная | JavaScript PL/SQL T-SQL | 423.23 | 775.92 | 54.5 % | 31.82 % | 1 | 9311 | 7 | 5 | 9 | 1 | 22 | 2.2 | |
| bb64d0c7-a7f6-4bbb-b638-9181e09df278 | 08/12/2023 15:59 | 5 | Минорная | JavaScript PL/SQL T-SQL | 404.87 | 716.31 | 56.5 % | 34.78 % | 0 | 9312 | 8 | 5 | 9 | 1 | 23 | 2.1 | |
| a0d85df8-c175-4d8a-aa3c-af7437af7f88 | 08/12/2023 15:38 | 4 | Минорная | JavaScript PL/SQL T-SQL | 404.09 | 714.92 | 56.5 % | 34.78 % | 0 | 9294 | 8 | 5 | 9 | 1 | 23 | 2.1 | |
| d0aa8d19-b5c3-4f87-83d6-54c433f8296b | 08/12/2023 15:33 | 3 | Минорная | JavaScript PL/SQL T-SQL | 403.30 | 713.54 | 56.5 % | 34.78 % | 0 | 9276 | 8 | 5 | 9 | 1 | 23 | 2.1 | |

росбанк

# Наши рекомендации

1

- Правильно выстроенный процесс важнее инструментов
- Не внедряйте всё и сразу — начните с того, что уже есть, и двигайтесь исходя из потребности: тип интеграции, назначение, бизнес-процесс
- Функциональные дефекты и уязвимости равны

2

- Выращивайте в своих командах **Security Champions** и задействуйте в процессе разработки — это повысит поток ценности увеличивая безопасность продукта,
- Помните, что качество продукта — это общая цель
- Важнее люди, а не процесс
- Разделяйте влияние обособленных команд

3

- Автоматизируйте по максимуму – **Infrastructure-as-a-Code** to **Everything-as-a-Code**
- Выбирайте процессы и инструменты, подходящие именно для вашего продукта
- Делайте упор на метрики здоровья и зрелости продукта, - нивелируйте **legging**

росбанк

# Вопросы & ответы

настоящие возможности

росбанк 30 лет

# Secure Software Development Life Cycle (SDLC) / DevSecOps ToolChain


**Статический анализ (SAST)**
- Скрытые уязвимости
- Логические ошибки
- Ошибки архитектуры
- Дефекты реализации
- «Мёртвый» код
- «Закладки»
- Общий уровень качества кода и отклонения от стандартов

**Контроль Open Source (OSA)**
- Контроль библиотек в периметре
- Известные уязвимости в Open Source
- Лицензионные ограничения
- Мониторинг новых уязвимостей в используемых компонентах

**Динамический анализ (DAST)**
- Недекларированное поведение
- Утечки памяти
- Перерасход ресурсов
- Ошибки аутентификации

**Тестирование на проникновение**
- Back doors, вредоносный код
- Ошибки конфигурации
- Несоответствие политикам ИБ компании

Requirements Gathering - Reviewed/Understanding

- Address Technical Security Debt
- DevSecOps Metrics (DASA DevOps)
- Threat Modeling
- Security Tool Training

Test Analysis, Planning & Test Design

IDE Security Plug-Ins

Test Execution (Manual/Automation) & Defect Reporting & Retesting

Test case writing and Auto-test Development

- Chaos Monkey
- Input Fuzzing
- Integration Test
- OWASP ZAP
- Software Signing

**01.PLAN (Adapt)**
- Threat Modeling
- Cases
  - Misuse Cases
  - Abuse Cases

**02.CODE (Create)**
- Analyzer
  - SAST
  - SCA
- Configuration Drift

**03.BUILD (Verify)**
- Analyzer
  - DAST
  - SCA
  - OSA
  - BCA
- Environment
  - Consistency in Environment
- Configuration Drift

**04.TEST (Preprod)**
- Analyzer
  - IAST
  - SAST
  - DAST
- Tests
  - Performance Testing
  - Infrastructure Cloud Testing

**05.RELEASE (Prevent)**
- Analyzer
  - SCA
  - BCA
- Environment
  - Configuration Drift
  - Consistency in Environment

- Signature Verify
- Integrity Checks
- Defense In-Depth Measure

**06.DEPLOY (Detect)**
- Secrets Management
- Tests
  - Compliance Test
- Tools
  - RASP
  - PKI
  - WAF
  - IDS
  - BCA

Final Testing and Preparing Final Test Executive Summary Report Preparation

- UEBA/Network Monitoring
- Penetration Testing
- Security Orchestration
- WAF Shielding
- Obfuscation

**07.OPERATE (Respond)**
- Secrets Management
- Tests
- Tools
  - RASP
  - PKI
  - WAF
  - IDS
- Resilience Testing
- Chaos Testing

- Dev Consumable
- Correlated Vulnerability Analysis
- TI STIX TAXII

**08.MONITOR (Predict)**
- Monitoring
  - Security Monitoring
  - Application Monitoring
- Tests
  - Performance Testing

Maintenance Testing

- Security Technical Debt
- Modify Incident Responce
- Modify DND

росбанк