BIS JOURNAL

ИНФОРМАЦИОННАЯ БЕЗОПАСНОСТЬ БИЗНЕСА

№1 (60) 2026

Анатолий АКСАКОВ
Госдума
«Это не будет навязчивой кампанией»
→ 4

Игорь ШЕРЕМЕТ
РАН
Что ждет мировую отрасль ИИ в 2026–30 гг.
→ 72

Цифровой рубль: «А от тебя, мошенница, я точно уйду!» → 4

Безопасность КИИ: эволюция доверия → 16

РБПО. Испытания закончились. Что дальше? → 48

Юрий ШАБАЛИН
AppSec.Sting
Программный модуль цифрового рубля → 8

Алексей ПЕТУХОВ
RuSCADASec
ЦУПС и сообщество RuSCADASec → 16

Сергей ГРУЗДЕВ
Аладдин Р.Д.
Аутентификация, ЭП и биометрия → 25

ИТОГИ ИСПЫТАНИЙ СТАТИЧЕСКИХ АНАЛИЗАТОРОВ ИСХОДНОГО КОДА ПРИ ОРГАНИЗАЦИОННОЙ И МЕТОДИЧЕСКОЙ ПОДДЕРЖКЕ ФСТЭК РОССИИ

Дмитрий ПОНОМАРЕВ
заместитель генерального директора — директор департамента внедрения и развития практик РБПО ООО НТЦ «Фобос-НТ», сотрудник ИСП РАН, преподаватель МГТУ им. Н. Э. Баумана

Одним из ключевых событий Открытой конференции Института системного программирования РАН им. В. П. Иванникова, прошедшей 9–10 декабря 2025 года, стало подведение итогов испытаний статических анализаторов, организованных при поддержке ФСТЭК России. В 2025 г. коллективы-участники испытаний и автор этой статьи посвятили организации и проведению испытаний значимую часть своего рабочего и личного времени. По итогам были получены интереснейшие и важные результаты, и одновременно с этим стало очевидно — мы только в начале пути. Приобретённый опыт позволит внести необходимые улучшения в ГОСТ Р 71207–2024, доработать тестовую базу и методологию испытаний, подойти к организации следующего цикла испытаний с существенно более точной оценкой ресурсозатрат. Наиболее подробную информацию о задачах и целях, ходе и результатах испытаний, с «человеческими» комментариями их непосредственных участников, вы можете получить, ознакомившись с «первоисточником» — видеозаписью подведения итогов на сайте Открытой конференции [1].

Дальнейшее развитие системы испытаний статических анализаторов (далее — СА) будет освещаться на портале РБПО.рф [2].

ОБОБЩЁННЫЙ ОПЫТ ОРГАНИЗАЦИИ И ПРОВЕДЕНИЯ ИСПЫТАНИЙ

Вместо предисловия важно отметить, что организация и модерация испытаний явились для меня личным вызовом, потребовавшим существенных затрат времени и нервов, в том числе на принятие решений в ситуациях, когда идеального решения не существует, и от практически любого принятого решения обязательно останутся недовольные. Тем не менее итоги испытаний признаны успешными и полезными, работа будет продолжена. Поэтому я хочу выразить искреннюю признательность всем, принявшим участие в планировании, организации, проведении и всестороннем обеспечении испытаний, и особенно — лично Д. Н. Шевцову и И. С. Гефнер (ФСТЭК России), А. А. Белеванцеву (ИСП РАН), А. В. Кузнецову и А. И. Буркэ (НТЦ Фобос-НТ / ИСП РАН), Е. А. Погорелко (МГТУ им. Н. Э. Баумана) и Н. А. Костригину (Базальт СПО) — порой нам даже было весело (рис. 1).

Далее представлены наиболее значимые тезисы, описывающие полученный опыт:

1. Задуманный объём испытаний существенно превысил возможности коллективов по формированию комплекта тестов и последующей оценке и кросс-верификации результатов работы инструментов СА.

Рисунок 1. Порой нам даже было весело

2. Необходимо существенно повышать автоматизацию выполнения анализа и оценки результатов инструментов, что требует выработки более строгих критериев и правил (стандарта) формирования базы тестов.

3. Важно фиксировать все обсуждения и принятые решения в протоколах — особенно в отношении тех решений, которые теоретически могут повлиять на подчёркивание тех или иных сильных (и в особенности — слабых) сторон инструментов.

4. Добровольный формат участия в испытаниях предполагает нестабильную и не в полной мере прогнозируемую результативность работы различных групп участников. Получение официального подтверждения от организаций, принявших решение выступить в роли экспертов, оценивающих результаты работы инструментов СА, о выделении соответствующих людских ресурсов, представляется правильным для планирования дальнейших работ.

5. Использование в качестве тестовой базы исходного кода программных компонентов с открытым исходным кодом требует филигранной точности в определении числа и объёма кодовой базы компонентов. Слишком большое число срабатываний инструментов СА на выбранной для испытаний очевидно избыточной кодовой базе не позволило выполнить разметку и кросс-верификацию срабатываний в объёме, позволяющем считать данную выборку репрезентативной. По этой причине результаты разметки открытого кода не вошли в итоговую оценочную формулу.

Тем не менее определённая польза от данной активности была получена в любом случае. Помимо того, что ряд коллективов в ходе разметки выявил некоторые недостатки в своих инструментах СА, коллеги ИСП РАН самостоятельно выполнили «вдумчивую разметку» всего объёма срабатываний детекторов, соответствующих категории ГОСТ (рис. 2).

Результаты данной разметки представлены на портале Центра исследований безопасности системного ПО [6] и могут быть использованы в качестве «эталона» при сравнительном анализе в ходе выбора инструмента СА.

	Всего срабатываний	True Positive	TP без Won't fix	False Positive
C/C++				
nginx	50	76.0%	**30,0%**	24,0%
unbound	151	64,2%	**41,1%**	35,8%
Java				
guava	220	93,2%	**12,3%**	6,8%
netty	440	94,8%	**7,3%**	5,2%
spring-framework	32	50,0%	**46,9%**	50,0%
C#				
FluentValidation	20	100,0%	**10,0%**	0,0%
SharpZipLib	349	85,7%	**12,3%**	14,3%
SignalR	866	67,3%	**29,8%**	32,7%
Go				
etcd	187	83,4%	**36,9%**	16,6%
minio	953	96,3%	**8,8%**	3,7%
nats-server	99	85,9%	**69,7%**	14,1%

Рисунок 2. Результаты данной разметки представлены на портале Центра исследований безопасности системного ПО

6. ГОСТ Р 71207–2024 «Статический анализ исходного кода» требует внесения ряда доработок и уточнений, в том числе в терминах и определениях, для устранения неоднозначностей трактовок.

7. Необходимо выполнить более точный маппинг детекторов, реализованных инструментами СА, на методы анализа и типы ошибок, заявленные в ГОСТ. Эта кропотливая, но необходимая работа позволит повысить доверие к результатам последующих циклов испытаний за счёт минимизации коллизий срабатывания «не того детектора» в «том месте» кода, где заложена ошибка.

8. Необходимо точно и однозначно определить в методологии оценки перечни методов анализа, свойств инструментов и типов подлежащих обнаружению ошибок, подлежащие оценке в ходе испытаний.

9. Штатные особенности работы ряда инструментов анализа должны получить оценку «баг или фича» с целью либо верного учета данных особенностей при оценке результатов испытаний, либо — доработки инструментов до стандартного для методологии испытаний уровня возможностей.

10. Необходимо выработать и зафиксировать в ГОСТ единый, стандартизованный формат SARIF, который не только будет существенно способствовать повышению автоматизации последующих циклов испытаний, но и позволит стандартизовать обмен информацией с Центром исследований безопасности системного ПО в ходе выполнения сертификационных испытаний на соответствие требованиям ФСТЭК России.

11. Необходимо развивать подход, предполагающий создание комплексных тестов, позволяющих проверить корректность работы инструмента СА не только на простых атомарных тестах, но и на кодовой базе, по своей сложности сопоставимой с зубодробительными конструкциями и сложнейшими контекстами, порой встречающимися в реальном коде.

РЕЗУЛЬТАТЫ ИСПЫТАНИЙ

Данный раздел практически полностью состоит из слайдов с итоговыми количественными оценками (выборочно), без каких-либо пояснений, поскольку полноценный контекст их формирования слишком объёмен для текстового представления в рамках одной статьи. Всех желающих разобраться подробнее с принципами подсчёта результатов, а также ознакомиться со всем множеством результатов, я вновь приглашаю ознакомиться с материалами по ссылкам [1] и [4].

Зафиксирую единственный принципиально важный для восприятия результатов тезис-предупреждение — приведённые далее оценки следует трактовать в качестве приблизительных, в силу:

- неоднозначности трактовок результатов прохождения конкретными инструментами некоторых конкретных тестов;
- неполного выполнения процедуры разметки и кросс-верификации.

