

Utilizando o reconhecimento de aplicativo com base em rede e ACLs para bloquear o worm código vermelho

Índice

[Introdução](#)

[Pré-requisitos](#)

[Requisitos](#)

[Componentes Utilizados](#)

[Convenções](#)

[Como bloquear o worm "Código Vermelho"](#)

[Plataformas suportadas](#)

[Detecte a tentativa de infecção nos registros da Web do IIS](#)

[Marcar hack recebido como "Código Vermelho" utilizando recurso de marcação com base em classe de IOS](#)

[Método A: Use um ACL](#)

[Método B: Use o Policy-Based Routing \(PBR\)](#)

[C do método: Usar vigilância baseada em classe](#)

[Restrições NBAR](#)

[Problemas conhecidos](#)

[Informações Relacionadas](#)

Introdução

Este original fornece um método obstruindo o worm do "código vermelho" em pontos de ingresso de rede com o Network-Based Application Recognition (NBAR) e o Access Control Lists (ACLs) dentro do software de Cisco IOS® no Roteadores de Cisco. Essa solução deve ser usada em conjunto com as correções recomendadas para os servidores IIS da Microsoft.

Note: Este método não trabalha em Cisco 1600 Series Router.

Note: Algum tráfego P2P não pode ser completamente obstruído devido à natureza de seu protocolo P2P. Estes protocolos P2P mudam dinamicamente suas assinaturas para contornar todos os motores DPI que tentam obstruir completamente seu tráfego. Consequentemente, recomenda-se limitar a largura de banda em vez completamente de obstruí-los. Estrangule a largura de banda para este tráfego. Dê muito menos largura de banda; contudo, deixe a conexão ir completamente.

Pré-requisitos

Requisitos

A Cisco recomenda que você tenha conhecimento destes tópicos:

- Políticas de serviços do Qualidade de Serviço (QoS) usando os comandos da [interface de linha do comando modular qos](#) (CLI).
- NBAR
- ACL
- Roteamento baseado em política

Componentes Utilizados

Este original não é restringido à versão de software e hardware específica. A configuração neste original foi testada no Cisco 3640 que executa a versão do Cisco IOS 12.2(24a)

As informações neste documento foram criadas a partir de dispositivos em um ambiente de laboratório específico. Todos os dispositivos usados neste original começaram com uma configuração cancelada (do padrão). Se sua rede está viva, certifique-se de que você compreende o impacto potencial do comando any.

Convenções

Consulte as [Convenções de Dicas Técnicas da Cisco](#) para obter mais informações sobre convenções de documentos.

Como bloquear o worm "Código Vermelho"

A primeira coisa que você deve fazer para combater o “código vermelho” é aplica a correção de programa disponível de Microsoft (veja as relações no [método A da seção: Use um ACL](#) abaixo). Isto protege sistemas vulneráveis e remove o worm de um sistema infectado. Contudo, aplicar a correção de programa a seus server impede somente que o worm contamine os server, ele não para os pedidos HTTP GET de bater os server. Há ainda o potencial para que o server obtenha bombardeado com uma inundação das tentativas de infecção.

A solução detalhada neste advisory é projetada trabalhar conjuntamente com o patch do microsoft para obstruir os pedidos HTTP GET do “código vermelho” em um ponto de ingresso de rede.

Esta solução tenta obstruir a infecção, porém não curará os problemas causados pelo acúmulo de um grande número entradas de cache, adjacências, e entradas NAT/PAT, desde que a única maneira de analisar os índices do pedido HTTP GET está seguindo o estabelecimento de uma conexão de TCP. O seguinte procedimento não ajudará a proteger contra uma varredura da rede. Contudo, protegerá um local da infestação de uma rede externa ou reduzirá o número de tentativas de infecção a que uma máquina deve prestar serviços de manutenção. Em combinação com o filtragem de entrada, o filtragem externa impede que os clientes contaminados espalhem o worm do “código vermelho” aos Internet globais.

Plataformas suportadas

A solução descrita neste original exige os recursos de marcação baseado em classe dentro do

pedido GET não será necessariamente consistente como apenas está tentando criar um excesso de buffer. Isto pode ser visto comparando as duas entradas acima.

Agora está relatando-se que a diferença entre estas duas assinaturas é devido a uma nova tendência do worm do “código vermelho”, do CodeRed.v3 dublado ou do CodeRed.C. A tensão original do “código vermelho” contém a corda “NNNNNNNN” no pedido GET, quando a nova tendência contiver o “”. Refira o [consultivo Symantec](#) para mais detalhes.

Em 6:24PM EDT, o 6 de agosto 2001, nós gravamos uma pegada nova. Nós temos aprendido desde que esta é a pegada que é deixada atrás pelo [scanner de vulnerabilidade do eEye](#) .

```
2001-08-06 22:24:02 10.30.203.202 - 10.1.1.9 80 GET /x.ida AAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA=X 403 HTTP/1.1 -
```

A técnica para obstruir o “código vermelho” fornecido neste advisory pode igualmente obstruir estas tentativas da exploração simplesmente apertando a definição de mapa de classe segundo as indicações da próxima seção.

