APPENDIX C

Property Settings

To navigate to the properties, known as Dynamic Component Properties Library (DCPL), navigate to the tab **Administration > Control Center > Hosts**. Then select a check box for a specific host and click the **Config** button.

None of these properties can be set on a per user basis, including logging.

Note More details about this are explained in the "Config" section on page 9-23.

When you click on the folder or subfolder, it expands to more subfolders or eventually to the property itself. Then you receive an explanation, default values, and in some cases range and rules. This table can help you understand all the properties available at a glance. The properties are listed alphabetically. When a / ends an entry, this means it can be expanded further. Also, if you are searching for a property and do not know the name, you can use some key words and do a Find on the pdf version.

Table C-1 DCPL Properties

Property	Default Value	Range/Rules	Explanation
AutoDiscovery Properties:			Controls the operation of Autodiscovery.
/DiscoveryTemplateFolder	/Discovery	string	Template folder under which the templates to be discovered for MPLS VPN Discovery will reside.
/TopologyHandler	Default	string	This property points to the topology handler for the discovery run.
/createVpnAndCustomerFromVRFName	true	The valid values are **true** and **false**.	This property controls whether the VPN and Customer objects can be created from the VRF names. This is valid only in certain scenarios when Service Providers have maintained such a mapping.
/performTemplateDiscovery	false	The valid values are **true** and **false**.	With this flag, the user can control the template discovery. For performance reasons, if the template discovery is not desired this should be set to false.
Cleanup Properties:			Cleans up various system resources such as log files and temporary files.
/Cleanup/TaskLogs/			This component cleans up old TaskLogs.

Table C-1 DCPL Properties (continued)

maxAgeInHours	168	integer	Maximum age of the TaskLogs in hours. TaskLogs older than this age will be deleted during the next cleanup cycle. Set to 0 to disable this feature.
sleepIntervalInHours	24	integer, 1-1000 hours	Time in hours for taskLog cleanup service to sleep between clean up cycles.
/Cleanup/Tasks/			This component cleans up old TaskLogs.
maxAgeInHours	0	integer	Maximum age of the Tasks in hours. Tasks that have not been modified in over maxAge hours and that have no Active schedules will be deleted during the next cleanup cycle. Set to 0 to disable this feature.
sleepIntervalInHours	24	integer, 1-1000 hours	Time in hours for task cleanup service to wait between clean up cycles. Changing this value initiates an immediate cleanup cycle.
/Cleanup/TempFiles/			This component cleans up old temporary files.
maxAgeInHours	168	integer	Maximum age of the temporary files in hours. Temporary files older than this age will be deleted during the next cleanup cycle. Set to 0 to disable this feature.
sleepIntervalInHours	24	integer, 1-1000 hours	Time in hours for tempFile cleanup service to sleep between clean up cycles.
/Cleanup/logLevel	CONFIG	selection	This log Level is used only if there is no log Level defined for a component. The log Level is the level at which logging is done for this component. These levels are identical to the logging levels defined for JDK1.4 logging package. The levels in descending order are: SEVERE (highest value) WARNING INFO CONFIG FINE FINER FINEST (lowest value).
CNS Properties			
/CNS/defaultVersion	1.4	1.3, 1.3.1, 1.3.2, 1.4, 1.5, and 2.0	Default version of CNS to be selected while creating a device. The supported versions are: 1.3, 1.3.1, 1.3.2, 1.4, 1.5, and 2.0.