Для предоставленных цифр доверительный интервал оценивается в 7%, за исключением особо обозначенных случаев. Вся разметка выполнялась публично и была доступна для всех участников испытаний, все спор-

Категория	Вес	Базальт	ИСП	ПВС	Позитив	Солар	Эшелон
1	0.1	98	130	58	12	90	94
2	0.25	122	207	42	73	28	39
3	0.25	100	169	54	39	33	98
4	0.25	89	165	68	66	37	25
8	0.15	62	117	28	32	7	74
Всего	**1**	**96.85**	**165.8**	**51**	**50.5**	**34.55**	**61**

Категория	Вес	Базальт	ИСП	ПВС	Позитив	Солар	Эшелон
1	0.1	16	19	16	14	13	19
2	0.25	10	15	14	7	10	15
3	0.25	9	15	5	7	8	15
4	0.25	35	32	24	12	15	41
8	0.15	4	13	8	4	12	6
Всего	**1**	**15.7**	**19.35**	**13.55**	**8.5**	**11.35**	**20.55**

Рисунок 3. Результаты по основной формуле для языка программирования (ЯП) Си/Си++

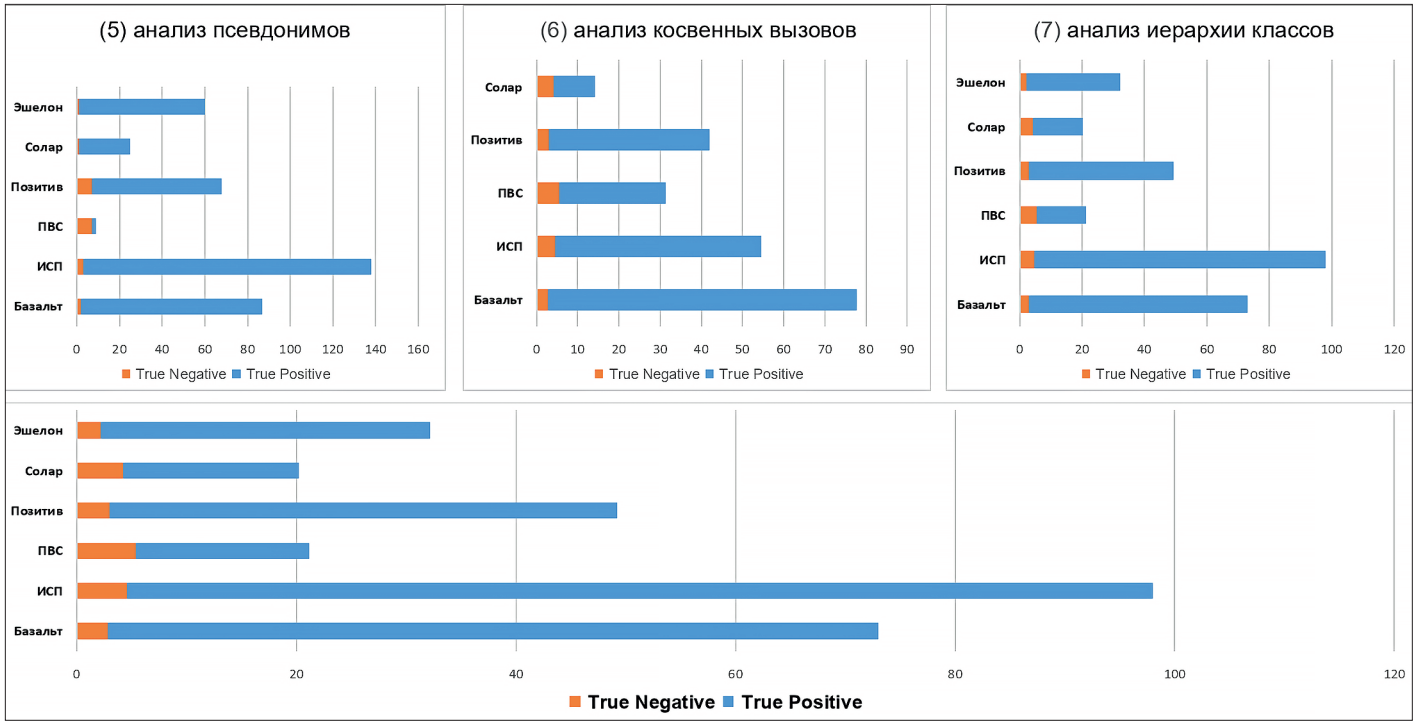


Рисунок 4. Результаты по дополнительной формуле для ЯП Си/Си++

ные вопросы разбирались участниками испытаний публично (рис. 3–6).

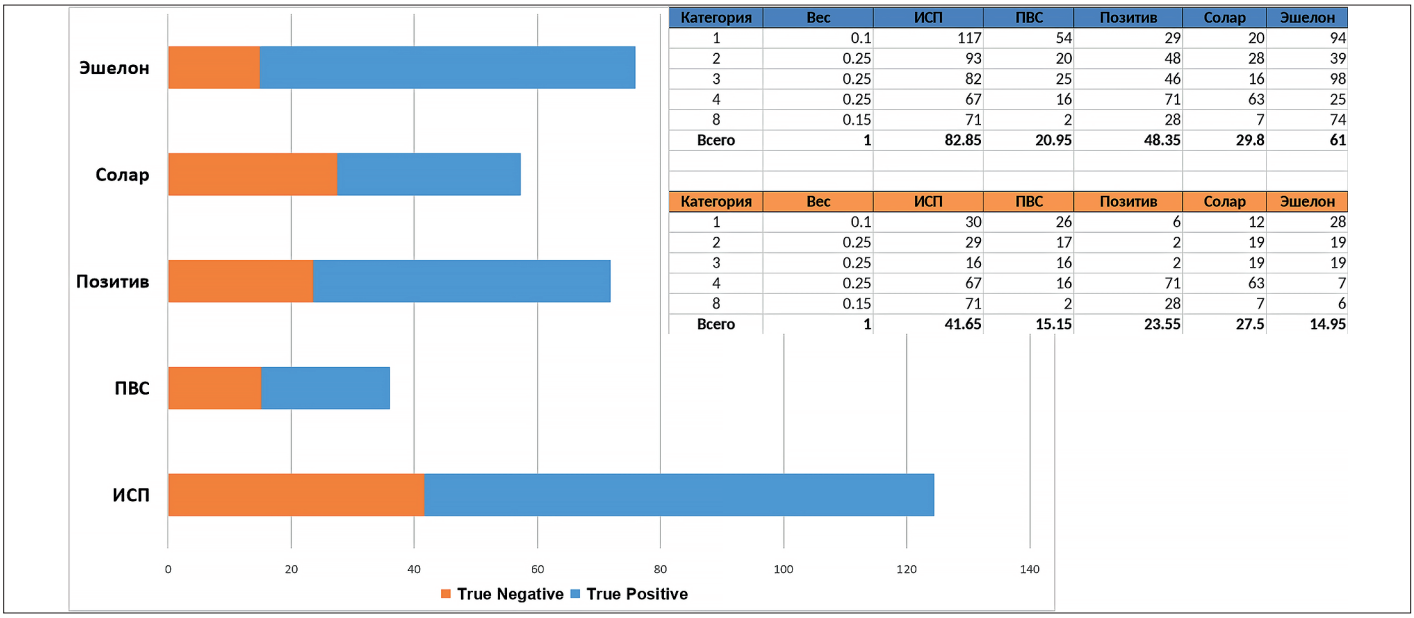
В теории на рис. 6 показано заявленное участниками испытаний число срабатываний детекторов по ГОСТ. На практике:

- не все инструменты имеют механизмы фильтрации по типам детекторов;
- не все разработчики качественно отделили детекторы out-of-scope от ГОСТ;
- некоторые инструменты действительно выдают очень много срабатываний.

