РАСУ
РОСАТОМ

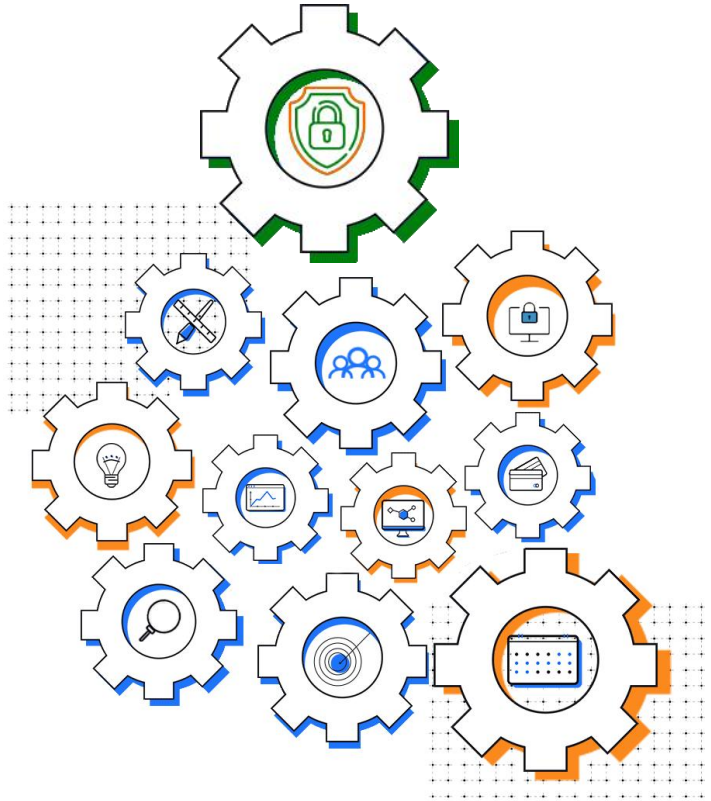
Методический подход к разработке и оценке безопасного программного обеспечения объектов КИИ и АСУ ТП

Конференция «Кибербезопаность цифрового предприятия в новой реальности: обеспечиваем непрерывность бизнеса и оптимизируем затраты на ИБ»

Сплюхин Денис Валерьевич
Главный специалист группы специальных лабораторных отработочных испытаний АО «РАСУ», к.т.н.

Обеспечение безопасности в процессе разработки

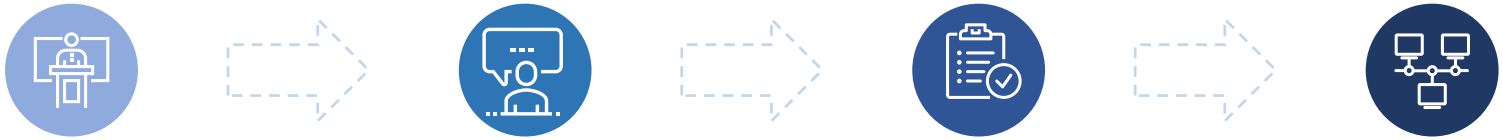
- Постановка требований
- Эскизное и техническое проектирование
- Разработка
- Тестирование и отладка
- Документирование
- Приёмочные испытания
- Выпуск в релиз (производство)
- Техническая поддержка в эксплуатации



Постановка требований к продукту

обязательные требования

дополнительные требования



Государственные регуляторы

требования определяются в соответствии с нормативными документами ФСТЭК России

Заказчик

требования формируются по итогам взаимодействия с Заказчиком, на основе утверждённого ТЗ/ТУ

Стандарты

проводится анализ национальных и международных стандартов на предмет необходимости включение в перечень специфичных требований

Меры для защиты ИС/АС

проводится анализ приказов и методических документов ФСТЭК России, определяющих требования к реализации мер защиты в целевых ИС/АС

