МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ

РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное бюджетное образовательное учреждение

высшего образования

«Московский государственный технический университет имени Н.Э. Баумана

(национальный исследовательский университет)»

**ВЫПУСКНАЯ КВАЛИФИКАЦИОННАЯ РАБОТА**

**по курсу**

«Data Science»

Слушатель Якобсон Екатерина Максимовна

Москва, 2023

Оглавление

[Введение 4](#_Toc130925117)

[Глава 1. Архитектура SIEM и основные компоненты 5](#_Toc130925118)

[1.1.1 Источники и датчики событий 7](#_Toc130925119)

[1.1.3 Механизм корреляции 10](#_Toc130925120)

[1.1.4 Хранилище событий 11](#_Toc130925121)

[1.1.5 Консоль SIEM 13](#_Toc130925122)

[1.1.6. Вспомогательная инфраструктура 13](#_Toc130925123)

[1.1.7 Сетевые каналы 14](#_Toc130925124)

[1.2 Смежные исследования в области обнаружения атак 16](#_Toc130925125)

[Глава 2. Неисправности в системе SIEM 21](#_Toc130925126)

[2.1.1 Источники событий 22](#_Toc130925127)

[2.1.2 Датчики 23](#_Toc130925128)

[2.1.3 Коллекторы событий 24](#_Toc130925129)

[2.1.4 Механизм корреляции 25](#_Toc130925130)

[2.1.5 Хранение событий 26](#_Toc130925131)

[2.1.6 Консоль SIEM 27](#_Toc130925132)

[2.2 Терминология правил корреляции 28](#_Toc130925133)

[2.3 Исследования в области обнаружения атак 35](#_Toc130925134)

[Глава 3 Устойчивые правила корреляции 40](#_Toc130925135)

[3.1 Элементарные правила корреляции 41](#_Toc130925136)

[3.1.1 Правила, использующие один источник событий 42](#_Toc130925137)

[3.1.2 Правила использования триггеров на основе времени 45](#_Toc130925138)

[3.1.3 Ограничения основных правил корреляции 46](#_Toc130925139)

[3.2 Совершенствование правил корреляции 47](#_Toc130925140)

[3.2.1 Метод улучшения правил корреляции 50](#_Toc130925141)

[3.2.2 Усиление правил корреляции 53](#_Toc130925142)

[3.2.3 Корреляция различных источников событий 57](#_Toc130925143)

[3.3 Ограничения корреляционных правил для обнаружения атак 63](#_Toc130925144)

[3.4 AutoRule: Автоматический анализ правил 65](#_Toc130925145)

[3.4.1 Разработка autorules 65](#_Toc130925146)

[3.4.2 Принципы внедрения 66](#_Toc130925147)

[3.4.3 Развёртывание и демонстрация 67](#_Toc130925148)

[Глава 4. Реализация 70](#_Toc130925151)

[4.2 Правила корреляции с функцией Autorules 73](#_Toc130925152)

[4.3 Алгоритмы определения атаки 76](#_Toc130925153)

[Заключение 79](#_Toc130925154)

[Список литературы 80](#_Toc130925155)

# Введение

В последние годы информационная безопасность стала актуальной темой, поскольку крупные компании и широкая общественность стали уделять ей больше внимания. Частые новости о целевых атаках и масштабных кражах информации, приводящих к крупным финансовым потерям, иногда даже к банкротству компании, оправдывают инвестиции в механизмы защиты.

В основе мониторинга безопасности в режиме реального времени лежит система управления информацией и событиями безопасности, широко известная как SIEM. Эти системы позволяют собирать события безопасности и выявлять закономерности, анализируя взаимосвязи между этими событиями в режиме реального времени. Однако, как и в случае со всеми компьютерными системами, злоумышленник, знающий о ее существовании, будет стремиться преодолеть действующие механизмы защиты, не давая экспертам по безопасности оповещать о происходящих атаках.

Представлю анализ возможных атак на SIEM-систему и ищу решения для предотвращения успешного использования этих атак, даже если злоумышленникам удастся взять под контроль часть инфраструктуры. Вместо того чтобы предлагать масштабные изменения в многочисленных системах и сетевых компонентах, предлагаю подход, основанный на возможностях системы SIEM по сбору и корреляции событий безопасности из различных источников. Предполагаю, что можно обнаружить неисправности, злонамеренные или случайные, путем анализа собранных событий в режиме реального времени с использованием тщательно разработанных и устойчивых правил корреляции.

Моей целью является определение систематического метода для устойчивого сбора и корреляции событий безопасности в SIEM-системе, несмотря на наличие компонентов, уже находящихся под контролем злоумышленников. Эффективность предложенной методики оценивается в реальной производственной среде, моделируя атаки и случайные сбои и наблюдая их влияние на способность SIEM-системы выявлять аномальное поведение. А также разработала приложение, способное автоматически анализировать правила корреляции, выявлять уязвимости и предлагать улучшения для повышения общей устойчивости.

Задачи:

1. Изучить теоретические основы и методы решения

2. Провести разведочный анализ данных

3. Разработать код, способный автоматически анализировать правила корреляции, выявлять уязвимости

# Глава 1. Архитектура SIEM и основные компоненты

Хотя одной из ключевых характеристик SIEM-системы является ее гибкость, можно определить эталонную архитектуру, на которой основано большинство реализаций SIEM. Компоненты могут быть переставлены в соответствии с конкретными ограничениями и целями, но информационные потоки сохраняются.

В этой главе представлена эталонная архитектура системы SIEM, подробно описаны ее основные компоненты. Также представлен синтаксис, используемый при определении правил корреляции, что позволяет лучше понять возможности и ограничения этого инструмента. Для контекстуализации работы также обратимся к смежным работам по обнаружению атак, подчеркивая полезный вклад, но также и существующие ограничения в этой области исследований, особенно при рассмотрении конкретного анализа SIEM-систем.

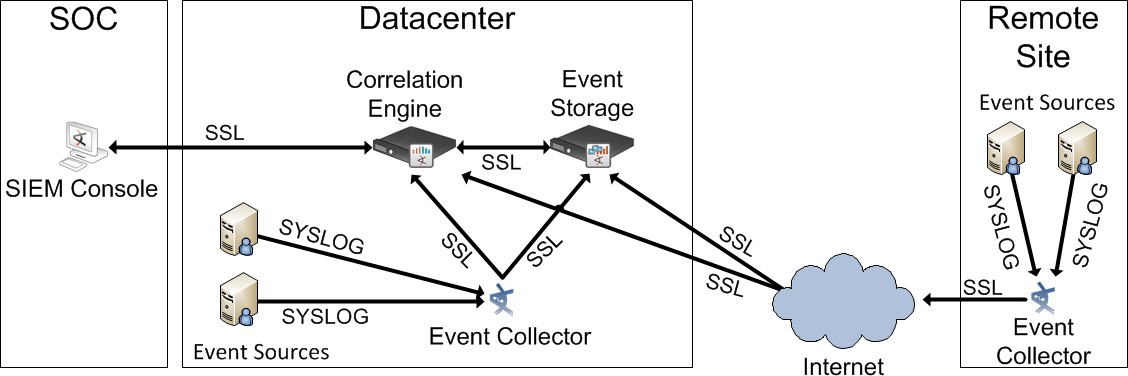


Рисунок 1 – SIEM архитектура

На рисунке 1 показана возможная высокоуровневая архитектура SIEM, включающая как центральную инфраструктуру, так и распределенные сетевые зоны с соответствующими источниками событий. Наиболее важные связи между компонентами SIEM представлены стрелочками для обозначения потоков информации, а также наиболее часто используемых протоколов.

В оставшейся части этого раздела описываем и анализируем компоненты, входящие в SIEM-систему, которые делают возможным непрерывный мониторинг безопасности, а также основные проблемы надежности системы, связанные с их работой.

## Источники и датчики событий

Хотя источники событий не являются частью системы SIEM, они представляют собой фундаментальную часть архитектуры, поскольку их возможности, свойства и расположение в сети имеют большое значение для успеха мониторинга. Источники событий соответствуют существующей инфраструктуре, из которой могут быть получены события. Сложность реализации компьютеризированной инфраструктуры настолько огромна, что даже в пределах одной области действия существует множество различий в способах установки и конфигурации компонентов.

Источники событий могут варьироваться от самой физической среды, например, температуры, регистрируемой датчиком, до сложных компьютерных систем, обрабатывающих большие объемы данных. Их объединяет то, что команды, управляющие ими, хотят централизовать сбор и обработку событий безопасности на единой платформе, что позволяет эффективно управлять оперативным контекстом безопасности. Такой мониторинг событий безопасности необходим для обнаружения аномалий в режиме реального времени, вызывая необходимые оповещения для дальнейшего расследования. Двумя возможными примерами источников событий являются серверы аутентификации, которые генерируют событие для каждого запроса на аутентификацию; и межсетевые экраны, которые регистрируют информацию из потоков данных, включая адреса источника и назначения, сетевые порты, а также одобрение или отклонение этого сетевого трафика.

Как правило, в контролируемой инфраструктуре имеется множество датчиков, охватывающих различные компоненты. Эти датчики отвечают за генерацию событий безопасности, обычно представленных в виде текстовых блоков, использующих заранее определенный формат, и отправку их в коллекторы. Аппаратные датчики обычно более просты по своей природе, обычно они измеряют физические переменные и выдают одно значение, которое меняется в течение времени. С другой стороны, программные датчики могут быть более сложными, с возможностью выполнения процедур аутентификации для доступа к защищенным данным.

Программные датчики могут быть частью существующих процессов, которые генерируют журналы безопасности внутри системы, как это обычно происходит в процессах операционной системы, или специальными компонентами, которые действуют как зонды для всей системы и обнаруживают любые изменения свойств безопасности. Журнал событий безопасности ведется датчиком таким образом, чтобы информация могла быть обработана SIEM. Примерами программных датчиков являются: процесс регистрации событий операционной системы, который записывает процедуры аутентификации и любые другие действия, выполняемые пользователями; и процессы, отвечающие за проверку соблюдения определенных условий, например, запуск антивирусного агента, ведение реестра для обнаружения изменений в контексте безопасности.

В данной работе решено считать, что датчики не являются частью платформы SIEM, а являются частью системы, за которой ведется наблюдение. Это объясняется тем, что датчики обычно являются частью процессов, являющихся источниками событий, и совершенно не зависят от существования SIEM. SIEM отвечает за сбор данных, генерируемых датчиками, либо активно собирая эту информацию, либо пассивно получая ее, в зависимости от типа источника событий и датчика.

1.1.2 Коллекторы событий

Коллектор событий является самым важным компонентом системы SIEM. Непосредственно подключенный к датчикам событий, коллектор получает все необработанные данные, необходимые для работы механизма корреляции, выполняя ряд задач, которые могут варьироваться от простой пересылки событий до их агрегирования. Как уже упоминалось ранее, способ сбора событий безопасности от их источников может варьироваться в зависимости от конкретного способа программирования и реализации датчиков. Некоторые датчики периодически связываются с коллекторами событий и отправляют данные о событиях, в то время как в других случаях коллектор обязан связаться с датчиками для получения информации. В любом случае, процесс сбора должен быть реализован безопасным образом, в идеале заставляя обе стороны взаимно аутентифицироваться и устанавливать безопасный канал, особенно если события безопасности содержат конфиденциальные данные, которые могут раскрыть ключевую информацию для возможных злоумышленников, раскрывая существующие уязвимости.

Что касается возможностей сборщика событий, то наиболее важными являются фильтрация, агрегация и нормализация. Поскольку датчики не являются частью платформы SIEM, команда безопасности может не иметь возможности настраивать их параметры. Более того, датчик может регистрировать дополнительную информацию, помимо событий безопасности, которая важна для оперативных целей, но не с точки зрения безопасности. В таких ситуациях коллектор событий, обычно расположенный ближе к датчикам, может отфильтровать ненужные данные, чтобы минимизировать использование каналов связи с другими компонентами SIEM, тем самым снижая нагрузку на платформу. С этой же целью коллекторы могут объединять похожие события, если это возможно сделать без потери необходимой информации. Коллекторы также несут ответственность за нормализацию событий, полученных от датчиков. Исходные события, также называемые необработанными, должны быть преобразованы в формат, который легче обрабатывается механизмом корреляции. Метаданные также добавляются для увеличения объема информации, доступной для правил корреляции. Разбор необработанных событий означает организацию текстовой информации в структурированные поля, определенные для конкретного коллектора, в зависимости от типа обрабатываемых событий. Коллектор отвечает за правильную пересылку событий другим компонентам SIEM.

## 1.1.3 Механизм корреляции

Механизм корреляции - это мозг SIEM-системы, способный непрерывно анализировать тысячи событий в секунду, сопоставляя их с набором заранее определенных правил. Этот компонент SIEM-системы отвечает за наполнение пользовательского интерфейса операторов информацией о текущем состоянии безопасности инфраструктуры.

Поскольку данные хранятся в локальной базе данных и используются для корреляции, важно определить разумную продолжительность хранения, так как большая база данных приводит к замедлению операций корреляции и загрузки. События в режиме онлайн хранятся во внутренней базе данных механизма корреляции и могут быть немедленно проверены на соответствие недавно внедренным правилам. События во внешних базах данных архивируются или находятся в автономном режиме, что означает, что они доступны для экспертного анализа, но механизм корреляции не может обрабатывать их напрямую. Временные рамки для каждого типа хранения должны быть определены курсом, например, указывающим, что события должны оставаться в режиме онлайн в течение одного месяца, а затем в автономном хранилище в течение еще одного года. Если создается новое правило корреляции и команда безопасности считает необходимым проверить это правило на прошлых событиях, можно загрузить события из хранилища обратно в механизм корреляции.

Также возникает вопрос о снижении производительности, если количество событий, поступающих в механизм корреляции, превысит заранее установленный порог. Возможности обработки механизма корреляции измеряются в событиях в секунду, и платформа должна быть разработана с учетом размера контролируемой инфраструктуры. Тем не менее, платформы SIEM могут масштабироваться либо за счет увеличения вычислительных ресурсов машин, либо за счет добавления новых узлов. В случае с механизмом корреляции, если добавление вычислительных ресурсов уже невозможно, решением может быть следующее: отдельные машины используют одну и ту же базу данных событий, каждая из которых имеет подмножество правил корреляции, гарантируя, что события будут сопоставлены со всеми правилами.

## 1.1.4 Хранилище событий

Хотя основной причиной развертывания SIEM-системы является возможность мониторинга событий безопасности в режиме реального времени, эти системы также играют важную роль в обеспечении соответствия требованиям, с которыми сталкиваются крупные компании. Долгосрочное хранение событий безопасности и журналов регистрации не только позволяет компании отвечать на требования законодательства, но и дает возможность ИТ-отделу проводить криминалистический анализ после обнаружения инцидента безопасности.