ПЛАНЫ ПО ПОДГОТОВКЕ И ПРОВЕДЕНИЮ СЛЕДУЮЩЕГО ЦИКЛА ИСПЫТАНИЙ

Дальнейшие планы, озвученные И. С. Гефнер в ходе Открытой конференции, включают в себя:

- доработку и уточнение ГОСТ Р 71207–2024 «Статический анализ исходного кода»;
- доработку методологии испытаний и комплекта тестов с приоритетом создания «сложных», комплексных тестов, позволяющих испытывать инструменты в условиях, приближенных к «боевым»;
- разработку и стандартизацию формата представления результатов



Категория	Вес	ИСП	ПВС	Позитив	Солар	Эшелон
1	0.1	117	54	29	20	94
2	0.25	93	20	48	28	39
3	0.25	82	25	46	16	98
4	0.25	67	16	71	63	25
8	0.15	71	2	28	7	74
Всего	1	82.85	20.95	48.35	29.8	61

Категория	Вес	ИСП	ПВС	Позитив	Солар	Эшелон
1	0.1	30	26	6	12	28
2	0.25	29	17	2	19	19
3	0.25	16	16	2	19	19
4	0.25	67	16	71	63	7
8	0.15	71	2	28	7	6
Всего	1	41.65	15.15	23.55	27.5	14.95

Рисунок 5. Результаты по основной формуле для ЯП Java

Пакет	ИСП РАН	Эшелон	Солар	ПВС	Позитив	Базальт
Cpython (C/C++)	414	88	813	2655	4882	3318
nginx (C/C++)	54	208	147	70	373	320
unbound (C/C++)	151	222	0	257	613	498
guava (Java)	224	317	3858	156	5519	0
netty (Java)	462	321	150491	144	61896	0
spring-framework (Java)	49	859	15078	353	1213	0
nats-server (Go)	104	25	24	0	2102	0
minio (Go)	994	58	221	0	724	0
etcd (Go)	198	22	3407	0	99	0
webpack (Go)	0	0	633	0	334	0
react (JavaScript)	6	0	1791	0	1	0
axios (JavaScript)	0	0	289	0	3	0
SharpZipLib (C#)	349	3	567	60	3553	0
SignalR (C#)	866	30	581	114	200	0
FluentValidation (C#)	20	0	1	2	551	0
Django (Python)	15	562	117	0	171	0
celery (Python)	3	61	2068	0	158	0
fastapi (Python)	0	103	1454	0	26	0

Рисунок 6. Общее число срабатываний детекторов для различных инструментов при анализе открытых компонентов

статического анализа (по аналогии с форматом SARIF);

- выработку механизма поддержки и развития репозитория комплекта тестов (действующий репозиторий [7] создан и поддерживается коллективом Базальт СПО),
- и, главное, подготовку и проведение следующего цикла испытаний в 2027 году.

ВМЕСТО ВЫВОДОВ. ЛИЧНАЯ ПОЗИЦИЯ МОДЕРАТОРА

Стратегической целью испытаний СА является создание и улучшение условий для развития отечественной научной-инженерной и производственной культуры создания сложнейших инструментов анализа.

Желания «перекрасить» и продавать открытый или даже проприетарный иностранный инструмент, либо взять опенсорс «as is» и, не вкладываясь в его развитие, получить коммерческий результат, понятны с тактической точки зрения конкретного бизнеса.

Но стратегический негативный эффект от таких действий для отрасли и государства противоречит поставленной Президентом РФ цели достижения технологической независимости.

Испытания СА должны способствовать развитию и конкуренции отечественных школ создания инструментов СА и усложнять рыночную деятельность в указанных выше случаях – разумеется, за исключением случаев, когда открытый инструмент СА «из коробки» существенно лучше платного.

Поэтому я искренне рассчитываю, что органы по сертификации, аккредитованные в области сертификации процессов РБПО, будут опираться на методологию и результаты испытаний, в качестве минимально приемлемой, при оценке реализации процесса № 10 «Статический анализ исходного кода» (ГОСТ Р 56939–2024) разработчиками программного обеспечения и, в особенности, средств защиты информации.

ИСТОЧНИКИ

[1] https://www.isprasopen.ru/ Секция «Разработка безопасного ПО: качество, доверие, сообщество»
[2] https://рбпо.рф/news/rbpo-isprasopen-2025
[3] https://ib-bank.ru/rbpos
[4] https://t.me/sdl_community/9420
[5] https://t.me/sdl_community/7859
[6] https://clck.ru/3RSsMi
[7] https://altlinux.space/rbpo-community

ЧТО НОВОГО В SVACE

Андрей БЕЛЕВАНЦЕВ
руководитель направления анализа и оптимизации программ ИСП РАН, доктор физико-математических наук, член-корреспондент РАН

Прошедший год для развития нашего статического анализатора Svace оказался очень продуктивным. Прежде всего хочется отметить прошедшие испытания статических анализаторов под эгидой ФСТЭК России и поблагодарить организаторов, площадку — МГТУ им. Н. Э. Баумана и отдельно кафедру ИУ10, коллег-вендоров, принявших участие. Для нас испытания стали поистине событием года, источником бесценного опыта, выразившегося в подготовке тестов анализаторов и изучении тестов, сделанных коллегами. Удалось увидеть множество деталей, которые раньше не всегда были сделаны на должном уровне. При этом надёжная основа в виде качественного масштабируемого межпроцедурного анализа, системы перехвата сборки, механизмов чувствительности к путям выполнения и других необходимых алгоритмов позволили Svace показать лучшие — с существенным отрывом — результаты для языков C/C++ и Java, полностью соответствующие ГОСТ Р 71207–2024.

Главное, на наш взгляд, что от испытаний выиграли все — организаторы, получившие обратные связи от сообщества вендоров анализаторов; сами вендоры, лучше узнавшие сильные и слабые стороны своих инструментов и обменявшиеся опытом с коллегами; пользователи, увидевшие объективную расстановку сил на рынке анализаторов и получившие более качественные инструменты.

Весь опыт, который мы вобрали в ходе испытаний, вошёл в последний релиз Svace 5.0. Помимо многочисленных улучшений анализа, мы отдельно выделили категорию детекторов, обнаруживающих критические по ГОСТ ошибки. Будем оперативно обновлять эту категорию по мере того, как будет дорабатываться и сам ГОСТ.

Отметим несколько важных новшеств анализа. Из новых языков Svace теперь поддерживает анализ программ на Lua, пока легковесный — на уровне универсального синтаксического дерева. Будем развивать его вместе с анализами других динамических языков — Python и JavaScript.

Продолжает развиваться интерфейс SvaceAPI для создания пользовательских детекторов. Его очень удобно использовать для поиска специфических ошибок, затрагивающих неизвестные анализу библиотеки или связанных с нарушением правил использования критических для безопасности компонент. При этом задействуется вся мощь основного движка анализа Svace. Мы сами использовали SvaceAPI для создания ряда специфичных детекторов для ГОСТ-ошибок и приглашаем пользователей активно его применять. Ваши обратные связи помогут нам развивать интерфейс в полезных для всех направлениях.

Серьёзно улучшен сервер удалённого анализа, позволяющий теперь работать с клиентами разных версий и конфигурировать различные дистрибутивы, гибко управляя пулом агентов анализа. Эта часть Svace востребована в крупных компаниях, которые могут единообразно настраивать анализ для различных подразделений с учётом их специфики.

Появилось два новых режима анализа. Анализ «сверху вниз» позволяет находить ошибки, для которых естественно распространение информации от источников к приёмникам. Его комбинация с привычным анализом «снизу вверх» позволяет увеличить полноту (количество находимых ошибок) за счёт большего времени работы. Покомпонентный анализ позволяет анализировать независимые подпрограммы большой программной системы отдельно, сокращая потребление памяти; при этом выделение независимых модулей выполняется автоматически по данным графа компоновки — зависимые компоненты по-прежнему будут анализироваться вместе.

Активно развивается направление по применению в Svace алгоритмов машинного обучения. Уже можно попробовать ИИ-ассистенты, которые предсказывают истинность срабатывания по данным метрик программы (собранным Svace) и изнутри сервера работы с результатами Svacer. В будущем интерфейсы Svacer работы с ИИ-ассистентами будут позволять подключать различные модели, которые исследуются в ИСП РАН, в том числе разрабатываемую в настоящий момент функциональность предсказывания истинности и объяснения предупреждений через специализированные большие языковые модели.

Следите за нашими обновлениями!

Инструменты анализа ИСП РАН

Блог Svace

УСКОРЕНИЕ И ТОЧНОСТЬ

КАК МЕНЯЮТСЯ SAST-ИНСТРУМЕНТЫ В ЭПОХУ БЫСТРОЙ РАЗРАБОТКИ

Антон ПРУСАК
руководитель продукта AppSec.Wave

Ускорение — важный тренд мировой разработки ПО, в том числе — российской. Объем российского ИТ-рынка, по прогнозам экспертов, в 2026 превысит 4,5 триллиона рублей. В условиях непрерывной интеграции и DevOps современным разработчикам ПО критически важно, чтобы инструменты статического анализа кода (SAST) отвечали требованиям бизнеса по скорости и удобству. Запрос 2026, пожалуй, можно обозначить как: еще быстрее и разложить по полочкам.

На первый план у ИБ-команд компаний-разработчиков вышли высокая скорость анализа, минимальное количество ложных срабатываний и интуитивно понятный процесс триажа. Однако сегодняшние запросы рынка идут дальше: ключевым становится требование к гибкости и адаптивности платформы. Именно эта потребность легла в основу амбициозной задачи по созданию для российского ИТ-сектора универсального SAST-решения, которое не только находит уязвимости, а органично встраивается в рабочий процесс, позволяя бесшовно интегрировать безопасность в цикл разработки. Совмещая скорость, качество и адаптивность, SAST-решения становятся естественной частью рабочего процесса, обеспечивая безопасность без потери темпа разработки.

Основной тренд последних полутора лет — активное внедрение в разработку генерируемого ИИ-кода. Массовое распространение вайб-кодинга требует серьезного ускорения и усиления кибербезопасности.

