

How To Use the CISCO-BULK-FILE-MIB

Índice

[Introdução](#)

[Antes de Começar](#)

[Requisitos](#)

[Componentes Utilizados](#)

[Convenções](#)

[Informações de Apoio](#)

[Usando o CISCO-BULK-FILE-MIB](#)

[Criando uma operação BILK-FILE](#)

[Instruções passo a passo](#)

[Transferindo o arquivo utilizando o CISCO-FTP-CLIENT-MIB](#)

[Instruções passo a passo](#)

[Verificando o resultado](#)

[Troubleshooting do Resultado](#)

[Caveats](#)

[Informações Relacionadas](#)

[Introdução](#)

Este documento explica como usar o CISCO-BULK-FILE-MIB e transfere arquivos criados por essa Base de informações de gerenciamento (MIB) usando o CISCO-FTP-CLIENT-MIB.

Partindo do Software Release 12.0 de Cisco IOS®, Cisco executou uma maneira de armazenar um objeto do Simple Network Management Protocol (SNMP) ou de apresentá-lo como um arquivo no dispositivo. Este arquivo pode então ser recuperado usando o CISCO-FTP-CLIENT-MIB. Essa tecnologia permite transferir grandes volumes de dados usando um método de transporte confiável.

[Antes de Começar](#)

[Requisitos](#)

Antes de tentar esta configuração, verifique se estes requisitos são atendidos:

- Você tem um dispositivo Cisco que executa o Software Release 12.0 ou Mais Recente de Cisco IOS®. Verifique a Ferramenta MIB Locator para certificar-se de que o CISCO-BULK-FILE-MIB seja suportado por seu dispositivo. Um link à ferramenta pode ser encontrado na página das [ferramentas do Cisco IOS MIB](#). **Note:** Esta MIB não é suportada nos dispositivos Catalyst OS.

- O SNMP deve ser configurado no dispositivo com as séries de comunidade de somente leitura e leitura/gravação. Este documento não trata dessa questão. Para obter informações sobre de configurar o SNMP nos dispositivos IOS®, lidos [como configurar séries de comunidade snmp no Roteadores, o Switches XL com Base no Cisco IOS Software, RS, MSFC e Catalyst Switches](#).

Componentes Utilizados

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

- O CISCO-BULK-FILE-MIB deve armazenar o ifTable de um roteador 7507 executando 12.1(12) em um arquivo, em seguida use o CISCO-FTP-CLIENT-MIB para transferir esse arquivo do roteador para um servidor FTP.
- A série do [comando net-snmp snmp](#) instalada em UNIX ou em Windows.
- Este MIBs é usado:SNMPv2-TCSNMPv2-SMISNMPv2-CONFv2-MIBIANAifType-MIBIF-MIBCISCO-SMICISCO-TCCISCO-BULK-FILE-MIBCISCO-FTP-CLIENT-MIB

As informações neste documento foram criadas a partir de dispositivos em um ambiente de laboratório específico. Todos os dispositivos utilizados neste documento foram iniciados com uma configuração (padrão) inicial. Se a sua rede estiver ativa, certifique-se de que entende o impacto potencial de qualquer comando.

Convenções

Para obter mais informações sobre convenções de documento, consulte as [Convenções de dicas técnicas Cisco](#).

Informações de Apoio

Certifique-se de você ter o MIBs nesta tabela carregada em sua plataforma de gerenciamento. Isso permite que você use os nomes de objeto e os valores listados acima em vez dos OIDs (identificadores de objeto). Em geral, este documento se refere aos nomes de objeto e não aos OIDs.

Versão 1 Formato SMI	Formato da versão 2 S I
SNMPv2-SMI-V1SMI.my	SNMPv2-SMI.my
SNMPv2-TC-v1SMI.my	SNMPv2-TC.my
	SNMPv2-CONF.my
SNMPv2-MIB-V1SMI.my	SNMPv2-MIB.my
IANAifType-MIB-V1SMI.my	IANAifType-MIB.my
IF-MIB-V1SMI.my	IF-MIB.my
CISCO-SMI-V1SMI.my	CISCO-SMI.my
CISCO-TC-V1SMI.my	CISCO-TC.my
CISCO-BULK-FILE-MIB-V1SMI.my	CISCO-BULK-FILE-MIB.my
CISCO-FTP-CLIENT-MIB-V1SMI.my	CISCO-FTP-CLIENT-MIB.my

Usando o CISCO-BULK-FILE-MIB

Criando uma operação BULK-FILE

Neste exemplo, nós capturamos o `ifTable` de um roteador, e armazenamo-lo em um arquivo maioria. Contudo, você pode usar todo o objeto MIB ou tabela.

