



Provisioning with the VPN Solutions Center Template Manager

The Template Manager

The Template Manager in the VPN Solutions Center software is a provisioning system that provides fast, flexible, and extensible Cisco IOS command generation capability. The Template Manager defines standard templates to generate Cisco IOS configurations for common provisioning tasks, such as common IPv4, QoS, and VPN provisioning.

- A *template file* is a file created by the Template Manager that stores a VPN Solutions Center template definition.
- A *template data file* is a text file that stores variable values to generate the template configuration file. A valid data file contains name-value pairs for all the variables defined in a template. Each template file can be associated with multiple data files; however, note that each data file can only be associated with a single template. You can select which data file to use to generate a template. The filename suffix for data files is *.dat*.
- A *template configuration file* is an IOS configuration file that stores the Cisco IOS commands created by the Template Manager. A template configuration file can be either a partial or complete configuration file. When you generate a template configuration file using a particular data file, the template configuration filename is the same as the data file's name.

The template data files are tightly linked with its corresponding template. You can use a data file and its associated template to create a template configuration file. The template configuration file is merged with (either appended to or prepended to) the VPNSC configlet. VPN Solutions Center downloads the combined configlet to the edge device router.

You can apply the same template to multiple edge devices, assigning the appropriate template data file for each device. Each template data file includes the specific data for a particular device (for example, the management IP address or host name of each device).

The template files and data files are in XML format. The template file, its data files, and all template configuration file files are mapped to a single directory.

- VPN Solutions Center creates the initial VPNSC configlet. Through the Template Manager, you can create a template configuration file. You can then associate a template configuration file with a service request, which effectively merges the VPNSC configlet and the template configuration file. For details on this process, see the “Integrating a Template with a Service Request” section on page 5-11. You can then download this merged VPNSC configlet to the target router (or routers).

**Tips**

Before you can integrate templates with VPN Solutions Center service requests, you must edit the *cs.m.properties* file and change the following property that is by default set to false to **true**:

1. Find the following statement in the *cs.m.properties* file:

```
netsys.vpn.serviceRequest.showTemplates=false
```

2. Change the statement to true:

```
netsys.vpn.serviceRequest.showTemplates=true
```

3. Restart VPN Solutions Center.

- You can also create a template configuration file and download it directly to a router as described in the “Provisioning a Template Configuration File Directly to a Router” section on page 8-22.

Uses for the Templating Function

Service providers can use the Template Manager to enhance VPN Solutions Center functionality. Because the Telnet Gateway Server (TGS) supports console access to VPN Solutions Center targets, you can use the Template Manager to provide initial configuration for any service provider core device or edge device.

The Template Manager can be used as a stand-alone tool to generate complete configuration files that you can download to any VPN Solutions Center target.

Some of the additional uses for templating are as follows:

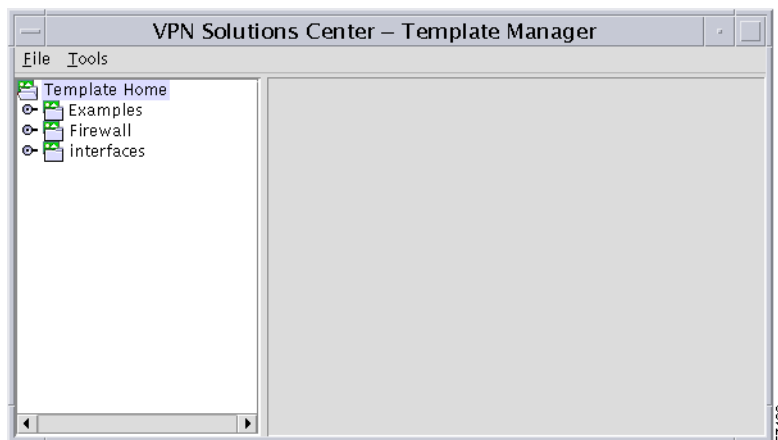
- IOS firewall provisioning
- Add a set of commands that VPN Solutions Center does not include to a service request; for example, provisioning ATM Class of Service.
- Use the templating feature to apply Class of Service using IP connectivity.
- Download a VPN Solutions Center service request and an Cisco IOS configuration file in one download operation through the console. This edge device staging method would create a template and apply the service request in one step.

Creating a Template for VPN Provisioning

To create a VPN Solutions Center template for VPN provisioning, follow these steps:

- Step 1** Start VPN Solutions Center and bring up the VPN Console.
- Step 2** From the VPN Console, choose **Tools > Template Console**.
The VPN Solutions Center Template Manager appears (see Figure 8-1).

Figure 8-1 The Template Manager Window



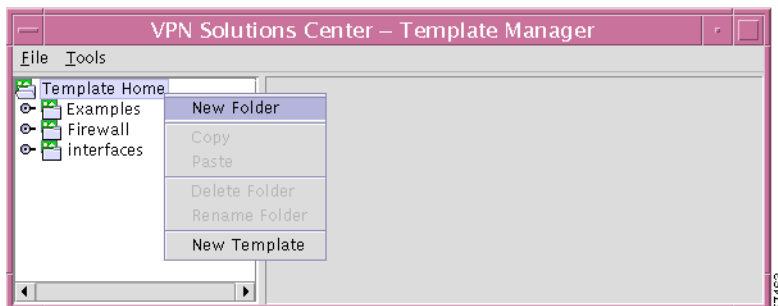
Template Home is the root folder for templates.

Template folders logically organize templates into a hierarchy that facilitates navigation. As in a file system, the Template folders are like directories and the templates are equivalent to files. Many data files can be associated with a template, and any of the data files can be used to generate a template.

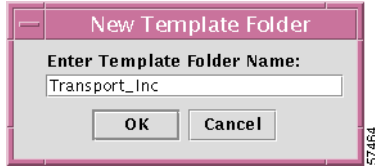
As you can see in Figure 8-1, the Template Manager provides several template examples.

- Step 3** Select the Template Home folder, then **right-click**.
The menu shown in Figure 8-2 appears.

Figure 8-2 Template Folders Menu



- Step 4** From the Template Folders menu, choose **New Folder**.
The New Template Folder dialog box appears (see Figure 8-3).

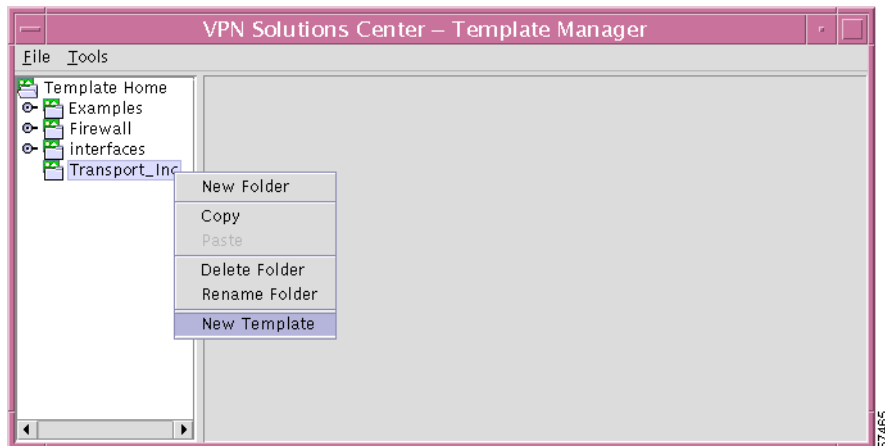
Figure 8-3 New Template Folder Dialog Box

Step 5 Enter the name of the new template folder, then click **OK**.

The new template folder is added to the folders in the template tree.

Step 6 Select the new folder, then right-click.