AppSec Solutions для этой задачи разработал универсальный SAST — AppSec.Wave, отличительной чертой которого является высокая скорость сканирования. Основные векторы развития AppSec.Wave — скорость анализа, качество и удобство триажа уязвимостей.

AppSec.Wave поддерживает множество языков программирования, включая JavaScript, Python, Go, C++, Java, PHP, Ruby, C# и другие. В 2026 планируется добавить и язык 1С, широко используемый отечественными компаниями.

Современная архитектура и агентская модель предполагают полный функционал и добавление любого количества агентов, это удобно для любого размера бизнеса. Менеджмент может подключить к процессу разработки 30+ команд, которым будет вполне комфортно работать совместно. Помимо этого, анализатор дает возможность добавлять кастомные правила, позволяя ИБ-команде настраивать SAST с учетом потребностей бизнеса. Система анализирует код не только по шаблонам, как опенсорс-решения статического анализа, но и использует усовершенствованный taint-анализ, который позволяет определить источник, например, SQL-инъекции, и проследить все части системы, на которые она повлияла.. Команда разработчиков будет работать над качеством анализа и прохождением всего AST-дерева для полноценного поиска уязвимостей. Также AppSec.Wave позволяет отключать предустановленные правила из комплекта поставки на усмотрение пользователя, а заниматься отладкой новых правил можно в самом сканере, не заполняя очередь реальных проектов. Также поддерживается импорт отчетов и правил из других сканеров, что важно при переходе с импортных SAST-инструментов на российское ПО.

Инструмент позволяет удобно группировать отчеты, а также импортировать их из сканеров других решений, чтобы собирать данные по уязвимостям воедино. История изменений при этом сохраняется. Среди функций: поддержка инкрементального анализа, групповая обработка результатов и интеграция с таск-трекерами. Сейчас разработчики AppSec.Wave работают над возможностью интеграции инструмента с MCP-сервером, который позволит сканировать код с помощью языковых моделей.

Ключевой фактор, который влияет на скорость разработки — распространение вайб-кодинга. По оценкам экспертов, от 60 до 80% разрабатываемого в России ПО сгенерировано искусственным интеллектом. К примеру, совместное исследование IT_ONE, Фонда «Сколково» и «Сколтеха» показало, что 62% разработчиков в России применяли технологии ИИ при написании кода в 2025, а к 2028 эта цифра вырастет до 98%

ТРАНСФОРМАЦИЯ ОТЕЧЕСТВЕННЫХ ИНСТРУМЕНТОВ БЕЗОПАСНОЙ РАЗРАБОТКИ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ

Сас АРУСТАМЯН
директор Центра оценки соответствия и тестирования АО «НПО «Эшелон»

Чтобы отвечать вызовам современных угроз безопасности информации, инструменты разработки безопасного ПО должны не только существовать, но и доказывать свою эффективность на практике. Так в конце 2025 года завершились масштабные испытания статических анализаторов исходного кода, проведённые при поддержке ФСТЭК России. Эти испытания стали важным этапом в развитии отечественных средств анализа кода и позволили сформировать более прозрачное представление о текущем уровне инструментов статического анализа и их соответствии требованиям ГОСТ Р 71207–2024.

Для нашей команды участие в испытаниях стало возможностью получить независимую оценку ключевых характеристик анализатора на реальных тестовых наборах, а также сопоставить результаты с другими решениями в рамках единых испытаний[1].

При оценке качества статических анализаторов использовались классические метрики, отражающие корректность их срабатываний.

- TP (True Positive) — ситуация, когда анализатор обнаружил дефект и этот дефект действительно присутствует в исходном коде. Высокое значение TP демонстрирует способность инструмента выявлять реально существующие ошибки в исходном коде.
- TN (True Negative) — ситуация, когда анализатор не выдал предупреждение и при этом в анализируемом участке кода действительно отсутствует дефект.

РЕЗУЛЬТАТЫ ПО C/C++

По основной формуле для языков C/C++ анализатор АК-ВС 3 показал следующие результаты:

- TN (True Negative) — 1-е место;
- TP (True Positive) — 3-е место;
- TN + TP — 3-е место.

[1] Сравнительный анализ и выбор статических анализаторов безопасности кода / Марков А. С., Антипов И. С., Арустамян С. С., Магакелова Н. А. // Вопросы кибербезопасности. 2024. № 5 (63). С. 79–88.

Первое место по показателю TN имеет особую практическую значимость, поскольку низкое количество ложных срабатываний напрямую снижает трудозатраты на анализ отчётов и повышает применимость инструмента в промышленных проектах с большими кодовыми базами

Первое место по показателю TN имеет особую практическую значимость, поскольку низкое количество ложных срабатываний напрямую снижает трудозатраты на анализ отчётов и повышает применимость инструмента в промышленных проектах с большими кодовыми базами.

РЕЗУЛЬТАТЫ ПО JAVA

Для языка Java результаты АК-ВС 3 распределились следующим образом:

- TN (True Negative) — 2-е место;
- TP (True Positive) — 5-е место;
- TN + TP — 2-е место.

Несмотря на более низкую позицию по показателю TP, суммарная метрика TN + TP демонстрирует второе место, что указывает на сбалансированное качество анализа в целом. Высокое значение TN подтверждает низкий уровень ложных срабатываний, что особенно важно для Java-проектов, использующих большое количество типовых библиотек, фреймворков и повторяющихся шаблонов кода.

При всём том, что результаты по части заявленных языков программирования (C#, Go, JavaScript, Python) не были озвучены, в совокупности результаты по C/C++ и Java показывают, что АК-ВС 3 демонстрирует высокие значения по ключевым метрикам.

РАЗВИТИЕ АК-ВС 3

Отдельную ценность представляет не только итоговая оценка, но и обратная связь, полученная в ходе испытаний. По сути, испытания стали источником данных о тех направлениях развития инструмента, которые требуют приоритетного внимания и способны дать наибольшую практическую пользу при реальном использовании.

Так, в рамках дальнейшего развития продукта планируется расширение перечня поддерживаемых языков программирования, включая Swift и Kotlin, что позволит применять инструмент при анализе более широкого круга современных программных проектов, включая мобильные приложения.

Особое внимание уделяется развитию модуля фаззинг-тестирования, нацеленного на существенное повышение эффективности динамического анализа[2]. В частности, реализована поддержка всех основных режимов инструментации AFL++, что позволяет более гибко адаптировать процесс фаззинга к особенностям исследуемого программного обеспечения. Для оптимизации использования вычислительных ресурсов разработаны механизмы распределения ядер процессора между различными режимами фаззинга, а также предоставлена возможность создания пользовательских схем распределения. Автоматизирован процесс сборки проекта во всех режимах инструментации, и, что немаловажно, в отчёте по результатам фаззинга теперь доступны графики скорости фаззинга и детальная информация о достигнутом покрытии кода, позволяющие оценить эффективность проведённого тестирования. Параллельно развивается модуль полносистемного динамического анализа Код2, который обеспечивает запись, последующий анализ полносистемных трасс, а также позволяет автоматизировано определять поверхность атаки исследуемого ПО.

Дополнительно планируется разработка веб-интерфейса для IDE «Эшелониум», которая в настоящее время доступна в виде десктопного приложения. Так процесс просмотра, анализа и разметки результатов статического анализа станет ещё более удобным.

[2] Интеллектуальные методы фаззинг-тестирования в рамках цикла безопасной разработки программ / Арустамян С. С., Антипов И. С. В сборнике: Безопасные информационные технологии. Сборник трудов Двенадцатой международной научно-технической конференции. Москва, 2023. С. 11–15.

Не стоит забывать о том, что практики разработки безопасного ПО не ограничиваются статическим и динамическим анализом. Всё большую роль играет управление рисками, связанными с заимствованными компонентами: библиотеками, фреймворками, сторонними пакетами

СЛЕДУЮЩИЙ ЭТАП В РАЗВИТИИ РБПО: КОМПОЗИЦИОННЫЙ АНАЛИЗ

Не стоит забывать о том, что практики разработки безопасного ПО не ограничиваются статическим и динамическим анализом[3]. Всё большую роль играет управление рисками, связанными с заимствованными компонентами: библиотеками, фреймворками, сторонними пакетами. Это находит отклик как в отечественных регуляторных требованиях, так и в международных практиках.

Если статический и динамический анализ уже реализованы и активно развиваются в составе АК-ВС 3, то композиционный анализ выделен в отдельное направление.

В настоящее время в разработке находится инструмент CAT (Composition Analysis Tool) — средство композиционного анализа, предназначенное для:

- автоматической идентификации всех зависимостей проекта;
- формирования Software Bill of Materials (SBOM);
- анализа компонентов по базам уязвимостей (включая NVD и БДУ ФСТЭК России);
- автоматического формирования ППК (Прикладных программных комплексов);
- разметки уязвимостей;
- представления результатов в виде наглядной статистики.