Создание защищённой инфраструктуры среды разработки ПО

- **Разработаны и утверждены:**
  - ✓ Спецификации требований к процессам разработки ПО и ПАК (24 шт.)
  - ✓ Единые требования по обеспечению безопасного процесса разработки ПО и ПАК для систем и подсистем АСУ ТП (ПП) для подрядчиков (поставщиков)
  - ✓ Модель угроз для инфраструктуры среды разработки
  - ✓ Руководящие и методические документы по описанию процессов разработки и производства ПО и ПАК
- **Внедрены процессы** разработки безопасных ПО и ПАК на текущей инфраструктуре АО «РАСУ»
- Спроектирована **защищённая инфраструктура среды разработки** ПО и ПАК
- **Обработка информации обеспечивается на технических и программных средствах** - спроектированная защищённая инфраструктура САПР ПК и ПАК
- **Пути дальнейшего развития:**
  - ➢ Развитие (оптимизация, улучшение) процессов разработки безопасных ПО и ПАК
  - ➢ Развитие защищённой инфраструктуры среды разработки ПО и ПАК с учётом включения в процесс разработки кооперации организаций-разработчиков и «удалённщиков»

Процессы разработки и тестирования ПО и ПАК
- Процесс определения требований
- Процесс проектирования
- Процесс кодирования
- Процесс интеграции
- Процесс верификации
- …

Инструментальные средства разработки и тестирования
- Системы управления репозиториями кода, системы автоматизации сборки ПО из исходных текстов, наборы компиляторов, интерпретаторов, трансляторов, системы управления версиями, отладчики и т.д.

Объекты инфраструктуры среды разработки
- Технические средства, общесистемные и прикладные программные средства (ОС, средства виртуализации, системы управления проектами и т.д.)

Инфраструктура среды разработки программного обеспечения

МЕРЫ И СРЕДСТВА ЗАЩИТЫ

Регламентация процессов разработки (1/2)

RASU ROSATOM

Национальные стандарты	Документы ФСТЭК России	Отраслевые требования

- Приказ ФСТЭК России от 25.12.2017 №239 «Об утверждении Требований по обеспечению безопасности значимых объектов критической информационной инфраструктуры РФ»
- Приказ ФСТЭК России от 02.06.2020 №76 «Об утверждении Требований по безопасности информации, устанавливающие уровни доверия к средствам технической защиты информации и средствам обеспечения безопасности информационных технологий»
- ГОСТ Р ИСО/МЭК 12207-2010 «Информационные технологии. Системная и программная инженерия. Процессы жизненного цикла программных средств»
- ГОСТ Р 56939-2016 «Защита информации. Разработка безопасного программного обеспечения. Общие положения»
- ГОСТ Р 54293-2020 «Анализ состояния производства при подтверждении соответствия»

На основе положений действующих нормативных и регламентирующих документов в области ИТ и защиты информации определены порядки по разработке ПО и ПАК, разработаны методические указания и руководства по обеспечению требований ИБ и проведению квалификационного тестирования ПО и ПАК

РАСУ РОСАТОМ

Локальные нормативно-правовые акты АО «РАСУ» по разработке ПО и ПАК с учётом требований по информационной безопасности

2021-2022

Регламентация процессов разработки (2/2)

- Порядок «Разработка и производство программных и программно-аппаратных комплексов»
- Руководство «Реализация мер по разработке безопасных программных и программно-аппаратных комплексов»
- Руководство «Организация и ведение системы управления конфигурацией программных и программно-аппаратных комплексов»
- Руководство «Проведение анализа функциональных требований по безопасности к разрабатываемым программных и программно-аппаратных комплексов»
- Порядок «Проведение квалификационного тестирования программных и программно-аппаратных комплексов по требованиям информационной безопасности»
- Методические указания «Проведение квалификационного тестирования программных и программно-аппаратных комплексов по требованиям информационной безопасности»
- Методические указания «Формирование свидетельств разработчика программных и программно-аппаратных комплексов»
- Руководство «Обеспечение технической поддержки программных комплексов и программно-аппаратных комплексов в части информационной безопасности»
- Методические указания «Управление отслеживанием и устранением уязвимостей программных и программно-аппаратных комплексов»