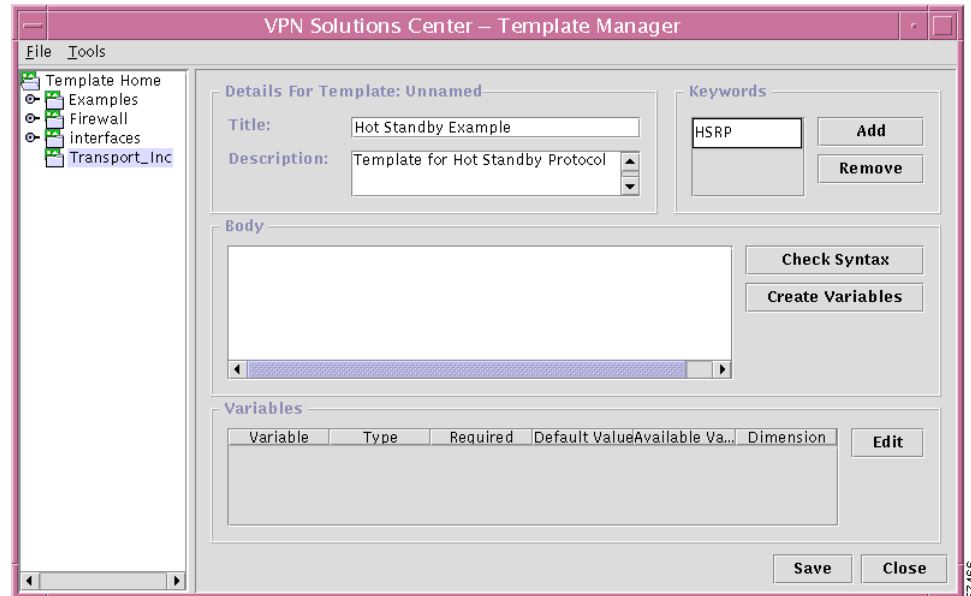
The Folders menu appears, but now additional commands are enabled, including **New Template** (see Figure 8-4).

Figure 8-4 Opening a New Template

Step 7 From the menu, choose **New Template**.

The Template Editor appears in the data pane (see Figure 8-5).

Figure 8-5 The Template Editor



Step 8 Complete the *Title* and *Description* fields as described here.

- a. In the *Title* field, enter a title for the new template.

The *Title* field is optional, but recommended. The title you enter here is not the name of the template; it is a high-level description of the template.

- b. In the *Description* field, enter a description of the template.

The *Description* field is optional, but recommended.

Step 9 Enter one or more keywords in the *Keywords* area.

The keywords are to assist you in finding templates after they have been defined.

- a. When you want to add a keyword, click **Add**.

A new keyword field appears in the *Keywords* area.

- b. Click into the *Keywords* area and enter the keyword.

Entering Configuration Commands in the Template Body

The Body area of the Template Manager is the place where you enter the configuration commands that you want to add to a configuration file. This area of the Template Manager allows you to enter the Cisco IOS commands, check the VPNSC template syntax of the command you have entered, and create the variables called by the IOS commands.

For a discussion of the template grammar and conventions in the VPN Solutions Center template language, see Appendix A, “Template Language and Syntax Reference.”

The result is a *template configuration file* that is added to the *VPNSC configlet* generated through standard VPN Solutions Center provisioning. VPN Solutions Center software downloads the combined template configuration file and VPNSC configlet to the target routers.

The use of interactive IOS commands is not supported in VPN Solutions Center templates.

- Step 1** In the Body area of the Template Manager, enter the Cisco IOS commands and the appropriate variables to create the template configuration file you need.



Tips For efficient data entry, you can copy existing commands from a terminal window and paste them (using **Ctrl-V**) into the Body area.

- Step 2** When you have completed entering the IOS commands to your satisfaction, click **Create Variables**. The Template Manager automatically checks the VPNSC template syntax—not the IOS syntax—and places the variables that were entered in the Body area into the Variables area (at the bottom of the dialog box)—see Figure 8-6.

If there is one or more errors in the template syntax, a syntax error message appears. The syntax error message states the type of error and the line on which the error occurs. Dismiss the error message and correct the syntax error.

Figure 8-6 Completed Template and Variables Displayed

The screenshot shows the 'Details For Template: Unnamed' dialog box. It has three main sections: 'Details', 'Body', and 'Variables'.
 - **Details:** Title: 'Hot Standby Example', Description: 'Template for Hot Standby Protocol'.
 - **Keywords:** A text box containing 'HSRP' with 'Add' and 'Remove' buttons.
 - **Body:** A text area containing the following Cisco IOS commands:

```
! HSRP per Interface
!
interface $ifname
standby $group preempt $priority
standby $group ip $virtual_ip
standby $group track GigabyteEthernet2/0 $track_priority
```

 To the right of the text area are 'Check Syntax' and 'Create Variables' buttons.
 - **Variables:** A table with columns: Variable, Type, Required, Default Val..., Available Values, Dimension, and Edit.

| Variable | Type | Required | Default Val... | Available Values | Dimension | Edit |
|-----------------|--------|-------------------------------------|----------------|------------------|-----------|------|
| priority | String | <input checked="" type="checkbox"/> | Undefined | | 0 | |
| track_priori... | String | <input checked="" type="checkbox"/> | Undefined | | 0 | |
| virtual_ip | String | <input checked="" type="checkbox"/> | Undefined | | 0 | |

 At the bottom right are 'Save' and 'Close' buttons. A small number '57467' is visible in the bottom right corner of the dialog box.

- Step 3** Save the changes you have made by choosing **File > Save**. The Save Template dialog box appears (see Figure 8-7).

Figure 8-7 Saving the Template Changes

The screenshot shows a 'Save' dialog box with the title 'Save'. It contains a label 'Enter Template File Name:' and a text box with the value 'Hot_Standby'. There are 'OK' and 'Cancel' buttons at the bottom. A small number '57468' is visible in the bottom right corner of the dialog box.

- Step 4** Enter the template filename, then click **OK**.

When you save a template, the name of the template is added to the template tree under the appropriate template folder.

Assigning Attributes to Template Variables

Variables are strings that start with a \$ sign. Variables are placeholders that are replaced with actual values from the template data files when the template is generated.

All variables entered into VPN Solutions Center templates are local variables. A variable declared in one template can only be used by that template.

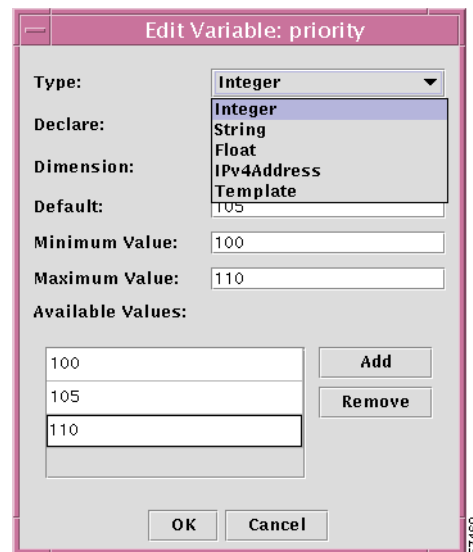
To set attributes for the variables, do the following:

Step 1 From the Variables area in the Template Manager (see Figure 8-6 on page 8-6), select the variable you want to edit.

Step 2 Click **Edit**.

The Edit Variable dialog box appears (see Figure 8-8).

Figure 8-8 Edit Variable Dialog Box



Setting the Variable Type

Step 3 From the Type drop-down menu, choose the variable type to be assigned to the selected variable. As shown in Figure 8-8, there are five variable types you can choose from:

- Integer
- String
- Float
- IPv4 Address
- Template

When you select a variable type, the fields displayed in the Edit Variable dialog box change to accommodate the parameters you can set for the selected type. Some of the parameters are common to all the variable types, and those are described in the following steps.

For each variable type, there are a set of predefined attributes associated with it. For details on each of the variable types, see the “About the Variable Types and Their Attributes” section on page 8-9.

Declaring the Variable as Required or Optional

- Step 4** From the Declare drop-down menu, specify whether the variable is *Required* or *Optional*.
- When you select **Required**, you must specify a value for this variable in the associated data file(s). This is the default setting.
 - When you select **Optional**, you can omit the value for this variable in the associated data file(s).