Table C-1 DCPL Properties (continued)

/CNS/deprecatedReboot	0	The valid values are **0** and **1**.	This is the flag to be used for reloading IOS 12.3 devices using cisco.mgmt.cns.config.reboot CNS event. Value 0 means IOS 12.3 devices may not be rebooted using cisco.mgmt.cns.config.reboot CNS event. So, IOS versions other than 12.3 can be rebooted. Value 1 means only IOS 12.3 devices are rebooted using cisco.mgmt.cns.config.reboot CNS event.
DCS Properties:			Device Configuration Service. This component corresponds to a library that is used by ISC to communicate with network devices using protocols such as telnet, ssh, tftp, and so forth.
/DCS/FTP/			FTP Settings.
ftpPassword		string	Password for FTP server login, used by DCS and GTL.
ftpRootDirectory		string	FTP root directory, used by DCS and GTL.
ftpServer		string	FTP Server host name or IP address, used by DCS and GTL.
ftpSubDirectory		string	FTP sub directory, used by DCS and GTL.
ftpUsername		string	Username for FTP server login, used by DCS and GTL.
/DCS/IOSUsePrimaryWarningExprOnly	false	The valid values are **true** and **false**.	If **true**, DCS uses only the primary warning expression list, specified in DCS/IOSWarningExpressions. If **false**, DCS uses the primary list specified in DCS/IOSWarningExpressions for add and modify operations and uses the list specified in DCS/ IOSWarningExpressionsRemoveCfg during delete (decommissioning) operations.

Table C-1 DCPL Properties (continued)

/DCS/IOSWarningExpressions		string	IOS warning expressions that can be safely ignored; case insensitive; . matches any char except newline, * means zero or more, + means one or more, ? means zero or one. All regular expressions except the last one should have a **$** at the end of the regular expression. %Aborting Save. Compress the config$.*Access Rules Download Complete$ % Access VLAN does not exist.$ Address aliases with.*$ % All RSA Keys will be removed.$ % All router certs issued using these keys will also be removed.*$ % Already found same .* statement in this profile$ % A profile is deemed incomplete until it has match identity statements$.*certificate accepted$ Certificate request sent$.?Changes to the System MTU will not take effect until the next reload.*$ CNS config partial agent is running already$ % Configuration buffer full, can't add command.*$.*Crypto EzVPN does not exist.*$ Enter configuration commands, one per line$ Explicit Path name .*$ % Generating .* bit RSA keys$ Global .* will be Port Address Translated.*$ Global Ethernet MTU is set to.*$ If the interface doesn't support baby giant frames.*$ Increasing .* burst size to$ % Interface .* IP address .* removed due to enabling VRF$ % Interface .* IP address .* removed due to disabling VRF$ % IP addresses from all interfaces in VRF .*have been removed$

Table C-1 DCPL Properties (continued)

/DCS/IOSWarningExpressions (Continued)		string	% IP routing table V.* does not exist. Create first$ % IP routing table g.*does not exist. Create first$ % No CEF interface information$ %No matching route to delete$ %Translation not found$.*Not all config may be removed and may reappear after reactivating$ ^%.?NOTE:$ OSPF: Unrecognized virtual interface .* Treat it as loopback stub route$ outside interface address added$ % Profile already contains this keyring$ %PVC is already defined$ Restarting RADIUS authentication service on port .* $ Restarting RADIUS accounting service on port .*$ Redundant .* statement$ security level for .* changed to$.*Service policy .* is already attached$ % Signature RSA Keys not found in configuration.$.*success$ The .*command will also show the fingerprint$ %The static routes in .* with outgoing interface .* will be removed$ Unable to disable parser cache$ % Unknown VPN$.* Unknown VRF specified$ % VRF .* does not exist or does not have a RD$.?warning.*
/DCS/IOSWarningExpressionsExitCfgMode		string	IOS warning expressions that can be safely ignored when exiting config term mode; regular expression must match whole warning message; for messages that wrap more than one line replace line terminations (CR and/or LF chars) with a single space character; replace each variable field with the meta-character sequence \\S+ that will match a single group of non-whitespace chars; literals are case insensitive; use $ to separate entries.