Важно подчеркнуть, что CAT будет доступен как в виде самостоятельного продукта, так и в качестве модуля в составе АК-ВС 3, что позволит разработчикам выстроить конвейер РБПО, включающий сразу три вида анализа.

[3] Вареница В. В., Марков А. С., Цирлов В. Л. [и др.]. Как избежать ошибок при безопасной разработке программного обеспечения / Москва: Квант-медиа, 2025.— 344 с.— ISBN978–5–605–33861–1.

Арустамян С. С., Вареница В. В., Марков А. С. Методические и реализационные аспекты внедрения процессов разработки безопасного программного обеспечения // Безопасность информационных технологий. 2023. Т. 30. № 2. С. 23–37.

В завершение отметим, что всем, кто интересуется статическим анализом исходного кода, практическими примерами и новостями в этой области, будет полезно подписаться на Telegram-канал **@AK_BC_3**.

Для приобретения знаний по особенностям внедрения процессов безопасной разработки программного обеспечения рекомендуем ознакомиться **с практическим пособием от экспертов «Эшелона»**.

БЕЗОПАСНАЯ АРХИТЕКТУРА

ОТ ПРИНЦИПА НАИМЕНЬШИХ ПРИВИЛЕГИЙ ДО ВЫЗОВОВ ИИ И ТРЕНДОВ 2026 ГОДА

Евгений ТОДЫШЕВ
руководитель направления безопасной разработки УЦСБ

С чего начинается проектирование защищённого ПО? Как изменяются подходы к проектированию безопасности при разработке AI-систем? Прогноз на ближайшее будущее строим с руководителем направления безопасной разработки УЦСБ Евгением Тодышевым.

— Как на практике применять принцип наименьших привилегий и что такое безопасные настройки по умолчанию?

— Принцип наименьших привилегий — одно из основных понятий в безопасной архитектуре, но его часто реализуют формально. Суть в том, чтобы выдавать права строго в соответствии с функцией. Это как пропускной режим: сотрудник отдела кадров не должен иметь доступ в серверную. То же самое применимо в разрезе сервисов и процессов. Например, служба для отправки почты не должна иметь полный доступ к базе данных (БД) пользователей. Для этого мы проектируем для каждого сервиса отдельные учётные записи с минимальным набором прав.

Безопасность настройки по умолчанию основана на принципе «запрещено пока не разрешено» и делает «сервис из коробки» максимально закрытым. Например, при создании новой учётной записи пользователя она по умолчанию не имеет привилегированных прав и доступ к программным интерфейсам (API) выключен. И только администратор системы должен сознательно их включать, понимая все риски. Также при первом входе система сама должна требовать смены пароля администратора, чтобы его нельзя было дальше использовать по умолчанию.

Все эти правила создают защитный слой: даже если злоумышленник получил доступ к какой-то части системы, его продвижение дальше будет затруднено, и он по цепочке привилегий не сможет добраться до самых ценных данных.

При проектировании прежде всего надо понять, какие основные бизнес-функции будет выполнять продукт. Ещё следует учитывать изменения, которые вносятся в софт, моделирование угроз для нового ПО, разработку ИБ-требований и различные рекомендации в адрес ИТ-архитектора и разработчиков.

— Как спроектировать систему разграничения прав в приложении?

— Ядром безопасности приложения является авторизация, и она должна быть централизованной. В системе необходимо создать единую точку принятия решения, которая фактически будет отвечать на вопрос, можно ли пользователю совершать определённое действие. Все запросы к данным должны проходить через неё, исключая разрозненную и противоречивую логику в разных подсистемах. За сами же решения должен отвечать отдельный сервис (или библиотека), в ко-

В системе необходимо создать единую точку принятия решения, которая фактически будет отвечать на вопрос, можно ли пользователю совершать определённое действие. Все запросы к данным должны проходить через неё, исключая разрозненную и противоречивую логику в разных подсистемах

тором и находятся непосредственно сами правила.

Для обеспечения гибкой модели доступа можно использовать связку RBAC и ABAC, то есть ролевой и атрибутной модели. Следует предусмотреть отказ от наследования привилегий по умолчанию и явно задавать разрешение для каждой новой сущности. Например, если создаётся новая папка, пользователь не должен на неё получить те же права, что и на родительскую, — это предотвратит случайное расширение прав. Кроме того, надо создать автоматические тесты на нарушение прав доступа, чтобы пользователю с низкими привилегиями не удалось выполнить действия администратора.

— Атаки на цепочки поставок — один из ключевых вызовов последних лет. На что ещё, помимо Supply Chain Attack, стоит обратить внимание в 2026 году?

— Я бы выделил три чётких направления. Во-первых, **безопасность мобильных приложений**. Их количество и зрелость растут, они работают с критичными данными и даже управляют инфраструктурой. Защита от компрометации устройства — must have.

Во-вторых, **API Security.** Количество API растёт лавинообразно, они становятся ядром цифровых продуктов. Необходимы централизованный контроль, анализ трафика и защита от злоупотреблений.

И главный тренд, который мы уже затронули чуть раньше, — **MLSecOps**. Компании массово внедряют модели машинного обучения (ML-модели) для автоматизации, но часто забывают про их безопасность. В ближайшей перспективе ожидается появление ГОСТа и усиление регуляторных требований к безопасности машинного обучения. Это станет отдельной дисциплиной.

Кстати, детально обсуждать тренды мы будем 19 марта **на квартирнике по безопасной разработке** в Москве. Приходите, будем рады пообщаться вживую с экспертами-практиками DevSecOps и всеми, кто хочет глубже погрузиться в эту тему.

В AI-системах появляется фундаментальный класс неуправляемых или слабоуправляемых рисков, связанных с самой природой модели. Также появляются уязвимости, которые присущи данным: то есть атаки смещаются с кода на данные, и тут требуется другой подход к защите

— Какие проблемы возникают при проектировании AI-систем?

— Проектирование AI-систем, особенно с точки зрения безопасности вообще, достаточно непривычная парадигма, которая порождает и новые вызовы. Если в традиционном софте мы имеем дело с детерминированной логикой и управляемыми рисками в виде известных уязвимостей и ошибок в коде, то здесь работаем с вероятностными моделями. И это вносит определённые корректировки в подходы. В AI-системах появляется фундаментальный класс неуправляемых или слабоуправляемых рисков, связанных с самой природой модели. Также появляются уязвимости, которые присущи данным: то есть атаки смещаются с кода на данные, и тут требуется другой подход к защите. Вдобавок модель сама по себе становится целевым активом и вектором атаки (к примеру, кража интеллектуальной собственности, инверсия модели и т.д.).

Задача архитектора по безопасности в данном случае — минимизировать риски утечек через саму модель. Ему приходится сталкиваться с проблемой чёрного ящика и проблемой тестирования. Сложные модели, например глубокие нейронные сети, часто не интерпретируемы. Недостаточно просто пройтись отладчиком — необходимо решать задачу верификации и валидации данных.

Проблемы с цепочками поставок имеют место как при классической разработке ПО, так и в случае применения AI-компонентов. Мы используем какие-то готовые модели, фреймворки, датасеты, которые нужно уметь анализировать, видеть проблемы, находя уязвимости в зависимостях AI-компонентов.

— Какова в идеальной модели роль специалиста по безопасности при проектировании нового ПО: консультант, куратор, диктующий условия, или часть команды?

— Здесь могут быть разные подходы. В одном случае мы идём в партнёрском тандеме с разработчиками и помогаем друг другу повысить уровень защищённости ПО. Бывает и так, что приходится всё-таки диктовать условия и настаивать на требованиях к безопасности. В идеале хотелось бы органично сочетать в проекте все три роли: консультанта, блюстителя регуляторных требований и «своего парня».

Без регулярного общения с командой, с архитекторами приложения обойтись нельзя. Специалист по безопасности должен осуществлять и консультационную роль. Да, возможно, он часто не скажет, как лучше написать тот или иной кусок кода, но зато компетентно разъяснит, какие практики безопасности необходимо внедрять. Кроме того, эти практики необходимо составить, утвердить и согласовать как с бизнесом, так и с архитекторами, памятуя об универсальных и специфических условиях их внедрения.

ДЕВУШКИ В РБПО

В ПРЕДДВЕРИИ 8 МАРТА BIS JOURNAL БЕСЕДУЕТ С ДЕВУШКАМИ, РАБОТАЮЩИМИ В СФЕРЕ РБПО

Карина НАПАДОВСКАЯ
руководитель Центра сертификации и соответствия стандартам АО «Лаборатория Касперского»

— Ваш путь в РБПО?

— Если бы в школе и институте мне сказали, что я буду работать в ИТ-компании, да еще иметь отношение к РБПО, я бы сильно посмеялась. С компьютером и новыми технологиями я всегда была на вы. Получала себе спокойно специальность экономиста-управленца в автомобильно-дорожном университете и собиралась стать министром транспорта. Но в одночасье жизнь кардинально изменилась, на 5-м курсе института мне нужно было срочно выйти на любую работу, и по воле судьбы я устроилась в одну из испытательных лабораторий и стала заниматься сертификацией ПО. За несколько лет карьеры стало очевидно, что моя сильная сторона — это умение руководить проектами, управлять командой и выстраивать коммуникации. Так я оказалась в Лаборатории Касперского, где искали именно такого специалиста, и создала центр сертификации, который на сегодняшний день является образцово-показательным в отрасли. Ну и здравствуй, РБПО!