Specifying the Dimension Attribute

- Step 5** From the Dimension drop-down menu, specify the Dimension attribute.
- The Dimension attribute is an optional attribute that creates an array (or list) of variables. The default value is 0, which indicates a single or enum variable.
- If you set the Dimension attribute to **1** or **2**, the variable becomes a 1- or 2-dimension array. In this case, the constraint attributes are applied to all the elements in the array.

Specifying the Default Attribute

- Step 6** In the *Default* field, specify the default value for the selected variable (if there is a default value). This is an optional attribute. If you also set the available values for this variable, the default value specified here must be one of the available values.

Specifying the Available Values

When you specify a set of available values for a variable, the values entered become the only values that are allowed to be assigned to the variable. If you also set a default value for this variable, the default value must be one of the available values.

- Step 7** To specify the available values for the selected variable, follow these steps:
- a. From the list of variables displayed in the Template Manager, select the variable you want to edit, then click **Edit**.
The Edit Variable: *variable_name* dialog box appears.
 - b. Click **Add**.
A new blank field is displayed in the Available Values area.
 - c. Enter the first available value.
 - d. To specify additional available values for the selected variable, click **Add**, then enter the additional available values until you have specified all the necessary values.
 - e. When finished entering the available values, click **OK**.

The values you entered in the Edit Variable dialog box are displayed for the selected variable in the Template Manager Variables area.

- Step 8** Save your changes by clicking **Save**, then click **Yes** to accept the save operation.
-

About the Variable Types and Their Attributes

A variable is a symbol or name that stands for a value. VPN Solutions Center converts most variables to strings when the template is created, but you can set attributes for each variable type. For example, for an integer variable, you can set the minimum and maximum values allowed in the data.

As shown in Figure 8-8, there are five variable types you can choose from:

- Integer
- String
- Float
- IPv4 Address
- Template

Integer Variable Type

An integer is a whole number. The attributes included in the Integer variable type are as follows:

Table 8-1 Integer Variable Attributes

| Attribute | Comments | Attribute | Comments |
|----------------------|--|-------------------------|---|
| Name | A required attribute. The name must be unique. | Minimum value | Optional |
| Maximum value | Optional | Available values | Optional |
| Default | The default value if the operator does not provide data. | Dimension | The Dimension attribute of 1 or 2 creates an array (or list) of variables. Specify whether the integer variable is an array. The options are: 0 , 1 , or 2 . |
| Declaration | Specify whether it is a required or optional variable. | | |

String Variable Type

A string is a combination of characters considered as a group. The attributes included in the String variable type are as follows:

Table 8-2 String Variable Attributes

| Attribute | Comments | Attribute | Comments |
|-----------------------|--|-------------------------|--|
| Name | A required attribute. The name must be unique. | Minimum length | Optional. If you specify a minimum length, the string cannot be less in length than the value specified here. |
| Maximum length | Optional. If you specify a maximum length, the string cannot be longer in length than the value specified here | Available values | Optional. |
| Default | The default value if the operator does not provide data. | Dimension | The Dimension attribute of 1 or 2 creates an array (or list) of variables. Specify whether the string variable is an array. The options are: 0 , 1 , or 2 . |
| Declaration | Specify whether it is a required or optional variable. | Pattern | The regular expression pattern of the string. For example, a pattern of vpnc[0-9]+ defines a string that starts with “vpnc” followed by a number from 0 to 9. |

Float Variable Type

A floating point number is a number that has no fixed number of digits before or after the decimal point. The attributes included in the Float variable type are as follows:

Table 8-3 Float Variable Attributes

| Attribute | Comments | Attribute | Comments |
|----------------------|--|-------------------------|---|
| Name | A required attribute. The name must be unique. | Minimum value | If you specify a minimum value, the values cannot be less than the value specified here. |
| Maximum value | If you specify a maximum value, the values cannot exceed the value specified here. | Available values | Optional |
| Default | The default value if the operator does not provide data. | Dimension | The Dimension attribute of 1 or 2 creates an array (or list) of variables. Specify whether the float variable is an array. The options are: 0 , 1 , or 2 . |
| Declaration | Specify whether it is a required or optional variable. | | |

IPv4 Address Variable Type

This variable type represents an IPv4 address. The default IPv4 Address variable is in string format, unless the operator provides the IP address data. Figure 8-9 shows an example of an IPv4 variable definition using the Edit Variable dialog box.

Figure 8-9 Example of an IPv4 Address Variable Definition

The screenshot shows a dialog box titled "Edit Variable: virtual_ip". It contains several configuration fields:

- Type:** IPv4Address (dropdown)
- Declare:** Required (dropdown)
- Dimension:** 0 (dropdown)
- Default:** 209.165.200.226 (text input)
- Subnet Mask:** 255.255.255.128 (text input)
- Class:** Class C (dropdown)
- Available Values:** A section with an empty list box, an "Add" button, and a "Remove" button.
- At the bottom are "OK" and "Cancel" buttons.



Note

The *Subnet Mask* attribute is not supported in the 2.0 release.

The attributes included in the IPv4 Address Variable type are as follows:

Table 8-4 IPv4 Address Variable Attributes

| Attribute | Comments | Attribute | Comments |
|--------------------|---|-------------------------|---|
| Name | A required attribute. The name must be unique. | Class | The class of the IP address. The options are: A , B , or C . |
| Subnet mask | The subnet mask of the IP address. This attribute is not supported in this release. | Available values | Optional. If desired, specify the addresses to be accepted as the available values. |
| Default | String: the default value if the operator does not provide data. | Dimension | The Dimension attribute of 1 or 2 creates an array (or list) of variables. Specify whether the address variable is an array. The options are: 0 , 1 , or 2 . |
| Declaration | Specify whether it is a required or optional variable. | | |

Template Variable Type

A template that is embedded within a template is called a *subtemplate*. This variable type specifies a subtemplate variable that can retrieve a subtemplate. VPN Solutions Center supports one level of subtemplates only. Thus, a subtemplate cannot include (or embed) another subtemplate.

Before you create a Template variable, you must create the subtemplate that the variable refers to. When you set the variable type to **Template**, you must browse for and specify the name and location of the indicated subtemplate.

The attributes included in the Template variable type are as follows:

Table 8-5 Template Variable Attributes

| Attribute | Comments | Attribute | Comments |
|-------------|--------------------------------------|--------------------------|---|
| Name | The subtemplate name must be unique. | Template location | Specifies the location of the template in the VPNSC template hierarchy. |

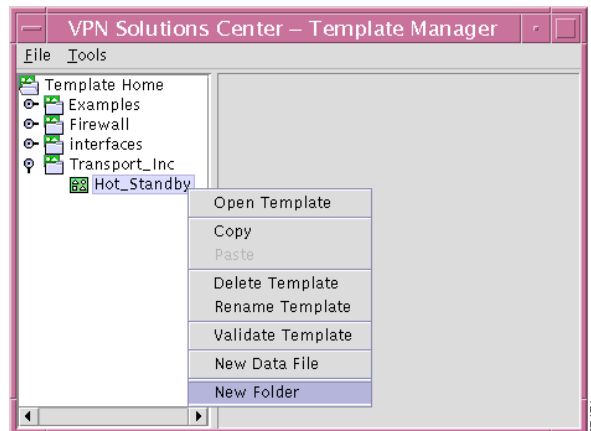
Creating a Template Data File Folder

You can create a folder to organize your template data files. Template data file folders logically organize data files into a hierarchy that facilitates navigation. The template data files are exclusively linked to the associated template.

To create a data file folder, follow these steps:

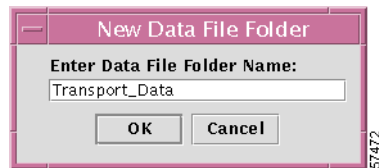
-
- Step 1** From the VPN Console menu bar, choose **Tools > Template Console**.
The Template Manager appears.
 - Step 2** In the Template Home hierarchy pane, expand the hierarchy until you can see the name of the template you want to create the new data file folder for.
 - Step 3** Select the template name, then **right-click**.
 - Step 4** From the menu, choose **New Folder** (see Figure 8-10).

