???????? ? Kaspersky EDR Expert 8.1: ?????? ???????????

> **i Информация:** Приведенная на данной странице информация, является разработкой команды pre-sales и/или AntiAPT Community и НЕ является официальной рекомендацией вендора.

Плейбук (Playbook) в Kaspersky EDR Expert — это автоматизированный сценарий реагирования на инциденты информационной безопасности. Плейбук запускает алгоритм, включающий последовательность действий по реагированию, которые помогают анализировать и обрабатывать алерты или инциденты.

????? ????? ????????

Плейбуки решают три ключевые задачи SOC:

1. **Скорость реагирования.** Вместо того чтобы аналитик вручную собирал хеши, искал хосты и удалял файлы — плейбук делает это за секунды.
2. **Стандартизация.** Один и тот же тип инцидента обрабатывается одинаково, независимо от того, кто дежурит в смену.
3. **Разгрузка аналитиков.** Рутинные действия автоматизируются, и аналитик может сосредоточиться на расследовании.

??????? ??????????

- **Автоматическое реагирование на критические угрозы** — ransomware, sandbox-детекты, компрометация учётных записей
- **Первичная обработка алертов** — сбор информации, обогащение данными TI
- **Ручное реагирование** — аналитик выбирает инцидент и запускает плейбук
- **Расследование** — сбор дампов памяти, образов дисков, ключей реестра

??????????

Эта статья основана на:

- Официальной справке **Kaspersky EDR Expert 8.1**

- Материалах **KUMA Community** (статьи «Триггеры в плейбуках» и «Действия в плейбуках»)
- Практическом опыте настройки плейбуков

Автор не является сотрудником «Лаборатории Касперского». Примеры и рекомендации носят характер best practices и могут требовать адаптации под конкретную инфраструктуру. Перед применением в продуктивной среде обязательно тестируйте плейбуки в режиме **«Обучение»**.

> **Важно:** Синтаксис jq-выражений, модель данных и имена полей могут меняться между версиями. Всегда проверяйте актуальную документацию для вашей версии.

???????? ????????????? ?????????

Плейбуки в Kaspersky EDR Expert 8.1 предоставляют следующие возможности:

???????? ???????????

Возможность	Описание
Автоматический запуск	Плейбук срабатывает по триггеру при обнаружении соответствующих алертов/инцидентов
Ручной запуск	Аналитик сам выбирает объекты и запускает плейбук
Триггеры на jq	Гибкая фильтрация по свойствам, активам, наблюдаемым объектам, событиям
19 действий по реагированию	От сбора форензики до изоляции хостов и блокировки учётных записей
Визуальный редактор	Создание алгоритмов без написания JSON вручную
Тестовый режим	Эмуляция запуска без выполнения реальных действий
Ручное подтверждение	Требование одобрения перед опасными действиями

Версионирование	Автоматическое сохранение истории изменений с возможностью отката
Наследование тенантов	Плейбуки автоматически доступны дочерним тенантам

???? ?????????

В системе существует два типа плейбуков:

??????????????? ???????? [KL]

Созданы специалистами "Лаборатории Касперского" и отмечены префиксом `[KL]` в названии.

Особенности:

- **Нельзя изменять алгоритм** (можно изменить только режим работы и триггер)
- **Нельзя удалять**
- Можно **дублировать** и настраивать копию
- По умолчанию работают в режиме **Обучение**
- Требуют предварительной настройки KUMA (обогащение событий)

??????????????? ????????

Создаются и настраиваются администраторами SOC под конкретные задачи.

Обязательные параметры при создании:

1. **Область действия** (алерт или инцидент)
2. **Триггер** (условия автоматического запуска)
3. **Алгоритм** (последовательность действий)

Импорт/экспорт: Можно использовать XDR REST API для переноса плейбуков между тенантами.

???? ????????? ? ??????? ????????

??????????? ???????????: ??????? ????????

Плейбук может работать **ТОЛЬКО** с алертами **ИЛИ** инцидентами. Нельзя создать плейбук, который обрабатывает оба типа объектов.

