УДК 004.056.53

ПОДХОД К МОНИТОРИНГУ БЕЗОПАСНОСТИ РАСПРЕДЕЛЁННЫХ ПРИЛОЖЕНИЙ

Автор - Пеньков Д.В. (ВКА)
Научный руководитель – кандидат технических наук Крюков Р.О. (ВКА)

Введение. Современные высоконагруженные распределённые приложения, функционирующие в средах контейнерной виртуализации (Kubernetes), сталкиваются с проблемой «слепых зон» безопасности. Традиционные SIEM-системы и средства мониторинга приложений (APM) работают изолированно: первые анализируют системные логи, вторые - метрики производительности. Это не позволяет восстановить полную картину инцидента, например, связать запуск вредоносного процесса в контейнере (системное событие) с конкретным HTTP-запросом, который его инициировал (прикладной контекст) [1, 2].

Основная часть. Технология eBPF предоставляет возможность глубокого и безопасного мониторинга ядра Linux (перехват системных вызовов execve, connect, open), оставаясь невидимой для атакующих и не требуя модификации кода приложений [3]. Однако события eBPF (например, от Falco или Tetragon) генерируются без привязки к бизнес-логике неизвестно, через какой микросервис и от какого пользователя пришла команда. С другой стороны, OpenTelemetry (OTel) стал стандартом де-факто для сбора трассировок, метрик и логов, описывающих путь запроса через распределённую систему, но он не видит действия на уровне ОС [4].

Новизна предлагаемого подхода заключается в создании единой плоскости анализа (Security Observability) путём симбиоза eBPF и OpenTelemetry. В отличие от существующих решений (Sysdig, Aqua Security), которые используют eBPF изолированно или требуют проприетарных агентов, предложенная архитектура впервые реализует обогащение системных событий прикладным контекстом через открытый стандарт OpenTelemetry. Это позволяет не просто зафиксировать факт execve("/bin/sh"), а идентифицировать его как неотъемлемую часть конкретной транзакции (trace), пришедшей, например, от неавторизованного клиента через уязвимый эндпоинт API. Для достижения этой цели разработана архитектура, интегрирующая eBPF-события от Cilium Tetragon в пайплайн OpenTelemetry Collector (рисунок 1).

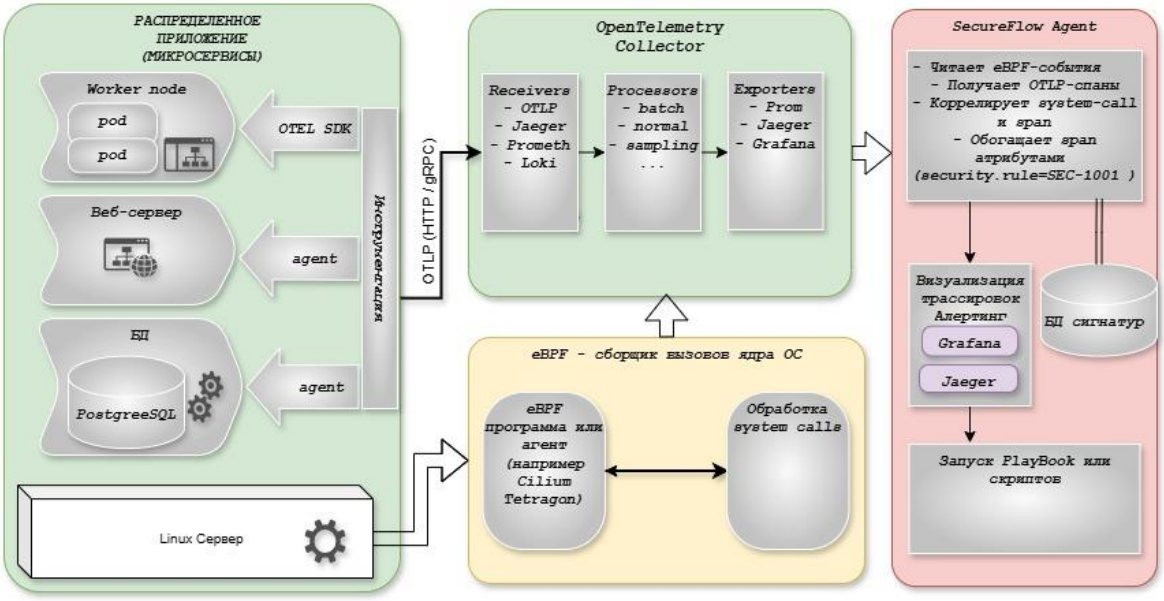

Рисунок 1 - Архитектура системы мониторинга безопасности распределенных приложений

Ключевым механизмом является корреляция данных на основе общих идентификаторов окружения (namespace, pod_name, PID) и, в перспективе, внедрение trace_id в пространство процессов. Предлагаемая система (рисунок 1) реализована на базе Kubernetes-кластера и включает в себя: микросервисное приложение (Gateway, Auth, Order), инструментированное OTel SDK; слой сбора системных событий (Tetragon/eBPF); центральный процессор — OpenTelemetry Collector, выполняющий обогащение и маршрутизацию данных; бэкенды хранения и визуализации (Jaeger для трассировок, Prometheus для метрик, Loki для логов и Grafana как единая панель).

Выводы. Таким образом, предложенная архитектура комплексного мониторинга на базе eBPF и OpenTeleplicity меняет парадигму обеспечения безопасности в облачных средах, превращая наблюдаемость из пассивного инструмента диагностики в активный механизм предотвращения атак. Решение является языко- и платформонезависимым, полностью масштабируемым и легко встраивается в существующий DevSecOps-стек организаций. Архитектура полностью соответствует принципам современных облачных систем, так как все компоненты от eBPF-проб до OTel Collector и Alertmanager поддерживают горизонтальное масштабирование, что позволяет использовать их в кластерах с тысячами узлов без потери производительности. Наконец, интеграция с уже устоявшимся стеком DevSecOps: Grafana для визуализации, Jaeger для трассировки, Alertmanager для оповещения гарантирует минимальные барьеры для внедрения и возможность поэтапной адаптации без замены существующих инфраструктур

Список использованных источников:

1. *Кузнецов, А. А.* Проблемы обеспечения безопасности в микросервисных архитектурах // Вопросы кибербезопасности, 2023 г. - стр 45-53.
2. *Smith, J.* The illusion of security: Why logs are not enough // ACM Queue, 2022 г. - стр. 30-40.
3. *Gregg, B.* BPF Performance Tools: Linux System and Application Observability, Addison-Wesley, 2019. - 300 стр.
4. OpenTelemetry Documentation: Concepts and Architecture [Электронный ресурс].