Детализация протоколирования зависит от целей и юридических ограничений, с которыми сталкивается организация. Поскольку неважные события отфильтровываются коллекторами, те, которые попадают в хранилище событий, должны быть релевантными и поэтому сохраняться, но для оптимизации процесса хранения могут применяться сложные курсы. Например, организация может решить хранить события из определенных источников в течение более длительного периода времени, если для этого существуют определенные рекомендации.

Поскольку хранение событий в некоторых случаях должно соответствовать строгому законодательству, события также хранятся в необработанном формате, именно в том виде, в котором они были переданы от датчиков. Обработка на уровне коллектора может не только удалить критически важную информацию, но и повлиять на целостность события, сделав его неприемлемым для юридических процедур. Чтобы дополнительно гарантировать это свойство безопасности, само хранилище может обладать проверкой целостности, например, с помощью криптографических подписей, чтобы подтвердить, что данные, находящиеся в состоянии покоя, не были подделаны. Решения для хранения данных с гарантией целостности информации известны как Write Once Read Many (WORM) и обычно используют криптографические контрольные суммы для обнаружения изменений в хранимых данных.

Команда, состоящая из аналитиков по безопасности и других технических специалистов, должна иметь доступ к событиям и сигналам тревоги для работы с системой SIEM. Консоль предоставляет такой доступ, а также параметры конфигурации и редакторы правил корреляции. Консоль - это программный компонент, устанавливаемый на ноутбук или настольный компьютер под управлением коммерческой операционной системы без каких-либо особых требований.

## 1.1.5 Консоль SIEM

Используя консоль, операционная команда SIEM может развертывать, регистрировать и удалять коннекторы при условии, что пользователь имеет необходимые привилегии доступа. Учитывая, что события обрабатываются перед отображением в консоли, целостность информации, представленной на экранах, имеет большое значение для правильной интерпретации этих событий безопасности. Любые ошибки могут привести к ложным срабатываниям или ложным отрицаниям механизма корреляции, что может привести к катастрофическим последствиям для общей безопасности инфраструктуры.

## 1.1.6. Вспомогательная инфраструктура

Система SIEM должна полагаться на вспомогательные системы для некоторых фундаментальных функций, которые, в случае отказа, могут поставить под угрозу ее общую работу.

Источники времени (time sources) - одной из наиболее важных вспомогательных систем является источник времени, с которым все компоненты должны связываться для синхронизации часов, обычно с использованием протокола сетевого времени (NTP). Если источник времени подвергается атаке и начинает действовать произвольно, несколько правил корреляции оказываются под угрозой, поскольку они полагаются на анализ временных меток. Чтобы гарантировать эффективность правил корреляции, необходимо, чтобы различные компоненты SIEM были идентично синхронизированы с уникальным источником времени, с минимальным дрейфом между их внутренними часами.

Службы аутентификации (authentication Services) - Система SIEM собирает события либо путем пассивного прослушивания данных, передаваемых датчиками, либо путем активного запроса информации у других систем. В последнем случае SIEM-системе может потребоваться предоставить действительные учетные данные, чтобы датчики подтвердили подлинность запроса, поступающего от сборщиков. Чтобы сделать возможным активный сбор событий, необходимо иметь органы аутентификации для проверки предоставленных учетных данных.

## 1.1.7 Сетевые каналы

Весь поток информации должен передаваться между различными модулями архитектуры SIEM. Количество и тип сетевых сегментов варьируется в зависимости от определенной архитектуры, при этом некоторые компоненты размещаются вместе в одной машине, что устраняет необходимость в сетевом канале связи. Тип каналов и протоколов связи должен быть адаптирован к конкретным технологиям. Обычно события передаются между датчиками и коллекторами в основном с использованием стандарта syslog (системный журнал) по протоколу User Datagram Protocol (UDP), при этом все соединения между компонентами SIEM используют протокол Secure Socket Layer (SSL) в целях безопасности.

Syslog - это стандарт для регистрации данных, который маркирует сообщения, указывая тип источника и степень серьезности сообщения. Используя этот стандарт для большинства источников событий безопасности можно определить единые правила фильтрации и агрегации для всех коллекторов событий, используя метки в каждом сообщении. Сам стандарт syslog не включает в себя защиту безопасности, так как данные передаются открытым текстом по сети без какой-либо гарантии конфиденциальности. Это видно на рисунке 2, где показана передача данных от датчика к коллектору событий. Более того, используемый UDP - это протокол, который не гарантирует доставку сообщений. Было предпринято несколько попыток повысить устойчивость syslog [9], но, поскольку стандарт не был изменен, большинство реализаций полагаются на стратегии снижения риска, такие как создание сквозных защищенных каналов.

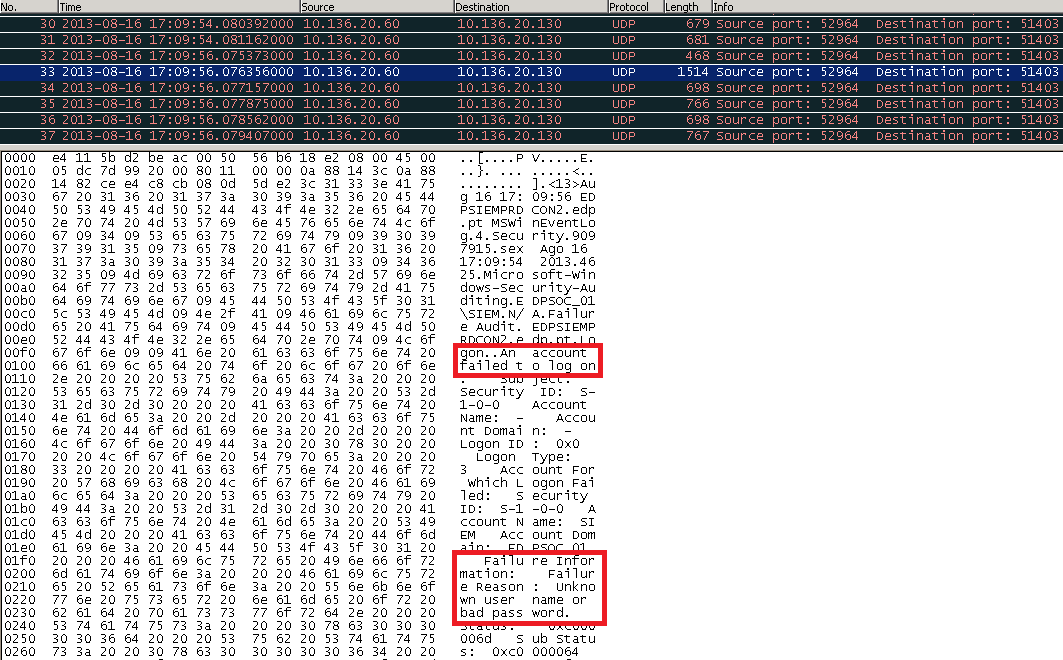


Рисунок 2 – передача данных от датчика к коллектору событий

## Смежные исследования в области обнаружения атак

Хотя системы SIEM получили достаточное количество внедрений в нескольких отраслях, основное внимание в научных публикациях по-прежнему уделяется надежному обнаружению шаблонов на уровне систем обнаружения вторжений (IDS) или систем предотвращения вторжений (IPS). В отличие от системы SIEM, эти платформы основаны исключительно на анализе данных от одного объекта, а не на сборе и корреляции событий из различных источников.

Тодд и др. рассматривают тему точного обнаружения шаблонов атак [14], где представлены методы уклонения от проверки предупреждений. В данной работе показано, что можно использовать этап проверки процесса обнаружения вторжения, изменяя поведение взломанного сервера путем создания поддельных ответных сообщений, чтобы успешная атака казалась неудачной. Авторы предлагают набор методик, которые могут улучшить обнаружение атак даже при наличии поддельных сообщений, анализируя полезную нагрузку каждого взаимодействия с сервером или полагаясь на преимущественно статические каталоги уязвимостей. Недостатками этого подхода являются сложность реализации и тот факт, что он не рассматривает атаки на саму IDS. Более того, даже в сценарии, где возможно реализовать предложенные усовершенствования, кодирование полезной нагрузки все равно может помешать им, и использование ранее не обнаруженных уязвимостей, также известных как уязвимости 0-day, все еще будет возможным.

Подавляющее число событий и оповещений, генерируемых все большим количеством внедренных IDS/IPS, привело к необходимости создания надежных правил корреляции, способных свести к минимуму число ложных срабатываний, тем самым повышая эффективность работы аналитиков безопасности и сетевых операторов. [16] Вальдес и Скиннер представляют вероятностный подход к корреляции оповещений, используя математическую основу для корреляции оповещений. Исходя из предпосылки, что современные IDS/IPS генерируют подавляющее количество данных о событиях из принципиально разных источников, в исследовании предполагается углубить эвристические подходы к решению этой проблемы (которые ранее были представлены теми же авторами [15]). Представлен метод слияния оповещений, использующий Байесовский вывод для объединения общих признаков и мер сходства, создавая потоки для агрегации событий. Наиболее ярким примером является приоритет ошибок, возникающих в результате некорректных запросов, при одновременном преуменьшении ошибок, вызванных запросами к уже недоступным сервисам, что позволяет снизить количество ложных тревог, когда известно, что компонент вышел из строя. Представив матрицу сходства классов инцидентов, авторы смогли четко сформулировать свой взгляд на то, как можно агрегировать события, что позволяет выявить закономерности атак на основе обычно последовательных действий злоумышленника. Хотя результаты обнадеживают: объем предупреждений сократился на 50% - 66%, фундаментальная проблема обеспечения надежной доставки событий в систему SIEM все еще существует. Исследование сосредоточено на снижении количества оповещений при нормальной работе SIEM, не решая проблему устойчивости при наличии случайных или злонамеренных сбоев.

На самом деле, лишь немногие авторы рассматривали проблему получения событий от множества датчиков к коллекторам и затем к механизму корреляции устойчивым образом. Отправная точка связана с обнаружением того, когда компоненты SIEM-решения подвергаются атаке или были скомпрометированы. Олинер, Кулкарни и Айкен предлагают интересное решение, основанное на обнаружении коррелированных по времени аномалий в группах идентичных активов, которые называются сообществами [11]. Используя информацию из агрегированного источника, а не из отдельных компонентов, авторы демонстрируют, что можно уменьшить склонность к ложным срабатываниям, которая обычно наблюдается при обнаружении вторжений на основе аномалий, а также надежно определить, когда подмножество сообщества имеет аномальное и потенциально вредоносное поведение. Для обнаружения аномального поведения путем корреляции событий от нескольких хостов предлагается подход, основанный на подсчете баллов, который может быть использован для создания механизмов голосования с целью исключения ложных данных из процесса сбора событий SIEM. Например, клиент, время отклика которого необычно велико, на основе исторических данных и информации, собранной от других членов сообщества, может указать на аномальную оценку. Это объясняется тем, что странно, если только подмножество клиентов в одной и той же среде начинают вести себя по-другому, и это можно выявить с помощью агрегированных данных и аномалий, коррелированных по времени, чтобы определить, становится ли это изменение более распространенным в данном сообществе. Хотя это полезно для создания сигналов тревоги о возможных нарушениях безопасности, в статье не рассматриваются вопросы доставки событий и избыточности. Система SIEM способна генерировать более конкретные сигналы тревоги, учитывая надежный и устойчивый поток событий, но она не узнает об атаке, если эти события подавляются или изменяются в источниках.

[3] Бриземейстер и др. уделяют особое внимание регулярности трафика и ограниченному количеству протоколов, присутствующих в среде ICS, демонстрируя сочетание обнаружения на основе сигнатур в сочетании с байесовскими методами и обнаружением аномалий на основе обучения. Также вводится понятие атак с обходом сети, определяющее, как противник может использовать доверительные отношения между узлами для атаки на высокоценные цели, к которым злоумышленник иначе не смог бы получить прямой доступ. Эти атаки становятся возможными не только подключением критически важных инфраструктур к сетям общего пользования, но и потому, что данные из этих критически важных систем должны быть импортированы в корпоративные приложения. Дело в том, что профессионалы в бизнесе сблизили корпоративные и промышленные сети, создав сетку взаимосвязей, которую трудно контролировать и еще труднее защитить от кибератак. Авторы излагают концептуальную архитектуру соединения корпоративных и управляющих сетей с помощью демилитаризованной зоны с двойным межсетевым экраном, или DMZ, где определенные системы доступны для обеих сетей контролируемым образом. Определенная критичность систем возрастает по мере того, как атака проникает все глубже во внутренний периметр, создавая предупреждения на основе коррелирующих событий в различных сегментах сети, как будто SIEM способен отслеживать перемещения атакующего по мере развития атаки. Результаты этого исследования указывают не только на важность правильной категоризации активов и определения правил корреляции, но и на актуальность продвинутого инструмента визуализации для содействия усилиям по мониторингу. Авторы дают указания по определению архитектуры SIEM-системы в среде ICS, но стремятся обогатить правила корреляции и информацию о событиях, полагаясь только на один источник, что не решает вопросы устойчивости.

Особенно интересным в контексте данной работы является понятие совместной работы SIEM между различными доменами, описанное Агирре и Алонсо [1]. Тот факт, что ИТ-сети и домены должны оставаться независимо управляемыми и контролируемыми, хотя и соединенными, ставит вопрос о том, как автоматизировать обмен и корреляцию соответствующих событий безопасности. В компаниях ICS и корпоративные сети четко разделены. Тем не менее, каналы связи между ними, а также общие уязвимости означают, что обмен тревожными сигналами между системами SIEM принесет пользу обеим сторонам. Хотя в данной статье не рассматривается вопрос надежного сбора событий, понятие обмена информацией из нескольких доменов, коррелируя ее в одной SIEM-системе, является полезным. Анализируя коммуникационные потоки между отдельными доменами, мы можем найти интересные альтернативные методы сбора информации, поскольку одновременная компрометация активов в более чем одном домене менее вероятна.

# Глава 2. Неисправности в системе SIEM

В этой главе определены возможные неисправности, которые могут возникнуть в различных компонентах системы SIEM. Модель неисправностей делится на два основных класса:

- Случайные неисправности: неисправности, в результате которых некоторые или все компоненты системы SIEM перестают работать правильно. Неисправности возникают без непосредственного участия злоумышленника;

- Намеренные атаки: неисправности, намеренно вносимые злоумышленниками с целью нарушения конфиденциальности, доступности, целостности или своевременности системы SIEM.