Почему это важно:

- Если выбрали "Алерт" → в jq-выражениях используем `alert.*`
- Если выбрали "Инцидент" → в jq-выражениях используем `incident.*`
- Несоответствие приведёт к ошибке: **"Expression does not match the selected scope"**

???????? ?????????

Плейбуки НЕ могут запускаться автоматически для дочерних инцидентов!

Что это значит:

- Если у вас настроена сегментация и создаются дочерние инциденты
- Автоматические плейбуки сработают только для **родительского** инцидента
- Для дочерних инцидентов нужно запускать плейбук **вручную**
- Перед ручным запуском нужно выбрать активы и наблюдаемые объекты

Обходное решение:
Использовать сегментацию так, чтобы критичные инциденты не становились дочерними, или запускать плейбуки вручную через интерфейс.

???????????? ????????

Плейбук принадлежит **одному тенанту** и **автоматически наследуется** всеми дочерними тенантами.

Особенности:

- Наследуются даже дочерние тенанты, добавленные **после** создания плейбука
- Можно **отключить наследование** при создании или изменении плейбука
- Дочерние тенанты могут дублировать и настраивать унаследованные плейбуки

Практический совет:
Если плейбук специфичен для конкретного тенанта (например, использует уникальные интеграции), сразу отключайте наследование, чтобы не засорять другие тенанты.

??????????? ? ?????? ??????

Прежде чем писать триггеры и действия, нужно понять, **как устроен плейбук внутри**. Это критически важно — 90% ошибок при создании плейбуков связаны с неправильным пониманием контекста выполнения и модели данных.

?????????? ?????? vs ????????????? ??????

В плейбуке есть два типа данных, и их **нельзя путать**:

Тип данных	Обращение	Доступ	Назначение
Глобальные	`alert.*` или `incident.*`	Только чтение	Информация об алерте/инциденте
Операционные	`.input.*`	Чтение и запись	Данные, передающиеся между шагами

Пример:

```
// Глобальные данные — имя инцидента
incident.Name

// Глобальные данные — хеши из наблюдаемых объектов инцидента
[incident.Alerts[].Observables[] | select(.Type == "sha256") | .Value]

// Операционные данные — то, что мы сохранили ранее через updateData
.input.hashes
```

> **Правило:** Глобальные данные доступны на любом шаге, но их нельзя изменить. Операционные данные создаются и меняются шагами `updateData`.

?????? ??????: ????? vs ????????

Это **фундаментальное различие**, от которого зависит весь синтаксис плейбука.

????? (Alert)

Алерт — это одиночное событие или группа связанных событий, сгенерированных правилом корреляции. Структура:

```
alert
├── Name, Severity, Status, MITRE...
```

```
├── Assets[]              ← активы (хосты, пользователи)
├── Observables[]         ← наблюдаемые объекты (хеши, IP, URL)
├── BaseEvents[]          ← нормализованные события
├── OriginalEvents[]      ← исходные события
└── ScannedFiles[]        ← просканированные файлы
```

???????? (Incident)

Инцидент — это агрегация нескольких алертов, объединённых по расследованию. Структура **вложенная**:

```
incident
├── Name, Severity, Priority, Status...
└── Alerts[]              ← массив алертов
    ├── Assets[]          ← активы каждого алерта
    ├── Observables[]     ← наблюдаемые объекты каждого алерта
    ├── BaseEvents[]
    └── OriginalEvents[]
```

> **Критически важно:** Чтобы получить активы из инцидента, нужно писать `incident.Alerts[].Assets[]`, а **не** `incident.Assets[]` — такого поля не существует!

???????? ???????????: ??????? vs ????????

Это самая частая причина ошибок «Expression does not match the selected scope».

? ???????? ????????