Use a versão rede-SNMP do `snmpset`. O endereço IP do roteador é 14.32.8.2. Sua série de comunidade de leitura/gravação é **privada**. A string de comunidade de somente leitura é pública.

Toda vez que você criar uma nova operação de arquivo de grande escala, escolha dois números aleatórios para a instância da fileira. Eles podem ser qualquer número entre 1 e 4294967295, inclusive. Para fins deste exemplo, use 333 e 444.

Instruções passo a passo

Para criar uma operação BULK-FILE, termine estas etapas:

1. Estabelecer o arquivo a ser criado.

```
$ snmpset -c private 14.32.8.2 cbfDefineFileEntryStatus.333 i 5
$ snmpset -c private 14.32.8.2 cbfDefineFileName.333 s ifTable.txt
$ snmpset -c private 14.32.8.2 cbfDefineFileFormat.333 i bulkASCII
```

2. Especifique o objeto MIB a ser capturado. Este objeto exige dois deslocamentos predeterminados para a operação correta. Os 333 são os 333 da tabela da criação do arquivo acima. 444 é um novo número aleatório usado para o índice principal em `cbfDefineObjectTable`. Este exemplo demonstra o uso de um nome de objeto para `cbfDefineObjectID` (`ifTable`). Você também poderia usar um OID totalmente qualificado aqui.

```
$ snmpset -c private 14.32.8.2 cbfDefineObjectID.333.444 o ifTable
```

3. Ative as filas recém-criadas. É necessário ter ambos os índices para a linha `cbfDefineObjectTable`.

```
$ snmpset -c private 14.32.8.2 cbfDefineObjectEntryStatus.333.444 i 1
$ snmpset -c private 14.32.8.2 cbfDefineFileEntryStatus.333 i 1
```

4. Crie o arquivo.

```
$ snmpset -c private 14.32.8.2 cbfDefineFileNow.333 i 3
```

O arquivo de grande escala é criado.

5. Verifique se o arquivo foi criado com êxito usando `snmpget` no objeto `cbfStatusFileState`. Este objeto requer dois índices. O primeiro índice é o número aleatório escolhido para a tabela File (333 neste exemplo). O segundo índice depende de quantos arquivos você criou em seu roteador. Desde que este é seu primeiro arquivo, o deslocamento predeterminado é 1. , Use conseqüentemente o comando:

```
$ snmpget -c public 14.32.8.2 cbfStatusFileState.333.1
```

O valor `running(1)` significa que o arquivo está em processo de ser criado. Um valor de `ready(2)` significa que o arquivo esteve criado com sucesso, e está esperando para ser lido.

Entretanto, esse arquivo não está diretamente acessível do roteador. Use o CISCO-FTP-CLIENT-MIB para ler esse arquivo.

Transferindo o arquivo utilizando o CISCO-FTP-CLIENT-MIB

Para cada operação de Cliente FTP, você deve selecionar um número aleatório para a instância de linha. Pode ser utilizado um dos mesmos números aleatórios utilizados acima. Este exemplo usa 555.

Instruções passo a passo

Para transferir o arquivo usando um CISCO-FTP-CLIENT-MIB, termine estas etapas:

1. Crie uma instância de linha do cliente FTP.

```
$ snmpset -c private 14.32.8.2 cfcRequestEntryStatus.555 i 5
```

2. Preencha os parâmetros necessários. O LocalFile deve ter o mesmo nome que o arquivo criado acima! Utilize o comando putASCII para transferir arquivos bulkASCII. Se você definir o cbfDefineFileFormat para o bulkBinary acima, terá que definir o cfcRequestOperation para putBinary.

```
$ snmpset -c private 14.32.8.2 cfcRequestOperation.555 i putASCII
$ snmpset -c private 14.32.8.2 cfcRequestLocalFile.555 s ifTable.txt
$ snmpset -c private 14.32.8.2 cfcRequestRemoteFile.555 s /home/Marcus/ifTable.txt
$ snmpset -c private 14.32.8.2 cfcRequestServer.555 s 172.18.123.33
$ snmpset -c private 14.32.8.2 cfcRequestUser.555 s Marcus
$ snmpset -c private 14.32.8.2 cfcRequestPassword.555 s marcus123
```