— Что Вы считаете важным для развития РБПО в отрасли?

— Всегда говорила, что самое важное — это осознание, принятие и про-

движение РБПО руководством компании. Без полной поддержки, как ресурсной, так и бюджетной, выстроить РБПО нельзя. Можно сделать пародию на РБПО, временно решить точечную задачу, но создать отлаженную машину — бесперебойно работающий механизм, коим и должна стать безопасная разработка в компании,— нельзя.

— Одна-две забавных истории из вашей карьеры?

— Когда в 2016 году активно начались разговоры про РБПО, появились соответствующий ГОСТ и инициатива ФСТЭК по внедрению РБПО вендорами, я понимала, что просто попросить команды разработки что-то поделать — очень плохой вариант. Нужен масштабный подход, правильная цель, полная поддержка. Соответственно, нужно не просто плыть по течению и делать что-то, как все, чтобы формально соблюсти требования по сертификации (важно заметить, что все начинали внедрять РБПО, чтобы не было проблем с получением сертификатов, не для реальной пользы в разработке продуктов!) — нужно возглавить это направление. Стать первыми. Показать реальный профит. Руководство компании поддержало эту позицию, ведь Лаборатория Касперского — это флагман в области защиты информации, чья миссия — спасать мир. Ну и безусловным преимуществом было то, что практики безопасной разработки в части анализа уязвимостей и отдельные другие уже были внедрены в компании на высоком уровне. Этот путь мы начинали не с нуля. Дальше нужно было культивировать эту историю (здесь мы вспоминаем, что я на вы с новыми технологиями, про РБПО я знала немного). И я создала многим известное мероприятие Certification Day, которое в этом году пройдет уже в 11-й раз. Цель — на единой площадке с вендорами, регуляторами, испытательными лабораториями, органами по сертификации обсуждать практические проблемы внедрения практик безопасной разработки. Многие, как и я, знали об этом немного (смеюсь). Мы учились, делились опытом, обсуждали проблемы и подходы с регулятором, который слышал боль отрасли и корректировал нормативные документы таким образом, чтобы они были актуальными, полезными и жизнеспособными. Также мы обкатали на себе и первую методику ВУ и НДВ, продемонстрировав отрасли её выполнимость. Закрепили успех, переписав ГОСТ по безопасной разработке в команде с другими передовыми компаниями. За десять лет РБПО буквально переродилось. Это было рискованно. Ну а что, кто не рискует, тот не пьет шампанское, а шампанское я люблю.

— «Женский взгляд»?

— Женщины — уникальные существа. В нас уживается бесконечное множество противоречивых натур, и нам проще найти нужный ключик к сложной задаче, неоднозначному проекту, неразрешимому конфликту, нереальной цели. Важно оставаться женщиной в любой ситуации, на любой позиции, каждый день. Желаю нам всем не растерять свою суть, нести в мир, на работу и домой прекрасное, доброе и светлое — для этого мы и созданы!

Елена БЫХАНОВА
начальник отдела аудиторско-консалтинга процессов РБПО НТЦ «Фобос-НТ»

— Ваш путь в РБПО?

— «Оказалась в нужное время, в нужном месте» — так я пришла в РБПО три года назад, когда впервые встретилась с евангелистом РБПО Дмитрием Пономаревым в МГТУ им. Баумана на студенческой секции. Я попала в сферу в отличное время — как раз в самый разгар набирающего популярность движения, а именно популяризации парадигмы безопасной и качественной разработки. Положительно сказывающиеся на развитии сферы события происходят одно за другим: нарастает регулярность проведения тематических конференций, ведутся работы по созданию стандартов в техкоме (ТК362), начинают приглашать экспертов на интервью в журналы и подкасты. Тут же по стечению обстоятельств начинает приобретать востребованность аудитоконсалтинг, который становится моим основным профилем. Считайте, что в РБПО мы с ним пришли в одно и то же время, так как набирать популярность эта услуга начала как раз за год до выхода ГОСТ Р 56939–2024.

— Что Вы считаете важным для развития РБПО в отрасли?

— В дискуссиях с коллегами по отрасли часто всплывает вопрос о том, что является важным для развития РБПО. Сперва на него хочется отреагировать обратным вопросом: «А кто спрашивает: регулятор, разработчик, потребитель?». Хотя, впрочем, это неважно, ответ один: «Не переставать видеть в этом ценность для России».

От потребителя должны исходить запросы, от разработчика — готовность и инициатива, от регулятора — неукоснительность и разумность. Развиваться будет то, на что есть спрос, что считается перспективным, в чем видят необходимость. А для того, чтобы РБПО было *тем самым*, требуется повышать уровень культуры всех заинтересованных сторон, демонстрируя ценность результатов для решения задачи повышения качества и конкурентоспособности отечественного программного обеспечения.

— Одна-две забавных истории из вашей карьеры?

— Что ж, их много, каждый день сплошные забавы :) А главная из них, даже не знаю, квалифицировать это в положительном или отрицательном ключе, следующая: «нет цели, есть только путь». Как аудитоконсалтер, я должна постоянно наращивать собственные (и членов моей команды) компетенции в вопросах выстраивания процессов РБПО и их качественной оценки. При этом с годами я начинаю осознавать, что «чем больше я знаю, тем больше я

понимаю, что ничего не знаю». Для меня, специалиста, занимающегося распространением знаний об РБПО в массы и несущего ответственность за качество передаваемых знаний, важно всегда обладать актуальной информацией и высококлассной экспертизой в сфере, которая развивается невероятно быстро. Так что факт того, что сколько ты в этом РБПО ни развивайся, всё равно конца не увидишь, остается таковым — «забавным». Отсюда, дорогие читатели-разработчики, второй «забавный» факт: и вы тоже со временем приходите к такому осознанию и затем начинаете понимать, что требуется в рамках процесса №1 ГОСТ Р 56939–2024, а именно постоянный анализ, развитие, реализация ;)

— «Женский взгляд»?

— Как я понимаю, здесь надо написать «от женщины для женщин», ну тогда: «Берегите мужчин!». Поддерживайте тех, кто выше вас, и помогайте тем, кто ниже вас. Всем дамам в преддверии праздника желаю осознать свою ценность и применить навыки во имя общего Дела.

Евгения РЕМЕЗОВА
начальник отдела технической экспертизы ООО НТЦ «Фобос-НТ»

— Ваш путь в РБПО?

— Моя история в РБПО началась более пяти лет назад. Тогда это была еще небольшая региональная компания, основанная выпускниками Орловской академии ФСО России в городе Орле. Начав как технический писатель, я получила ценный опыт в сфере информационной безопасности, изучила сертификацию, специфику ИТ-компаний и важность качественной документации. Затем перешла в отдел технической экспертизы, где занималась стандартизацией документации, участвовала в разработке нормативных документов по вопросам информационной безопасности и проведении аудитов на соответствие требованиям РБПО. Сейчас я возглавляю этот отдел, что позволяет мне не только решать конкретные задачи компании, но и формировать будущее отрасли, определяя стандарты и подходы к обеспечению безопасности ПО.

— Что вы считаете важным для развития РБПО в отрасли?

— Развитие практики безопасной разработки ПО является критически важной задачей для современных компаний. На мой взгляд, ключевыми факторами успешного внедрения РБПО являются три основных направления:

1. Стандартизация подходов. Крайне важна единая система стандартов и регламентов, регулирующих процесс разработки и эксплуатации ПО. Отсутствие четких критериев оценки, различие в понимании терминов — всё это ведет к разночтениям между разработчиками и регулятором. Единая терминология необходима для правильного понимания требований и корректного исполнения задач всеми участниками процесса. Стандарты должны охватывать широкий спектр аспектов, начиная от выбора инструментов анализа уязвимостей и заканчивая процедурами тестирования и контроля качества.

2. Развитие и улучшение существующих стандартов. Любые стандарты требуют постоянного обновления и совершенствования. Технологии развиваются стремительно, появляются новые угрозы и методы атак, соответственно, требования к защите должны регулярно пересматриваться и обновляться. Для этого необходимо тесное взаимодействие разработчиков с регулятором. Важно поддерживать открытый диалог, способствующий формированию наиболее эффективных рекомендаций и решений. Регулярные семинары, конференции и рабочие группы способствуют обмену опытом и созданию единой базы знаний, необходимой для качественного развития стандарта.

3. Автоматизация процессов документирования. Одним из ключевых факторов повышения эффективности разработки является автоматизация процессов создания и поддержки технической документации. Инструменты автоматизированного документирования позволяют сократить временные затраты на рутинные операции, повысить точность информации и снизить вероятность человеческих ошибок. Все это способствует повышению прозрачности процессов и упрощению управления проектами.