Триггер определяет, **на какие алерты/инциденты срабатывает плейбук**. Контекст — это **сам алерт или инцидент**, поэтому обращение идёт **без префикса**:

```
// ПРАВИЛЬНО для триггера на алерте
.Severity == "Critical"
.Name | contains("Ransomware")

// ПРАВИЛЬНО для триггера на инциденте
.Alerts[] | .Name | contains("Sandbox")
.Severity == "High"

// НЕПРАВИЛЬНО для триггера
incident.Alerts[] | ...   // префикс incident в триггере не нужен
```

```
alert.Severity            // префикс alert в триггере не нужен
```

? ????????? ???????? (executionFlow)

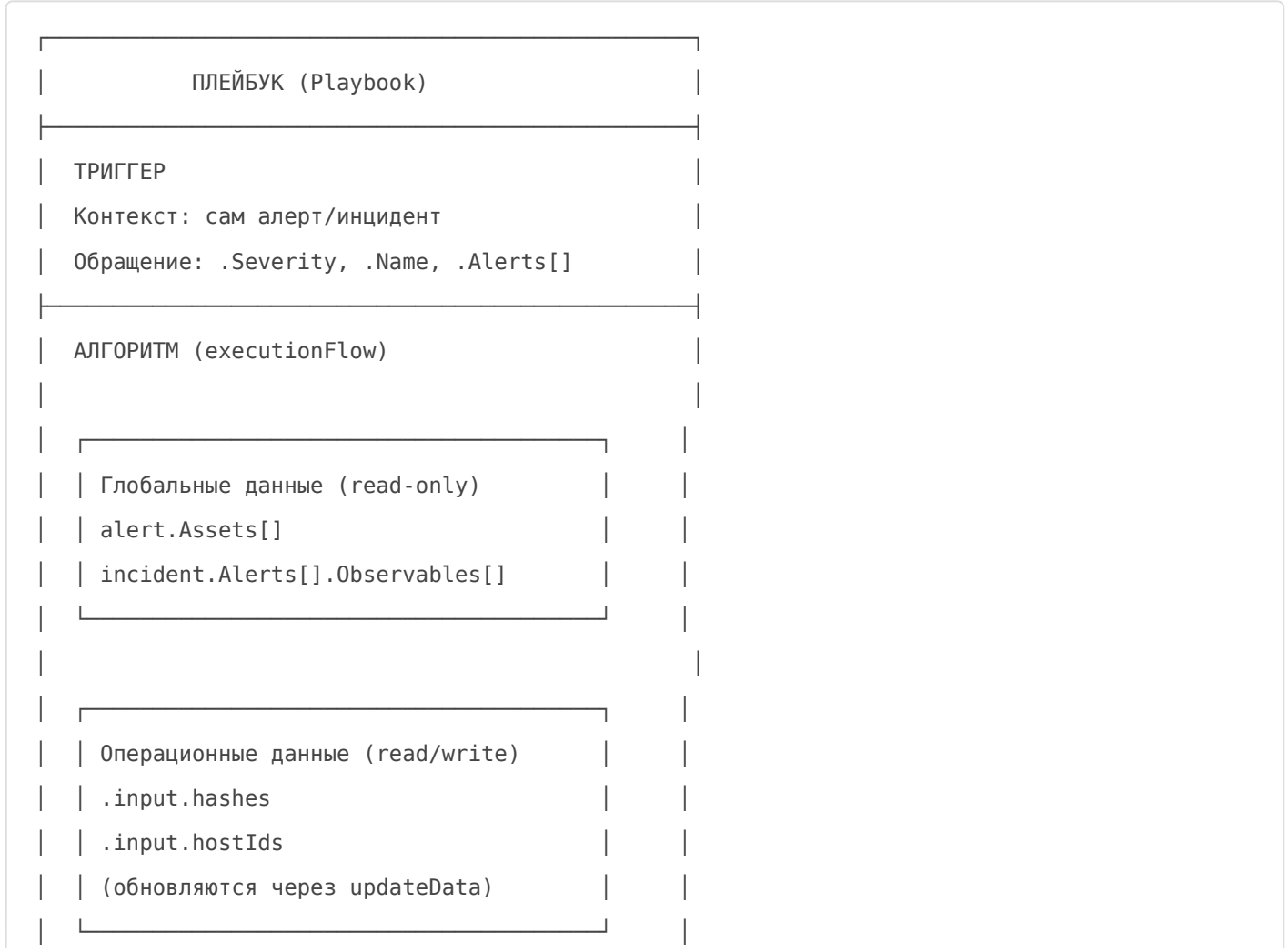
Алгоритм — это шаги, которые выполняются после срабатывания триггера. Здесь контекст другой, и нужно использовать **полные пути**:

```
// ПРАВИЛЬНО для алгоритма, работающего с инцидентом
incident.Alerts[].Assets[] | select(.Type == "host") | .ID

// ПРАВИЛЬНО для алгоритма, работающего с алертом
alert.Assets[] | select(.Type == "host") | .ID

// ПРАВИЛЬНО для доступа к операционным данным
.input.hashes
.input.hostIds
```