3. Comece a transferência definindo a fila a ser ativada.

```
$ snmpset -c private 14.32.8.2 cfcRequestEntryStatus.555 i 1
```

Transferência por FTP começa. Quando completo, o arquivo salvar a **/home/Marcus/ifTable.txt**.

4. Para obter o estado de transferência por FTP, use o **snmpget** outra vez no objeto do **cfcRequestResult**. Esse objeto utiliza o mesmo índice usado com os outros objetos FTP.

```
$ snmpget -c public 14.32.8.2 cfcRequestResult.555
```

Um valor de pending(1) significa que o arquivo ainda está em transferência. Um valor de sucesso (2) significa que o arquivo foi transferido com êxito. Todo o outro valor é um [erro](#).

5. Quando a transferência do arquivo for concluída, teste o snmpget do objeto **cbfStatusFileState** novamente. Agora ele tem um valor diferente.

```
$ snmpget -c public 14.32.8.2 cbfStatusFileState.333.1
enterprises.cisco.ciscoMgmt.ciscoBulkFileMIB.ciscoBulkFileMIBObjects.cbfStatus.
cbfStatusFileTable.cbfStatusFileEntry.cbfStatusFileState.333.1 = emptied(3)
```

O valor de emptied(3) significa que o arquivo foi lido com sucesso. Não é possível transferir o arquivo novamente.

6. Agora é seguro excluir esse arquivo destruindo sua linha de status. Este objeto toma os mesmos deslocamentos predeterminados que o **cbfStatusFileState** acima.

```
$ snmpset -c private 14.32.8.2 cbfStatusFileEntryStatus.333.1 i 6
```

7. Depois que o arquivo é excluído, exclua as filas Object e File correspondentes.

```
$ snmpset -c private 14.32.8.2 cbfDefineObjectEntryStatus.333.444 i 6
$ snmpset -c private 14.32.8.2 cbfDefineFileEntryStatus.333 i 6
```

Desse modo, você pode usar o CISCO-FTP-CLIENT-MIB para transferir fora todo o arquivo do roteador que usa o FTP.

Verificando o resultado

Esta seção guia-o com da leitura de alguma da sintaxe para este arquivo.

1. A primeira linha é a linha de prefixo. Para nosso exemplo iftable, é:

```
$ snmpset -c private 14.32.8.2 cbfDefineObjectEntryStatus.333.444 i 6
$ snmpset -c private 14.32.8.2 cbfDefineFileEntryStatus.333 i 6
```

Isso corresponde ao OID do objeto ifEntry. O ifTable é composto de uns ou vários ifEntries.

2. A próxima linha lista o número de objetos da tabela. A linha consiste de uma tabela de palavras-chaves seguida pelo número de objetos na tabela e pelo índice de cada objeto. Por exemplo:

```
$ snmpset -c private 14.32.8.2 cbfDefineObjectEntryStatus.333.444 i 6
$ snmpset -c private 14.32.8.2 cbfDefineFileEntryStatus.333 i 6
```

Esta linha indica que a tabela contém 22 objetos, e cada objeto tem um deslocamento predeterminado de incremento. Estes objetos são do exemplo iftable:

```
$ snmpset -c private 14.32.8.2 cbfDefineObjectEntryStatus.333.444 i 6
$ snmpset -c private 14.32.8.2 cbfDefineFileEntryStatus.333 i 6
```

3. Depois dessa linha, há várias entradas de linha. No exemplo de ifTable, cada linha corresponde a uma interface. As fileiras começam com a fileira da palavra-chave, seguida por seu identificador do deslocamento predeterminado, e seguida pelos objetos enumerados pela entrada de tabela precedente. Por exemplo:

```
$ snmpset -c private 14.32.8.2 cbfDefineObjectEntryStatus.333.444 i 6
$ snmpset -c private 14.32.8.2 cbfDefineFileEntryStatus.333 i 6
```

