Содержание

Введение

Этот документ описывает, как использовать регулярные выражения (regex) с **командой grep** для поиска журналов.

Предварительные условия

Сведения, содержащиеся в данном документе, касаются следующих версий программного обеспечения и оборудования:

- Cisco Web Security Appliance (WSA)
- Cisco Email Security Appliance (ESA)
- Устройство менеджмента Cisco Security (SMA)

Grep с Regex

Regex может быть мощным программным средством, когда используется с **командой grep** перерыть журналы, доступные на устройстве, такие как Журналы Доступа, Журналы Прокси и другие. Можно искать журналы на основе веб-сайта, или любую часть URL и имена пользователей с командой CLI **grep**.

Вот некоторые общие сценарии, где можно использовать regex с **командой grep** для помощи с устранением проблем.

Сценарий 1: Найдите определенный веб-сайт в журналах доступа

Наиболее распространенный сценарий - когда вы пытаетесь найти запросы, которые выполнены к веб-сайту в журналах доступа WSA.

Например:

Соединитесь с устройством через Secure Shell (SSH). Как только у вас есть приглашение, введите **команду grep** для распечатки доступных журналов.

```
CLI> grep
```

Введите номер журнала, которого вы желаете к **grep**.

```
[]> 1 (Choose the # for access logs here)
```

Введите регулярное выражение в **grep**.

```
[]> website\.com
```

Сценарий 2: Попытайтесь найти определенное расширение файла или домен верхнего уровня

Можно использовать **команду grep** для обнаружения определенного расширения файла (.doc, .pptx) в URL или домене верхнего уровня (.com, .org).

Например:

Для обнаружения всех URL, которые заканчиваются .crl, используют этот regex:

```
[]> website\.com
```

Для обнаружения всех URL, которые содержат расширение файла .pptx, используют этот regex:

```
[]> website\.com
```

Ситуация 3: Попытайтесь найти определенный блок для веб-сайта

При поиске определенного веб-сайта вы могли бы также искать определенный Ответ HTTP.

Например:

Если вы хотите искать все сообщения TCP_DENIED/403 для domain.com, используйте этот regex:

```
[]> website\.com
```

Сценарий 4: найдите имя машины в журналах доступа

При использовании схемы проверки подлинности NTLMSSP вы могли бы встретиться с экземпляром, куда Клиент User Agent (Microsoft NCSI наиболее распространена) неправильно передает учетные данные машины вместо учетных данных пользователя, когда это аутентифицируется. Для разыскивания URL/клиента User Agent, который вызывает эту проблему, используйте regex с **grep** для изоляции запроса, выполненного, когда произошла аутентификация.

Если у вас нет имени машины, которое использовалось, используйте **grep** и найдите все имена машины, которые использовались в качестве имен пользователей при аутентификации с этим regex:

```
[]> website\.com
```
Как только у вас есть линия, где это происходит, grep для определенного имени машины, которое использовалось с этим regex:

```
[]> website\.com
```
Первая запись, которая появляется, должна быть запросом, который был выполнен, когда пользователь аутентифицировался с именем машины вместо имени пользователя.

Сценарий 5: найдите определенный период времени в журналах доступа

По умолчанию подписки журнала доступа не включают поле, которое показывает человекочитаемую дату/время. Если вы хотите проверить журналы доступа в течение периода определенного времени, выполните эти шаги:

1. Поиск метка времени UNIX от узла, такого как Онлайновое Преобразование.
2. Как только вы имеете метку времени, ищете специфическое время в Журналах Доступа.

Например:

Метка времени Unix **1325419200** эквивалентна **01.01.2012 12:00:00**.

Можно использовать эту запись regex для поиска журналов доступа близко к 12:00 1 января 2012:

13254192

Сценарий 6: поиск критических или предупреждающих сообщений

Можно искать критические или предупреждающие сообщения в любых доступных журналах, таких как журналы прокси или системные журналы, с регулярными выражениями.

Например:

Для поиска предупреждающих сообщений в журналах прокси введите этот regex:

```
CLI> grep
```
Введите номер журнала, которого вы желаете к **grep**.

```
[]> 17 (Choose the # for proxy logs here)
```
Введите регулярное выражение в **grep**.

```
[]> warning
```