Table C-1 DCPL Properties (continued)

/DCS/IOSWarningExpressionsRemoveCfg		string	IOS warning expressions that can be safely ignored during decommissioning; case insensitive; . matches any char except newline, * means zero or more, + means one or more, ? means zero or one.
/DCS/RCP/			RCP Settings.
rcpDirectory	/tmp	string	Directory to use for uploaded/downloaded config files.
/DCS/SSH/			SSH Client Settings.
overWriteSSHKeys	true	The valid values are **true** and **false**.	Overwrite SSH Keys: If **true**, will allow new keys to overwrite existing keys in the key file for a given host. If **false**, an error will be displayed if host sent key does not match the server sent key.
sshEncryptionCipher	3DES->DES	selection	Cipher to use for SSH Encryption/Decryption; requires restart on change. Values: 3DES->DES first tries 3DES then if not available falls back to DES; 3DES, only tries 3DES; DES, only tries DES.
/DCS/SSHv2/			SSHv2 Client Settings.
overWriteSSHv2Keys	true	The valid values are **true** and **false**.	Overwrite SSHv2 Keys: If **true**, will allow new keys to overwrite existing keys in the key file for a given host. If **false**, an error will be displayed if host sent key does not match the server sent key.
/DCS/TFTP/			TFTP Settings.
tftpCreateFileOnServerBeforeUpload	true	The valid values are **true** and **false**.	Some TFTP servers require a file to exist on the server with write access before a TFTP client can upload it. This is sometimes called write-replace or overwrite mode. Other TFTP servers require a that a file NOT exist, this is sometimes called write-create or no overwrite mode. When **true**, DCS will create the file on the TFTP server before uploading device configuration.
tftpRootDirectory	/tftpboot	string	TFTP Root Directory used by DCS and GTL.
tftpServerIPAddress		string	TFTP Server host name or IP Address used by DCS and GTL must be the same as that of the ISC server.
tftpSubDirectory		string	TFTP Sub Directory used by DCS and GTL.
/DCS/XR			IOS XR properties.

Table C-1 DCPL Properties (continued)

WarningExpressions	^.?.?warning$	string	IOS XR warning expressions that can be: safely ignored; case insensitive; . matches any character except newline, where: * means zero or more, + means one or more, ? means zero or one.
commitConfigTimeout	120	integer, 30-600	Maximum time in seconds to commit config target buffer to running config.
maxRetriesEnterCfgExcIMode	3	integer, 0-10	Maximum number of times to retry entering configure exclusive mode. 0 = no retries. Retry delay interval is fixed at 30 seconds.
/DCS/allowCommandDownloadOnError	false	The valid values are **true** and **false**.	Continue command download on error.
/DCS/cnsEventTimeout	120	integer, 0-120 seconds	CNS event wait time in seconds
/DCS/configUploadTimeout	300	integer, 60-900	Maximum time in seconds to wait for a device configuration to be uploaded.
/DCS/customPasswordPrompt	Password:		Device custom password prompt.
/DCS/customUsernamePrompt	Username:		Device custom username prompt.
/DCS/getCommitCLIConfigAfterDownload	true	The valid values are **true** and **false**.	Retrieve the committed CLI configuration after an XML configuration download. If the default of **true** is set, whenever a Service Request is deployed on an IOS XR device, a transaction is created. This transaction gets the configlet deployed in the CLI mode and stores it in the repository. This creation of a new transaction adds to the time of Service Request deployment. If this property is set to **false**, no transaction to retrieve the CLI configlet is created.
/DCS/logLevel	CONFIG	selection	The log Level is the level at which logging is done for this component. These levels are identical to the logging levels defined for JDK1.4 logging package. The levels in descending order are: SEVERE (highest value) WARNING INFO CONFIG FINE FINER FINEST (lowest value).
/DCS/maxDeviceConnectCompleteTime	60	integer, 15-600 seconds	Maximum time in seconds to wait for a terminal session connection to a device.