Проектирование архитектуры и разработка

Определение структуры на уровне подсистем
логическая структура, основные интерфейсы и сопоставление подсистем с функциональными требованиями к продукту

Определение структуры на уровне модулей
модульная структура, внутренние и внешние интерфейсы, взаимодействие модулей и реализация информационных потоков

Обеспечение защиты и невозможности обхода ФБО
исключение возможности влияния пользователей на реализацию ФБО и выполнения действий в обход правил разграничения доступа

Определение порядка оформления исходного кода
стандарты языков программирования, требования к использованию структур и функций, исключения из правил

Прослеживание архитектуры к исходному коду
сопоставление архитектуры на уровне модулей и функциональной спецификации с файлами исходного кода

Экспертный анализ архитектуры в разрезе ИБ

RASU
ROSATOM

Документальная экспертиза

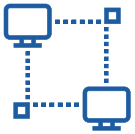
Проводится оценка проекта архитектуры программной реализации, полнота выполнения требований ИБ

Оценка архитектуры в разрезе угроз ИБ

Проводится оценка достаточности решений, направленных на устранение угроз ИБ, определяются требования к безопасной настройке и среде функционирования

Анализ уязвимостей архитектуры

Выполняется поиск в открытых источниках сведений об уязвимостях выбранных архитектурных и интерфейсных решений



Анализ скрытых каналов

Моделируются схемы реализации скрытых каналов, проводится оценка их пропускной способности, разрабатываются меры противодействия

Проведение квалификационного тестирования. Статический анализ

RASU ROSATOM

Статический анализ – обеспечительный метод, используемый при разработке программного обеспечения для сокращения вероятности возникновения уязвимостей, ошибок в бизнес-логике, нарушений штатной работы

Целью проведения статического анализа является автоматизированное обнаружение потенциальных дефектов (недостатков) программного обеспечения

Задачи:
- Поиск разнообразных паттернов ошибок
- Раннее обнаружение проблем
- Контроль качества кода в проектах
- Уменьшение трудозатрат программистов на код-ревью
- Контроль оформления кода

Встраивание в процессы CI/CD

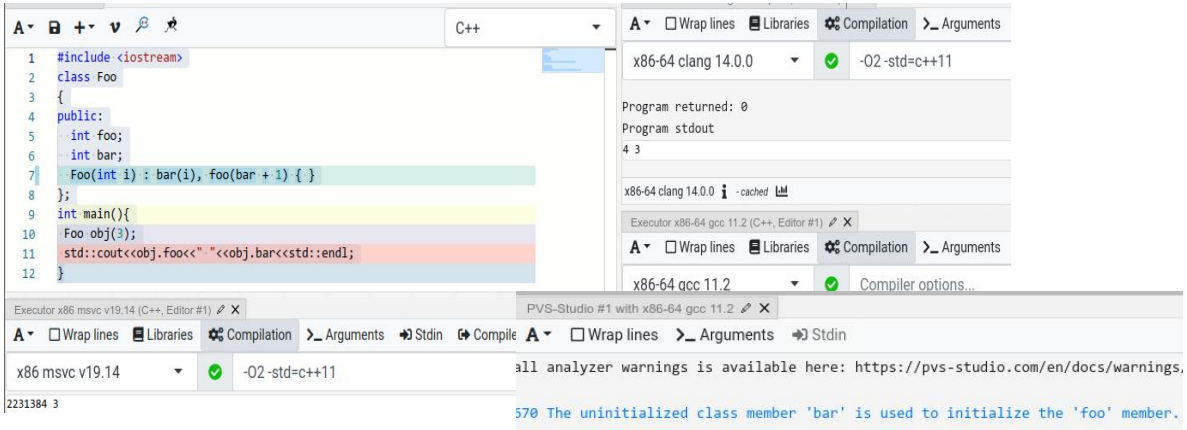
Проведение испытаний на тестовой инфраструктуре