— Одна-две забавных истории из вашей карьеры?

— Курьёзы случаются. Из последних: при проверке руководства пользователя на последней странице на всю ширину была подпись: «Этот раздел намеренно оставлен пустым». Я удаляю текст и оставляю страницу действительно пустой. Отправляю заявителю, на что приходит ответ: «Почему пропал мой секретный раздел?» Оказалось, что изначально этот раздел должен был содержать важную техническую информацию, но отдел разработки забыл заполнить его содержимым, решив временно вставить шутливую надпись.

В документации по одному из процессов РБПО, в разделе, где подробно должны были быть описаны методы проверки безопасности, встречается фраза: «Использованы лучшие практики безопасной разработки». Что, конечно, говорило о непонимании процесса и несерьезном отношении к документации и работе в целом.

— «Женский взгляд»?

— Цитируя слова песни It's a Man's Man's Man's World: «Этот мир создан мужчиной, но он был бы бессмысленным без женщин», хочу добавить, что будущее индустрии информационной и кибербезопасности зависит в первую очередь от профессионалов, готовых брать ответственность и создавать условия для дальнейшего развития отрасли. Так почему бы таким профессионалом не стать тебе?

Елена МАНЬЖОВА
эксперт органа по сертификации ИСП РАН

— Ваш путь в РБПО?

— Мой путь в РБПО стартовал 10 лет назад, когда я начала заниматься сертификацией ПО в рамках систем сертификации средств защиты информации. Быстро стало очевидным, что один скромный абзац в техническом задании о соответствии разрабатываемого ПО требованиям по безопасности информации, расположенный где-то вдали от функциональных требований,— это только верхушка айсберга, под которой скрывается огромная глыба требований, способная привести к срыву работ. Выполнение этих требований без пересмотра подходов и процессов непосредственно разработки — затруднительно и очень затратно, особенно на последнем этапе.

— Что вы считаете важным для развития РБПО в отрасли?

Интересно, что в настоящее время, даже если организация не учитывает требования безопасности при разработке ПО, так как обязательная сертификация как будто бы не нужна, а разработчик не задумывается о безопасности кода, который пишет, или о безопасности заимствованных компонентов, которые использует,— это не освобождает их впоследствии от необходимости этим требованиям соответствовать. Ни одному уважающему себя заказчику не нужно небезопасное ПО. Терять выручку или доверие пользователей после компьютерного инцидента с использованием уязвимостей небезопасного ПО — вот что реально болезненно.

Наверное, осознание этого является наиболее важным для развития РБПО в отрасли. РБПО — это инвестиция в свою репутацию (или в свое резюме), в качество и место на рынке. И это не так сложно, как кажется, но требует в первую очередь правильной мотивации, а во вторую — правильных проводников.

— Одна-две забавных истории из вашей карьеры?

— Из своего профессионального опыта хочется сказать следующее: чем быстрее удастся перейти из стадии отрицания в стадию принятия и начать перестраивать свои процессы с учетом концепции РБПО, тем лучше.

Иногда разработчики пытаются обосновать отсутствие необходимости в реализации некоторых процессов РБПО, проявляя удивительную смекалку: ссылаются на статьи гражданского кодекса или на тот факт, что разработка является коммерческой и «без обязательств», что их ПО не является средством защиты информации, приводят интересные толкования нормативной базы. Спойлер: это не помогает, когда начинают срываться сделки. Также ничем хорошим не заканчивается замалчивание, когда разработка ведется вне концепции РБПО, а разработчики приходят к соответствующим специалистам уже после выпуска продукта на рынок при возникновении проблем с реализацией.

Поэтому важно последовательно выстраивать культуру и процессы РБПО в организации в целом, с определённой очередностью и дифференциацией требований в зависимости от вида, назначения и области применения ПО.

— «Женский взгляд»?

— В преддверии праздника 8 Марта отдельно хочется сказать всем женщинам, которые занимаются вопросами информационной безопасности, сертификации и обеспечением безопасности разработки,— никогда не сомневаться в себе, с юмором относиться к предрассудкам и даже использовать их в свою пользу, ведь женщина способна добиться успеха там, где мужчины встречают сопротивление. Желаю всем легкой безопасности!

Натали ДУБОТОЛКОВА
руководитель группы обеспечения безопасности приложений ООО Базис

— Ваш путь в РБПО?

— Я начинала свою карьеру как обычный инженер-программист .NET, но на стажировке выпала возможность дополнительно занять место Security Champion в проекте. На тот момент я даже не представляла, во что вписалась, и как это изменит мою карьеру. Именно там я впервые погрузилась в моделирование угроз и попробовала свои силы в проектировании безопасной архитектуры. Там же я впервые столкнулась с непониманием со стороны разработки необходимости усложнять и менять существующую устоявшуюся архитектуру. После стажировки я сознательно выбрала роль AppSec-инженера в той компании. И поняла: вот он, мой мир, где есть чёткие правила, строгость и однозначность. Мне это оказалось гораздо ближе, чем творческая неопределённость в чистой разработке.

— Что вы считаете важным для развития РБПО в отрасли?

— Я очень рада, что сейчас всё больше и больше говорят о важности безопасности в жизненном цикле разработки. В целом для отрасли я считаю ключевым смещение фокуса с формальных показателей на суть. Нам нужна гонка за качеством и реальной безопасностью продукта, а не за красивыми сертификатами на стене или погоней за новыми функциями любой ценой. К сожалению, пока это часто остаётся мечтой о мире розовых единорогов.

— Одна-две забавных истории из вашей карьеры?

— Любовь к учёбе — мой конёк, бакалавриат и две магистратуры тому доказательство. Параллельно с рабо-

той специалистом по безопасности кода я изучала и другие направления информационной безопасности. Так я познакомилась с тестированием на проникновение и начала использовать разные сканеры для работы. Изучив простые виды уязвимостей, я переходила к более сложным и изощрённым и, конечно же, искала именно такие на практике. Однажды мне выпала возможность провести свой первый реальный пентест на сайте внутренних услуг компании. Я, полная амбиций, сразу нырнула в недра сайта в надежде найти что-то «серьёзное» и показать, на что я способна. Не найдя ничего «достойного», я уверенно отписала в заключении, что сайт безопасен. А через неделю пришло уведомление: на том самом сайте обнаружена обычная, классическая XSS. Та самая, куда можно вставить alert(1) — и это сработает. Сказать, что я была в шоке — ничего не сказать. История обошлась небольшим выговором и фразой «горе от ума», которая за мной закрепилась до конца работы в той компании. Зато с тех пор я твёрдо усвоила правило: проверку всегда начинать с самого простого и элементарного. Потому что иногда самое очевидное оказывается самым опасным.

— «Женский взгляд»?

Часто говорят про какую-то особую «женскую интуицию». Я бы назвала это скорее натренированным вниманием к мелочам. И особенно — к моментам, когда всё вокруг слишком идеально. Именно в такие периоды затишья у меня включается режим максимальной боевой готовности. Потому что опыт уже научил: самые критичные уязвимости и сломанные пайплайны появляются чаще всего в этой обманчивой тишине. Так что советую и себе прошлой, и всем, кто чувствует подобный внутренний звоночек,— прислушиваться к нему. Возможно, это просто голос вашего же опыта, который тихо подсказывает, куда смотреть. В безопасности нет мелочей, и такая интуиция — это ваш первый и самый важный сканер аномалий.

И ещё один момент, важный лично для меня. Говоря о «женском взгляде», я не сталкивалась с ситуациями, где мой пол создавал бы трудности. В безопасности, как я её вижу, нет «мужских» или «женских» ролей. Есть специалист, который либо видит архитектурные риски, уязвимости, угрозы и умеет их объяснить, либо нет. И я верю, что будущее именно за такой средой, где главный критерий — не кто ты, а что ты знаешь и как умеешь применять свои знания.

Анастасия КАЛУГИНА
руководитель направления безопасной разработки и инфраструктуры компании «ИнфоТеКС»

— Путь в РБПО или из каких кирпичиков он строится?

— Мой путь в безопасной разработке начался вполне классически — с программирования. При техническом складе ума, родителях-программистах и активно развивающейся ИТ-отрасли у меня не было ни единого шанса. С профессией я определилась ещё в школе, выучилась по этому направлению и продолжаю получать новые знания в дополнение к базе, с которой все начиналось. Во время учёбы на программиста я нашла подработку с гибким графиком в области системного администрирования, поэтому практический опыт в ИТ начался с этого направления. Позднее к нему прибавилась работа по направлению backend-разработки.