?????????? ?????????



```
│                                                  │
│  Шаги: action → decision → loop → parallel       │
└──────────────────────────────────────────────────┘
```

?????????????? ? ???????? (????????)

Kaspersky EDR Expert **чрезвычайно чувствителен к регистру символов** во всех jq-выражениях. Даже одна буква в неправильном регистре приведёт к ошибке выполнения плейбука.

??????? ?????? ??-?? ????????

```
// ❌ ОШИБКА: Incident с заглавной буквы
Incident.Alerts[].Assets[]

// ✅ ПРАВИЛЬНО: incident строчными буквами
incident.Alerts[].Assets[]

// ❌ ОШИБКА: assets строчными буквами
alert.assets[]

// ✅ ПРАВИЛЬНО: Assets с заглавной буквы
alert.Assets[]

// ❌ ОШИБКА: type строчными буквами
select(.type == "host")

// ✅ ПРАВИЛЬНО: Type с заглавной буквы
select(.Type == "host")

// ❌ ОШИБКА: value строчными буквами
.Observables[].value

// ✅ ПРАВИЛЬНО: Value с заглавной буквы
.Observables[].Value
```

??????? ?????????? ????????? ?????

??? ??????? (alert):

Поле	Правильное написание	Неправильные варианты
Активы	`alert.Assets[]`	alert.assets[], alert.ASSETS[]
Наблюдаемые объекты	`alert.Observables[]`	alert.observables[]
Тип актива	`.Type`	.type, .TYPE
Значение наблюдаемого	`.Value`	.value, .VALUE
Имя	`.Name`	.name
Важность	`.Severity`	.severity
Базовые события	`alert.BaseEvents[]`	alert.baseEvents[]
Исходные события	`alert.OriginalEvents[]`	alert.originalEvents[]
Просканированные файлы	`alert.ScannedFiles[]`	alert.scannedFiles[]
MITRE тактики	`alert.MITRETactics[]`	alert.mitretactics[]

??? ?????????? (incident):

Поле	Правильное написание	Неправильные варианты
Алерты инцидента	`incident.Alerts[]`	incident.alerts[], Incident.Alerts[]
Активы через алерты	`incident.Alerts[].Assets[]`	incident.Alerts[].assets[]
Важность инцидента	`incident.Severity`	incident.severity
Приоритет инцидента	`incident.Priority`	incident.priority

?????: ?????? ??????? ??? ?????? ??????????

Обратите внимание на **критическое различие**:

В глобальных данных (alert/incident):

```
// Заглавные буквы
alert.Assets[]
```

```
alert.Observables[]
select(.Type == "host")
.Value
```

В операционных данных (.input):

```
// Строчные буквы!
.input.assets[]
.input.observables[]
select(.type == "host")
.value
```

????????????? ?????????????

1. **Всегда копируйте имена полей из документации** — не пишите по памяти
2. **Используйте подсказки интерфейса** — при вводе `alert.` или `incident.` система покажет список доступных полей с правильным регистром
3. **Проверяйте регистр при возникновении ошибок** — 80% ошибок связаны с неправильным регистром
4. **Создайте шпаргалку** с правильными написаниями полей для вашей команды

Раздел 1. Что такое плейбук

1.1. ??? ????? ???????

При расследовании инцидентов аналитик SOC часто выполняет одни и те же действия: проверяет источник события, получает информацию об устройстве, запускает дополнительные проверки, изолирует хост или уведомляет ответственных сотрудников. Выполнение этих операций вручную занимает время и увеличивает вероятность ошибки.