Table C-1 DCPL Properties (continued)

/DCS/maxDeviceConnectRetryCount	3	integer, 0-5	Maximum number of times to retry connecting to a device when the maxDeviceConnectCompleteTime expires. 0= no retries.
/DCS/maxOperationTimeout	30	integer, 5-300 minutes	Maximum time in minutes to wait for a device operation to complete.
/DCS/maxPromptTimeout	60	integer, 15-300 seconds	Maximum time in seconds to wait for a prompt during a terminal session with a device.
/DCS/maxSocketReadTimeout	30	integer, 10-300 seconds	Maximum time in seconds to wait for data on a socket connection read operation.
/DCS/misc			Miscellaneous settings.
ConfigForMergeXML		string, file name	Configuration file to be used for the merging of two XMLs.
allowPromptCharsInBanner	false	The valid values are **true** and **false**.	Controls if prompt characters, such as # and >, are allowed in banners. If **true**, a minimum of 2 seconds (default of loginSocketReadTimeout) is added to each login. Note that selecting this option requires “aaa authentication attempts login n” to be set to a minimum of 2.
loginSocketReadTimeout	2	integer, 1-45	Number of seconds to WAIT for a login authentication username or password prompt. Applicable if DCS\misc\allowPromptCharsInBanner is **true**. Increasing this value slows down device logins and counts against DCS\maxDeviceConnectCompleteTime who’s default is 60 seconds.
readBufferSize	32	integer, 4-96	Size in KBytes of the buffers used while reading device input streams with telnet and SSH. Increasing size might improve performance. Decrease size if there are memory issues.
DeploymentFlow Property:			Deployment flow Component: Used to create a flow of different types of steps such as mpls.
/DeploymentFlow/logLevel	CONFIG	selection	The log Level is the level at which logging is done for this component. These levels are identical to the logging levels defined for JDK1.4 logging package. The levels in descending order are: SEVERE (highest value) WARNING INFO CONFIG FINE FINER FINEST (lowest value).
Discovery Properties:			ISC auto discovery framework.

Table C-1 DCPL Properties (continued)

/Discovery/DeviceDiscovery			
continueOnError	false	The valid values are **true** and **false**.	A Boolean flag indicating whether device discovery should try to continue on an error. When the value is **true**, device discovery ignores the device and attempts to create other devices discovered. In this case, the device discovery is marked as SUCCESS, but indicates there were errors. The default behavior is device discovery is marked FAILED at the first error encountered. This property applies only to errors encountered during the device creation phase of device discovery like duplicate or missing hostnames in case of CDP and file based discovery options and invalid device configurations or insufficient read permissions for configurations files and so on, for the configuration file based discovery option. Any errors encountered during CDP discovery itself or while parsing XML files still result in the device discovery step being marked as FAILED. WARNING: If this property is set to **true**, discovery continues if there are any device creation errors, ignoring the device that caused the error, but only partial NPCs and services are discovered.
logLevel	CONFIG	selection	The log Level is the level at which logging is done for this component. These levels are identical to the logging levels defined for JDK1.4 logging package. The levels in descending order are: SEVERE (highest value) WARNING INFO CONFIG FINE FINER FINEST (lowest value).

Table C-1 DCPL Properties (continued)

mgmtIpAddressLoopkupPattern		string	A comma separated list of interface name patterns to look for to determine the management IP address of the device discovered using the import configuration option. The configuration is parsed for the interface information, and the first available IP address of the interface from the given list is used as the management IP address of the device. For example, if the IP address of the loopback 0 interface should be used as the management IP address, the value of the property should be set to “loopback0”. If the first available loopback should be used, set the value of the property to “loopback”. A comma separated list can be specified as “Loopback0,Ethernet0”. In this case, the first available IP address among the list of interfaces specified in that order is used as the management IP address.
/Discovery/DataCollection			
continueOnError	false	The valid values are **true** and **false**.	A Boolean flag indicating whether data collection should try to continue on an error. When the value is **true**, the data collection step does not collect discovery data for the failed device, but attempts to collect configuration for other devices discovered. In this case, the configuration collection step is marked as SUCCESS, but indicates there were errors. The default behavior is discovery data collection step is marked FAILED at the first error encountered. WARNING: If this property is set to **true**, discovery continues if there are any collection or parsing errors, ignoring the device that caused the error, but only partial NPCs and services are discovered.
logLevel	CONFIG	selection	The log Level is the level at which logging is done for this component. These levels are identical to the logging levels defined for JDK1.4 logging package. The levels in descending order are: SEVERE (highest value) WARNING INFO CONFIG FINE FINER FINEST (lowest value).

