Особенности выбора систем обнаружения вторжений: всегда ли "больше" означает "лучше"?

Владимир Воротников, *руководитель отдела интеграционных решений, ООО "С-Терра СиЭсПи"*

При выборе системы обнаружения вторжений (СОВ) необходимо учитывать множество параметров: удобство интерфейса управления и мониторинга, производительность, частоту обновления и актуальность сигнатурных баз, возможности централизованного управления и многое другое. Но все ли из этих параметров так очевидны, как кажется? Как показывает практика, подход, основанный сугубо на количественном сравнении характеристик, в случае технически сложных продуктов работает плохо. В данной статье рассмотрим некоторые параметры систем обнаружения вторжений (или, как их еще называют, систем обнаружения атак (СОА)), выбор по которым не так очевиден, как кажется на первый взгляд.

Производительность

Казалось бы с чем-чем, а с производительностью все ясно: чем больше, тем лучше. Типовые: "40 гигабит лучше, чем 10, потому что есть запас по прочности на случай пиковых нагрузок", и тому подобные с первого взгляда технически безупречные аргументы позволяют дороже продать избыточно мощное решение. Да, действительно, если вы анализируете трафик с 10-гигабитного канала, то вам необходим продукт, способный переварить этот объем трафика. Но только ли скорость обработки отдельных пакетов ограничивает общую производительность СОВ? Тот, кто работал с системами обнаружения вторжений, сразу скажет, что нет. В ходе анализа пакетов генерируются записи об инцидентах (событиях). И работа с базой этих инцидентов очень легко может стать узким местом системы. При определенных условиях объем генерируемых записей о событиях может составлять гигабайты и десятки гигабайт в сутки. Через месяц работы запросы к такой базе могут стать слишком "неторопливыми". Тогда производительность по обработке трафика будет слабо полезна: администратор просто не сможет оперативно добраться до записей о нужных инцидентах.

Что нужно делать в таком случае? Как и к любой другой технически сложной системе, следует подойти к делу с умом и должным профессионализмом. Есть ли хоть малейший шанс, что администратор обработает все гигабайты данных об инцидентах, созданные в течение суток? Конечно, нет. Но очевидно, что ему это и не необходимо: ну не может в его сети произойти столько интересующих его событий. А значит, администратор должен посвятить первые недели работы с системой "притирке". Откинуть правила и группы правил (всегда удобно, когда это можно сделать группой, см. рис. 1), которые для его сети не актуальны. Разобраться с наиболее шумными оповещениями: если угрозы в них актуальны, то устранить их, если же не актуальны – отключить правила.

Для наглядности: классическим примером сильно шумящего протокола является torrent. Во многих системах обнаружения атак по умолчанию он генерирует множество событий. Если администратор считает использование торрентов в своей сети допустимым, то соответствующие правила можно отключить, и события перестанут возникать. Если он считает их недопустимыми, то их необходимо заблокировать теми или иными средствами, и, опять-таки, события перестанут возникать. Как результат, администратор сможет сконцентрироваться на действительно важных и уникальных инцидентах. В идеале хорошо настроенный СОВ должен генерировать не более нескольких

Выберите класс правил:

shellcode-detect | Поиск по описанию: | Поиск

Включить все | Выключить все | Открыть SID: | Открыть

После включения/отключения правил необходимо перезагрузить правила (Управление -> Общее -> Сервисные функции)

SID правила	Класс	Файл правила	Описание правила (msg)
2010408	shellcode-detect	emerging-deleted.rules	ET DELETED METASPLOIT BSD Reverse shell (PexFnstenvSub Encoded 2)
2010422	shellcode-detect	emerging-deleted.rules	ET DELETED METASPLOIT BSD Reverse shell (PexFnstenvMov Encoded 2)
2003117	shellcode-detect	emerging-deleted.rules	ET DELETED SHELLCODE CLET polymorphic payload
2003118	shellcode-detect	emerging-deleted.rules	ET DELETED SHELLCODE Shikata Ga Nai polymorphic payload
2003119	shellcode-detect	emerging-deleted.rules	ET DELETED SHELLCODE ADMutate polymorphic payload
2001369	shellcode-detect	emerging-exploit.rules	ET EXPLOIT MS04-032 Windows Metafile (.emf) Heap Overflow Exploit

Рис. 1. Всегда удобно, когда есть возможность отключить сразу группу правил

событий в день, каждому из которых администратор сможет уделить достаточно времени, чтобы разобраться в причинах и предпринять адекватные действия.

Итак, производительность – это важно, но гораздо важнее – грамотно настроить процесс, и тогда даже более дешевое и менее производительное решение принесет реальной практической пользы в разы больше монструозного терминатороподобного многокиловаттного пожирателя трафика.

IDS или IPS?

Еще один вопрос, который может стоять при выборе решения, а должны ли у системы обнаружения вторжений быть функции предотвращения атак. Другими словами, нужен нам пассивный наблюдатель или активное оборудование, которое сможет автоматически генерировать правила для межсетевого экрана (МЭ) и блокировать подозрительный трафик. На первый взгляд, чем функциональнее устройство, тем лучше: если мы можем возложить реакцию на инциденты на автоматику, то почему бы и нет? Но и здесь не все так просто.

