Конфигурация переменной SNORT_BPF на Центре Защиты

Содержание

Введение

Можно использовать Фильтр пакета Беркли (BPF) для исключения хоста или сети от того, чтобы быть осмотренным Центром Защиты. Фырканье использует переменную **`Snort_BPF`** для исключения трафика из политики проникновения. Этот документ предоставляет инструкции по тому, как использовать переменную **`Snort_BPF`** в различных сценариях.

> **Совет:** Строго рекомендуется использовать трастовое правило в Политике контроля доступа для определения то, что трафик и не осмотрен, а не BPF в политике проникновения. Переменная **`Snort_BPF`** доступна на версии программного обеспечения 5.2 и осуждается на версии программного обеспечения 5.3 или выше.

Предварительные условия

Требования

Cisco рекомендует ознакомиться на Центре Защиты, Политике Проникновения, Фильтре пакета Беркли и правилах Фырканья.

Используемые компоненты

Сведения в документе приведены на основе данных версий аппаратного и программного

обеспечения:

- Центр защиты
- Версия программного обеспечения 5.2

Сведения, представленные в этом документе, были получены от устройств, работающих в специальной лабораторной среде. Все устройства, описанные в этом документе, были запущены с чистой (стандартной) конфигурацией. В рабочей сети необходимо изучить потенциальное воздействие всех команд до их использования.

Порядок действий для настройки

Для настройки переменной **`Snort_BPF`** выполните действия ниже:

1. Обратитесь к интерфейсу веба - пользователя своего Центра Защиты.
2. Перейдите к **Политике> Проникновение> Политика Проникновения**.
3. Нажмите *значок карандаша* для редактирования политики проникновения.
4. Щелкните **по Variables** из меню слева.
5. Как только переменные настроены, необходимо будет сохранить изменения и повторно применить политику проникновения для нее для вступления в силу.

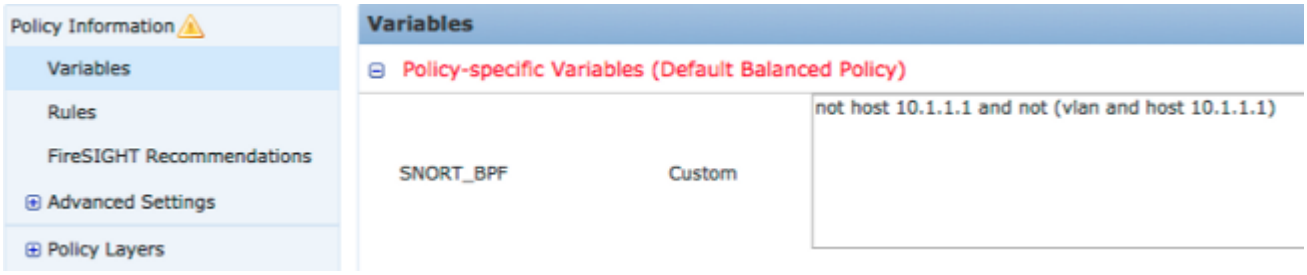

Рисунок: Снимок экрана страницы переменной конфигурации Snort_BPF

Примеры конфигураций

Некоторые базовые примеры предоставлены ниже для ссылки:

Сценарий 1: Проигнорируйте весь трафик, к и FROM сканнер уязвимости

1. У нас есть сканнер уязвимости в IP-адресе 10.1.1.1
2. Мы хотим проигнорировать весь трафик к и FROM сканер
3. Трафик может или может не иметь 802.1q (vlan) метка

`SNORT_BPF` :
`not host 10.1.1.1 and not (vlan and host 10.1.1.1)`СРАВНЕНИЕ: трафик *не* помечен VLAN, но указывает 1, и 2 остаются истинными, был бы: **`not host 10.1.1.1`**На простом английском языке это проигнорировало бы трафик, где одна из оконечных точек 10.1.1.1 (сканер).

