



Интеграция SAST

как первый шаг
внедрения подходов DevSecOps

Деев Сергей
Менеджер продукта Solar appScreener

Современные подходы в разработке

Критерии успеха при разработке
сервисов и ПО:

- Скорость выпуска новых релизов
- Качество (надежность и безопасность)
публикуемых сервисов/разрабатываемых
продуктов



Безопасность – важное свойство процесса разработки

Актуальные практики:

Разработка



AGILE, SCRUM и т.д.

DevOps

Безопасность



SDLC

DevSecOps

SAST – базовый инструмент в практиках безопасной разработки, который поддается автоматизации



Ключевые преимущества SAST



Полное покрытие кода

Показывает все потенциальные ошибки схожего типа.



Простая интеграция

Может быть интегрирован в сборочную инфраструктуру на разных этапах разработки ПО в рамках процесса DevSecOps.



Конкретные выводы

Указывает, где конкретно находится ошибка в коде, и описывает, к чему она может привести.

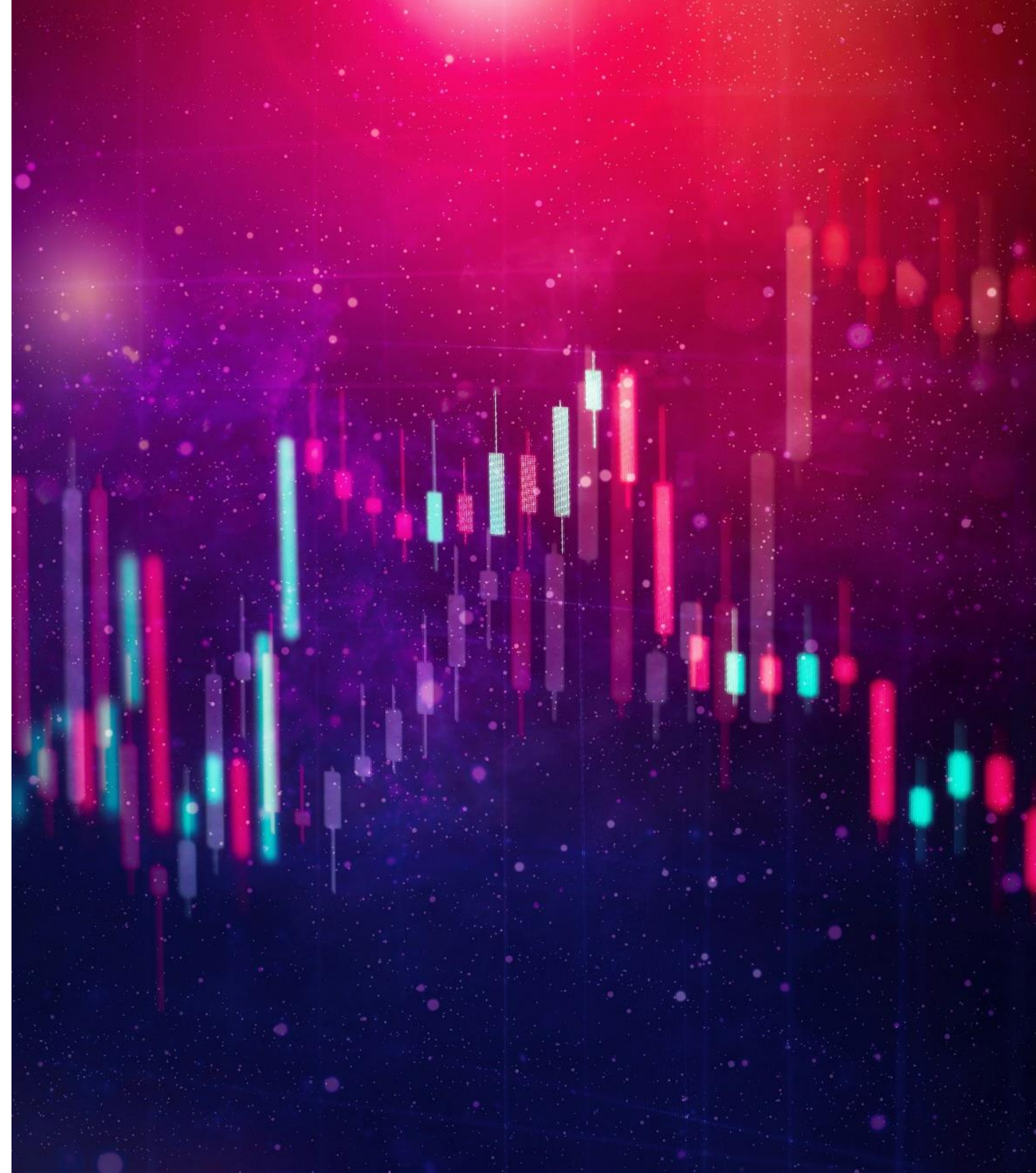


Выявление 0-day

Сохраняется возможность обнаружения уязвимостей нулевого дня.