[Marcar hack recebido como "Código Vermelho" utilizando recurso de marcação com base em classe de IOS](#)

Para obstruir o worm do “código vermelho”, use um dos três métodos descritos abaixo. Todos os três métodos classificam o tráfego malicioso usando a característica do Cisco IOS MQC. Este tráfego é deixado cair então como descrito abaixo.

[Método A: Use um ACL](#)

Este método usa um ACL na interface de saída para deixar cair os pacotes marcados do “código vermelho”. Deixe-nos usar o diagrama de rede seguinte para ilustrar as etapas neste método:



Estão aqui as etapas a configurar este método:

1. Classifique cortes de entrada do “código vermelho” com os recursos de marcação baseado em classe no software do Cisco IOS, como mostrado abaixo:

```
Router(config)#class-map match-any http-hacks
Router(config-cmap)#match protocol http url "*/default.ida*"
Router(config-cmap)#match protocol http url "*/cmd.exe*"
Router(config-cmap)#match protocol http url "*/root.exe"
```

O mapa acima da classe olha dentro dos URL do HTTP e combina algumas das séries especificadas. Observe que nós incluímos outros nomes de arquivo além do default.ida do “código vermelho”. Você pode usar esta técnica para obstruir tentativas de hack similares, tais como o vírus sadmind, que é explicado nos seguintes

documentos:<http://www.microsoft.com/technet/treeview/default.asp?url=/technet/security/bulletin/MS00-078.aspx><http://www.sophos.com/virusinfo/analyses/unixsadmind.html>

2. Construa uma política e use o **comando set** marcar cortes de entrada do “código vermelho” com um mapa de política. Este original usa um valor DSCP de 1 (no decimal) desde que é improvável que algum outro tráfego de rede está levando este valor. Aqui nós marcamos cortes de entrada do “código vermelho” com um mapa de política nomeado “Mark-de entrada-HTTP-cortes”.

```
Router(config)#policy-map mark-inbound-http-hacks
Router(config-pmap)#class http-hacks
Router(config-pmap-c)#set ip dscp 1
```

3. Aplique a política como uma política de entrada na interface de entrada para marcar pacotes de chegada do “código vermelho”.

```
Router(config)#interface serial 0/0
Router(config-if)#service-policy input mark-inbound-http-hacks
```

4. Configurar um ACL que combine no valor DSCP de 1, como ajusta-se pela política de serviços.

```
Router(config)#access-list 105 deny ip any any dscp 1
Router(config)#access-list 105 permit ip any any
```

Note: Cisco IOS Software Releases 12.2(11) e 12.2(11)T introduzem o apoio para a palavra-chave do **log no ACL** na definição em mapas da classe para o uso com NBAR (CSCdv48172). Se você está usando uma versão anterior, não use a palavra-chave do **log no ACL**. Fazer força assim todos os pacotes processo-para ser comutado em vez de comutado por CEF, e o NBAR não trabalhará desde que exige o CEF.

5. Aplique o ACL de partida na interface de saída que conecta aos servidores da Web de destino.

```
Router(config)#interface ethernet 0/1
Router(config-if)#ip access-group 105 out
```

6. Verifique que sua solução trabalha como esperado. Execute o **comando show access-list** e assegure-se de que o valor dos “fósforos” para a instrução de negação esteja incrementando.

```
Router#show access-list 105
Extended IP access list 105
  deny ip any any dscp 1 log (2406 matches)
  permit ip any any (731764 matches)
```

Na etapa de configuração, você pode igualmente desabilitar a emissão de mensagens do IP unreachable com o **comando no ip unreachable interface-level** evitar fazer com que o roteador gaste recursos excessivos. Este método não é recomendado se você pode política-rotar o tráfego DSCP=1 anular 0, como descrito na seção do método B.

[Método B: Use o Policy-Based Routing \(PBR\)](#)

Este método usa o roteamento baseado em política para obstruir pacotes marcados do “código vermelho”. Você não precisa de aplicar os comandos neste método se os métodos A ou o C são configurados já.