В этой главе описываются последствия неисправностей компонентов, представленных в главе 1, а также предлагаются возможные методы их устранения. Идея заключается в том, чтобы представить четкие и систематические методы борьбы с неисправностями, позволяющие своевременно обнаружить эти неисправности и, по возможности, обеспечить дальнейшее функционирование систем. При рассмотрении злонамеренных атак мы обращаемся к концепции устойчивости к вторжениям [17]. Принятие возможности успешной атаки на часть инфраструктуры заставляет системного архитектора разрабатывать решения, которые позволят продолжить работу в этих неблагоприятных ситуациях.

## 2.1.1 Источники событий

При анализе сбоя важно определить его влияние, выяснив, привел ли он к полному отключению компонента или сбой повлиял только на генерацию событий. Например, если мы рассматриваем компьютерный узел - сервер или рабочую станцию, - важно определить, привела ли неисправность к полному выключению этого узла или сбой произошел только в процессе ведения журнала операционной системы. Поскольку в некоторых случаях датчик собирает данные из журналов, при сбое процесса протоколирования может показаться, что хост остановлен, тогда как на самом деле он продолжает нормально работать.

Таким образом, чтобы правильно определить, вышел ли из строя хост, необходимо рассмотреть сбор событий от этого источника более чем одним способом, либо используя два датчика, работающих по-разному, либо используя данные, полученные от других подключенных источников. Рассмотрим ситуацию, когда датчик, подключенный к хосту, перестает получать события в течение длительного периода времени, но при этом продолжает отвечать на сообщения о периодичном сигнале, что указывает на то, что причина не в неисправности датчика. Первым шагом должна быть конфигурация, указывающая, допустимо ли или ожидается, что этот узел будет молчать. Если это не так, то следует считать, что узел вышел из строя. Чтобы подтвердить или опровергнуть это предположение, можно использовать правила корреляции, чтобы выяснить, собираются ли какие-либо события, связанные с этим узлом, в другом источнике, например, в сетевом компоненте, к которому подключен узел. Если хост генерирует сетевую активность или взаимодействует с другими хостами, это доказывает, что он все еще активен, а процесс, генерирующий события мониторинга или журналы, либо потерпел крах, либо был отключен злонамеренно.

## 2.1.2 Датчики

Поскольку в большинстве случаев на каждый контролируемый узел или сегмент сети приходится только один датчик, поломка датчика может привести к потере событий, генерируемых контролируемым компонентом. С точки зрения коллектора это выглядит так, как будто этот компонент больше не проявляет никакой активности. Для учета аварийных сбоев протокол связи между датчиками и коллектором должен включать периодические сообщения (heartbeat), либо отправляемые датчиком проактивно, либо в ответ на запрос коллектора. С помощью этого механизма коллектор может идентифицировать сбойные датчики и генерировать предупреждение, которое будет получено механизмом корреляции.

Один из наиболее часто предлагаемых подходов к повышению отказоустойчивости системы заключается в предоставлении избыточных услуг путем дублирования компонентов. Если мы используем два идентичных и независимых датчика и учитываем только случайные неисправности, то результирующая надежность повысится, поскольку вероятность того, что оба датчика откажут одновременно, вероятностно меньше. У такого подхода есть два непосредственных следствия: необходимо разработать способы борьбы с дублированием событий; и стоимость решения может резко возрасти, поскольку датчики являются одним из доминирующих компонентов с точки зрения количества.

Если датчик основан на аппаратном обеспечении, то дублирование датчиков влечет за собой большие затраты, если рассматривать большие и сложные системы. С другой стороны, если датчик является программным компонентом, возрастает потребность в вычислительных ресурсах, и они могут иметь общие уязвимости, что может привести к корреляции сбоев. Более того, наличие большего количества датчиков, собирающих данные путем опроса или подталкивания, приводит к увеличению нагрузки на систему на хосте. В уже нагруженных системах дополнительная нагрузка может оказаться непосильной.

Альтернативой является использование корреляции событий, интеллектуального набора правил для вывода событий с хоста, когда датчик, контролирующий его, выходит из строя. Хотя с помощью этого подхода невозможно получить такой же объем информации, он позволяет смягчить последствия отказа датчика, хотя и с пониженным качеством данных, при небольших дополнительных затратах или нагрузке на систему.

## 2.1.3 Коллекторы событий

Хотя в большинстве случаев датчик может направлять события нескольким коллекторам, большинство продуктов SIEM не способны эффективно справляться с дублирующимися событиями. Поэтому результатом внедрения более чем одного коллектора для одного и того же источника событий будет неуправляемый поток данных, поступающих в механизм корреляции, не говоря уже о проблемах с производительностью и пропускной способностью. Учитывая это ограничение, следует считать, что основным решением проблемы сбоя коллектора является применение более интеллектуальных правил корреляции, позволяющих собирать информацию из источника путем анализа событий из других смежных источников.

Например, можно рассматривать сервер UNIX и клиентскую рабочую станцию Windows как источники событий, каждый из которых имеет свой датчик, передающий информацию независимым сборщикам событий. Несмотря на то, что операционные системы разные, взаимодействие между ними генерирует события на обоих концах. Поэтому отдельные коллекторы для систем Windows и UNIX будут получать одинаковые события. В случае если коллектор для UNIX-системы имеет неисправность, приводящую к сбою, все равно можно определить часть активности сервера, анализируя события, передаваемые клиентом. Если распространить этот принцип на все компоненты контролируемой инфраструктуры, то можно обнаружить аналогичные ситуации, в которых события из одного источника могут быть соотнесены и обработаны для определения активности из других источников в случае случайного сбоя в одном из коллекторов событий.

## 2.1.4 Механизм корреляции

Поскольку хранение и архивирование событий безопасности гарантируется другими компонентами SIEM, авария механизма корреляции не должна привести к потере информации о событиях. Используя буферные возможности коллекторов, механизм корреляции получит все события после возобновления работы после сбоя и обработает их в это время. Однако во время простоя механизма корреляции операторы безопасности не будут получать сигналы тревоги или какие-либо коррелированные события. Помимо дорогостоящей и сложной репликации основного механизма, существует возможность прямого доступа к хранилищу событий, анализируя необработанные журналы событий. Хотя это не идеальное решение и значительно повышает сложность обнаружения вредоносного поведения, можно повысить эффективность операций мониторинга за счет упреждающего определения сложных запросов в компонентах хранилища, имитирующих правила корреляции в механизме корреляции. Хотя запросы, выполняемые по необработанным событиям, работают медленнее и менее эффективны, это приемлемый вариант для ситуаций, когда механизм корреляции недоступен в течение короткого периода времени в результате сбоя.

## 2.1.5 Хранение событий

Поскольку хранение и архивирование событий безопасности гарантируется другими компонентами SIEM, авария механизма корреляции не должна привести к потере информации о событиях. Используя буферные возможности коллекторов, механизм корреляции получит все события после возобновления работы после сбоя и обработает их в это время. Однако во время простоя механизма корреляции операторы безопасности не будут получать сигналы тревоги или какие-либо коррелированные события. Помимо дорогостоящей и сложной репликации основного механизма, существует возможность прямого доступа к хранилищу событий, анализируя необработанные журналы событий. Хотя это не идеальное решение и значительно повышает сложность обнаружения вредоносного поведения, можно повысить эффективность операций мониторинга за счет упреждающего определения сложных запросов в компонентах хранилища, имитирующих правила корреляции в механизме корреляции. Хотя запросы, выполняемые по необработанным событиям, работают медленнее и менее эффективны, это приемлемый вариант для ситуаций, когда механизм корреляции недоступен в течение короткого периода времени в результате сбоя.

## 2.1.6 Консоль SIEM

Консоль SIEM сама по себе представляет риски для безопасности, связанные с распространенными уязвимостями программного обеспечения. В то время как основные компоненты системы SIEM размещаются внутри защищенного периметра, консоль обычно развертывается на обычных рабочих станциях, подключенных к частной сети компании и Интернету. Затем между сетями офиса и центра обработки данных создаются специальные правила брандмауэра, гарантирующие необходимый доступ.

Как и любой другой программный компонент, консоль имеет уязвимости, некоторые из которых присутствуют на уровне промежуточного программного обеспечения и операционной системы. Например, консоль ArcSight основана на JAVA, технологии, в которой часто обнаруживаются многочисленные проблемы безопасности, некоторые из них критические [12]. Поэтому, если злоумышленник сможет получить доступ к рабочей станции, подключенной к Интернету, и скомпрометировать консоль SIEM, он может либо полностью нарушить представление информации, либо, что более интересно, представить ложную информацию, чтобы обмануть операторов. Даже если доступ в Интернет перекрыт, вредоносная программа все равно может быть внедрена с помощью такой простой вещи, как зараженная флешка. Целый сценарий, основанный на заражении компьютерной сети вредоносным ПО с целью отображения ложной информации для операторов системы, уже был реализован в печально известной атаке Stuxnet [4].

Что еще более важно, как мы уже упоминали, рабочая станция, на которой установлена консоль, может представлять собой путь между Интернетом и сетью центра обработки данных. Если злоумышленнику удастся обойти существующие средства защиты, он может получить прямой доступ к основным компонентам SIEM из внешней сети.

## 2.2 Терминология правил корреляции

Здесь будут описаны основные концепции и синтаксис для определения событий и правил корреляции.

Events (события) - Событием может быть действие или измерение, собранное непосредственно из исходной системы, что в этом случае эквивалентно записи в файле журнала.

Каждое событие имеет набор свойств, переведенных в буквенно-цифровые или булевы поля. Эти свойства включают всю необходимую информацию для определения происхождения события, например, IP-адреса источника и назначения, тип события и результат события, который позволяет определить, было ли описанное действие успешным или нет. Серьезность события определяется по его свойствам и может варьироваться в зависимости от определенных факторов (например, один и тот же тип события может быть классифицирован по-разному в зависимости от его исхода и/или сегмента сети, в котором оно произошло).

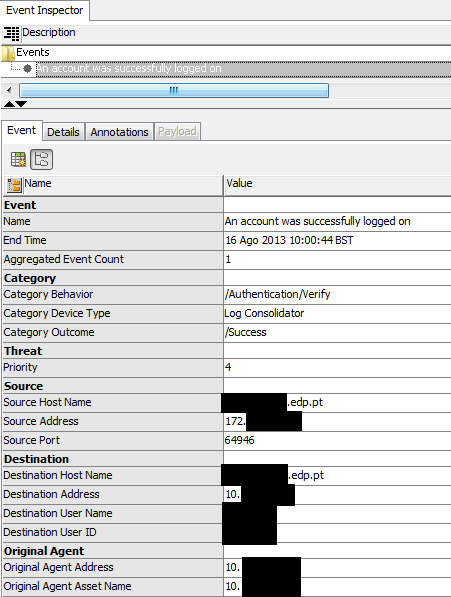


Рисунок 3 – Пример события

Рисунок 3 иллюстрирует подмножество свойств, составляющих событие безопасности. В данном случае событие было сгенерировано на файловом сервере, поэтому наиболее важные поля определяют тип события, источник и место назначения запроса, а также некоторую дополнительную информацию о процессе аутентификации. Из-за соображений конфиденциальности некоторые поля на рисунке 3 изменены.

Правила корреляции - Одна из основных целей системы SIEM - использовать события безопасности, собираемые из различных источников, объединяя связанную информацию для подачи сигналов тревоги. За эту задачу отвечает механизм корреляции, поэтому он должен быть запрограммирован на выполнение этих действий с помощью правил корреляции. Правило основано на наборе операторов и выражений для обработки событий, проверки условий и, при необходимости, запуска результирующих действий.

На рисунке 4 показан пример правила корреляции, определенного в ArcSight. Показанные здесь условия определяют параметры фильтрации для запуска правила, используя информацию из различных полей двух отдельных событий.

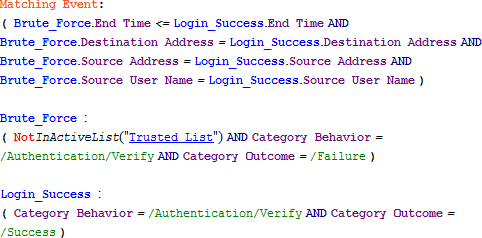


Рисунок 4 – пример правил корреляции

Если условия выполнены, правило корреляции способно выполнять автоматические действия для немедленного реагирования на возможные атаки. На рисунке 5 показан пример таких действий, включая установку информации в полях события и добавление подозреваемого источника в список контролируемых объектов.

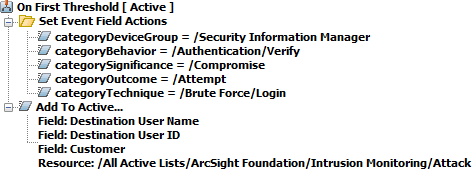


Рисунок 5 – действия по правилам корреляции

Корреляция события - событием также может быть изменение переменной или условия, рассчитанного системой SIEM, например, изменение потока событий или комбинации событий. Коррелированные события всегда создаются в результате правила корреляции, что означает, что они являются следствием предполагаемых закономерностей. Рассматривая пример на рисунке 4, SIEM обнаружит успешную аутентификацию после попытки выполнить атаку грубой силы с использованием тех же учетных данных. Действия, определенные на рисунке 5, пометят это коррелированное событие, заявив, что оно соответствует атаке, а также добавят возможные скомпрометированные учетные данные в активный список мониторинга.

На рисунке 6 показан пример корреляции события с указанием предыдущих событий, которые были идентифицированы и объединены правилом корреляции.

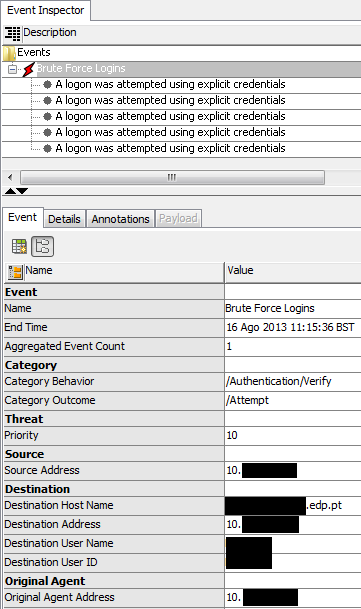


Рисунок 6 пример корреляции события

Действующие события - события могут иметь информацию от одного до трех действующих лиц, в зависимости от типа события. Если это внутреннее событие, сгенерированное внутри SIEM или полученное непосредственно от источника, например, изменение потока событий, единственным действующим лицом является источник этого события. С другой стороны, если событие представляет собой некое взаимодействие между системами, есть два действующих лица: источник и получатель. Примером может быть изменение определенной конфигурации или транзакция клиент-сервер, где событие покажет как место, где был сделан запрос, так и источник этой инструкции. Наконец, существуют определенные ситуации, когда одно действие имеет источник, пункт назначения, а также третью сторону. Наиболее очевидным примером является процесс аутентификации с использованием метода, основанного на репозитории пользователя. Системе может понадобиться подтвердить предоставленные учетные данные с помощью третьей стороны, например, контроллера домена или сервера RADIUS. В результирующем событии будут зарегистрированы источник, место назначения и хранилище, к которому обращались в процессе. Если событие собирается с сервера аутентификации, оно будет идентифицировать источник и место назначения, как упоминалось ранее, а также свой собственный адрес.