Сценарий 2: Проигнорируйте весь трафик, к и FROM два сканнера уязвимости

1. У нас есть сканнер уязвимости в IP-адресе 10.1.1.1
2. У нас есть второй сканнер уязвимости в IP-адресе 10.2.1.1
3. Мы хотим проигнорировать весь трафик к и FROM сканер
4. Трафик может или может не иметь 802.11 (vlan) метка

SNORT_BPF :
not (host 10.1.1.1 or host 10.2.1.1) and not (vlan and (host 10.1.1.1 or host 10.2.1.1))Сравнение: Трафик *не* помечен VLAN, но указывает 1, и 2 остаются истинными, был бы: **not (host 10.1.1.1 or host 10.2.1.1)**Таким образом, это проигнорировало бы трафик, где одной из оконечных точек является 10.1.1.1 OR 10.2.1.1.

> Примечание: Следует отметить, что тег VLAN, в почти всех случаях, должен произойти только однажды в данном BPF. Единственные времена необходимо видеть его несколько раз, то, если использования сети вложили маркирование VLAN (иногда называемый 'QinQ').

Ситуация 3: Проигнорируйте помеченный трафик VLAN, к и FROM два сканнера уязвимости

1. У нас есть сканнер уязвимости в IP-адресе 10.1.1.1
2. У нас есть второй сканнер уязвимости в IP-адресе 10.2.1.1
3. Мы хотим проигнорировать весь трафик к и FROM сканер
4. Трафик является 802.11 (vlan), помеченный, и вы хотите использовать определенное (vlan) метка, как в VLAN 101

SNORT_BPF :

```
not (host 10.1.1.1 or host 10.2.1.1) and not (vlan 101 and (10.1.1.1 or host 10.2.1.1))
```

Сценарий 4: Проигнорируйте трафик от сервера резервного копирования

1. У нас есть сервер резервирования сети в IP-адресе 10.1.1.1
2. Машины в сети connect к этому серверу на порту 8080 для выполнения их ночной резервной копии
3. Мы хотим проигнорировать этот резервный трафик, поскольку он зашифрован и большой объем

SNORT_BPF :

not (dst host 10.1.1.1 and dst port 8080) and not (vlan and (dst host 10.1.1.1 and dst port 8080)) **Сравнение:** Трафик *не* помечен VLAN, но указывает 1, и 2 остаются истинными, был бы: **not (dst host 10.1.1.1 and dst port 8080)**

Преобразованный, это означает, что трафик к 10.1.1.1 (наш гипотетический сервер резервного копирования) на порту 8080 (порт прослушивания) не должен быть осмотрен механизмом обнаружения IPS.

Также возможно использовать `сеть` вместо `хоста` для определения сетевого блока, а не одного хоста . Пример:

not net 10.1.1.0/24

В целом это - полезный прием для создания BPF максимально определенным; исключая трафик от контроля, который должен быть исключен, в то время как не, исключая любой не связанный друг с другом трафик, который мог бы содержать попытки использования.

Сценарий 5: Для использования диапазонов сети, а не отдельных хостов

Можно задать диапазоны сети в переменной BPF, а не хостах для сокращения длины переменной. Чтобы сделать так, вы будете использовать `сетевое` ключевое слово вместо хоста и задавать диапазон CIDR. Ниже представлен пример:

not (dst net 10.8.0.0/16 and dst port 8080) and not (vlan and (dst net 10.8.0.0/16 and dst port 8080))

> **Примечание:** Гарантируйте ввод сетевого адреса с помощью системы обозначений CIDR и применимого адреса в адресном пространстве блока CIDR. Например, используйте сеть 10.8.0.0/16 , а не сеть 10.8.2.16/16.

Переменная **SNORT_BPF** используется, чтобы препятствовать тому, чтобы определенный трафик был осмотрен механизмом обнаружения IPS; часто для целей повышения производительности. Эта переменная использует стандартный формат Фильтров пакета Беркли (BPF). Трафик, совпадающий с переменной **SNORT_BPF**, будет осмотрен; в то время как трафик, НЕ совпадающий с переменной **SNORT_BPF**, НЕ будет осмотрен механизмом обнаружения IPS.