Что примечательно — две ипостаси («Dev» и «Ops») «жили» у меня параллельно, практически не пересекаясь, около десяти лет. Аббревиатура DevOps уже появилась, опыт в обоих направлениях накопился немаленький, но не было комплексного понимания этого термина. А потом я поменяла работу. Пришла на очень интересный инновационный проект, где не существовало готовых решений, и подход и механизмы для решения поставленных задач пришлось прорабатывать с нуля. Спустя какое-то время я поняла, что ранее абстрактный эпитет «DevOps» применительно к разрабатываемой системе перестал быть для меня сухой аббревиатурой и трансформировался во вполне определенную концепцию. И эта концепция сильно отличается от общепринятого определения. Для меня DevOps — это подход, где единственный критерий — оптимальность: если конкретную задачу эффективнее решить набором конфигураций или скриптов — автоматизируем, если лучше использовать приложение — кодируем. Хотя это уже скорее «DevArchOps», так как оценивать нужно архитектуру разрабатываемой системы в целом.

Понимание, как соединить разноплановые компетенции и какие это дает выгоды, выкристаллизовалось не сразу, но оно очень пригодилось для развития и обогащения знаний. И далее, формирование уже третьего направления до полноценного DevSecOps было для меня вполне осознанным. А потом активно стал развиваться подход разработки безопасного ПО (РБПО), и DevSecOps органично обогатился необходимостью:

- обеспечения безопасности в привязке к процессам разработки и на каждом этапе жизненного цикла ПО;
- обеспечения безопасности цепочки поставок;
- управления уязвимостями разрабатываемого и эксплуатируемого ПО;
- управления обучением сотрудников.

На мой взгляд, РБПО — на данный момент единственный системный подход для комплексного обеспечения безопасности разрабатываемого ПО. Нельзя выделить приоритетное требование или главный этап жизненного цикла ПО, только системный подход позволяет максимально обезопасить разработку продукта.

А теперь к теме женщин в РБПО. Еще пять лет назад я целенаправленно искала хотя бы пару девушек, чтобы разбавить коллектив из 40 мужчин-безопасников. Тогда результат был печальным — чтобы найти пер-

вую кандидатуру, потребовалось полгода интенсивных поисков. Девушки в ИБ просто не шли. Сейчас ситуация наконец выравнивается — отрасль безопасной разработки стала активнее развиваться, ее необходимость на рынке стала очевидно продвигаться, и женщины в безопасность стали приходить активнее. И это отлично! Когда над решением задачи работают специалисты с разноплановым опытом и подходами, когда привносят большее количество идей — это всегда позволяет достичь лучшего результата.

Альбина АСКЕРОВА
руководитель направления по взаимодействию с регуляторами Swordfish Security

— Ваш путь в РБПО?

— Мой путь, как это часто бывает, не был прямым, но вектор своего развития в ИТ я определила ещё в школе. У меня были прекрасные учителя математики и физики, которые показали «магию» точных наук и зажгли интерес на годы вперед.

Я поступила в технический вуз на направление сетей связи. Любимый Бонч (СПбГУТ им. проф. М. А. Бонч-Бруевича) стал важным этапом моего профессионального становления. На последнем курсе я уже работала по специальности, а ближе к защите диплома поняла, что задачи по защите информации увлекают меня сильнее. Сразу после выпуска поступила в магистратуру по информационной безопасности и начала работать в ИБ. И вот тут всё закрутилось!

Старт в отделе ИБ крупной госкомпании был классическим: политики, антивирусы, инциденты. Со временем пришло понимание, что мы чаще «латаем дыры», реагируя на проблемы. Это было объяснимо: молодой отдел, ограниченные ресурсы, масштабная инфраструктура и множество регуляторных требований.

Постепенно команда взрослела: выстроили процессы, начали автоматизацию, внедряли инструменты. Благодарна своим руководителям за доверие и пространство для развития.

Следующим этапом развития процессов уже моего Управления ИБ стала разработка — внутренний код и заказные решения. Это закономерно привело к погружению в мир практик безопасной разработки, а затем и в РБПО. Оглядываясь назад, понимаю, что сегодня стратегия была бы другой, но именно этот опыт дал мне прочную опору в текущих проектах.

Сложно переоценить, сколько сил, времени и души вкладывают люди, развивающие ИБ и РБПО в нашей стране, и я рада сегодня быть частичкой этого!

— Что вы считаете важным для развития РБПО в отрасли?

— На мой взгляд, ключевое — культура. Можно внедрить лучшие инструменты и стандарты, но без сдвига в отношении к созданию продуктов это упрётся в стену. Безопасность кода со временем должна стать частью профессиональной гордости и осознанной ответственности разработчика. Сейчас многое делается: развиваются сообщества, проводятся мероприятия, обновляется регулирование, меняются образовательные программы. Наверное, остаётся заложить основы безопасного ПО ещё в обучении в школе в инженерных классах, чтобы это стало естественной частью мышления будущих специалистов.

— Одна-две забавных истории из вашей карьеры?

— Один забавный случай произошел недавно. Я опаздывала с одной встречи регулятора на другую. Забежав в уже полный кабинет, заняла одно из последних свободных мест с краю стола. Оказалось, это место рядом с главой стола. Вошел инициатор встречи, и по законам жанра первые вопросы адресовали тем, кто сидел рядом, включая меня. Со временем начинаешь относиться к таким ситуациям с юмором и здоровым спокойствием.

— «Женский взгляд»?

— Убеждена, что ключ к успеху — любознательность и искренний интерес к своему делу. В технических сферах женщинам часто помогают качества: чуткость, умение работать в команде, гибкость, способность понимать не только слова, но и мотивы людей. Мы все разные, и в этом наша сила. Важно осознать и использовать свою индивидуальность, слышать друг друга и действовать сообща.

Нелли МАГАКЕЛОВА
руководитель группы Центра оценки соответствия и тестирования АО «НПО «Эшелон»

— Ваш путь в РБПО?

— Уже более четырёх лет моя профессиональная деятельность связана с информационной безопасностью. Мой путь начался с тестирования различных программ, преимущественно относящихся к банковскому сектору. Спектр задач был широким: от анализа кода и поиска уязвимостей до разработки документации. Этот опыт дал мне возможность не только находить слабые места, но и видеть, как команды реагируют на них, как принимаются решения и как устраняются проблемы в продуктах, которыми пользуются миллионы людей. Со временем пришло понимание, что безопасность — это не только про жизненный цикл уязвимости, но и про культуру команды. Позже мне удалось участвовать в полноценной разработке программного обеспечения с нуля, где мне приходилось быть одновременно разработчиком, аналитиком и архитектором. Этот опыт дал понимание того, как рождаются решения, какие компромиссы приходится искать для баланса между безопасностью и промышленной разработкой.

— Что вы считаете важным для развития РБПО в отрасли?

Что же действительно движет РБПО вперёд? На мой взгляд, ключ не только в регламентах и стандартах, а прежде всего в осознанности людей. Безопасность не должна быть обособленным этапом, про который обычно вспоминают перед релизом. Она должна стать способом мышления всей команды разработки. Не просто формальное выполнение требований, а постоянные вопросы в стиле: «Что будет, если злоумышленник попробует вот так?». Когда вопрос «Для чего нам РБПО?» становится общим языком разработчиков, аналитиков, тестировщиков и менеджеров, РБПО перестаёт быть бюрократией и превращается в инструмент создания качественных продуктов. И, что немаловажно, такой подход почти всегда оказывается экономически выгоднее, поскольку он снижает количество критических инцидентов, доработок и репутационных потерь.

— Истории из карьеры?

— Было немало случаев, когда даже продукт с высоким уровнем зрелости оказывался проблемным с точки зрения безопасности. Однажды с помощью одного отправленного вредоносного запроса удалось вывести из строя многопользовательский сервис с реальными клиентами. Как позже выяснилось, это произошло из-за неправильно настроенной инфраструктуры. Также об одном забавном случае рассказал слушатель нашего курса. При обсуждении фаззинг-тестирования в контексте непредсказуемости поведения программы коллега поделился историей, как оператор системы систематически выводил из строя программу из-за тремора рук, поскольку он отправлял запрос чаще, чем система могла бы обработать. Такие примеры демонстрируют, что порой аварийное завершение программы может возникнуть не только из-за ошибок, допущенных при разработке или проектировании, но и из-за банального человеческого фактора.

— «Женский взгляд»?

— Для девушек, которые думают об этом пути, хочется сказать следующее: не позволяйте стереотипам определять ваш выбор. История развития информационных технологий свидетельствует о том, что женщины всегда играли значимую роль в развитии этой области: Ада Лавлейс, Грейс Хоппер, Маргарет Гамильтон, Хеди Ламарр — и это лишь самые известные имена из тех, кто внёс огромный вклад в развитие компьютерных наук. Важно помнить, что в любой профессии ценится в первую очередь глубина мышления, любознательность, инициатива. Люди без индивидуальности — это всего лишь копии копий копий, а миру особенно нужны те, кто способен мыслить самостоятельно и видеть дальше шаблонов. Поэтому, если вас интересует мир разработки безопасного программного обеспечения, не бойтесь идти вперёд и создавать будущее, в котором технологии будут служить безопасно и надёжно.