Table C-1 DCPL Properties (continued)

reuseConfigsIfAvailable	false	The valid values are **true** and **false**.	If the Boolean flag is **true**, the discovery data collection step uses the config from the repository if available. If the configs are not in the repository, an attempt is made to contact the device to collect the current running configuration. The default behavior is discovery tries to collect the current running configs from the device.
/Discovery/MPLSService			MPLS services discovery.
logLevel	CONFIG	selection	The log Level is the level at which logging is done for this component. These levels are identical to the logging levels defined for JDK1.4 logging package. The levels in descending order are: SEVERE (highest value) WARNING INFO CONFIG FINE FINER FINEST (lowest value).
/Discovery/MetroEService			Metro Ethernet services discovery.
logLevel	CONFIG	selection	The log Level is the level at which logging is done for this component. These levels are identical to the logging levels defined for JDK1.4 logging package. The levels in descending order are: SEVERE (highest value) WARNING INFO CONFIG FINE FINER FINEST (lowest value).
meConfigParsingRegistry		string	List of handlers to be invoked at collect config time for Metro Ethernet services.
meDiscoverIntraPopVPWS	false	The valid values are **true** and **false**.	Set this to **true** if local switched VPWS services are to be discovered. Do this only if you wish to discover VPWS services switched at NPE. If not, set this to **false** for performance reasons.
/Discovery/NPCDiscovery			NPC discovery.
logLevel	CONFIG	selection	The log Level is the level at which logging is done for this component. These levels are identical to the logging levels defined for JDK1.4 logging package. The levels in descending order are: SEVERE (highest value) WARNING INFO CONFIG FINE FINER FINEST (lowest value).
/Discovery/RoleAssignment			

Table C-1 DCPL Properties (continued)

logLevel	CONFIG	selection	The log Level is the level at which logging is done for this component. These levels are identical to the logging levels defined for JDK1.4 logging package. The levels in descending order are: SEVERE (highest value) WARNING INFO CONFIG FINE FINER FINEST (lowest value).
/Discovery/Workflow			ISC auto discovery workflow.
logLevel	CONFIG	selection	The log Level is the level at which logging is done for this component. These levels are identical to the logging levels defined for JDK1.4 logging package. The levels in descending order are: SEVERE (highest value) WARNING INFO CONFIG FINE FINER FINEST (lowest value).
/Discovery/configs.location	*<vpnsc_tmp>*/ Discovery/ configs		The directory name where the temporary device configurations are stored during the collect config process.
/Discovery/logLevel	CONFIG	selection	The log Level is the level at which logging is done for this component. These levels are identical to the logging levels defined for JDK1.4 logging package. The levels in descending order are: SEVERE (highest value) WARNING INFO CONFIG FINE FINER FINEST (lowest value).
/Discovery/logLocation	*vpnsc_tmp*>/ Discovery/ logs	string	The directory name where discovery logs files are kept.
/Discovery/restart	false	The valid values are **true** and **false**.	With this property, you can clear out all network objects from the repository that was created by the Discovery process and you can restart the Discovery process. Be very cautious in setting this value to **true**.
/Discovery/tmpdir	*<vpnsc_tmp> /Discovery*	string	A directory to store the temporary results of the discovery process.
DistributionFramework Properties:			Distribution Framework. This component handles the distribution of work (jobs) between different servers in a ISC distributed installation.
/DistributionFramework/Dispatcher/			Service that dispatches jobs to workers.
DefaultUnitDuration	1000	integer	The unit duration (in milliseconds) used to estimate jobs without a profile.
PingInterval	1000	integer	The interval (in ms) dispatcher pings the workers to get the load.