Отличие Enterprise решений SAST от Open Source

- Более точные алгоритмы выявления ошибок, потенциально являющихся уязвимостями
- Охват языков программирования
- Возможность интеграции с компонентами сборочной инфраструктуры
- Техническая поддержка



Принципы работы Solar appScreener



Преимущества Solar appScreener: 36 языков программирования



Преимущества Solar appScreener: 9 типов бинарных файлов

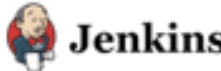


Преимущества Solar appScreener: Гибкая настройка выводимых данных

1. Механизм **Fuzzy Logic Engine** – интеллектуальная настройка вывода информации о выявленных ошибках.
2. Калибровка информации об уязвимостях в просканированных проектах.
3. Настройка представления отчетных данных в более сокращенном или расширенном виде.
4. Поддерживается вывод данных:
 - Через интерфейс
 - В отчете форматов PDF/CSV
5. Правила для **WAF**

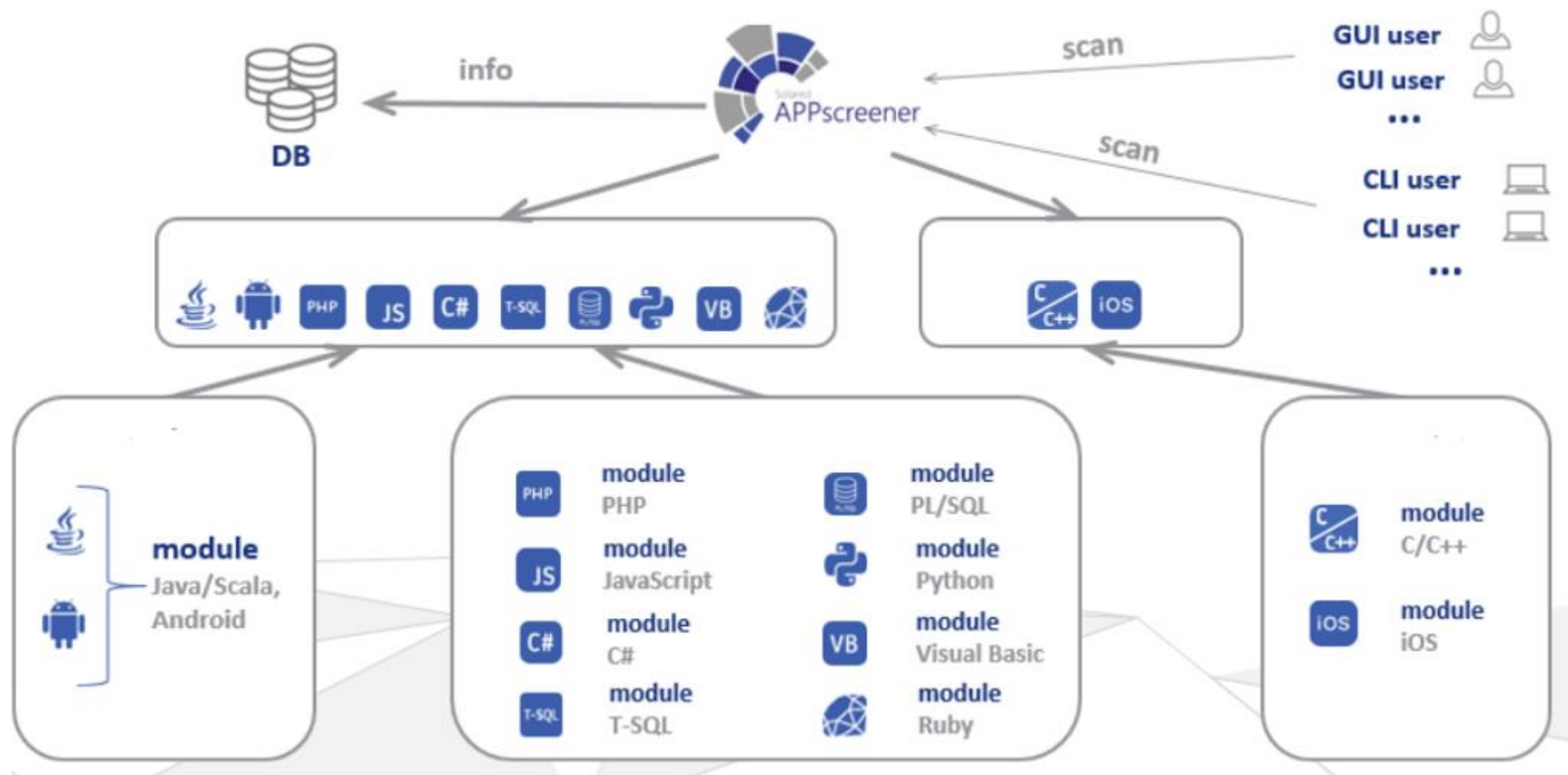


Поддерживаемые инструменты для интеграции

Репозитории	 git	 SUBVERSION					
VCS хостинги	 GitHub	 GitLab	 Bitbucket				
Средства разработки	 eclipse	 Visual Studio					
Средства сборки	 Xcode	 GNU (Autotools, Make)	 sbt	 Maven	 Gradle	 CMake	 Visual Studio
Серверы CI/CD	 Jenkins	 TeamCity	 Azure DevOps Server				
Отслеживание задач	 JIRA						
Анализ кода	 sonarqube						

Открытый API предоставляет широкие возможности по дополнительной интеграции и автоматизации. Включает в себя **JSON API** и **CLI**.

Пример интеграции Solar appScreener в сборочную инфраструктуру



Интеграция SAST– важный шаг к повышению безопасности ПО и сервисов

- Повышает качество решений
- Обеспечивает серьезное конкурентное преимущество для организаций



Для более подробного знакомства с Solar appScreener

Узнать информацию о продукте
Solar appScreener на официальной
странице решения:



rt-solar.ru/products/solar_appscreener/

Для проведения **пилотного
тестирования** можно оставить
запрос на email:

presale.appscreener@rt-solar.ru



Деев Сергей

Менеджер продукта Solar appScreener

s.deev@rt-solar.ru

rt-solar.ru