Figure 8-10 New Data File Folder Option



The New Data File Folder dialog box appears (see Figure 8-11).

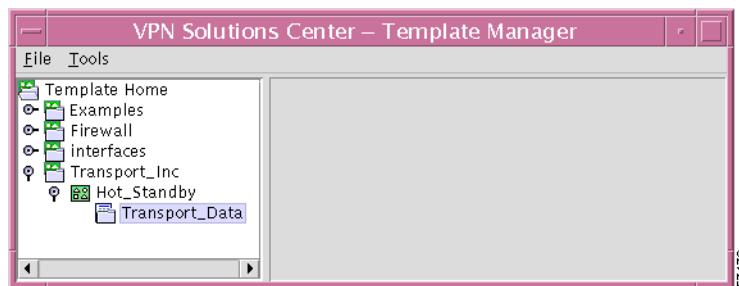
Figure 8-11 Entering the Name of the New Data File Folder



Step 5 Enter the name of data file folder, then click **OK**.

The new data file folder is added to the template hierarchy (see Figure 8-12).

Figure 8-12 New Data File Folder Added



Creating a New Template Data File

A *template data file* is a text file that stores the variable values necessary to generate a template file. A valid template data file contains name-value pairs for all the variables defined in a template.

Each template can be associated with multiple data files; however, note that each data file can only be associated with a single template. You can select which template data file to use to generate a template.

To create a new template data file, follow these steps:

Step 1 From the VPN Console menu bar, choose **Tools > Template Console**.

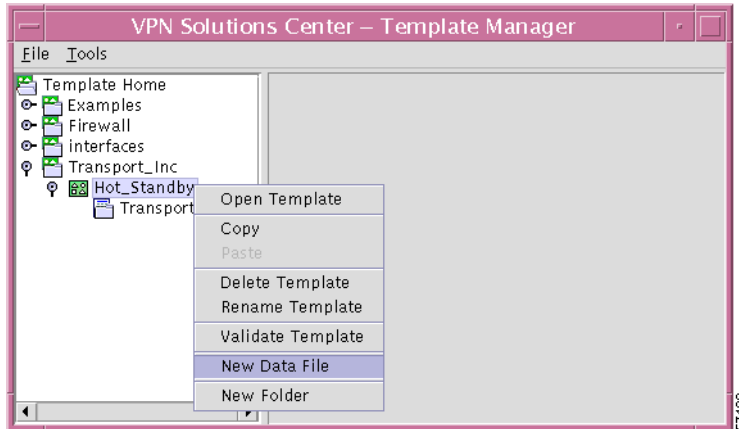
The Template Manager appears.

Step 2 In the Template Home hierarchy pane, expand the hierarchy until you can see the name of the template you want to create the new data file for.

Step 3 Select the template name, then **right-click**.

Step 4 From the menu, choose **New Data File** (see Figure 8-13).

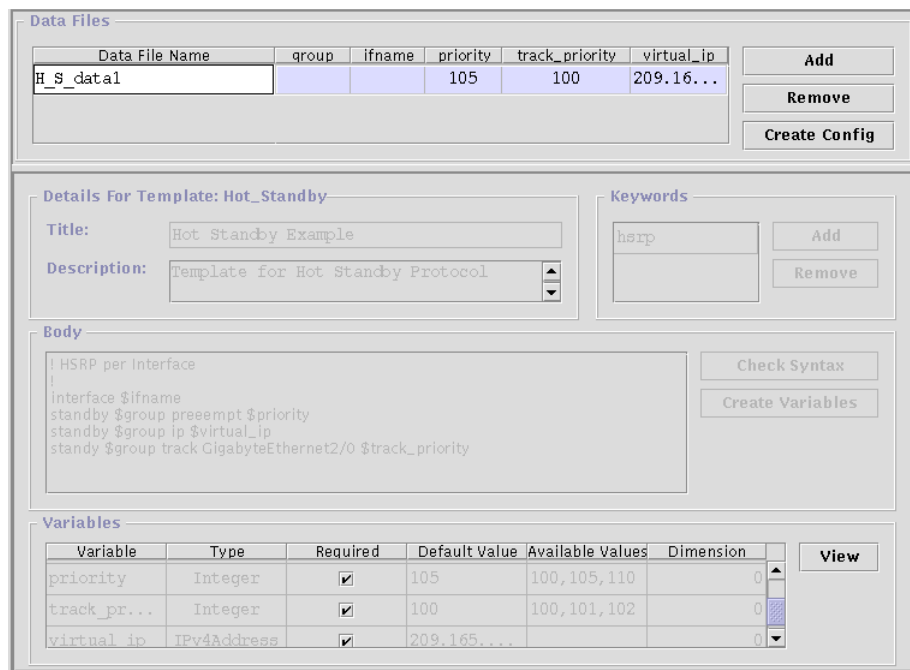
Figure 8-13 New Data File Menu Option



The new Template Data Files dialog box appears in the data pane on the right (see Figure 8-14).

In the Data Files area of the dialog box, the name of each of the variables appears with a cell beneath it for the designated value of that variable.

Figure 8-14 New Template Data Files Dialog Box



- Step 5** If desired, change the name of the data file from the default name (Data0) to a more identifiable name.
- Click the *Data File Name* cell.
 - Change the data file name as desired.

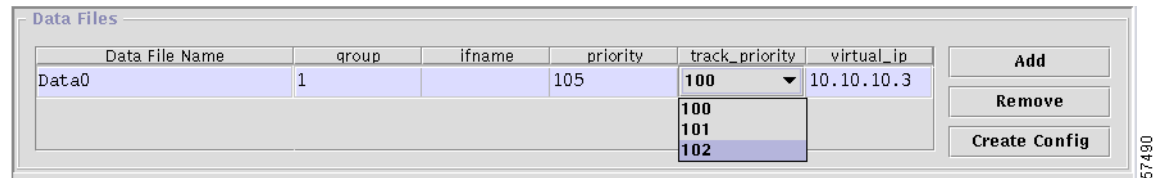
- Step 6** Enter the appropriate data for each of the variables called in the Template body by clicking the cell below each variable name and entering the data.

The Template Manager displays the variables in standard spreadsheet format. The initial default values are displayed for each variable.

Depending on the type of variable, and its settings and attributes, the data values you enter will vary. The Template Manager provides type checking when you save the data; for example, if you enter non-numeric characters in a field defined as an integer, the Template Manager does not accept the data and alerts you to the discrepancy.

As shown in the `track_priority` variable in Figure 8-15, you can see the available values set for a particular variable by clicking the name of the variable.

Figure 8-15 A Variable's Available Values Listed



Entering Values for a One-Dimensional Array

A one-dimensional array creates a list of values that you can apply in your template for that variable. The number of values that can be substituted for any given one-dimensional variable is determined by the number of values you define for it (as shown below in Figure 8-17 on page 8-16).

The following is an example of program statements entered into a template that make use of a one-dimensional array:

```
!
{
access-list $ACL[$i] permit ip $Source_IP_Address[$i] $Source_Mask[$i]
$Dest_IP_Address[$i] $Dest_Mask[$i] precedence 7
}
```

The `$i` specified in the variables declared in the substatements above such as:

```
$Source_IP_Address[i]
```

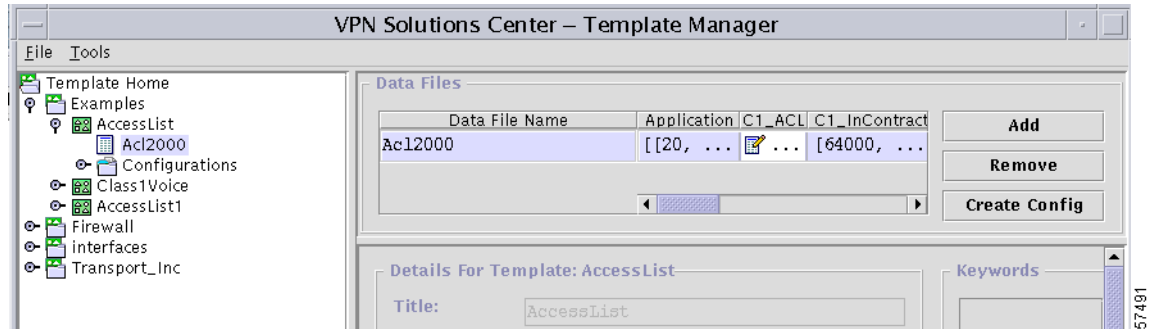
indicates that it is a one-dimensional variable. For example, for each value specified for the **Source_IP_Address** variable, the template substitutes one of the values for **Source_IP_Address**.