Table C-1 DCPL Properties (continued)

ProcessorEpsilon	10	integer	If two proccessors differ in usage by an amount less than this, they are considered identical from the point of view of the load balancer.
ProfileUpdateThreshold	10	integer	The percent change of a profile that triggers an update of the dispatcher.
logLevel	CONFIG	selection	The log Level is the level at which logging is done for this component. These levels are identical to the logging levels defined for JDK1.4 logging package. The levels in descending order are: SEVERE (highest value) WARNING INFO CONFIG FINE FINER FINEST (lowest value).
/DistributionFramework/NamingHost	*<master_server>*	string	The hostname or ip address of the name server.
/DistributionFramework/NamingPort	*<naming_port>*	string	The port of the name server.
/DistributionFramework/RemoteUtil/			Layer abstracting the remote call functionality.
logLevel	CONFIG	selection	The log Level is the level at which logging is done for this component. These levels are identical to the logging levels defined for JDK1.4 logging package. The levels in descending order are: SEVERE (highest value) WARNING INFO CONFIG FINE FINER FINEST (lowest value).
/DistributionFramework/ServiceLauncher/			Manages the execution of multiple services in the same VM.
logLevel	CONFIG	selection	The log Level is the level at which logging is done for this component. These levels are identical to the logging levels defined for JDK1.4 logging package. The levels in descending order are: SEVERE (highest value) WARNING INFO CONFIG FINE FINER FINEST (lowest value).
/DistributionFramework/ThreadPool/			Thread pool component used by the worker to execute jobs.
logLevel	CONFIG	selection	The log Level is the level at which logging is done for this component. These levels are identical to the logging levels defined for JDK1.4 logging package. The levels in descending order are: SEVERE (highest value) WARNING INFO CONFIG FINE FINER FINEST (lowest value).

Table C-1 DCPL Properties (continued)

/DistributionFramework/Worker/			Worker.
Groups		string	The groups this worker belongs to. This property is deprecated because groups are stored in the database rather than being provided by the worker.
ThreadPoolSize	100	integer, 25-250	The maximum number of threads. Set it to 0 to allow the pool to use as many thread as necessary.
logLevel	CONFIG	selection	The log Level is the level at which logging is done for this component. These levels are identical to the logging levels defined for JDK1.4 logging package. The levels in descending order are: SEVERE (highest value) WARNING INFO CONFIG FINE FINER FINEST (lowest value).
GSAM Property:			Generic Service Access Model to get an XML dump from the repository for the provisioning driver.
/GSAM/logLevel	CONFIG	selection	The log Level is the level at which logging is done for this component. These levels are identical to the logging levels defined for JDK1.4 logging package. The levels in descending order are: SEVERE (highest value) WARNING INFO CONFIG FINE FINER FINEST (lowest value).
GTL Properties:			Generic Transport Layer. This library provides an API to different jobs (such as provisioning, collection etc.) to access Device Configuration Service (DCS). The jobs do not interface with DCS directly (to access the devices), but work with the API provided by GTL.
/GTL/CSL/			Configuration Services Layer
ios/			IOS related properties.
cmdsRequiringDelay		string	List of the IOS commands that execute asynchronously and require time to be processed before they are reflected in the running configuration. Matching rules: case insensitive, .matches any char except newline, * means zero or more, + means one or more, ? means zero or one.

Table C-1 DCPL Properties (continued)