В любой сколь угодно совершенной системе обнаружения атак обязательно будут присутствовать ложные срабатывания. В том случае, когда устройство способно вмешаться в передаваемый трафик, ценой таких ложных срабатываний может быть недоступность сервиса для добропорядочных пользователей. Тут нужен индивидуальный подход к каждой конкретной системе: где-то конфиденциальность и целостность данных в разы важнее их доступности, и тогда отказ сервиса является разумной платой за сохранность данных, но встречаются и обратные ситуации, когда именно доступность данных представляет главную ценность. Также следует понимать, что автоматические блокирующие правила IPS часто "лечат" симптоматически, что далеко не всегда устраняет источник проблемы. И в большинстве случаев только квалифицированный администратор может по-настоящему эту проблему решить.

Таким образом, IPS – это не панацея, а инструмент, который имеет свою область применения и свои ограничения. IPS не лучше, чем IDS, у него немного другие задачи и возможности.

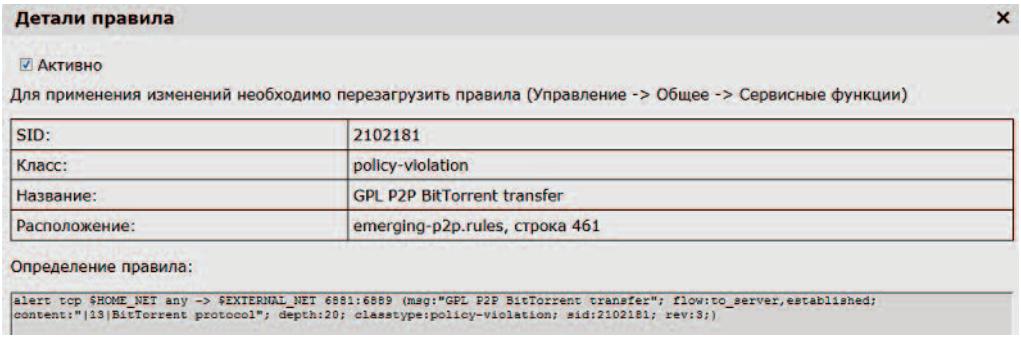

Рис. 2. Пример сигнатуры

Нужны ли они – зависит от каждого конкретного случая.

Сигнатурный или эвристический?

Практически все современные СОВ обладают возможностью сигнатурного анализа трафика. Этот подход относительно прост в понимании, и в этом его главная сила: всегда возможно точно установить, какой конкретно пакет (или группа пакетов) вызвал срабатывание сенсора. Все правила четко определены, для многих из них можно проследить всю цепочку: от информации о деталях уязвимости и методах ее эксплуатации до результирующей сигнатуры. Сама база правил огромна и регулярно пополняется. Сигнатурный подход не без слабых мест: критическим является интервал между появлением способа эксплуатации новой уязвимости и появлением соответствующей сигнатуры. Кроме того, сигнатурный анализ не позволяет обнаруживать новые, ранее неизвестные атаки. Но в целом можно сказать, что сигнатурный подход работает хорошо (рис. 2).

Эвристический анализ, в свою очередь, призван компенсировать некоторые минусы сигнатурного подхода. Справляется ли он с этим? Отчасти. В основе работы эвристического анализатора всегда заложена схема, в которой в режиме обучения формируются "правильные" шаблоны поведения системы, а в режиме анализа – обнаруживаются отклонения от этих шаблонов. За счет этого эвристический анализатор действительно может обнаружить вредоносную активность, не попавшую ни под какую конкретную сигнатуру. Но и минусов в этом подходе предостаточно. И особенно эти минусы становятся явными, когда эвристический анализ выходит за пределы лабораторий и попадает в реальные сети. Во-первых, необходимо проводить обучение на своей сети и при этом быть уверенным, что в ней на данный момент нет никакой вредоносной активности (иначе она "запишется" в шаблоны и не будет определяться как аномальная). На практике это не всегда просто реализовать. Кроме того, реальная сеть находится в постоянном развитии: появляются новые пользователи, добавляются новые сервисы, а старые закрываются, меняется количество и распределение трафика. При резких изменениях вы будете получать огромное количество ложных срабатываний.

Да и с плавными изменениями не все гладко: конечно, эвристический анализатор будет при них корректно переобучаться. Но на этой его особенности построен целый класс атак, которые постепенно, малыми изменениями переучивают ваш анализатор распознавать вредоносное поведение как корректное. Также есть ряд сложностей с расследованием инцидентов, сгенерированных эвристическим анализатором.

Безусловно, у эвристического анализа трафика есть будущее. Но на данный момент на практике он работает хорошо только в ограниченном ряде относительно простых случаев и пока не является опцией must have для СОВ, хотя и может оказать помощь квалифицированному администратору, понимающему все его возможности и ограничения.

Подводя итоги, можно сказать, что подход "чем больше, тем лучше" не работает в случае технически сложных устройств, и в частности СОВ. Все сети и их потребности уникальны. Если инженеры будут учитывать особенности конкретной сети, то выбрана будет не "самая навороченная", а "самая полезная" система обнаружения вторжений. ●

Ваше мнение и вопросы присылайте по адресу is@groteck.ru

Классическим примером сильно шумящего протокола является torrent. Во многих системах обнаружения атак по умолчанию он генерирует множество событий. Если администратор считает использование торрентов в своей сети допустимым, то соответствующие правила можно отключить, и события перестанут возникать. Если он считает их недопустимыми, то их необходимо заблокировать теми или иными средствами, и, опять-таки, события перестанут возникать.