To enter the values for a one-dimensional array, follow these steps:

- Step 1** If the Template Data Files dialog box is not already open, expand the Template Manager hierarchy until you can see the name of the data file of interest.
- Step 2** Select the template data file, then **right-click**.
- Step 3** From the menu, choose **Open**.

The Template Manager Data Files dialog box is displayed (see Figure 8-16).

Figure 8-16 Selecting One-Dimensional Variable



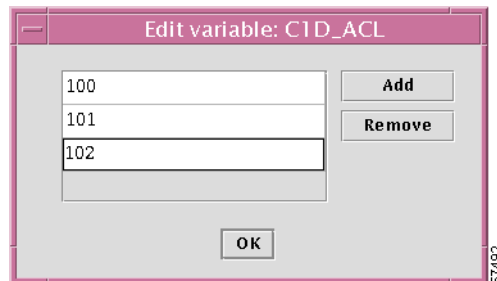
The name of the selected template data file and each of its variables are displayed. You can use the mouse to adjust the width of each variable cell.

Step 4 Scroll to the cell that displays the name of the variable you want to edit.

Step 5 Select the variable cell, then **double-click**.

The Edit Variable: *variable_name* dialog box appears (Figure 8-17).

Figure 8-17 Entering Data for a One-Dimensional Array Variable



Step 6 In the Edit Variable dialog box, click **Add**.

A new blank field appears in the editing area.

Step 7 Enter the first value to be assigned to the variable.

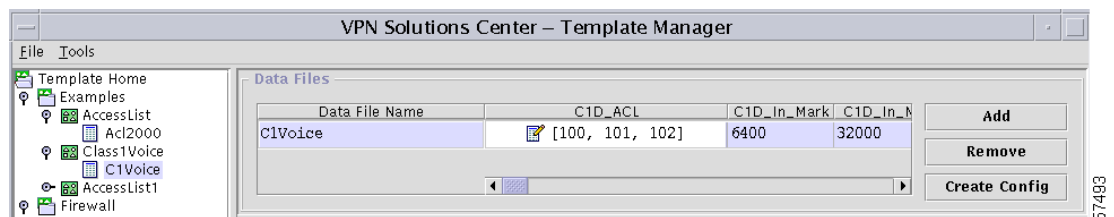
Step 8 Click **Add**, then enter the appropriate values for each additional value to be assigned to the variable.

- To delete a value from the list, select the value, then click **Remove**.

Step 9 When the data is entered to your satisfaction, click **OK**.

The values you entered are displayed in the corresponding *Data Files* variable cell, as shown in Figure 8-18.

Figure 8-18 One-Dimensional Array Added to Data File Variable Display



Step 10 To save your work, choose **File > Save**, then when prompted to save the file, click **Yes**.

Entering Values for a Two-Dimensional Array

A two-dimensional array creates a table of values that you can apply in your template for that variable. The number of values that can be substituted for any given two-dimensional variable is determined by the number of rows and columns that you define in the Edit Variable dialog box.

The indexing convention uses pairs of numbers that refer to positions in the indexing table, as shown in Table 8-6:

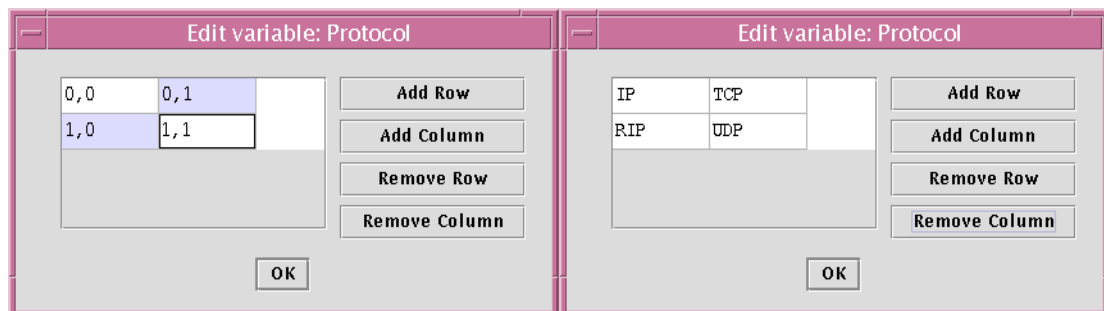
Table 8-6 Indexing Convention for a Two-Dimensional Array

| | | |
|------|------|------|
| 0, 0 | 0, 1 | 0, 2 |
| 1, 0 | 1, 1 | 1, 2 |
| 2, 0 | 2, 1 | 2, 2 |

The rows and columns that you create with the Template Manager Variable Editor creates a table of cells that correspond to the indexing convention for two-dimensional variables. Substituting values occurs by linking the positions indicated by the indexing convention with two-dimensional variable statements.

For example, let us say that you have a Protocol variable that has four possible values: IP, TCP, RIP, and UDP. The examples of the Edit Variable dialog box shown in Figure 8-19 show the index positions and the four protocol values in their corresponding relative positions.

Figure 8-19 Entering a Two-Dimensional Array



- “IP” is in the 0,0 index position
- “TCP” is in the 0,1 index position
- “RIP” is in the 1,0 index position
- “UDP” is in the 1,1 index position

To substitute the appropriate variable values, the variables defined in the template body would be defined as follows:

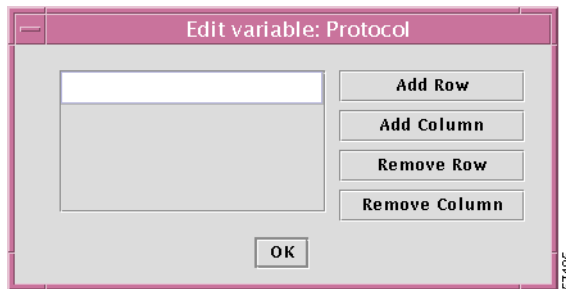
- To substitute the **IP** value: **Protocol [0] [0]**
- To substitute the **TCP** value: **Protocol [0] [1]**
- To substitute the **RIP** value: **Protocol [1] [0]**

- To substitute the **RIP** value: **Protocol [1] [1]**

To enter the values for a two-dimensional array, follow these steps:

- Step 1** If the Template Data Files dialog box is not already open, expand the Template Manager hierarchy until you can see the name of the data file of interest.
- Step 2** Select the template data file, then **right-click**.
- Step 3** From the menu, choose **Open**.
- The Template Manager Data Files dialog box is displayed. The name of the selected template data file and each of its variables are displayed. You can use the mouse to adjust the width of each variable cell.
- Step 4** Scroll to the cell that displays the name of the variable you want to edit.
- Step 5** Select the variable cell, then **double-click**.
- The Edit Variable: *variable_name* dialog box appears (Figure 8-20).

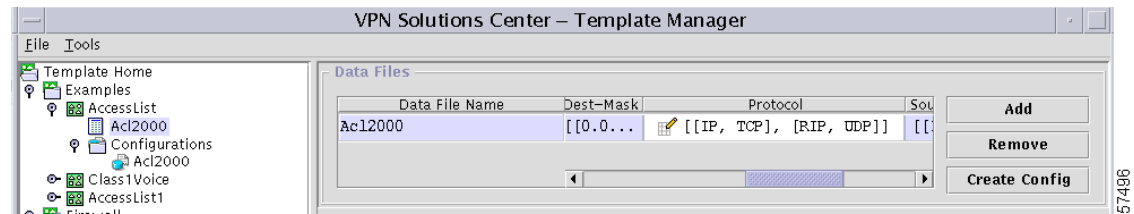
Figure 8-20 Editing Two-Dimensional Array



When the Edit Variable dialog box for a two-dimensional array first appears, there are no fields displayed in the edit area.