Операторы - операторы используются при определении правил корреляции. Они представляют собой обычные логические предлоги, наиболее частыми из которых являются: EQ (равно), NE (не равно), GE (больше или равно), LE (меньше или равно), GT (больше чем), LT (меньше чем), AND, OR, NOT, IN.

Активные списки - ещё один аспект, который необходимо учитывать при определении правил корреляции, - это возможность использования и обновления динамических наборов информации, известных как активные списки. Например, можно определить набор действующих событий, которые имеют одинаковую классификацию или характеристики. Списки можно использовать для управления исключениями (например, белыми списками доверенных систем) или ограничения области действия некоторых правил настроенными источниками. Динамическая природа списков делает их мощным инструментом для обогащения правил корреляции. Активные списки могут пополняться правилами корреляции, которые добавляют информацию из собранных событий, например, IP-адреса или имена хостов. Информация может быть удалена из активного списка либо явным действием, либо с помощью триггеров на основе времени, например, путем определения момента истечения срока действия записи в списке. Простым примером преимущества наличия активных списков является ведение набора подозрительных субъектов на основе прошлых событий. Рассмотрим, что была обнаружена возможная атака; даже в том случае, если источник этих событий не был скомпрометирован, а подозрительные действия возникли из-за неправильной конфигурации системы, все равно может быть полезно добавить эту систему в список подозрительных источников. Некоторые сложные угрозы используют скрытые действия для маскировки атак, тщательно исследуя окружающие компоненты перед началом атаки. В этом случае обнаружить, что что-то не так, можно только путем непрерывного мониторинга данного актива в течение длительного периода времени, подавая сигнал тревоги исключительно на основе повторения аномального поведения.

Модель сети и активов - Основываясь на концепции активных списков, SIEM позволяет создать модель сети и активов контролируемой инфраструктуры. Модель активов состоит из информации о контролируемой инфраструктуре, начиная от операционной системы до критических характеристик конкретной системы или оборудования. Сетевая модель включает не только сетевые адреса компонентов, но и архитектуру и способ подключения систем. Информация в этих моделях может повысить эффективность правил корреляции за счет ассоциации источников событий и определения взаимосвязи между ними. Например, можно рассмотреть соединения, установленные между серверами и сетевым оборудованием, чтобы определить возможные пути атак или обнаружить альтернативные каналы связи для выполнения сбора событий при наличии неполадок в сети. Модель актива также является важным источником информации, позволяя SIEM определить, является ли определенный актив, например, веб-сервером, маршрутизатором или брандмауэром. Тип атак и допустимое поведение могут меняться в зависимости от категории актива, что расширяет возможности определения правил корреляции. Серьезность события также может быть адаптирована в зависимости от местоположения цели, увеличиваясь по мере того, как злоумышленник способен скомпрометировать компоненты, расположенные ближе к ядру инфраструктуры, поскольку обнаружить аномальное поведение на сервере с внешним интерфейсом в демилитаризованной зоне (DMZ) или на внутреннем сервере за брандмауэром можно по-разному. Модели сети и активов могут обновляться автоматически с использованием информации из таких источников, как сканеры уязвимостей.

## 2.3 Исследования в области обнаружения атак

Хотя системы SIEM получили достаточное количество внедрений в нескольких отраслях, основное внимание в научных публикациях по-прежнему уделяется надежному обнаружению шаблонов на уровне систем обнаружения вторжений (IDS) или систем предотвращения вторжений (IPS). В отличие от SIEM-систем, эти платформы основаны исключительно на анализе данных от одного объекта, а не на сборе и корреляции событий из различных источников.

Тодд и др. затрагивают тему точного обнаружения шаблонов атак [14], где представлены методы уклонения от проверки предупреждений. В этой статье показано, что можно использовать этап проверки процесса обнаружения вторжения, изменяя поведение взломанного сервера путем создания поддельных ответных сообщений, чтобы успешная атака казалась неудачной. Авторы предлагают набор методик, которые могут улучшить обнаружение атак даже при наличии поддельных сообщений, анализируя полезную нагрузку каждого взаимодействия с сервером или полагаясь на преимущественно статические каталоги уязвимостей. Недостатками этого подхода являются сложность реализации и тот факт, что он не рассматривает атаки на саму IDS. Более того, даже в сценарии, где возможно реализовать предложенные усовершенствования, кодирование полезной нагрузки все равно может помешать им, и использование ранее не обнаруженных уязвимостей, также известных как уязвимости 0-day, все еще будет возможным.

Подавляющее число событий и оповещений, генерируемых все большим количеством внедренных IDS/IPS, привело к необходимости создания надежных правил корреляции, способных свести к минимуму число ложных срабатываний, тем самым повышая эффективность работы аналитиков безопасности и сетевых операторов. [16] Вальдес и Скиннер представляют вероятностный подход к корреляции оповещений, используя математическую основу для корреляции оповещений. Исходя из предпосылки, что современные IDS/IPS генерируют подавляющее количество данных о событиях из принципиально разных источников, в исследовании предполагается углубить эвристические подходы к решению этой проблемы (которые ранее были представлены теми же авторами [15]). Представлен метод слияния оповещений, использующий Байесовский вывод для объединения общих признаков и мер сходства, создавая потоки для агрегации событий. Наиболее ярким примером является приоритет ошибок, возникающих в результате некорректных запросов, при одновременном преуменьшении ошибок, вызванных запросами к уже недоступным сервисам, что позволяет снизить количество ложных тревог, когда известно, что компонент вышел из строя. Представив матрицу сходства классов инцидентов, авторы смогли четко сформулировать свой взгляд на то, как можно агрегировать события, что позволяет выявить закономерности атак на основе обычно последовательных действий злоумышленника. Хотя результаты обнадеживают: объем предупреждений сократился на 50 – 66 процентов, фундаментальная проблема обеспечения надежной доставки событий в систему SIEM все еще существует. Исследование сосредоточено на снижении количества оповещений при нормальной работе SIEM, не решая проблему устойчивости при наличии случайных или злонамеренных сбоев.

На самом деле, лишь немногие авторы рассматривали проблему получения событий от множества датчиков к коллекторам и затем к механизму корреляции устойчивым образом. Отправная точка связана с обнаружением того, когда компоненты SIEM-решения подвергаются атаке или были скомпрометированы. Олинер, Кулкарни и Айкен предлагают интересное решение, основанное на обнаружении коррелированных по времени аномалий в группах идентичных активов, которые называются сообществами [11]. Используя информацию из агрегированного источника, а не из отдельных компонентов, авторы демонстрируют, что можно уменьшить склонность к ложным срабатываниям, которая обычно наблюдается при обнаружении вторжений на основе аномалий, а также надежно определить, когда подмножество сообщества имеет аномальное и потенциально вредоносное поведение. Для обнаружения аномального поведения путем корреляции событий от нескольких хостов предлагается подход, основанный на подсчете баллов, который может быть использован для создания механизмов голосования с целью исключения ложных данных из процесса сбора событий SIEM. Например, клиент, время отклика которого необычно велико, на основе исторических данных и информации, собранной от других членов сообщества, может указать на аномальную оценку. Это объясняется тем, что странно, если только подмножество клиентов в одной и той же среде начинают вести себя по-другому, и это можно выявить с помощью агрегированных данных и аномалий, коррелированных по времени, чтобы определить, становится ли это изменение более распространенным в данном сообществе. Хотя это полезно для создания сигналов тревоги о возможных нарушениях безопасности, в статье не рассматриваются вопросы доставки событий и избыточности. Система SIEM способна генерировать более конкретные сигналы тревоги, учитывая надежный и устойчивый поток событий, но она не узнает об атаке, если эти события подавляются или изменяются в источниках.

Важно понимать особенности промышленных систем управления (ICS), таких как системы диспетчерского контроля и сбора данных (SCADA). Все организации должны серьезно относиться к безопасности своих информационных систем, но последствия успешной атаки совершенно иные в этих критически важных инфраструктурах. Большинство организаций оценивают воздействие на бизнес исключительно по финансовым потерям, прямым или косвенным, например, через кражу интеллектуальной собственности или влияние на общественный имидж. С другой стороны, в таких областях, как военное дело или коммунальное хозяйство, атака может привести к гибели людей, что повышает требования к точности оповещения. Однако даже при наличии таких рисков экономические причины подтолкнули эти отрасли к переходу от закрытых, проприетарных систем, протоколов и сетей к более открытым средам. Этот переход подверг критически важные системы кибератакам, подключив некогда изолированные системы к сетям общего пользования, что вызвало необходимость в новом подходе к обеспечению безопасности ИКС.

[3] Бриземейстер и др. сосредоточили внимание на регулярности трафика и ограниченном количестве протоколов, присутствующих в среде ИКС, показав сочетание обнаружения на основе сигнатур в сочетании с байесовскими методами и обнаружением аномалий на основе обучения. Также вводится понятие атак с обходом сети, определяющее, как противник может использовать доверительные отношения между узлами для атаки на высокоценные цели, к которым злоумышленник иначе не смог бы получить прямой доступ. Эти атаки становятся возможными не только подключением критически важных инфраструктур к сетям общего пользования, но и потому, что данные из этих критически важных систем должны быть импортированы в корпоративные приложения. Дело в том, что профессионалы в бизнесе сблизили корпоративные и промышленные сети, создав сетку взаимосвязей, которую трудно контролировать и еще труднее защитить от кибератак. Авторы излагают концептуальную архитектуру соединения корпоративных и управляющих сетей с использованием демилитаризованной зоны с двойным межсетевым экраном, или DMZ, где определенные системы доступны для обеих сетей контролируемым образом. Определенная критичность систем возрастает по мере того, как атака проникает все глубже во внутренний периметр, создавая предупреждения на основе коррелирующих событий в различных сегментах сети, как будто SIEM способен отслеживать перемещения атакующего по мере развития атаки. Результаты данного исследования указывают не только на важность правильной категоризации активов и определения правил корреляции, но и на актуальность продвинутого инструмента визуализации для содействия усилиям по мониторингу. Авторы дают указания по определению архитектуры SIEM-системы в среде ICS, но стремятся обогатить правила корреляции и информацию о событиях, полагаясь только на один источник, что не решает вопросы устойчивости.

Особенно интересным для контекста моей работы является понятие совместной работы SIEM между различными доменами, описанное Агирре и Алонсо в [1]. Тот факт, что ИТ-сети и домены должны оставаться независимо управляемыми и контролируемыми, хотя и соединенными, поднимает вопрос о том, как автоматизировать обмен и корреляцию соответствующих событий безопасности. Коммунальные компании четко разделяют свои ИКС и корпоративные сети. Тем не менее, каналы связи между ними, а также общие уязвимости означают, что обмен тревожными сигналами между системами SIEM принесет пользу обеим сторонам. Хотя в данной статье не рассматривается вопрос надежного сбора событий, понятие обмена информацией из нескольких доменов, коррелируя ее в одной SIEM-системе, является полезным. Анализируя коммуникационные потоки между отдельными доменами, мы можем найти интересные альтернативные методы сбора информации, поскольку одновременная компрометация активов в более чем одном домене менее вероятна.

# Глава 3 Устойчивые правила корреляции

В предыдущих главах мы обрисовали возможную архитектуру SIEM, описали ее компоненты, обсудили возможные неисправности и представили способы их устранения. Эти неисправности, злонамеренные или случайные по своей природе, приводят к тому, что события не достигают механизма корреляции SIEM и, следовательно, важные сигналы тревоги не срабатывают. В этой главе представляем методы повышения устойчивости правил корреляции, сердца SIEM, и способ рассмотреть более тщательные подходы к устранению атак, не ограничиваясь защитой процесса сбора событий.

Повышение устойчивости правил корреляции имеет решающее значение для обеспечения сбора всей необходимой информации и сохранения ее целостности. Это также является ступенькой к конечной цели - приобретению возможности мониторинга безопасности, которая может направлять команду безопасности через анализ текущих атак, повышая их эффективность за счет доступности информации и сокращая время реагирования за счет подачи соответствующих сигналов тревоги по мере возникновения событий. Для достижения этой цели крайне важно выполнять корреляцию с использованием различных данных, включенных в собранные события, используя таким образом преимущества избыточности, присущей миллионам обрабатываемых событий.

Правила корреляции лежат в основе работы SIEM, что делает их определение важной частью реализации SIEM, способствующей предотвращению атак в обход срабатывания сигналов тревоги, а также возможности того, что атака останется незамеченной. Цель - улучшить текущие реализации правил SIEM, повысив их устойчивость к атакам, даже при наличии скомпрометированных датчиков или других пограничных компонентов, способных прервать, задержать или подделать поток событий, поступающих в SIEM.

Чтобы сделать подход более конкретным, в оставшейся части главы, рассмотрим несколько примеров правил корреляции. Эти правила построены с использованием синтаксиса системы ArcSight SIEM.

## 3.1 Элементарные правила корреляции

Наиболее простой целью собранных событий является подача сигналов тревоги на основе их содержания, используя события из каждого источника для определения конкретных условий срабатывания. Например, можно определить типы событий, которые никогда не должны наблюдаться, поскольку они противоречат определенной политике безопасности, или, что более распространено, подавать сигнал тревоги, когда событие обнаруживается более определенного количества раз подряд. В этом разделе мы приводим примеры таких "готовых" правил, которые представляют собой то, что можно считать текущим состоянием правил корреляции SIEM, а также указываем на их недостатки и ограничения.

## 3.1.1 Правила, использующие один источник событий

Каждое правило корреляции начинается с указания параметров частоты, которые должны вызывать тревожный сигнал. В правиле 1 продемонстрировано нарушение политики, которое должно вызвать тревогу у команды безопасности, даже если оно произойдет только один раз. Политика данных гласит, что все изменения учетных записей пользователей должны производиться с помощью системы управления идентификацией (IdM), а это значит, что при любом изменении, происходящем не из этой системы, должен быть подан сигнал тревоги.

В строке 1 задается временное ограничение для срабатывания правила, обязательное поле, и указывается, что оно должно срабатывать при первом событии, соответствующем критериям в остальных строках. Критерием для срабатывания правила является конъюнкция трех условий. Строка 2 выражает, что имя пользователя злоумышленника отличается от учетной записи, используемой IdM, строка 3 соответствует типу события известной категории аутентификации и, наконец, строка 4 указывает, что результат события был успешным. Все правило можно сформулировать следующим образом: сопоставлять все успешные события, которые привели к изменениям в учетной записи пользователя и не были выполнены учетной записью IdM.