delayAfterDownloadingCmd		command name**:** integer, 0-1800 seconds	List of the IOS commands that require a delay after they are downloaded using a terminal session protocol, such as Telnet. The character **;** delimits the list elements. The IOS command in each list element must be followed by the character **:** followed by a maximum integer of 1800, which indicates the number of seconds to delay, thus indicating 0-1800 seconds (0-30 minutes). The command matching rules: case insensitive, .matches any char except newline, * means zero or more, + means one or more, ? means zero or one. The default is a blank field.
delayBeforeDownloadingCmd			List of the IOS commands that require a delay before they are downloaded using a terminal session protocol, such as Telnet. The character **;** delimits the list elements. The IOS command in each list element must be followed by the character **:** followed by a maximum integer of 1800, which indicates the number of seconds to delay, thus indicating 0-1800 seconds (0-30 minutes). The command matching rules: case insensitive, .matches any char except newline, * means zero or more, + means one or more, ? means zero or one.
delayBeforeUpload		integer, 0-30 seconds	The delay in seconds to wait after downloading a configlet that contains asynchronous commands before uploading the new configuration.
delayBeforeWriteMem	0	integer, 0-300 seconds	The delay in seconds to wait after downloading a configlet before performing a write memory command.
/GTL/PAM/			
args		string	Invocation argument to be used.
className		string	PAM Class name.
usePAM	false	The valid values are **true** and **false**.	When the value is **true**, the selected PAM is used for device authentication. When the value is **false**, the standard authentication credentials are used in the ISC repository for each device.
/GTL/device-config-access-protocol	1	integer, 1-3	Protocol to use for device configuration uploads and downloads. 1= TERMINAL (Use the device-terminal-session-protocol for config access) 2= TFTP 3= FTP.

Table C-1 DCPL Properties (continued)

/GTL/device-terminal-session-protocol	1	integer, 1-2	Protocol to use for device terminal sessions. 1= TELNET 2= SSH.
/GTL/echo-mode	false	The valid values are **true** and **false**.	Flag indicating whether to run GTL in **ECHO** mode or **DCS** mode. Setting ISC to run in echo mode allows ISC to perform Service provisioning tasks without downloading the resulting commands to the physical hardware. The resulting Service Provisioning is stored only in the Repository and no attempt is made to connect to the target devices. When echo mode is enabled (set to **true**), no attempt to audit the Service Request is performed. From a production environment, you are able to perform service provisioning on devices that are either temporarily offline or not yet commissioned. Once these devices become active, you can Force Deploy the already provisioned Service Requests and ISC downloads the configurations.
/GTL/ios/			IOS related GTL properties.
copy-running-to-startup	true	The valid values are **true** and **false**.	Flag indicating whether to copy running config to startup config when downloading configlets. Write Mem flag.
/GTL/logLevel	CONFIG	selection	The log Level is the level at which logging is done for this component. These levels are identical to the logging levels defined for JDK1.4 logging package. The levels in descending order are: SEVERE (highest value) WARNING INFO CONFIG FINE FINER FINEST (lowest value).
copy-running-to-startup	true	The valid values are **true** and **false**.	Flag indicating whether to copy running config to startup config when downloading configlets. Write Mem flag.
GUI Properties:			The component for GUI-based properties.
/GUI/Common/			Generic GUI component. Use it if you do not have any specific component requirements, such as L2VPN.
logFileViewThreshold	10000000	integer	The maximum log file size in bytes that can be viewed in the GUI Log Viewer.

Table C-1 DCPL Properties (continued)

logLevel	FINE	selection	The log Level is the level at which logging is done for this component. These levels are identical to the logging levels defined for JDK1.4 logging package. The levels in descending order are: SEVERE (highest value) WARNING INFO CONFIG FINE FINER FINEST (lowest value).
/GUI/L2VPN/			L2VPN related GUI component. Use it with L2VPN related operations only.
logLevel	SEVERE	selection	The log Level is the level at which logging is done for this component. These levels are identical to the logging levels defined for JDK1.4 logging package. The levels in descending order are: SEVERE (highest value) WARNING INFO CONFIG FINE FINER FINEST (lowest value).
/GUI/MPLSOAM/			The MPLS OAM component.
logLevel	FINEST	selection	The log Level is the level at which logging is done for this component. These levels are identical to the logging levels defined for JDK1.4 logging package. The levels in descending order are: SEVERE (highest value) WARNING INFO CONFIG FINE FINER FINEST (lowest value).
/GUI/MplsVPN/			MPLS VPN related GUI component. Use it with MPLS VPN related operations only.
logLevel	SEVERE	selection	The log Level is the level at which logging is done for this component. These levels are identical to the logging levels defined for JDK1.4 logging package. The levels in descending order are: SEVERE (highest value) WARNING INFO CONFIG FINE FINER FINEST (lowest value).
/GUI/Performance/			For monitoring GUI performance.