- Step 6** To begin entering data, click **Add Row**.
- A row appears, as shown in Figure 8-20.
- Step 7** To add a column to the variable table, click **Add Column**.
- Step 8** Add rows and columns as necessary to build a variable index table that provides the number of cells you need to hold the values required by the variable.
- You can also remove unneeded rows and columns by selecting a cell in the row or column you want to remove and clicking **Remove Row** or **Remove Column**.
- Step 9** Click each cell to enter the variable values as necessary.
- Step 10** When the data is entered to your satisfaction, click **OK**.
- The values you entered are displayed in the corresponding Data Files variable cell, as shown in Figure 8-21.

Figure 8-21 Two-Dimensional Array Added to Data File Variable Display



Step 11 To save your work, choose **File > Save**, then when prompted to save the file, click **Yes**.

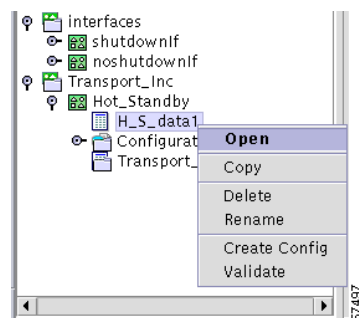
Copying a Template Data File

You can copy a template data file from one location to another within the Template Manager. Keep in mind that template data files are tightly coupled with their associated template. Therefore, you would normally copy a template data file to the same template, then modify the variables accordingly.

To copy a template data file, follow these steps:

- Step 1** Expand the template hierarchy until you can see the name of the data file you want to copy.
- Step 2** Select the data file name, then **right click**.

Figure 8-22 Copying a Template Data File



- Step 3** From the menu, choose **Copy**.
- Step 4** Select the template folder (or the data file folder) that you want to copy the data file to.
- Step 5** From the menu, choose **Paste**.

The template data file is copied to the selected folder.

Deleting a Template Data File

You can delete a template data file from the Template Manager. Keep in mind that template data files are tightly coupled with their associated template. Therefore, deleting a data file can effectively disable a template.

To delete a template data file, follow these steps:

-
- Step 1** Expand the template hierarchy until you can see the name of the data file you want to delete.
 - Step 2** Select the data file name, then **right click**.
 - Step 3** From the menu, choose **Delete**.

You receive the following warning:

Warning: Are you sure that you want to delete <name> file?

- Step 4** To delete the selected data file, click **Yes**.
To cancel the data file deletion operation, click **No**.
-

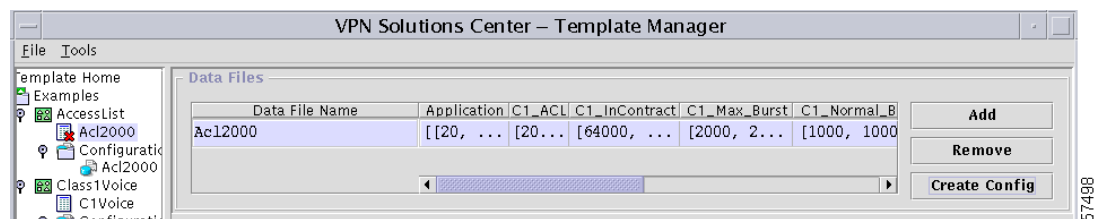
Creating a Template Configuration File

You can create a template configuration file based upon a particular template and a specific data file.

To create a template configuration file, follow these steps:

-
- Step 1** From the VPN Console menu bar, choose **Tools > Template Console**.
The Template Manager appears.
 - Step 2** In the Template Home hierarchy pane, expand the hierarchy until you can see the name of the template and then the name of the associated template data file that you want to create the template configuration file from.
 - Step 3** Select the template data file of interest, then **right-click**.
 - Step 4** From the menu, choose **Open**.
The Template Data Files dialog box appears.
 - Step 5** Select the row where the data file name and variables are displayed (see Figure 8-23).

Figure 8-23 Selecting the Template Data File



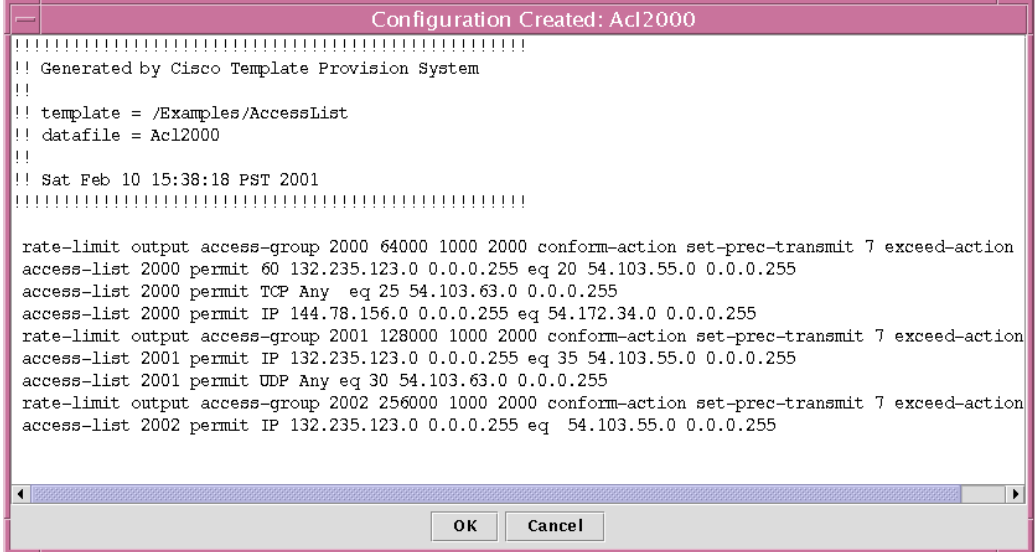
- Step 6** Click **Create Config**.
You receive the following message:

The Data File will be saved before creating a configuration.

Step 7 Click **OK** to proceed.

The Configuration Created window appears, which displays the template configuration file that was created from the selected template and configuration file (see Figure 8-24).

Figure 8-24 Template Configuration File Displayed



```

!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!! Generated by Cisco Template Provision System
!!
!! template = /Examples/AccessList
!! datafile = Acl2000
!!
!! Sat Feb 10 15:38:18 PST 2001
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!

rate-limit output access-group 2000 64000 1000 2000 conform-action set-prec-transmit 7 exceed-action
access-list 2000 permit 60 132.235.123.0 0.0.0.255 eq 20 54.103.55.0 0.0.0.255
access-list 2000 permit TCP Any eq 25 54.103.63.0 0.0.0.255
access-list 2000 permit IP 144.78.156.0 0.0.0.255 eq 54.172.34.0 0.0.0.255
rate-limit output access-group 2001 128000 1000 2000 conform-action set-prec-transmit 7 exceed-action
access-list 2001 permit IP 132.235.123.0 0.0.0.255 eq 35 54.103.55.0 0.0.0.255
access-list 2001 permit UDP Any eq 30 54.103.63.0 0.0.0.255
rate-limit output access-group 2002 256000 1000 2000 conform-action set-prec-transmit 7 exceed-action
access-list 2002 permit IP 132.235.123.0 0.0.0.255 eq 54.103.55.0 0.0.0.255

```

The Template Manager creates a folder called “Configurations” in the appropriate template folder, and places the configuration file in the Configurations folder.

Step 8 To close the configuration file window, click **OK**.

Copying a Template

You can copy a VPN Solution Center template from one folder to another within the template hierarchy. When you copy a template, all the files and folders associated with that template—its data files and configuration files—are copied to the new location.