* + - 1. Matching 1 events in 1 Minute with conditions(
      2. NE(event1.sourceUserName,IdMAccount);And;
      3. EQ(event1.categoryBehavior,/Authentication/Add);And;
      4. EQ(event1.categoryOutcome,/Success))

Правило 1 – изменение данных пользователя вне IdM

Это правило опирается на события из одного источника - каталога пользователей предприятия, сканируя журналы для обнаружения команд изменения определенного типа, а затем проверяя их происхождение на основе имени пользователя. Тот факт, что правило зависит только от имени пользователя для определения того, является ли изменение авторизованным, означает, что подмена этой информации может привести к тому, что атаки останутся незамеченными, если злоумышленник знает имя пользователя IdM и способен выдать себя за него.

Пример в правиле 2 немного сложнее, в нем используются вспомогательные правила для маркировки событий, уже определяя их как атаки или успешные операции. Задача состоит в том, чтобы определить, была ли атака грубой силы (атака методом перебора или brute force) успешной.

В строке 1 снова указаны временные условия, запускающие правило при каждом событии. В строке 2 указано, что успешный вход в систему должен произойти в то же время или после попыток перебора, а в строках 3-5 проверяется, что происхождение и место назначения событий совпадают. Строка 6 исключает подмножество доверенных субъектов, что означает, что если источник события находится в этом списке, правило не срабатывает. Строки 7-10 соответствуют типу события и его результату, который должен быть неудачным для попыток методом перебора и успешным для выполненного запроса аутентификации. В результате правило выглядит следующим образом: сопоставлять любые случаи атак методом перебора с успешной аутентификацией из того же источника, если этот источник не находится в списке доверенных лиц.

1. Matching 1 events in 1 Minutes with conditions(
2. LE(Brute\_Force.endTime,Login\_Success.endTime);And;
3. EQ(Brute\_Force.sourceAddress,Login\_Success.sourceAddress);And;
4. EQ(Brute\_Force.destinationAddress,Login\_Success.destinationAddress);And;
5. EQ(Brute\_Force.sourceUserName,Login\_Success.sourceUserName);And;
6. “Not" InActiveList(Brute\_Force.sourceAddress, Trusted List);And;
7. EQ(Brute\_Force.categoryBehavior,/Authentication/Verify);And;
8. EQ(Brute\_Force.categoryOutcome,/Failure);And;
9. EQ(Login\_Success.categoryBehavior,/Authentication/Verify);And;
10. EQ(Login\_Success.categoryOutcome,/Success))

Правило 2 – вероятная успешная атака методом перебора

Как и в первом примере, это правило основано на анализе событий из одного источника - сервера аутентификации. Набор событий был предварительно проанализирован и классифицирован как атака грубой силы с помощью правила 3. Затем правило использует эту информацию и соотносит ее с событиями успешной аутентификации, чтобы определить, достиг ли злоумышленник своей цели.

1. Matching 5 events in 2 Minute with conditions(
2. "Not" InActiveList(Auth\_Fail.sourceAddress, Trusted List);And;
3. EQ(Auth\_Fail.categoryBehavior,/Authentication/Verify);And
4. EQ(Auth\_Fail.categoryOutcome,/Failure))

Правило 3 – вход в систему методом перебора

В правиле 3 ограничение по времени в строке 1 указывает, что правило срабатывает только в том случае, если в течение двух минут произойдет пять событий, соответствующих критериям. Строка 2 освобождает доверенных субъектов от срабатывания правила, разрешая этот тип поведения от заранее определенных источников. Наконец, строки 3 и 4 относятся к типу события и неудачному исходу. Таким образом, правило звучит следующим образом: соответствовать пяти неудачным попыткам аутентификации в течение двух минут, исходящим от источника, не входящего в список доверенных лиц.

В этом правиле снова есть четкие ограничения, например, тот факт, что оно срабатывает только в том случае, если адрес атакующей и целевой систем совпадают при атаке грубой силы и при успешной аутентификации. Если злоумышленник знает об этих рассуждениях, он может использовать несколько серверов аутентификации, обычно присутствующих на крупном предприятии, чтобы рассредоточить атаку, не выходя за границы временного ограничения, чтобы избежать обнаружения.

Для большей эффективности правило должно учитывать адреса всех серверов аутентификации, хотя даже в этом случае злоумышленник все равно может подменить свой собственный адрес при каждой попытке, чтобы скрыть истинное происхождение событий. Чтобы справиться с этими более продвинутыми атаками, необходимы более сложные правила, что и будет продемонстрировано далее.

# 3.1.2 Правила использования триггеров на основе времени

Хотя правило 3 учитывает не только адрес атакующего и цели, но и временной интервал между событиями, существуют более простые правила, которые классифицируют события как подозрительные или даже вызывают тревожный сигнал, основываясь исключительно на временных соображениях.

1. Matching 1 events in 1 Hours with conditions(
2. EQ(event1.deviceEventClassId,Security:630);And;
3. InActiveList(event1.destinationUserName, CreatedAccountsActiveList))

Правило 4 – учётная запись создаётся и удаляется в течение одного часа

Правило 4 использует информацию о событиях из каталога пользователей и опирается на связанное правило, которое добавляет вновь созданные учетные записи пользователей в активный список, упомянутый в строке 3 ("CreatedAccountsActiveList"). Записи, добавленные в этот список, имеют время ожидания (TTL) в один час, после чего они автоматически удаляются из активного списка системой SIEM. Если в течение этого часа будет удалена учетная запись, обозначенная в строке 2 кодом события 630 в контроллерах домена на базе Windows, это правило сработает, и действие может быть помечено как подозрительное или даже отобразить сигнал тревоги для команды безопасности, которая затем приступит к проверке действий, выполненных с использованием этой учетной записи.

Не полагаясь на временные ограничения и адреса источника или назначения, это правило может быть несколько прочнее, чем предыдущие примеры. Тем не менее, как и во всех правилах, основанных на временных ограничениях, если злоумышленник знает об ограничениях, налагаемых такими триггерами, он может легко обойти правило и последующие сигналы безопасности. В данном конкретном случае, создав фиктивную учетную запись и подождав один час до ее использования, можно успешно осуществить атаку.

# 3.1.3 Ограничения основных правил корреляции

Команда безопасности сильно подвержена возможным ошибкам, полагаясь только на один источник событий и/или в условиях временных ограничений для определения необходимости срабатывания правила. Как только злоумышленник узнает, как построены правила корреляции, лазейки становятся очевидными, что делает возможной целенаправленную атаку.

Базовые правила обычно легко обойти, подменив часть информации о событии, например, имя пользователя или IP-адрес, что можно сделать без особых усилий. Аналогичным образом, триггеры, основанные на времени, прошедшем между событиями, также можно обойти, если злоумышленник сможет изменить темп атаки, расширить круг целей или просто задержать отправку информации о событиях датчиками.

Даже если злоумышленник не в состоянии скомпрометировать компоненты SIEM-системы, он может обойти основные правила, просто скомпрометировав датчик, собирающий события от атакуемого источника. Единственный доступный вариант минимизировать количество пропущенных сигналов безопасности, когда один или несколько датчиков скомпрометированы, - это собирать информацию из разных источников, используя присущие этим источникам связи как способ обогащения правил корреляции. Модели сети и активов могут быть очень полезны при разработке надежного набора правил корреляции, поскольку они содержат ценную информацию об источниках событий, их характеристиках, расположении в сети и каналах связи между ними.

# 3.2 Совершенствование правил корреляции

Как было показано в предыдущих разделах, стандартные правила корреляции могут быть неэффективны даже против умеренно сложных злоумышленников и не способны справиться ни со случайными, ни со злонамеренными неисправностями, такими как скомпрометированные датчики. Цель - устранить как можно больше недостатков в правилах корреляции, улучшив их устойчивость и не усложняя их, чем это необходимо.

Как и при повышении безопасности в конфигурации системы, первым шагом должно быть усиление правила корреляции с учетом непрямых сценариев даже при использовании одного источника событий. Этот подход заключается в обогащении правил корреляции с помощью дополнительной информации, также известной как свойства, включенные в события, и использовании преимущества ресурсов SIEM, таких как модели активов и сетей. Набор правил по умолчанию воспринимает целостность информации как должное и ориентируется в основном на лучший сценарий, что приводит к недостаткам, упомянутым ранее. Вместо того чтобы рассматривать только часть информации, составляющей событие, нужно использовать как можно больше информации для обнаружения вредоносного поведения, даже если злоумышленник принимает некоторые меры предосторожности, чтобы не быть замеченным. Кроме того, расширяя круг свойств, учитываемых при определении правил, мы увеличиваем сложность подделки информации о событиях.

Понимание свойств событий и их идиосинкразии важно при разработке более устойчивых правил корреляции. Подмножество этих свойств является общим для событий из разных источников, например, адреса источника и назначения, тип события или результат события. Эти поля могут быть использованы в любом правиле и составляют основу большинства правил корреляции. Однако существует множество других, присущих только конкретным технологиям, что делает их чрезвычайно важными при разработке устойчивых правил корреляции. Признавая специфику свойств событий, можно расширить возможности обнаружения неисправностей и углубить уровень детализации, что поможет улучшить правила.

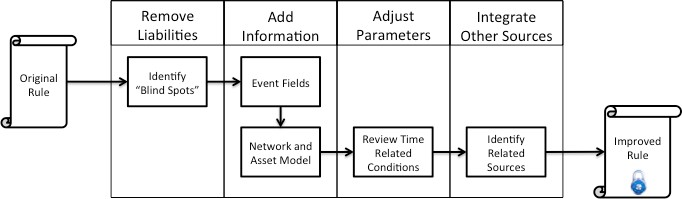
Хотя усиление правил корреляции позволяет SIEM обнаруживать ранее ненаблюдаемые аномальные действия, базирование оценки событий на одном источнике делает систему уязвимой к успешным атакам на этот источник. Эта уязвимость приводит к ситуациям, когда злоумышленник, способный скомпрометировать один компонент или систему, может полностью контролировать события, генерируемые в этом источнике, тем самым нарушая правила корреляции и позволяя атаке пройти незамеченной. Продемонстрируем возможность объединения информации из нескольких источников для усиления правил корреляции, что делает их эффективными даже при наличии частично скомпрометированной инфраструктуры.

Идея корреляции событий из нескольких источников заключается в том, что все системы взаимосвязаны, и поэтому большинство действий приводит к возникновению соответствующих событий в нескольких источниках, что создает определенный уровень избыточности информации, поступающей в механизм SIEM. Предположим, что злоумышленнику удается незаметно скомпрометировать сервер и впоследствии отключить сбор событий из этого источника. Если злоумышленник начнет использовать этот скомпрометированный сервер для новой атаки, каждый обмен данными с другими серверами будет генерировать события на этих целевых серверах, а также в сетевых компонентах, соединяющих оба ресурса. Таким образом, даже при наличии скомпрометированного источника можно собирать события из других источников, которые передают информацию, необходимую для обнаружения атаки. Используя преимущества этих связанных событий из разных источников, можно разработать устойчивые правила корреляции, которые не только повышают эффективность обнаружения атак, но и позволяют команде безопасности идентифицировать возможные скомпрометированные системы.

Еще более интересным является использование событий из нескольких источников не только для обнаружения, но и для маскировки неисправностей. Как уже говорилось ранее, некоторые действия должны генерировать события как в исходной, так и в конечной системах, поэтому несоответствия между этими источниками достаточно, чтобы получить сигнал, обнаружив возможную неисправность. Однако в случае действий, генерирующих события в более чем двух источниках, можно пойти дальше, например, использовать механизм голосования для определения того, какой из источников сообщает несогласованную информацию. Затем оставшиеся источники могут быть использованы для обнаружения продолжающейся атаки, так что недействительный источник может быть идентифицирован как сообщающий некогерентную информацию.

# 3.2.1 Метод улучшения правил корреляции

Систематическое улучшение правил корреляции может быть выполнено в соответствии с методологией, представленной на рисунке 7. В зависимости от типа правила и источников событий, некоторые из этапов методики могут быть неприменимы. Из каждого правила есть исключения, и в данном случае выбран общий подход, который подходит для большинства случаев, адаптируя его для конкретных ситуаций, когда это необходимо.

Рисунок 7 – схема улучшения правил корреляции

Чтобы устранить существующие обязательства в исходном, менее устойчивом правиле, необходимо определить "слепые зоны" или возможные уязвимости в условиях правила. Наличие белого списка или другого механизма исключений нежелательно, за исключением случаев крайней необходимости, поскольку это позволяет злоумышленнику обойти обнаружение, выдавая себя за доверенное лицо, иногда просто подделывая IP-адрес источника. Существуют ситуации, когда использование белых списков оправдано и необходимо, например, при рассмотрении конкретных систем, выполняющих запрещенные задачи, как сканеры уязвимостей. Однако в таких ситуациях крайне важно тщательно определить исключения, предоставив исчерпывающую информацию, не ограничиваясь IP-адресами или именами хостов.

Добавление соответствующей информации осуществляется по двум направлениям, причем первый и наиболее простой подход заключается в использовании преимуществ неиспользуемых полей в событиях. Такая информация, как имя хоста, порт или агент, который собрал событие, может дать ценные сведения при анализе событий, увеличивая необходимые знания о системе, которыми должен обладать злоумышленник, чтобы успешно манипулировать информацией, поступающей в SIEM. Вторым шагом будет включение информации из моделей сети и активов, с преимуществами, о которых мы говорили выше. Поскольку целостность информации в этих моделях проверена, и только администратор SIEM может обновлять их содержимое, модели можно использовать в качестве базовых линий для сравнения с данными о событиях.

Большинство правил корреляции частично основаны на промежутке времени между событиями или количестве схожих событий в определенный промежуток времени, и, следовательно, эти временные ограничения должны быть тщательно определены и пересмотрены. Временные интервалы, определенные для подачи сигнала об опасности, могут стать разницей между обнаружением или не обнаружением атаки. С другой стороны, это может привести к ложным срабатываниям или ложным отрицаниям, которые снижают доверие к возможностям SIEM по оповещению и заваливают команды безопасности информацией. Оба риска необходимо учитывать при построении правил корреляции, поскольку они приводят к потере жизненно важной информации либо из-за того, что сигналы тревоги не срабатывают, либо из-за переполнения информацией, превышающего возможности обработки. Решение должно опираться на тщательно разработанный процесс обучения, настраивая параметры в соответствии с нормальной работой инфраструктуры. Тем не менее, можно определить приблизительные значения по умолчанию для каждого сценария, основываясь на опыте команд безопасности.