Estão aqui as etapas a executar este método:



1. Classifique o tráfego e marque-o. Use os **comandos class-map and policy-map** mostrados no método A.
2. Use o **comando service-policy** aplicar a política como uma política de entrada na interface de entrada para marcar pacotes de chegada do “código vermelho”. Veja o método A.
3. Crie um IP extendido ACL esse fósforos nos pacotes marcados do “código vermelho”.

```
Router(config)#access-list 106 permit ip any any dscp 1
```

4. Use o **comando route-map** construir uma política de roteamento.

```
Router(config)#route-map null_policy_route 10
Router(config-route-map)#match ip address 106
Router(config-route-map)#set interface Null0
```

5. Aplique o mapa de rotas à interface de entrada.

```
Router(config)#interface serial 0/0
Router(config-if)#ip policy route-map null_policy_route
```

6. Verifique seus trabalhos da solução como esperado com o **comando show access-list**. Se você está usando emissores ACL e permitiu o logging ACL, você igualmente pode usar os **comandos show log**, como mostrado abaixo:

```
Router#show access-list 106
Extended IP access list 106
  permit ip any any dscp 1 (1506 matches)
```

```
Router#show log
Aug 4 13:25:20: %SEC-6-IPACCESSLOGP:
  list 105 denied tcp A.B.C.D.(0) -> 10.1.1.75(0), 6 packets
Aug 4 13:26:32: %SEC-6-IPACCESSLOGP:
  list 105 denied tcp A.B.C.D.(0) -> 10.1.1.75(0), 6 packets
```

Nós podemos fazer a decisão de descarte na interface de ingresso do roteador, um pouco do que precisando um emissor ACL em cada interface de saída. Além disso, nós recomendamos desabilitar as mensagens de emissão do IP unreachable com o comando `no ip unreachable`.

[C do método: Usar vigilância baseada em classe](#)

Este método é geralmente o mais escalável porque não depende do PBR ou dos emissores ACL.

1. Classifique o tráfego usando os **comandos class-map** mostrados no método A.
2. Construa uma política usando o **comando policy-map** e use o **comando police** especificar uma ação de queda para este tráfego.

```
Router(config)#policy-map drop-inbound-http-hacks
Router(config-pmap)#class http-hacks
Router(config-pmap-c)#police 1000000 31250 31250
  conform-action drop exceed-action drop violate-action drop
```

3. Use o **comando service-policy** aplicar a política como uma política de entrada na interface de entrada para deixar cair os pacotes do “código vermelho”.

```
Router(config)#interface serial 0/0
```

```
Router(config-if)#service-policy input drop-inbound-http-hacks
```

4. Verifique que sua solução trabalha como esperado com o comando **show policy-map interface**. Assegure-se de que você ver o incremento de valores para a classe e os critérios de verificação de repetição de dados individuais.

```
Router#show policy-map interface serial 0/0
```

```
Serial0/0
```

```
Service-policy input: drop-inbound-http-hacks
```

```
Class-map: http-hacks (match-any)
```

```
5 packets, 300 bytes
```

```
5 minute offered rate 0 bps, drop rate 0 bps
```

```
Match: protocol http url "*default.ida*"
```

```
5 packets, 300 bytes
```

```
5 minute rate 0 bps
```

```
Match: protocol http url "*cmd.exe*"
```

```
0 packets, 0 bytes
```

```
5 minute rate 0 bps
```

```
Match: protocol http url "*root.exe*"
```

```
0 packets, 0 bytes
```

```
5 minute rate 0 bps
```

```
police:
```

```
1000000 bps, 31250 limit, 31250 extended limit
```

```
conformed 5 packets, 300 bytes; action: drop
```

```
exceeded 0 packets, 0 bytes; action: drop
```

```
violated 0 packets, 0 bytes; action: drop
```

```
conformed 0 bps, exceed 0 bps, violate 0 bps
```

```
Class-map: class-default (match-any)
```

```
5 packets, 300 bytes
```

```
5 minute offered rate 0 bps, drop rate 0 bps
```

```
Match: any
```

Restrições NBAR

Ao usar o NBAR com os métodos neste original, note que as seguintes características não estão apoiadas pelo NBAR:

- Mais de 24 URL simultâneas, anfitriões ou MIMICAM o tipo fósforos
- Harmonização além dos primeiros 400 bytes em uma URL
- Tráfego não-IP
- Multicast e outros modos de switching NON-CEF
- Pacotes fragmentados
- Pedidos do HTTP persistentes canalizados
- Classificação URL/HOST/MIME/com HTTP seguro
- Fluxos assimétricos com protocolos stateful
- Pacotes que originam de ou destinado ao roteador que executa o NBAR

Você não pode configurar o NBAR nas seguintes interfaces lógica:

- Fast EtherChannel
- Relações que usam o Tunelamento ou a criptografia
- VLAN
- Interfaces do discador

- Multilink PPP

Note: O NBAR é configurável em VLAN até à data de Cisco IOS Release 12.1(13)E, mas apoiado no trajeto de switching do software somente.

Desde que o NBAR não pode ser usado para classificar o tráfego da saída em uma relação MACILENTO onde a escavação de um túnel ou a criptografia sejam usadas, aplique-a pelo contrário a outras relações no roteador, tal como a interface de LAN, para executar a classificação de entrada antes que o tráfego esteja comutado à relação MACILENTO para a saída.

Para mais informação NBAR, veja as relações na [informação relacionada](#)