C/C++:
- ❑ ИСП РАН Svace
- ❑ PVS-Studio
- ❑ Cppcheck
- ❑ Clang Static Analyzer/Clang-Tidy

Golang:
- ❑ Golangci-lint
- ❑ ИСП РАН Svace
- ❑ Sonarqube Community Edition

Python:
- ❑ Pylint
- ❑ Semgrep
- ❑ Bandit

Проведение квалификационного тестирования. Статический анализ. Опыт в рамках проектов



Языки программирования:	C/C++, Go
Объёмы исследуемого кода:	от 200 до 2500 строк кода
Срабатывания анализаторов:	Плотность возможных дефектов от 1,5 до 10 на 1000 строк кода
Основные виды ошибок:	Разыменование потенциально нулевых указателей, логические ошибки, чтение за границей массива, несоответствие аллокации и деаллокации памяти, использование неинициализированных переменных

Основные трудности:

- Наличие ложных срабатываний
- Трудоемкость экспертной разметки результатов анализа
- Требование обширного пула знаний при анализе сложных дефектов
- Сложности при выявлении ошибок, связанных с многопоточностью и иных высокоуровневых ошибок
- Большие затраты на приобретение лицензий на использование проприетарных инструментов анализа

Проведение квалификационного тестирования. Динамический анализ

Динамический анализ – метод исследования продукта с целью выявления уязвимостей и ошибок, включающий в себя модульное и фаззинг-тестирование объекта оценки. Тестированию подвергаются модули, составляющие поверхность атаки

Целью проведения динамического анализа является выявление уязвимостей программного обеспечения, которые могут привести к нарушению безопасности информации:

- несоответствие требованиям ИБ;
- нарушение функциональных спецификаций;
- недостатки архитектуры;
- недостатки интерфейсов;
- ошибки/отсутствие контроля входных данных.

ИСП РАН

Встраивание в процессы CI/CD

Проведение испытаний на тестовой инфраструктуре

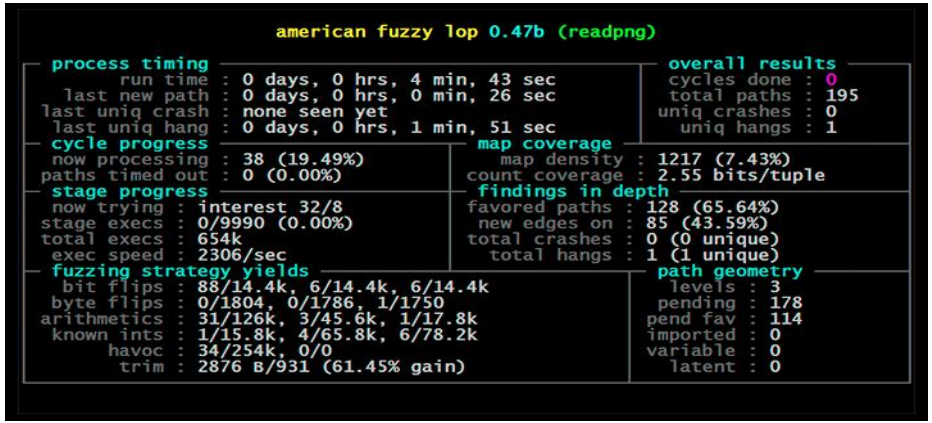
Статическая инструментация:
- ❑ LLVM
- ❑ GCC
- ❑ AddressSanitizer

Динамическая инструментация:
- ❑ Valgrind
- ❑ DynamoRIO

Эмуляция:
- ❑ Qemu

Фаззеры:
- ❑ AFL/AFL++
- ❑ Crusher
- ❑ libFuzzer
- ❑ AFLNet
- ❑ SoftTouch

Проведение квалификационного тестирования. Динамический анализ. Опыт в рамках проектов