Последним шагом при совершенствовании правила будет определение связанных источников, которые могут способствовать проверке достоверности информации, обрабатываемой SIEM, тем самым делая ее устойчивой к ограниченному числу скомпрометированных датчиков. Предыдущие методы эффективны против злоумышленников, пытающихся внедрить фальшивые данные в сеть, и повышают общую устойчивость инфраструктуры, но они не способны справиться со скомпрометированными компонентами, где злоумышленник способен проникнуть через внешний защитный слой, обладая всей необходимой информацией для обмана правил SIEM.

Возможным решением является сбор информации из нескольких связанных источников, что повышает устойчивость системы, учитывая, что злоумышленник способен скомпрометировать только часть систем, по крайней мере, на начальном этапе.

Учитывая модель повреждений, представленную в главе 2, можно утверждать, что все этапы методологии способствуют снижению риска, связанного с вредоносными атаками. Первые два шага в большей степени направлены на попытки персонификации, благодаря чему действия злоумышленника сложнее оставить незамеченными. Чтобы добиться успеха, злоумышленник должен обладать глубокими знаниями о системе SIEM и правилах корреляции, составляя подробные и последовательные события, соответствующие критериям правил. Два последних шага направлены на обнаружение скомпрометированных источников или компонентов SIEM. Если злоумышленнику удается нарушить часть процесса создания или сбора событий, он может изменить содержимое, отложить или удалить события. Применяя временные ограничения и сопоставляя события из разных источников, можно улучшить обнаружение манипулируемой информации. Последний шаг методики также имеет первостепенное значение для преодоления аварийных сбоев, добавляя избыточность источников, чтобы гарантировать, что как можно больше информации все еще поступает в механизм корреляции SIEM в случае сбоев, частично выводящих инфраструктуру из строя.

# 3.2.2 Усиление правил корреляции

В этом разделе применим предложенную методику для повышения устойчивости правил корреляции, представленных в разделе 3.1. Мы установили, что правило 1 уязвимо к подделке, поскольку знание имени учетной записи IdM позволяет обойти его. Эта атака возможна, поскольку правило проверяет только учетную запись источника, игнорируя любую дополнительную информацию. Таким образом, правило может быть улучшено путем добавления информации из других полей событий. Эти свойства могут быть использованы для включения дополнительных сведений о системе-источнике, заставляя возможного злоумышленника подделывать больше информации, тем самым усложняя атаку.

1. Matching 1 events in 1 Minute with conditions(
2. (NE(event1.sourceUserName,IdMAccount);Or;
3. NE(event1.sourceAddress,IdMAddress);Or;
4. NE(event1.sourceOS,IdMOS));And;
5. EQ(event1.categoryBehavior,/Authentication/Add);And;
6. EQ(event1.categoryOutcome,/Success))

Правило 5 – улучшение к правилу 1

Правило 5 - это улучшенная версия правила 1, где добавили подчеркнутые условия в строках 3 и 4 для принудительной дополнительной проверки. Используя адрес атакующего и сигнатуру операционной системы, можно сделать правило более надежным, заставив возможного атакующего подделать не только имя учетной записи, но и адрес и отпечаток ОС IdM-системы.

Однако для выполнения этих условий придется настроить большой набор переменных в SIEM, что, соответственно, увеличит операционные усилия и сложность конфигурации. К счастью, система SIEM включает вышеупомянутые модели сети и активов, которые могут быть автоматически обновлены соответствующей информацией из инфраструктуры, включая, в частности, адрес и отпечаток ОС серверов. Методология предлагает использовать модели активов и сети для облегчения управления правилами корреляции, добавляя информацию, поддерживаемую и обновляемую SIEM, для более точной идентификации систем источника или назначения.

Таким образом, усиленное правило становится не только более мощным, но и более простым в управлении. Поскольку единственным источником, уполномоченным выполнять действия по добавлению нового пользователя в домен, является IdM, необходимо, чтобы источник таких действий был частью модели активов. Свойства источника событий должны сверяться с информацией, присутствующей как в модели актива, так и в модели сети, что можно легко обеспечить.

1. Matching 1 events in 1 Minute with conditions(
2. (NE(event1.sourceUserName,IdMAccount);Or;
3. NE(event1.asset, AssetModel.IdMAsset);Or;
4. NE(event1.sourceAddress, NetworkModel.IdMAddress));And;
5. EQ(event1.categoryBehavior,/Authentication/Add);And;
6. EQ(event1.categoryOutcome,/Success))

Правило 6

Условие в строках 3 проверяет, что система-источник является частью модели актива и соответствует активу, заявленному как IdM, а строка 4 фокусируется на сетевой информации для установления соответствия. В результате Правило 6 сможет проверять не только конкретные атрибуты события, но и рассматривать свойства двух объектов, сопоставляя их для выявления соответствующих несоответствий, указывающих на то, что источник атаки не один и тот же.

Можно использовать аналогичные методы улучшения в Правиле 2 и Правиле 4. В первом случае первоначальным шагом будет рассмотрение атак, исходящих из различных источников и направленных против распределенных серверов аутентификации. Если злоумышленнику удается скомпрометировать несколько компьютеров с целью проведения атаки грубой силы на привилегированную учетную запись, например, с помощью компьютерного вируса или червя, на зараженные компьютеры могут быть отправлены инструкции для выполнения атаки грубой силы на несколько серверов аутентификации, что позволит избежать ограничений по времени, предусмотренных правилом.

1. Matching 1 events in 1 Minutes with conditions(
2. LE(Brute\_Force.endTime,Login\_Success.endTime);And;
3. EQ(Brute\_Force.sourceAddress,Login\_Success.sourceAddress);And;
4. EQ(Brute\_Force.destinationAddress,Login\_Success.destinationAddress);And;
5. EQ(Brute\_Force.sourceUserName,Login\_Success.sourceUserName);And;
6. (“Not" InActiveList(Brute\_Force.sourceAddress, Trusted List));And;
7. EQ(Brute\_Force.categoryOutcome,/Failure);And;
8. EQ(Login\_Success.categoryBehavior,/Authentication/Verify);And;
9. EQ(Login\_Success.categoryOutcome,/Success))

Правило 7 вероятная успешная атака методом перебора (усиленная)

Усовершенствованное Правило 7 будет учитывать количество неудачных попыток аутентификации одной и той же учетной записи, независимо от адреса происхождения, за которыми следует успешная аутентификация этой же учетной записи. Первоначальное правило также содержало лазейку, рассматривая доверенные источники, фактически игнорируя события, исходящие от систем с адресами из этого списка, которые могут быть диапазонами адресов внутри периметра доверенной сети, создавая слепое пятно, если злоумышленник сможет проникнуть в эту якобы безопасную зону. Устранение белых списков, которые могут создавать уязвимости, является первым шагом предлагаемой методологии для улучшения правил корреляции. Поскольку мы фокусируемся на сетевой информации, поля событий, используемые для построения и улучшения этих правил, являются частью набора свойств, общих для событий из всех источников, без необходимости прибегать к специфическим свойствам данного источника событий.

Усиление правила 4 требует дополнительных усилий, так как простота цели будет подорвана более сложной конструкцией, что, возможно, увеличит количество ложных срабатываний. Единственное предложение, следующее методологии улучшения, заключается в расширении временного окна между созданием и удалением учетной записи, так как не ожидается, что учетная запись пользователя будет активна менее 48 часов при рассмотрении нормального жизненного цикла учетных записей домена. Хотя исследования показывают, что нарушение безопасности остается необнаруженным в среднем в течение 416 дней [6], порог в 48 часов кажется подходящим для борьбы с наиболее очевидными угрозами. Более длительный временной интервал или любые другие условия резко увеличили бы количество ложных срабатываний и объем информации, которую должна обработать команда безопасности, фактически снижая вероятность обнаружения атаки.

1. Matching 1 events in 48 Hours with conditions(
2. EQ(event1.deviceEventClassId,Security:630);And;
3. InActiveList(event1.destinationUserName, CreatedAccountsActiveList))

Правило 8 – удаление учетной записи в течение 48 часов (по ОС «Windows»)

Результирующее правило 8 использует свойство deviceEventClassId события для определения исходного действия. Это свойство специфично для событий от серверов Windows, точнее, контроллеров домена, поэтому не является частью общего набора, разделяемого всеми событиями.

# 3.2.3 Корреляция различных источников событий

Даже после процесса ужесточения базовых правил, несколько ограничений все еще присутствуют. Как уже упоминалось выше, полагаться на один источник событий для подачи сигнала опасности неэффективно при рассмотрении модели неисправности, подобной той, которую мы представили в главе 2, где генерация событий может быть затронута. Последний шаг методологии предполагает корреляцию событий из нескольких источников, извлечение данных из отдельных систем или устройств для повышения устойчивости процесса.

Проверка с использованием сетевых событий - Компьютерные сети повсеместно распространены в любой современной ИТ-инфраструктуре, при этом каждый узел подключен к одному или нескольким сетевым компонентам для связи с приложениями, базами данных или другими системами. Это означает, что каждый запрос или команда от системы-источника обязательно пройдет через несколько сетевых узлов, прежде чем достигнет места назначения, что позволяет соотнести события из этих источников.

Первыми источниками событий для включения в SIEM-систему являются сетевые брандмауэры, поскольку они ведут обширный журнал установленных соединений, подробно описывая классификацию трафика, информацию о протоколах и используемых портах. Используя эту информацию, а также определенную модель сети, можно обнаружить попытки скрыть реальное происхождение трафика путем подмены адреса источника.

Продемонстрировать эту возможность в правиле 9, основанном на уже измененном правиле 6, для обнаружения изменений в учетных записях пользователей, не выполненных авторизованным приложением IdM. Первым шагом будет определение правила, которое обрабатывает журналы брандмауэра и идентифицирует команды от приложения IdM к серверу каталога пользователей, добавляя эти команды в активный список на одну минуту. Активный список проверяется в строке 4, чтобы гарантировать, что действие было основано на предыдущей команде от приложения IdM.

С помощью IPSec можно установить криптографический туннель между брандмауэром и SIEM, гарантируя происхождение и целостность событий, зарегистрированных в активном списке.

1. Matching 1 events in 1 Minute with conditions(
2. (NE(event1.sourceUserName,IdMAccount);Or;
3. NE(event1.asset, AssetModel.IdMAsset);Or;
4. NE(event1.sourceAddress, NetworkModel.IdMAddress)) ;Or;
5. (Not InActiveList(event1.command, IdMCommandsInLastMinute));And;
6. EQ(event1.categoryBehavior,/Authentication/Add);And;
7. EQ(event1.categoryOutcome,/Success))

Привило 9 - Изменения пользователя вне IdM (с использованием событий брандмауэра)

Указывая, что при невыполнении одного из условий срабатывает сигнал опасности, исключаем возможность того, что злоумышленник использует взломанную рабочую станцию в корпоративной сети, чтобы выдать себя за IdM-сервер и успешно создать учетную запись пользователя. Если бы злоумышленник попытался скомпрометировать датчик, собирающий события брандмауэра, изменение в каталоге пользователей вызвало бы тревогу, поскольку, блокируя события брандмауэра, злоумышленник также препятствует обновлению активного списка, что все равно вызывает тревогу. Последнее средство, доступное атакующему, - незаметно скомпрометировать машину в той же сетевой зоне, что и IdM-система, что уже является более безопасным периметром, и только затем подделать происхождение команды.

Однако в правиле 9 есть проблема, которая сделает его применение неэффективным. Как уже говорилось, сбор событий от источников иногда осуществляется с помощью ненадежных протоколов, без гарантий упорядочения или доставки. Поскольку это правило полагается на правильное упорядочивание событий, его эффективность ограничена и может генерировать ложные срабатывания.

Более надежной возможностью является объединение событий от контроллеров домена, где выполняется действие, с событиями из базы данных IdM. Когда IdM создает учетную запись, генерируется событие, которое сохраняется во внутренней базе данных. Собирая эти события в SIEM, можно генерировать событие каждый раз, когда есть совпадающие действия для одной и той же целевой учетной записи.

Чтобы объединить события из нескольких источников с общими полями, правило 10 использует теги событий, определенные в строке 3 и строке 8, соответственно идентифицируя событие от контроллера домена как "Action", а событие от базы данных IdM как "DatabaseOperation".

1. Matching 1 events in 1 Minute with conditions(
2. MatchingEvent(Action.destinationUserName, DatabaseOperation.destinationUserName);
3. Action {(EQ(Action.sourceUserName,IdMAccount));And;
4. EQ(Action.sourceAddress, NetworkModel.IdMAddress);And;
5. EQ(Action.assetID, AssetModel.IdMAsset);And;
6. EQ(Action.categoryBehavior,/Authentication/Add);And;
7. EQ(Action.categoryOutcome,/Success)}
8. DatabaseOperation{EQ(DatabaseOperation.assetID, AssetModel.IdMDBServer);And;
9. EQ(DatabaseOperation.categoryBehavior,/Authentication/Add);And;
10. EQ(DatabaseOperation.categoryOutcome,/Success)})

Правило 10- Изменения пользователя вне IdM (с использованием событий базы данных)

В строке 2 выражено условие для сопоставления полей из отдельных событий, метод, аналогичный операции Join в операторах SQL. Информация из моделей Asset и Network используется для гарантии целостности данных о событиях. Поскольку это правило объединяет информацию из двух событий без использования информации из активных списков, порядок поступления событий в SIEM не имеет значения. В отличие от правила 9, которое вызывает сигнал безопасности, когда условие не выполняется, это правило генерирует событие, когда выполняются оба условия. В результате анализа безопасности создание учетной записи домена должно привести к трем связанным событиям - одно от контроллера домена, одно от базы данных IdM и событие, сгенерированное правилом 10.

Учитывая повышенную сложность этого правила, мы представляем его графическое представление на рисунке 8, как оно отображается в консоли SIEM.

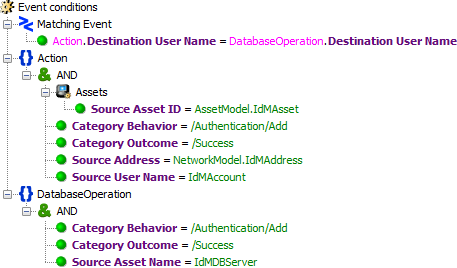


Рисунок 8 – графическое изображение правила 10

Обнаружение неисправностей с помощью корреляции - корреляция событий может быть использована непосредственно для обнаружения несогласованной информации из нескольких источников, регистрирующих одно и то же событие, как мы видели в правиле 9. Используя преимущества моделей сети и активов, можно определить не только допустимые команды, но и проверить, как эти команды согласуются с определенными рабочими процессами и процедурами. шаг за шагом с определенными рабочими процессами и процедурами. SIEM может интерпретировать такую информацию, как тип актива, чтобы обнаружить аномальное поведение, анализируя события, исходящие от этого актива.

