Entender as regras do Snort3

Contents

Introduction

Este documento descreve as regras para a **Snort3** no Cisco **Secure Firewall Threat Defense (FTD)**.

Prerequisites

Requirements

A Cisco recomenda que você tenha conhecimento destes tópicos:

- Cisco **Secure Firewall Threat Defense (FTD)**
- **Intrusion Prevention System (IPS)**
- **Snort2** sintaxe

Licenciamento

Não há nenhum requisito específico de licença, a licença básica é suficiente e os recursos mencionados estão incluídos no mecanismo **Snort** dentro do FTD e nas versões de código aberto do **Snort3**.

Componentes Utilizados

As informações neste documento são baseadas nestas versões de software e hardware:

- Cisco **Secure Firewall Threat Defense (FTD)**,Cisco **Secure Firewall Management Center (FMC)** versão 7.0+ com **Snort3**.

The information in this document was created from the devices in a specific lab environment. All of the devices used in this document started with a cleared (default) configuration. Se a rede estiver ativa, certifique-se de que você entenda o impacto potencial de qualquer comando.

Informações de Apoio

Snort O mecanismo Cisco IPS é capaz de analisar o tráfego em tempo real e registrar pacotes.

Snort A pode executar análise de protocolo, pesquisa de conteúdo e detectar ataques.

Snort3 é uma versão atualizada do Snort2 IPS com uma nova arquitetura de software que melhora o desempenho, a detecção, a escalabilidade e a usabilidade.

Regras do Snort3

Eles usam esse formato LUA para fazer o **Snort3** regras mais fáceis de ler, gravar e verificar.

Ações da regra

Essa nova versão altera as ações da regra, as novas definições são:

- **Pass**: Parar a avaliação das regras subsequentes em relação ao pacote
- **Alert**: Gerar apenas evento
- **Block**: Descartar pacote, bloquear sessão restante
- **Drop**: Descartar apenas pacote
- **Rewrite**: Obrigatório se a opção replaces for usada
- **React**: Enviar página de resposta de bloco HTML
- **Reject**: Injetar TCP RST ou ICMP inalcançável

Anatomia da regra

A anatomia é:

O cabeçalho da regra contém a ação, o protocolo, a(s) rede(s) de origem e de destino e a(s) porta(s).

IN **Snort3**, o cabeçalho da regra pode ser uma das próximas opções:

- Cabeçalho da regra de serviço

```
<iline" lang="lua">alert http ( msg:"Alert HTTP rule"; flow:to_client,established;
content:"evil", nocase; sid:1000001; )
```

- Cabeçalho da regra de arquivo

```
alert file ( msg: “Alert File example”; file_data; content:”malicious_stuff”; sid:1000006; )
```

- Cabeçalho de regra convencional

```
alert tcp $EXTERNAL_NET any -> $HOME_NET $HTTP_PORTS ( msg:"Alert HTTP rule";
flow:to_client,established; content:"evil", nocase; sid:1000001; )
```

Recursos da regra

Alguns dos novos recursos são:

- Espaço em branco arbitrário (cada opção em sua própria linha)

```
alert tcp $EXTERNAL_NET any -> $HOME_NET $HTTP_PORTS ( msg:"Alert TCP rule";
flow:to_client,established; content:"evil", nocase; sid:1000000; )
```

- Uso consistente de , e ;

```
content:"evil", offset 5, depth 4, nocase;
```

- Redes e portas são opcionais

```
alert http ( Rule body )
```

- Adiciona mais buffers sticky (esta não é a lista completa)

```
http_uri http_raw_uri http_header http_raw_header http_trailer http_raw_trailer http_cookie
http_raw_cookie http_true_ip http_client_body http_raw_body http_method http_stat_code
http_stat_msg http_version http2_frama_header script_data raw_data
```

- Comentários do estilo C

```
alert http ( msg:"Alert HTTP rule"; /* I can write a comment here */ ... )
```

- Palavra-chave (rem) Remark

```
alert http ( msg:"Alert HTTP rule"; flow:to_client,established; rem:"Put comments in the rule
anywhere"; content:"evil", nocase; sid:1000001; )
```

- palavras-chave appids

```
alert tcp $HOME_NET any -> $EXTERNAL_NET any ( msg:"Alert on apps"; appids:"Google, Google
Drive"; content:"evil", nocase; sid:1000000; )
```

- sd_pattern para filtragem de dados confidenciais
- Palavra-chave Regex com o uso da tecnologia hyperflex
- Palavra-chave Service substitui metadados

Examples

Exemplo com cabeçalho de serviço http e buffer sticky http_uri

Tarefa: Escrever uma regra que detecte a palavra **malicious** no URI HTTP.

Solução:

```
alert http ( msg:"Snort 3 http_uri sticky buffer"; flow:to_server,established; http_uri;
content:"malicious", within 20; sid:1000010; )
```

Exemplo com cabeçalho de serviço de arquivo

Tarefa: Escrever uma regra que detecte arquivos PDF.

Solução:

```
alert file ( msg:"PDF File Detected"; file_type: "PDF"; sid:1000008; )
```

Links relacionados

Download do software Snort Rules e IDS

Github

Sobre esta tradução
A Cisco traduziu este documento com a ajuda de tecnologias de tradução automática e humana para oferecer conteúdo de suporte aos seus usuários no seu próprio idioma, independentemente da localização.

Observe que mesmo a melhor tradução automática não será tão precisa quanto as realizadas por um tradutor profissional.

A Cisco Systems, Inc. não se responsabiliza pela precisão destas traduções e recomenda que o documento original em inglês (link fornecido) seja sempre consultado.