Языки программирования: C/C++, Go

Примеры программных компонент, исследованных методом фаззинга: Парсеры конфигурации Net-SNMP, утилита командной строки SUDO, парсеры скриптов интерпретируемых языков, сетевые интерфейсы Netkit-Telnet, OpenSSH, HTTP-сервисов, парсеры опций ключей OpenSSH, библиотека OpenSSL, библиотека FMT

Среднее покрытие: 10%-30%

Основные трудности:

- Сложности архитектуры и недостаточная инкапсуляция, что требует существенных ресурсов для разработки фазз-тестов
- Сложность исследуемых протоколов
- Недостаточная документированность исходного кода готовых продуктов и компонентов
- Высокая сложность реализации алгоритмов обработки структур данных, например за счёт применения криптографических алгоритмов
- Формирование начальных корректных наборов входных данных (корпусов)

Проведение квалификационного тестирования. Анализ уязвимостей и тестирование на проникновение

RASU
ROSATOM

Анализ уязвимостей проводится по базам данных в открытых источниках с целью обнаружения уязвимостей и сценариев эксплуатации данных уязвимостей для модулей используемых в исследуемом ПО

01 Составление списка модулей и версий исследуемого объекта оценки

02 Осуществление поиска в открытых источниках и базах данных уязвимостей

03 Анализ потенциальных уязвимостей

Тестирование на проникновение использует различные методы и инструменты для поиска и эксплуатации уязвимостей

01 Использование данных о потенциальных уязвимостях с этапа анализа открытых источников и других видов тестирования

02 Использование сканеров

03 Разработка сценариев атак и использование готовых сценариев при их наличии

04 Анализ результатов

metasploit®

KALI
BY OFFENSIVE SECURITY

Wireshark

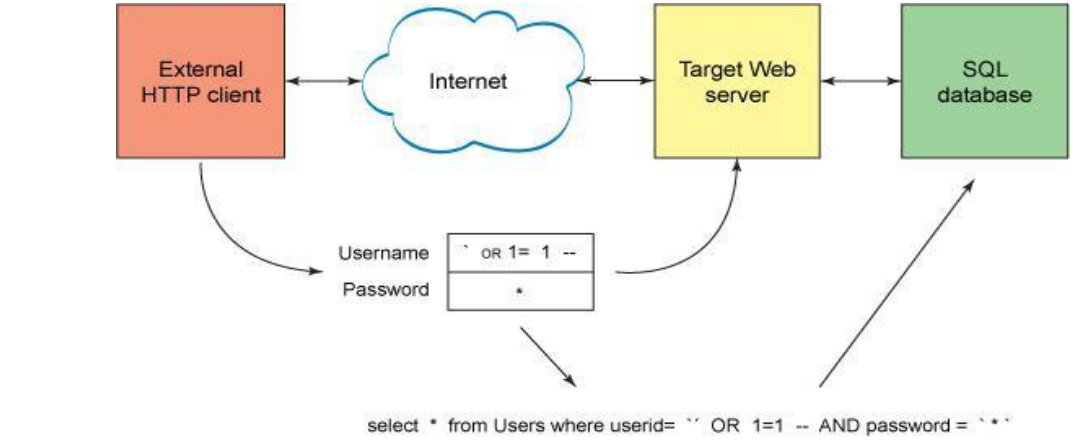
BURPSUITE

Сканер-ВС
анализ защищенности

NMAP PROJECT

HYDRA

Проведение квалификационного тестирования.
АУ и тестирование на проникновение. Опыт в рамках проектов



Примеры видов проводимых тестирований на проникновений:	Внешние тестирование на проникновение, web-тестирование, тестирование на проникновение в АСУ ТП
Основные методики тестирования:	Black Box, Grey Box, White Box
Используемые базы данных уязвимостей:	Реестр БДУ ФСТЭК России, MITRE CVE, база данных NVD, база данных VND, Exploit DB, VulnDB и т.д.