Например, учреждение может решить, что некоторые операции, такие как развертывание правил брандмауэра или обновление программного обеспечения, могут выполняться только в рабочее время, чтобы избежать влияния на производительность. Для определения того, что определенные типы событий не происходят вне разрешенного временного окна, можно использовать простое правило. Однако это правило может применяться только к производственным системам, в то время как среды разработки или тестирования имеют менее строгие политики. Модель активов можно использовать для обогащения правил корреляции информацией об инфраструктуре, в которой развернуты системы, что позволяет учесть эти нюансы.

Используя тот же принцип, SIEM также может обнаружить, если конкретные изменения конфигурации систем выполняются из операционного центра или с личных ноутбуков технических специалистов через VPN-соединение. Политика компании может диктовать, что критические операции могут выполняться только локально, чтобы обеспечить прямой доступ к системам в случае необходимости отката, поэтому правила SIEM могут быть определены для обнаружения таких случаев и запуска необходимых сигналов безопасности.

Используя тот же подход, можно также обнаружить неполадки в инфраструктуре, заметив отсутствие ожидаемых событий. Предположим, что злоумышленник решил атаковать датчик, подключенный к веб-серверу, с целью изменения его содержимого, которое, в свою очередь, хранится в отдельной базе данных. Если злоумышленник не сможет успешно скомпрометировать датчики в базе данных и в брандмауэре, отделяющем демилитаризованную зону от внутренних серверов базы данных, следы вредоносных действий все равно останутся. Сигнал опасности может быть подан после проверки того, что события из базы данных и брандмауэра указывают на действие, исходящее от веб-сервера, в то время как соответствующее событие из этого источника отсутствует в SIEM. Сигнал безопасности будет означать, что ожидаемое событие не поступило в SIEM, что указывает на возможный сбой в этом источнике или в процессе сбора.

Учитывая этот сценарий, целью будет создание правил корреляции на основе шаблонов, которые, опять же используя модель сети и активов, смогут сопоставить связанные события, тем самым обнаружив отсутствующие события, которые должны были быть получены SIEM.

Маскировка неисправностей с помощью корреляции - помимо простого обнаружения неисправностей, в определенных ситуациях правила корреляции могут позволять маскировать неисправности, то есть SIEM-система может прийти к тем же выводам и вызвать сигнал безопасности даже при наличии взломанных компонентов. наличии скомпрометированных компонентов. Принцип анализа не только отдельных событий, но и целых рабочих процессов, как мы описали выше, позволяет SIEM обрабатывать сложную информацию, связывая сведения из нескольких источников.

Мы показали, как использовать корреляцию для обнаружения неисправностей и запуска соответствующих сигналов безопасности, но давайте рассмотрим ситуацию, когда командный рабочий процесс должен генерировать события в трех различных компонентах. Если после корреляции информации из всех источников обнаружится, что одно из событий отсутствует или не соответствует остальным, SIEM может проигнорировать весь поток и просто направить сигнал безопасности. Однако, используя алгоритм голосования, можно также предположить, что в источнике с отклонениями произошла ошибка или он был взломан.

# 3.3 Ограничения корреляционных правил для обнаружения атак

Существуют неотъемлемые ограничения для обнаружения атак, полагаясь только на правила корреляции, поскольку избыточность информации присутствует не всегда. Представим сценарий, в котором программный компонент, установленный поверх операционной системы, действует как датчик событий в этом источнике. Если злоумышленник сможет внедрить вредоносное ПО в эту машину, например, с помощью зараженного USB-накопителя, то вредоносное ПО может сразу же нацелиться на датчик, подобно известным вредоносным программам, которые отключают антивирусный агент. Если представить, что система является целью атаки, например, для кражи информации, хранящейся на жестком диске, то атакующему не нужно будет использовать сеть, что сделает невозможным обнаружение атаки другими датчиками. Отсутствие избыточности информации, т.е. информации, поступающей только из одного источника, означает, что эти события не попадут в SIEM в случае компрометации этого источника.

Противодействие этим целенаправленным атакам невозможно осуществить с помощью системы SIEM, ответ заключается в более строгих политиках безопасности, таких как отключение USB-портов. Однако, поскольку мы разрабатываем способы повышения устойчивости SIEM, а не расширения ее возможностей, мы сосредоточимся на более распространенных атаках, которые используют сеть для доступа к удаленным системам и распространяются по всей ИТ-инфраструктуре.

Повышение устойчивости правил корреляции - это также упражнение для увеличения возможностей обнаружения атак и неисправностей при сохранении или, предпочтительно, улучшении показателя ложных срабатываний. Поскольку правила корреляции становятся более детальными, используют конкретную информацию из источников и включают в себя события из нескольких источников, знания инфраструктуры также должны быть на одном уровне. знания инфраструктуры также должны соответствовать этим требованиям к информации. Чем более конкретным является правило, тем более оно восприимчиво к изменениям в контролируемых системах, а это значит, что обновления или изменения в процессах могут привести к необходимости пересмотра правил корреляции во избежание ошибочных тревог. Например, когда используется информация из модели активов или сети, необходимо убедиться, что изменения в системах легко обновляются в этих моделях, одним из вариантов является заполнение моделей с помощью инструментов автоматического сканирования.

# 3.4 AutoRule: Автоматический анализ правил

При определенных условиях можно автоматизировать анализ правил корреляции, что поможет выявить необходимость пересмотра правил перед их внедрением. Учитывая сложность некоторых правил корреляции, ожидается, что автоматический процесс будет иметь ограничения по сравнению с человеческим анализом, проводимым экспертами по безопасности. Тем не менее, систематический подход позволит выявить наиболее распространенные ошибки при построении правил корреляции, а также укажет на возможности их улучшения.

# 3.4.1 Разработка autorules

Автоматический анализатор может начать с разбора правил и выявления ключевых слов. Затем можно применить эвристический анализ, чтобы выявить возможные недостатки и предложить улучшения. Предложенная методология должна применяться пошагово: сначала выявляется использование белых списков, затем отсутствие разнообразия информации о событиях, затем отсутствие ссылок на модели сети и активов в сочетании с другими свойствами событий.

Условия, связанные со временем, можно сравнить со стандартными значениями, основанными на типе правила, однако, как мы уже упоминали ранее, должен быть процесс обучения для корректировки параметров в соответствии со специфическими характеристиками инфраструктуры. Наконец, для определения возможных связанных источников автоматический процесс должен иметь возможность импортировать данные из моделей активов и сетей, создавая внутреннюю базу знаний, способную добавлять соответствующую информацию о событиях в правила корреляции. Таким образом, инструмент должен обеспечивать возможность настройки командой безопасности, адаптируясь к контролируемым системам.

# 3.4.2 Принципы внедрения

Реализация была основана на понимании синтаксиса корреляционных правил, выявлении структуры, лежащей в основе их определения. Необходимо было установить соответствие между понятиями в предлагаемой методологии и конкретным способом, которым они объединяются в правила корреляции.

Автоправила основаны на оценке ключевых слов, выявлении основных повторов и методов, используемых для обработки собранной информации. Выявление элементов, которые могут иметь недостатки или быть исследованы вредоносными сущностями, позволяет рекомендовать улучшения, а также рассчитать оценку устойчивости. Тот же принцип применяется для выявления важных элементов, которые заранее предусмотрены методологией и отсутствуют в правилах корреляции. В результате получается статический анализатор, способный оценить устойчивость правил путем выявления наличия или отсутствия соответствующих компонентов в их определении.

### 3.4.3 Развёртывание и демонстрация

Для демонстрации автоматического систематического анализа правил SIEM, следуя представленной ранее методологии, мы разработали AutoRule (Автоматический анализ правил), приложение на языке Java для анализа правил корреляции, предложения улучшений и расчета общего балла устойчивости в соответствии с проверенным уровнем избыточности. Оценка производится в соответствии с выявленными недостатками правила, при этом различные случаи имеют разный вес, а оценка, близкая к нулю, указывает на более устойчивое корреляционное правило.

Первым шагом, как и предполагает методология, является выявление наличия исключений из правила путем проверки использования доверенных списков. Как мы уже объясняли, если злоумышленник знает об этой потенциальной лазейке, он может подделать данные, чтобы атака прошла незамеченной для SIEM. Являясь важным источником ошибочной идентификации атак, наличие списка доверенных агентов оказывает большое влияние на общую оценку.

Autorule также проверяют информацию о сети или учетной записи, используемую индивидуально, что делает правила более слабыми. Рекомендуется сочетание обоих условий для одновременного выполнения идентификации учетной записи и проверки происхождения сети. Кроме того, предпочтительнее использовать модели сети и активов вместо явных и задаваемых пользователем переменных. Чтобы проиллюстрировать использование AutoRule для проверки корреляционных правил, мы прибегнем к оценке предыдущих примеров для обнаружения изменений учетной записи вне системы IdM.

Лазейка в исключениях из активного списка:

Line 6: "Not" InActiveList(Brute\_Force.sourceAddress, Trusted List);And;

Warning - Network conditions should rely on the network model Warning - Account verifications should rely on the asset model Warning - The rule does not use multiple event sources

Final Score: 12

AutoRule правила 2

Autorule к правилу 2 представляют собой результат автоматизированного процесса проверки при применении к правилу 2. Сразу видно, что это правило не очень устойчиво, учитывая, что оно включает ссылку на список доверенных лиц, идентифицирует агента-источника только на основе сетевого адреса и не использует модели сети или активов. Также важным является тот факт, что правило не обладает избыточностью источника, полагаясь только на события от контроллера домена.

Применение той же методики проверки к правилу 7, улучшенной версии правила 2, показывает различия в надежности и, следовательно, в приписываемой оценке, Autorule к правилу 7. Благодаря устранению возможной лазейки, созданной исключениями в списке доверенных лиц, а также не ограничивая идентификацию злоумышленника одним сетевым источником, общая оценка значительно улучшается.

Некоторые предупреждения остаются, поскольку правило по-прежнему пытается идентифицировать злоумышленника без какой-либо проверки модели актива. Благодаря использованию только одного источника событий, устойчивость правила остается низкой, и срабатывание сигналов безопасности может быть прервано случайными сбоями или успешными атаками.

Warning - Account verifications should rely on the asset model Warning - The rule does not use multiple event sources

Final Score: 6

AutoRule к правилу 7

В таблице 1 приведены результаты и оценки, полученные при анализе всех правил корреляции, представленных в этой главе, с помощью AutoRule. Здесь указываем на улучшения, чтобы продемонстрировать преимущества, полученные с помощью предложенной методики.

|  |  |  |  |
| --- | --- | --- | --- |
| **Rule** | **Improves** | **Output** | **Score** |
| **№1** | **N/A** | Username reference should be complemented with network information:  Line 2: NE(event1.sourceUserName,IdMAccount);And;  Warning - Account verifications should rely on the asset model Warning - The rule does not use multiple event sources | **9** |
| **№2** | **N/A** | Possible loophole in active list exceptions:  Line 6: "Not" InActiveList(Brute\_Force.sourceAddress, Trusted List);And;  Warning - Network conditions should rely on the network model Warning - Account verifications should rely on the asset model Warning - The rule does not use multiple event sources | **12** |
| **№3** | **N/A** | Possible loophole in active list exceptions:  Line 2: "Not" InActiveList(Auth\_Fail.sourceAddress, Trusted List);And;  Warning - The rule does not use multiple event sources | **10** |
| **№4** | **N/A** | Warning - Account verifications should rely on the asset model Warning - The rule does not use multiple event sources | **6** |
| **№5** | **№1** | Warning - Network conditions should rely on the network model Warning - Account verifications should rely on the asset model Warning - The rule does not use multiple event sources | **7** |
| **№6** | **№5** | Warning - The rule does not use multiple event sources | **5** |
| **№7** | **№2** | Warning - Account verifications should rely on the asset model Warning - The rule does not use multiple event sources | **6** |
| **№8** | **№4** | Warning - Account verifications should rely on the asset model Warning - The rule does not use multiple event sources | **6** |
| **№9** | **№6** | Warning - The rule could be impacted by out-of-order events | **2** |
| **№10** | **№9** |  | **0** |

Таблица 1 – Autorules анализ разработанных правил корреляции

### Глава 4. Реализация

При рассмотрении возможностей улучшения корреляционных правил мы постарались концептуализировать общие сценарии и обсудить теоретические возможности, чтобы учесть множество контекстов реализации. Однако, не только предлагая структуру для улучшения правил корреляции, мы также стремились к результату, который мог бы воплотиться в практическом применении. Реализация в реальной производственной среде создает дополнительные проблемы, учитывая недостатки сложной инфраструктуры, где функциональность является конечной целью, а безопасность - лишь желаемым, иногда игнорируемым свойством.

Поскольку целью SIEM-системы является сбор событий безопасности, первая трудность заключается в наличии доступа к этой информации. Существует множество причин, по которым доступ к полной и исчерпывающей информации может быть ограничен, например, недостаточный уровень протоколирования из-за проблем с производительностью или того, что часть инфраструктуры управляется третьей стороной. Кроме того, растет уровень экстернализации услуг, либо путем передачи управления ИТ-инфраструктурой на аутсорсинг, либо путем обращения к поставщикам облачных услуг, что представляет собой проблему, когда командам безопасности необходимо получить доступ к конфигурациям безопасности или событиям.

Часть функций операционного центра безопасности заключается в том, чтобы донести эту информацию до руководства компании, способствуя созданию политики безопасности, включающей конкретные требования, которые необходимо учитывать при заключении контрактов на предоставление услуг с третьими сторонами. Тем не менее, представленная нами схема включает в себя множество пунктов действий по улучшению правил корреляции, чтобы такие технические или контрактные ограничения можно было хотя бы частично обойти.