To copy a VPN Solution Center template, follow these steps:

- Step 1** Expand the template hierarchy until you can see the name of the template you want to copy.
- Step 2** Select the template name, then **right click**.
- Step 3** From the menu, choose **Copy**.
- Step 4** Select the template folder that you want to copy the template to.
- Step 5** From the menu, choose **Paste**.

The template, along with the files and folders associated with it, is copied to the selected folder.

Deleting a Template

You can delete any template from the VPN Solutions Center Template Manager. When you delete a template, all the files and folders associated with that template—its data files and configuration files—are also deleted.

To delete a VPN Solution Center template, follow these steps:

-
- Step 1** Expand the template hierarchy until you can see the name of the template you want to delete.
 - Step 2** Select the template name, then **right click**.
 - Step 3** From the menu, choose **Delete Template**.

You receive the following warning:

WARNING: All files in the selected template directory will be deleted. Do you want to continue?

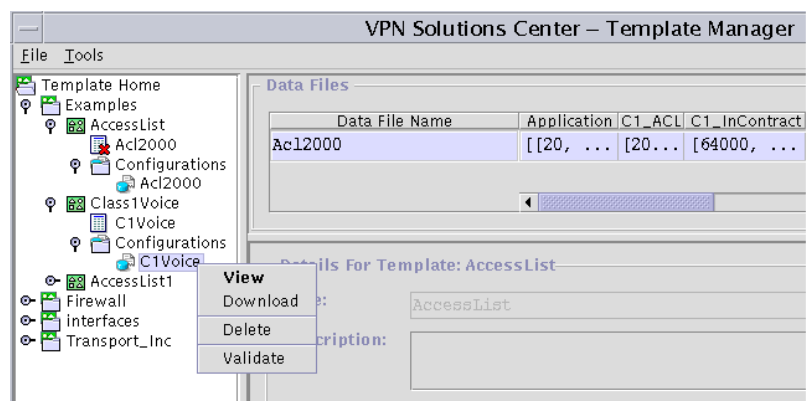
- Step 4** To delete the selected template, click **Yes**.
To cancel the template deletion operation, click **No**.
The template, along with the files and folders associated with it, is deleted from the Template Manager.
-

Provisioning a Template Configuration File Directly to a Router

You can download a template configuration file directly to a router. To do so, follow these steps:

-
- Step 1** From the VPN Console menu bar, choose **Tools > Template Console**.
The Template Manager appears.
 - Step 2** In the Template Home hierarchy pane, expand the hierarchy until you can see the name of the template and then the name of the associated template configuration file that you want to download to a router.
 - Step 3** Select the template configuration file of interest, then **right-click** (see Figure 8-25).

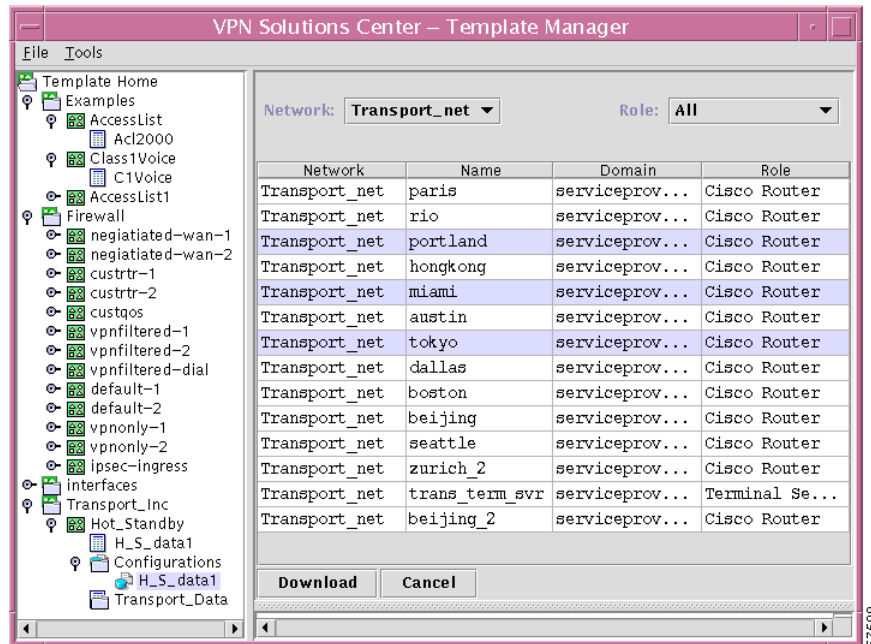
Figure 8-25 Template Configuration File Menu



- Step 4** From the menu, choose **Download**.

The VPN Solutions Center Network window appears (see Figure 8-26).

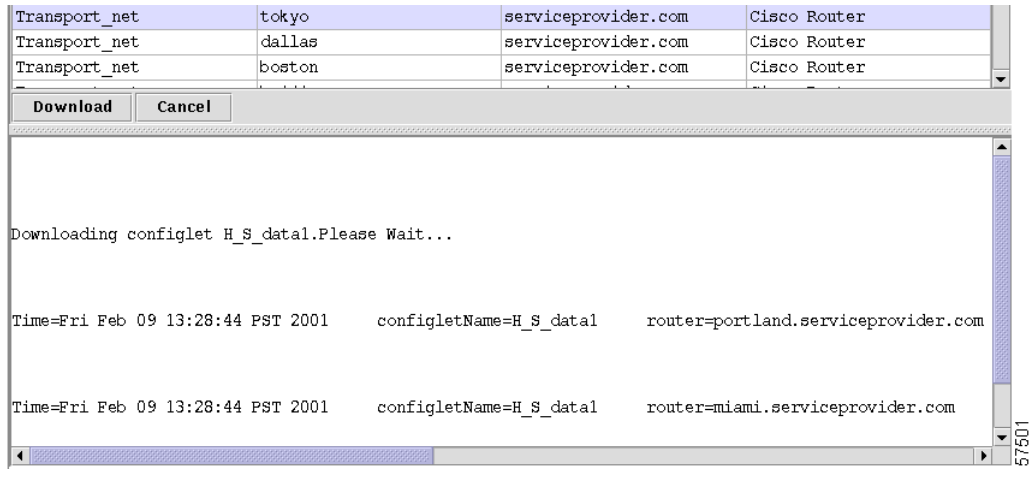
Figure 8-26 *Selecting Router from the VPNSC Network Window*



- Step 5** From the Network window, specify the following:
- From the Network drop-down menu, choose the appropriate network.
 - From the Role drop-down menu, choose **Cisco Router**.
 - From the list of routers, select one or more routers that you want to download the template configuration file to.
- Step 6** When satisfied with your selections, click **Download**.

An informational window appears, which shows the status of the template configuration file download operation (see Figure 8-27).

Figure 8-27 Template Configuration File Download Status Window



Using VPN Solutions Center Repository Variables as Template Data

You can choose from a list of IPsec (or MPLS) variables that are part of the VPN Solutions Center service definition. Their data values are substituted when you associate a template with a service request. For details on this procedure, see the “Integrating a Template with a Service Request” section on page 5-11.



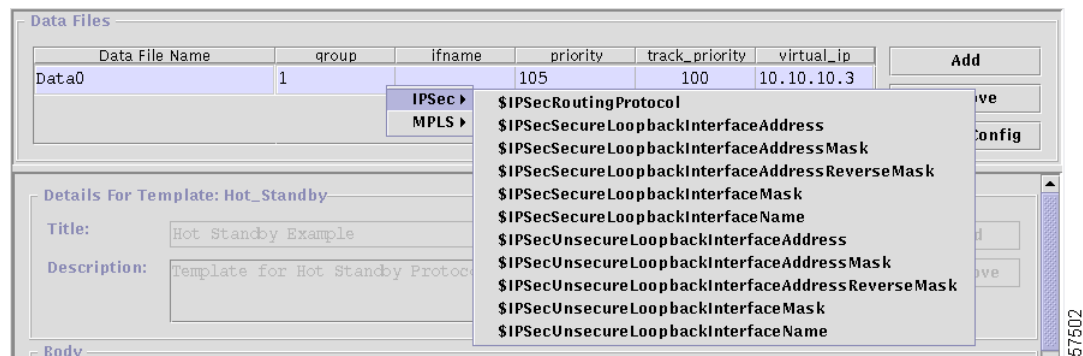
Caution

When you use the Repository variables in a template, you cannot use the Template Manager to download the template configuration file to the router. This is because the data values for that variable are substituted when the VPN Solutions Center provisions a service request. Since this process bypasses the service request, the appropriate data is not substituted for the assigned Repository variable.