Плейбук — это сценарий автоматизации, который позволяет системе самостоятельно выполнить заранее определенную последовательность действий при наступлении заданного события.

Проще говоря, плейбук можно представить как инструкцию:

> **«Если произошло определенное событие, последовательно выполни указанные действия и получи необходимый результат».**

Например, если в системе появился критический алерт, плейбук может:

1. определить устройство, на котором произошло событие;
2. получить сведения о пользователе;
3. изолировать устройство от сети;
4. запустить дополнительную проверку;
5. отправить уведомление ответственному специалисту.

В результате аналитик получает уже обработанный инцидент и может сосредоточиться на принятии решений, а не на выполнении однотипных операций.

1.2. ??? ??????? ??????????? ?? ???????? ????????

Отдельное действие выполняет только одну операцию:

- изолировать устройство;
- получить информацию о процессе;
- отправить уведомление.

Плейбук объединяет множество таких действий в единый алгоритм, где результат одного шага используется следующим.

Именно поэтому плейбуки являются одним из основных инструментов автоматизации реагирования в Kaspersky EDR Expert и OSMP.

????? ????? ????????

При расследовании инцидентов аналитик регулярно выполняет повторяющиеся действия: открывает карточку алерта, получает информацию об устройстве и пользователе, проверяет индикаторы компрометации, запускает дополнительные проверки, изолирует хост или отправляет уведомления.

По отдельности эти операции занимают немного времени, однако при большом количестве событий они превращаются в однообразную рутинную работу. Кроме того, выполнение одинаковых действий вручную увеличивает вероятность ошибки: можно пропустить важный шаг, выбрать неверный объект или просто забыть выполнить необходимую проверку.

Плейбуки позволяют автоматизировать такие сценарии. Вместо того чтобы выполнять каждое действие самостоятельно, аналитик описывает последовательность шагов один раз, после чего система воспроизводит ее при наступлении заданного условия.

Например, при обнаружении вредоносного процесса плейбук может автоматически:

- определить устройство, на котором произошло событие;
- получить информацию о пользователе;
- изолировать устройство от сети;
- запустить IOC Scan;
- отправить уведомление ответственному сотруднику.

В результате все типовые операции выполняются одинаково и без участия аналитика, а специалист может сосредоточиться на анализе инцидента и принятии решений.

????? ?????? ?????? ????????

Плейбуки не заменяют аналитика, а помогают автоматизировать повторяющиеся этапы реагирования. Их можно использовать для решения самых разных задач:

?????????????? ???????????

Выполнение заранее определенных действий сразу после появления алерта или инцидента.

Например:

- изоляция устройства;
- завершение вредоносного процесса;

- запуск дополнительной проверки;
- создание правила блокировки.

?????????? ??????????

Перед выполнением действий плейбук может собрать дополнительные сведения, необходимые для расследования:

- получить информацию об устройстве;
- определить пользователя;
- найти связанные алерты;
- извлечь хэш, IP-адрес или имя процесса.

???????? ???????

Плейбук способен анализировать данные и принимать решения на основе заданных условий.

Например:

- выполнить действия только для критических алертов;
- реагировать только на события определенного правила корреляции;
- запускаться только для устройств из конкретной группы.

??????????? ????????

Один плейбук может объединять несколько независимых операций в единый сценарий, где результат предыдущего шага используется следующим.

```
Получен алерт        |        ▼Получить HostID        |        ▼Получить информацию о
пользователе        |        ▼Изолировать устройство        |        ▼Запустить IOC
Scan        |        ▼Отправить уведомление
```

????? ????? ????????????? ????????

Плейбук имеет смысл создавать, если один и тот же набор действий регулярно выполняется по одинаковым правилам. Если сценарий требует постоянного участия аналитика и принятия решений на каждом этапе, автоматизация может оказаться избыточной.

Практика показывает, что лучше всего плейбуки подходят для стандартных процессов реагирования, обогащения данных и выполнения рутинных операций, позволяя сократить время обработки инцидентов и обеспечить единообразное выполнение действий независимо от того, кто занимается расследованием.

Revision #2
Created 23 June 2026 15:53:13 by Николай
Updated 1 July 2026 14:58:57 by Николай