4.1 Правила корреляции по умолчанию

* 1. Matching 1 events in 5 Minute with conditions(
  2. "Not" InActiveList(event1.sourceUserName,/All Active Lists/ArcSight Solutions/IdentityView/Book Keeping/Badged In Actors);And;
  3. EQ(event1.name,Successful Building Access Event))

Активность сотрудника

1. Matching 1 events in 5 Minute with conditions(
2. EQ(event1.name,Failed Building Access Events))

Невозможность доступа

1. Matching 3 events in 3 Minute with conditions(
2. (EQ(Protocol\_Deny.categoryBehavior,/Access);Or;
3. EQ(Protocol\_Deny.categoryBehavior,/Access/Start));And;
4. "Not" InActiveList(Protocol\_Deny.sourceAddress,/All Active Lists/ArcSight System/Attackers/Trusted List);And;
5. ("Not" InActiveList(Protocol\_Deny.sourceAddress,/All Active Lists/ArcSight System/Threat Tracking/Reconnaissance List);Or;
6. "Not" InActiveList(Protocol\_Deny.destinationAddress,/All Active Lists/ArcSight System/Targets/Scanned List));And;
7. EQ(Protocol\_Deny.categoryDeviceGroup,/Firewall);And;
8. EQ(Protocol\_Deny.categoryOutcome,/Failure))

Проверка протокола приложении

1. Matching 1 events in 5 Minute with conditions(
2. "Not" InActiveList(FirewallAcceptsMovingAverageEvent.sourceAddress,/All Active Lists/ArcSight System/Attackers/Trusted List);And;
3. EQ(FirewallAcceptsMovingAverageEvent.name,Firewall Accepts);And;
4. EQ(FirewallAcceptsMovingAverageEvent.deviceEventCategory, rising))

Прием больших объемов данных

1. Matching 5 events in 3 Minute with conditions(
2. (EQ(Deny\_TCP\_UDP.categoryBehavior,/Access);Or;
3. EQ(Deny\_TCP\_UDP.categoryBehavior,/Access/Start));And;
4. "Not" InActiveList(Deny\_TCP\_UDP.sourceAddress,/All Active Lists/ArcSight System/Attackers/Trusted List);And;
5. ("Not" InActiveList(Deny\_TCP\_UDP.sourceAddress,/All Active Lists/ArcSight System/Threat Tracking/Suspicious List);Or;
6. "Not" InActiveList(Deny\_TCP\_UDP.destinationAddress,/All Active Lists/ArcSight System/Targets/Scanned List));And;
7. EQ(Deny\_TCP\_UDP.categoryDeviceGroup,/Firewall);And;
8. EQ(Deny\_TCP\_UDP.categoryOutcome,/Failure))

Сканирование сетевых портов

1. Matching 3 events in 1 Minute with conditions(
2. NE(event1.type,Correlation);And;
3. EQ(event1.categoryBehavior,/Authentication/Verify);And;
4. EQ(event1.categoryObject,/Host/Application/Database);And;
5. EQ(event1.categoryOutcome,/Failure))

Неудачные попытки доступа к базе данных

1. Matching 5 events in 2 Minute with conditions(
2. EQ(event1.type,Base);And;
3. EQ(event1.deviceProduct,Microsoft Windows);And;
4. EQ(event1.deviceVendor,Microsoft);And;
5. EQ(event1.deviceEventClassId,Security:531))

Неудачные попытки доступа в заблокированный аккаунт

1. Matching 1 events in 1 Minute with conditions(
2. GE(event1.LoginCountActiveList,5);And;
3. EQ(event1.name,Successful Windows Login))

Несколько попыток входа одним и тем же аккаунтом

1. Matching 1 events in 1 Minute with conditions(
2. (InActiveList(SuspiciousFirewallPass.sourceAddress,/All Active Lists/ArcSight Foundation/Intrusion Monitoring/Attackers/Repetitive Firewall Block List);Or;
3. InActiveList(SuspiciousFirewallPass.sourceAddress,/All Active Lists/ArcSight System/Attackers/Untrusted List));And;
4. (EQ(SuspiciousFirewallPass.categoryBehavior,/Access);Or;
5. EQ(SuspiciousFirewallPass.categoryBehavior,/Access/Start));And;
6. "Not" InActiveList(SuspiciousFirewallPass.sourceAddress,/All Active Lists/ArcSight System/Attackers/Trusted List);And;
7. "Not" InActiveList(SuspiciousFirewallPass.sourceAddress,/All Active Lists/ArcSight System/Threat Tracking/Suspicious List);And;
8. EQ(SuspiciousFirewallPass.categoryDeviceGroup,/Firewall);And;
9. EQ(SuspiciousFirewallPass.categoryOutcome,/Success))

Проход после повторяющихся блоков

1. Matching 1 events in 1 Second with conditions(
2. EQ(event1.name,Windows Event);And;
3. (EQ(event1.deviceEventClassId,Security:643);Or;
4. EQ(event1.deviceEventClassId,Microsoft-Windows-Security-Auditing:4739));And;
5. EQ(event1.message,Password Policy);And;
6. EQ(event1.type,Base);And;
7. EQ(event1.categoryOutcome,/Success))

Изменения политики в отношении паролей

1. Matching 1 events in 2 Minute with conditions(
2. EQ(event1.name,Address or Username Present);And;
3. NE(event1.type,Correlation);And;
4. EQ(event1.categoryBehavior,/Authentication/Verify);And;
5. EQ(event1.categoryDeviceGroup,/VPN);And;
6. InActiveList(event1.ActorByAccountID,/All Active Lists/ArcSight Solutions/IdentityView/Book Keeping/Badged In Actors))

Доступ к VPN

1. Matching 1 events in 2 Minute with conditions(
2. EQ(event1.deviceProduct,ArcSight);And;
3. EQ(event1.deviceVendor,ArcSight);And;
4. EQ(event1.deviceCustomNumber1,3);And;
5. EQ(event1.filePath,Disabled Accounts);And;
6. EQ(event1.name,ActiveList entry updated))

Повторное отключение учетной записи

### 4.2 Правила корреляции с функцией Autorules

Ссылка на имя пользователя должна быть дополнена информацией о сети:

2 "Not" InActiveList(event1.sourceUserName,/All Active Lists/ArcSight Solutions/IdentityView/Book Keeping/Badged In Actors);And;

Warning - Account verifications should rely on the assetmodel

Warning - The rule does not use multiple event sources

Final score: 8

Autorules действий по правилам для отстранённого сотрудника

Possible loophole in active list exceptions:

"Not" InActiveList(Protocol\_Deny.sourceAddress,/All Active Lists/ArcSight System/Attackers/Trusted List);And;

Address reference should be complemented with account information

("Not" InActiveList(Protocol\_Deny.sourceAddress,/All Active Lists/ArcSight System/Threat Tracking/Reconnaissance List);Or;

Address reference should be complemented with account information

"Not" InActiveList(Protocol\_Deny.destinationAddress,/All Active Lists/ArcSight System/Targets/Scanned List));And;

Warning - Network conditions should rely on the network model Warning - The rule does not use multiple event sources

Final score: 15

Сканирование протокола приложений

Possible loophole in active list exception:

2 "Not" InActiveList(FirewallAcceptsMovingAverageEvent.sourceAddress,/All Active Lists/ArcSight System/Attackers/Trusted List);And;

Warning - The rule does not use multiple event sources

Final score: 10

Приём большого объёма информации

Possible loophole in active list exceptions:

"Not" InActiveList(Deny\_TCP\_UDP.sourceAddress,/All Active Lists/ArcSight System/Attackers/Trusted List);And;

Address reference should be complemented with account information

("Not" InActiveList(Deny\_TCP\_UDP.sourceAddress,/All Active Lists/ArcSight System/Threat Tracking/Suspicious List);Or;

Address reference should be complemented with account information

"Not" InActiveList(Deny\_TCP\_UDP.destinationAddress,/All Active Lists/ArcSight System/Targets/Scanned List));And;

Warning - Network conditions should rely on the network model Warning - The rule does not use multiple event sources

Final score: 15

Сканирование сетевых портов

Possible loophole in active list exceptions:

"Not" InActiveList(SuspiciousFirewallPass.sourceAddress,/All Active Lists/ArcSight System/Attackers/Trusted List);And;

Address reference should be complemented with account information

(InActiveList(SuspiciousFirewallPass.sourceAddress,/All Active Lists/ArcSight Foundation/Intrusion Monitoring/Attackers/Repetitive Firewall Block List);Or;

Address reference should be complemented with account information

InActiveList(SuspiciousFirewallPass.sourceAddress,/All Active Lists/ArcSight System/Attackers/Untrusted List));And;

Address reference should be complemented with account information

"Not" InActiveList(SuspiciousFirewallPass.sourceAddress,/All Active Lists/ArcSight System/Threat Tracking/Suspicious List);And;

Warning - Network conditions should rely on the network model Warning - The rule does not use multiple event sources

Final score: 17

Пропуск после повторных блокировок

Address reference should be complemented with account information

2 EQ(event1.name,Address or Username Present);And;

Warning - Network conditions should rely on the network model Warning - The rule does not use multiple event sources

Final score: 8

Доступ к VPN

### 4.3 Алгоритмы определения атаки

В системе SIEM (Security Information and Event Management) используются следующие алгоритмы определения атак:  
  
1. Анализ поведения: Система SIEM может анализировать поведение пользователей и устройств в сети, чтобы обнаружить любые необычные или подозрительные действия, которые могут указывать на атаку.  
  
2. Корреляция событий: SIEM может коррелировать данные из разных источников, таких как журналы, устройства защиты, системы обнаружения вторжений, чтобы выявить связи между событиями и обнаружить потенциальные атаки.  
  
3. Использование правил: SIEM может использовать заранее определенные правила для обнаружения конкретных типов атак. Например, если система обнаруживает несколько попыток входа в систему с неправильным паролем, она может сработать правило на предмет подбора паролей.  
  
4. Машинное обучение: Некоторые системы SIEM используют машинное обучение для обнаружения новых и неизвестных атак. Система может обучаться на основе исторических данных и использовать эту информацию для определения потенциальных угроз.  
  
5. Анализ угроз: SIEM может использовать базы данных угроз и информацию о новых уязвимостях, чтобы обнаружить потенциальные атаки, которые могут быть связаны с конкретными уязвимостями.  
  
Эти алгоритмы могут использоваться в комбинации для обнаружения и предотвращения различных типов атак на сеть.

В нашем случае, используется комбинация из всех пунктов, с упором на правила корреляции и использование правил, с их дальнейшим усовершенствованием. В целом, каждая компания может выбирать любую комбинацию или конкретное правило для своей системы.

Блок схема определения обычной кибератаки:

1. Мониторинг сетевой активности: отслеживание входящего и исходящего трафика на сетевых устройствах.  
2. Проверка журналов событий: анализ журналов событий на наличие подозрительной активности, такой как неудачные попытки входа в систему или изменения конфигурации.  
3. Анализ файлов: сканирование файлов на наличие вредоносных программ и анализ подозрительных файлов.  
4. Анализ поведения: мониторинг активности пользователей и устройств на предмет необычных действий или поведения.  
5. Сравнение с базой данных угроз: сравнение обнаруженной активности с базой данных известных угроз и вредоносных программ.  
6. Оповещение о возможной кибератаке: при обнаружении подозрительной активности, генерируется оповещение для ответственных лиц для дальнейшего расследования.

Предоставить сведения о других вариантах кибератак (по типу 0-day/ 1 day и прочих) я не могу, поскольку это является уголовно наказуемым. Я могу предоставить лишь базовую схему, ибо каждая новая атака, как правило отличается, совершенствуется и по закону распространять их нельзя.

## Заключение

Таким образом, была изучена и проанализирована система SIEM, а именно её составляющие, вспомогательная структура, их корректная работа. Также подробно расписаны возможные неисправности в каждой из систем, имеются исследования в этой области.

Более подробно расписаны выбранные правила корреляции, их работа, преимущества и недостатки. Опытным путём мы постарались продемонстрировать варианты их улучшения, а именно: более сложный вариант входа, который снижает вероятность взлома путем метода грубой силы. Аналогично был усовершенствован вход, путем увеличения количества времени между попытками. Также автоматическое удаление аккаунта в течение 30 минут неудачных попыток.

Все эти усовершенствования способны помочь хотя бы на время задержать преступление, и если сигнал будет распознан и быстро обработан, то высока вероятность того, что атака не увенчается успехом и доступ к данным преступники не получат, в худшем случае пострадает небольшое количество данных.

Естественно, способы атак, как и способы защиты с каждым днём совершенствуются, следовательно, необходимо регулярно совершенствовать систему безопасности. И защита с использованием правил корреляции, это всего лишь один из вариантов защиты, который также необходимо совершенствовать. Проблема этого способа лишь в том, что он не имеет огромной популярности, и многие книги и пособия уже устарели, в то время как атаки стали намного жестче и продуманней.

# Список литературы

1. Агирре, И., Алонсо, С., "Улучшение автоматизации управления информацией о безопасности: Совместный подход", IEEE Security and Privacy Magazine, январь / февраль 2012 г.

2. Эйдемарк, Дж., Карлссон, Дж., "Использование массивной временной избыточности для достижения устойчивости к переходным отказам на уровне узла", конференция аспирантов ARTES, 2000.

3. Брисемайстер, Л., Чунг, С., Линдквист, У., Вальдес, А., "Обнаружение, корреляция и визуализация атак на системы критической инфраструктуры", в материалах Восьмой ежегодной конференции по конфиденциальности, безопасности и доверию, Оттава, Канада, август 2010 г.

4. Falliere, N., Murchu, L., Chien, E. "W32.Stuxnet Dossier", Symantec Tec Rep. February 2011.

5. Gartner, http://www.gartner.com/it-glossary/security-information-and-event- management-siem/, доступ получен в июле 2013 года.

6. Haddix, J. et al, "HP 2012 Cyber Risk Report", март 2013.

7. Проект MASSIF FP7, http://www.massif-project.eu/.

8. Миллер, Д., Харрис, С., Харпер, А., Вандайк, С., Бласк, К., "Внедрение управления информацией и событиями безопасности (SIEM)", МакГроу-Хилл Осборн, 2010.

9. Навин, К., "Анализ безопасности регистрации системных событий с помощью Syslog", Институт SANS, 2003.

10. Николетт, М., Каванаг, К., "Магический квадрант для управления информацией и событиями безопасности", май 2012 г.

11. Олинер, А., Кулкарни, А., Айкен, А., "Обнаружение эпидемии сообщества с помощью коррелированных по времени аномалий", в материалах 13-й Международной конференции по последним достижениям в области обнаружения вторжений, Оттава, Канада, 2010.

12. Oracle Security Alert for CVE-2013-0422. [http://www.oracle.com/technetwork/topics/security/alert-cve-2013-0422- 1896849.html](http://www.oracle.com/technetwork/topics/security/alert-cve-2013-0422-%201896849.html)

13. Сталлингс В., "Криптография и сетевая безопасность", пятое издание, Pearson Education, 2011.

14. Тодд, А., Рейнс, Р., Болдуин, Р., Маллинз, Б., Роджерс, С., "Уклонение от проверки оповещений путем подделки ответов сервера", в материалах 10-й Международной конференции по последним достижениям в области обнаружения вторжений, Голд-Кост, Австралия, 2007.

15. Вальдес, А., "Голубые датчики, корреляция датчиков и объединение оповещений", в материалах 3-й Международной конференции по последним достижениям в области обнаружения вторжений, Тулуза, Франция, 2000.

16. Вальдес, А., Скиннер, К., "Вероятностная корреляция оповещений", в материалах 4-й Международной конференции по последним достижениям в области обнаружения вторжений, Дэвис, Калифорния, США, 2001.

17. Вериссимо, П., Невес, Н., Коррейя, М., "Устойчивые к вторжениям архитектуры: Концепции и дизайн", технический отчет DI/FCUL TR-08-5, факультет компьютерных наук, Оксфордский университет, апрель 2003 года.