Основные трудности:

- Отсутствие эксплойтов для уязвимостей или использование собственных модулей, разработка уникальных сценариев атак требует большего количество времени
- Отсутствие универсальных (или полное отсутствие для некоторых объектов исследования) инструментов со всей необходимой функциональностью
- Сложность эмуляции ПО и среды функционирования

Проведение квалификационного тестирования. Основные особенности

Использование при создании ПО и ПАК компонентов с открытым исходным текстом (OpenSource)

- Отсутствие регрессионных тестов и фаз-тестов и наличие избыточной функциональности заимствованных компонентов

Испытания встраиваемого ПО

- Сложность создания виртуальной инфраструктуры с учётом особенностей аппаратной части

Разработка ПО и ПАК с привлечением внешнего исполнителя

- Отсутствие регламентации сроков и способов передачи исходного текста и отсутствие процессов РБПО и исполнителя

Отсутствие единых инструментов коллективной разработки

- Снижение оперативности взаимодействия с разработчиками в части устранения недостатков

Подготовка документации

RASU
ROSATOM

Свидетельства разработчика

- ❑ Задание по безопасности
- ❑ Технические условия
- ❑ Модель безопасности
- ❑ Описание архитектуры
- ❑ Описание архитектуры безопасности
- ❑ Свидетельство прослеживаемости архитектуры к исходному коду
- ❑ Функциональная спецификация
- ❑ Порядок оформления исходного кода
- ❑ Описание средств, используемых для разработки и тестирования
- ❑ Тестовая документация (план, ПМИ, отчёты, протоколы)

Документация по ГОСТ

- ❑ Формуляр (паспорт)
- ❑ Руководство пользователя (руководство по эксплуатации)
- ❑ Руководство администратора (инструкция по установке и настройке)
- ❑ Описание программы
- ❑ Описание применения
- ❑ Спецификация
- ❑ Текст программы
- ❑ Конструкторская документация (для ПАК)
- ❑ Руководство по техническому обслуживанию (для ПАК)

Документация на процессы разработки и производства

- ❑ Руководство по РБПО
- ❑ План управления конфигурацией
- ❑ Регламент технической поддержки
- ❑ Регламент выявления ошибок и уязвимостей и выпуска обновлений
- ❑ Регламент обучения сотрудников
- ❑ Регламент производства

Вектора развития

- Техническое и программное обеспечение единой среды разработки ПО и ПАК
- Развитие защищённой инфраструктуры среды разработки ПО и ПАК с учётом включения в процесс разработки кооперации организаций-разработчиков и «удалённщиков»
- Развитие (оптимизация, улучшение) процессов разработки безопасного ПО и ПАК, внедрение процессов на самых первых этапах разработки
- Разработка средств автоматизации тестирования с целью оптимизации времени проведения испытаний
- Апробация и внедрение новых инструментов тестирования
- Разработка единых отраслевых методических указаний по внедрению процессов разработки безопасного ПО и организации защищённой инфраструктуры среды разработки
- Разработка методики и критериев оценки поставщиков в соответствии с требованиями российского и зарубежного законодательства
- Создание и регламентация центра технической поддержки

РАСУ
РОСАТОМ

Методический подход к разработке и оценке безопасного программного обеспечения объектов КИИ и АСУ ТП

Конференция «Кибербезопаность цифрового предприятия в новой реальности: обеспечиваем непрерывность бизнеса и оптимизируем затраты на ИБ»

Сплюхин Денис Валерьевич
Главный специалист группы специальных лабораторных отработочных испытаний АО «РАСУ», к.т.н.

Тел.: +7 (495) 933-43-40
E-mail: dvsplyukhin@rasu.ru
www.rasu.ru