4. A quarta entrada é o ifDescr para a relação 1. No entanto, esse é o ifDescr no ASCII hexadecimal criptografado. Para traduzir esta linha em mais formato legível, use este comando perl:

```
$ perl -e 'print pack("H*", "546F6B656E52696E67302F30")'
TokenRing0/0
```

Esta entrada corresponde para conectar TokenRing0/0. Todos os objetos que são normalmente cordas são indicados como o ascii codificado hexadecimal nos arquivos maiorias. É possível utilizar este comando Perl para converter qualquer tipo de string ASCII hexadecimal em texto legível. Se você não tem o Perl, use esta [tabela do caractere ASCII](#) para traduzir a corda.

5. Algumas entradas mostram ~ caracteres para valores. Isto significa que o valor para esse objeto é NULO. Ou seja, o objeto não está instanciado no dispositivo. Por exemplo:

```
$ perl -e 'print pack("H*", "546F6B656E52696E67302F30")'
TokenRing0/0
```

Isto corresponde à relação da camada ATM1/0/0-atm. Observe que o ifMtu é NULO para esta relação. Desde que esta é uma interface virtual, faz o sentido que não tem um MTU. Se você prefere, você pode substituir estes NULL com o 0 adicionando este comando à configuração de dispositivo:

```
Router(config)#no snmp-server sparse-table
```

Troubleshooting do Resultado

Ao eleger o objeto `cbfStatusFileState`, se você receber outro valor que não `running(1)` [em execução(1)], `ready(2)` [pronto(2)] ou `emptied(3)` [vazio(3)], isto significará que sua operações encontrou um erro. Estas são causas para os erros:

```
Router(config)#no snmp-server sparse-table
```

Se o número de objetos no arquivo é menos do que você espera, os `cbfDefineMaxObjects` do CISCO-BULK-FILE-MIB podem ser ajustados demasiado baixos. Para determinar o valor atual do objeto, use o `snmpget`.

```
$ snmpget -c public 14.32.8.2 cbfDefineMaxObjects.0
```

Um valor de 0 significa que nenhum limite está configurado. O valor pode ser ajustado a qualquer inteiro entre 0 e 4294967295, inclusivo. Para ajustar os objetos do máximo pelo arquivo ao 10, use o comando `snmpset`. O índice para esse objeto é sempre 0.

```
$ snmpset -c private 14.32.8.2 cbfDefineMaxObjects.0 u 10
```

Esse objeto talvez não seja configurável em todas as plataformas. Se o `snmpset` falha com este erro, o objeto não é configurável em sua plataforma:

```
$ snmpset -c private 14.32.8.2 cbfDefineMaxObjects.0 u 10
```

Ao executar `poll` do objeto `cfcRequestResult`, se você receber um valor diferente de `pendente(1)` ou `sucesso(2)`, a operação de FTP encontrou um erro. Estas são causas para os erros:

```
$ snmpset -c private 14.32.8.2 cbfDefineMaxObjects.0 u 10
```

Caveats

- Atualmente não há nenhuma maneira apoiada de alcançar diretamente os arquivos maiorias. Você deve percorrer o CISCO-FTP-CLIENT-MIB para ler os arquivos.
- O objeto do `cbfDefineFileStorage` define três tipos: `efêmero`, `temporário`, e `permanent`. Atualmente, o único tipo suportado no IOS é `efêmero`. Há arquivos `efêmeros` em pequenas quantidades até a leitura.
- Uma vez que os arquivos são lidos, não podem ser relidos. Antes, eles devem ser recriados.
- O objeto do `cbfDefineFileFormat` define três tipos: `standardBER`, `bulkBinary`, e `bulkASCII`. Os únicos formatos suportados são `bulkBinary` e `bulkASCII`. O formato padrão é `bulkBinary`.
- O servidor FTP Chameleon para Windows é conhecido por não funcionar com o CISCO-FTP-CLIENT-MIB, já que não retorna códigos de resultados corretos.

Informações Relacionadas

- [Como configurar séries de comunidades do SNMP em roteadores, Switches XL baseados no Cisco IOS Software, RSMs, MSFCs e Switches Catalyst](#)
- [Suporte Técnico - Cisco Systems](#)