To use VPN Solutions Center Repository values as template data, follow these steps:

- Step 1** From the VPN Console menu bar, choose **Tools > Template Console**.
- Step 2** From the Template Manager hierarchy pane, expand the template hierarchy until you can see the name of the data file of interest.
- Step 3** Select the name of the pertinent data file, then **right-click**.
- Step 4** From the menu, choose **Open**.
The Data Files dialog box appears.
- Step 5** Click the cell for the variable of interest, then **right-click**.
A menu appears with two options: **IPsec** and **MPLS**.
- Step 6** From the menu, choose **IPsec**.
The list of IPsec service values is displayed (see Figure 8-28).

Figure 8-28 Menu of IPsec Repository Variables



- Step 7** From the list of variable values, choose the appropriate value.

Summary of IPsec Repository Variables

Table 8-7 provides a summary of the IPsec Repository variables available from the VPN Solutions Center Template Manager.

Table 8-7 IPsec Repository Variables

| Repository Variable | Dimension | Description | Example |
|--|-----------|---|---|
| IPSecLocalProtectedIPAddress | 1 | List of IP address ranges protected by this edge device | 209.165.202.129 209.165.202.130... |
| IPSecLocalProtectedIPAddressMask | 1 | List of IP address ranges and subnet masks protected by this edge device | 209.165.20.129/30 209.165.20.130/30... |
| IPSecLocalProtectedIPAddressReverseMask | 1 | List of reverse subnet masks of IP address ranges protected by this edge device | 209.165.202.129 209.165.202.130... |
| IPSecLocalProtectedIPMask | 1 | List of subnet masks of IP address ranges protected by this edge device | 209.165.20.129/30 209.165.20.130/30... |
| IPSecRoutingProtocol | 0 | IPSec VPN routing protocol | NONE, OSPF |
| IPSecSecureInterfaceAddress | 1 | Edge device's secured interface IP address | 209.165.202.129 |
| IPSecSecureInterfaceAddressMask | 1 | Edge device's secured interface IP address and mask | 209.165.20.130/30 |
| IPSecSecureInterfaceAddressReverseMask | 1 | Edge device's secured interface reverse subnet mask | 0.0.0.3 |
| IPSecSecureInterfaceMask | 1 | Edge device's secured interface subnet mask | 255.255.255.252 |
| IPSecSecureInterfaceName | 1 | Edge device's secured interface name | Serial0/0 |
| IPSecSecureLoopbackInterfaceAddress | 0 | Edge device's secured tunnel endpoint interface IP address | 171.23.45.33 |
| IPSecSecureLoopbackInterfaceAddressMask | 0 | Edge device's secured tunnel endpoint interface IP address and mask | 171.23.45.33/32 |
| IPSecSecureLoopbackInterfaceAddressReverseMask | 0 | Edge device's secured tunnel endpoint interface reverse subnet mask | 0.0.0.0 |
| IPSecSecureLoopbackInterfaceMask | 0 | Edge device's secured tunnel endpoint interface subnet mask | 255.255.255.255 |
| IPSecSecureLoopbackInterfaceName | 0 | Edge device's secured tunnel endpoint interface name | Loopback0 |
| IPSecUnsecureInterfaceAddress | 1 | Edge device's nonsecured interface IP address | 209.165.202.131 |
| IPSecUnsecureInterfaceAddressMask | 1 | Edge device's nonsecured interface IP address and mask | 171.23.45.33/24 |
| IPSecUnsecureInterfaceAddressReverseMask | 1 | Edge device's nonsecured interface reverse subnet mask | 0.0.0.255 |

Table 8-7 IPsec Repository Variables (continued)

| Repository Variable | Dimension | Description | Example |
|---|-----------|---|---|
| IPSecUnsecureInterfaceMask | 1 | Edge device's nonsecured interface subnet mask | 255.255.255.0 |
| IPSecUnsecureInterfaceName | 1 | Edge device's nonsecured interface name | Ethernet1 |
| IPSecUnsecureLoopbackInterface Address | 0 | Edge device's nonsecured loopback interface IP address | 10.1.93.1 |
| IPSecUnsecureLoopbackInterface AddressMask | 0 | Edge device's nonsecured loopback interface IP address and mask | 10.1.93.1/32 |
| IPSecUnsecureLoopbackInterface AddressReverseMask | 0 | Edge device's nonsecured loopback interface reverse subnet mask | 0.0.0.0 |
| IPSecUnsecureLoopbackInterfaceMask | 0 | Edge device's nonsecured loopback interface subnet mask | 255.255.255.255 |
| IPSecUnsecureLoopbackInterfaceName | 0 | Edge device's nonsecured loopback interface name | Loopback1 |
| IPSecRemoteProtectedIPAddress | 2 | List of IP address ranges protected by remote peers | 209.165.202.129 209.165.202.130... |
| IPSecRemoteProtectedIPAddressMask | 2 | List of IP address ranges and subnet masks protected by remote peers | 209.165.20.129/30 209.165.20.130/30... |
| IPSecRemoteProtectedIPAddressReverse Mask | 2 | List of reverse subnet masks of IP address ranges protected by remote peers | 0.0.0.255... |
| IPSecRemoteProtectedIPMask | 2 | List of subnet masks of IP address ranges protected by remote peers | 209.165.20.129/30 209.165.20.130/30... |
| IPSecRemoteHostName | 1 | List of remote peer host names | |

Summary of MPLS Repository Variables

Table 8-8 provides a summary of the MPLS Repository variables available from the VPN Solutions Center Template Manager.

Table 8-8 MPLS Repository Variables

| Repository Variable | Dimension | Description | Example |
|--|-----------|---|---------|
| MPLSAsNumber | 0 | Domain autonomous system number | |
| MPLScableModemHelperAddr | 0 | Cable modem helper address | |
| MPLScableModemHostHelperAddr | 0 | Cable modem host helper address | |
| MPLScableModemMaintenanceInterfaceHelperAddr | 0 | Cable modem maintenance interface helper address | |
| MPLScableModemSecondaryIpAddrList | 1 | Cable modem secondary IP address list | |
| MPLSCeATM | 0 | Customer edge router ATM virtual circuit descriptor | |
| MPLSCeATM-vci | 0 | Customer edge router virtual circuit identifier | |
| MPLSCeATM-vpi | 0 | Customer edge virtual path identifier | |
| MPLSCeDLCI | 0 | Customer edge router data link circuit identifier | |
| MPLSCeInterfaceAddress | 0 | Customer edge router interface address | |
| MPLSCeInterfaceName | 0 | Customer edge router interface name | |
| MPLSCeVlanId | 0 | Customer edge router virtual LAN identifier | |
| MPLSPeCeLinkProtocol | 0 | PE-CE link routing protocol | |
| MPLSPeATM-vcd | 0 | Provider edge router ATM virtual circuit descriptor | |
| MPLSPeATM-vci | 0 | Provider edge router virtual circuit identifier | |
| MPLSPeATM-vpi | 0 | Provider edge router virtual path identifier | |
| MPLSPeDLCI | 0 | Provider edge router ATM link circuit identifier | |
| MPLSPeInterfaceName | 0 | Provider edge router interface name | |
| MPLSPeInterfaceAddress | 0 | Provider edge router interface address | |
| MPLSPeVlanId | 0 | Provider edge router virtual LAN identifier | |
| MPLSVrfRouteDistinguisher | 0 | MP-BGP attribute: VRF route distinguisher | |
| MPLSVrfName | 0 | VPN routing and forwarding table name | |