Table C-1 DCPL Properties (continued)

logLevel	INFO	selection	The log Level is the level at which logging is done for this component. These levels are identical to the logging levels defined for JDK1.4 logging package. The levels in descending order are: SEVERE (highest value) WARNING INFO CONFIG FINE FINER FINEST (lowest value).
GUI/Ping			Ping related GUI component. Use it with Ping related operations only.
logLevel	CONFIG	selection	The log Level is the level at which logging is done for this component. These levels are identical to the logging levels defined for JDK1.4 logging package. The levels in descending order are: SEVERE (highest value) WARNING INFO CONFIG FINE FINER FINEST (lowest value).
/GUI/Topology/			Component related to the web start topology application.
logLevel	CONFIG	selection	The log Level is the level at which logging is done for this component. These levels are identical to the logging levels defined for JDK1.4 logging package. The levels in descending order are: SEVERE (highest value) WARNING INFO CONFIG FINE FINER FINEST (lowest value).
/GUI/VPLS/			VPLS related GUI component. Use it with VPLS related operations only.
logLevel	SEVERE	selection	The log Level is the level at which logging is done for this component. These levels are identical to the logging levels defined for JDK1.4 logging package. The levels in descending order are: SEVERE (highest value) WARNING INFO CONFIG FINE FINER FINEST (lowest value).
/GUI/srRefreshRate	30000	integer	The refresh rate (in milliseconds) for the SR List screen.
/GUI/workflowSteps	<*vpnsc_home*>/etc/workflowSteps.csv	string	The predefined workflow steps.
/GUI/workflows	<*vpnsc_home*>/etc/workflows.csv	string	The predefined workflows.

Table C-1 DCPL Properties (continued)

JavaWebStart Properties			Java Web Start components.
/JavaWebStart/InventoryManager/			Component to create and manage Devices.
MaxDevicesPerSaveTransaction	25	integer, 1-500	Specifies the maximum number of devices per transaction when performing save operation.
/JavaWebStart/TaskManager/			Component to create and monitor scheduled tasks.
MaxDevicesPerCollectionTask	25	integer, 1-500	Specifies the maximum number of devices per Collect Config task. More devices can be specified for a single task and they will be managed as such from a user perspective. However, there may be more than on Collect Config task created and executed in the repository.
Logging Properties:			This contains different properties needed by the logging framework. There are a set of default values for logging parameters. These values can be overridden for a specific server.
/Logging/Defaults/			This contains the default values for the logging framework.
logFileNumber	2	integer, 1-10	Maximum number of log files for a process. Each of these files can be of size **logFileSize**. When the maximum number for log files is reached for a process, the log files are rotated by deleting the oldest log file for that process.
logFileSize	2000000	integer, 1000000-10000000 bytes	Size in bytes of a single log file for a process. Each process will have a number of log files (see **logFileNumber** property), where each of these files can grow to this size.
logFormatter	java.util.logging.XMLFormatter	string	Class name for the default formatter of log records.
logLevel	CONFIG	selection	NOTE: This log Level is used only if there is no log Level defined for a component. The log Level is the level at which logging is done for this component. These levels are identical to the logging levels defined for JDK1.4 logging package. The levels in descending order are: SEVERE (highest value) WARNING INFO CONFIG FINE FINER FINEST (lowest value).
logLocation	*<vpnsc_tmp>*	string	The directory name where log files are kept.

Table C-1 DCPL Properties (continued)

/Logging/TaskLogs/			This contains logging properties for task logs.
logLocation	*<vpnsc_tmp>*/ TaskLogs	string	The directory name where all the task logs are kept.
logMessageSize	100	integer, 100-300	This property sets the number of lines of message to be displayed for each log entry.
Provisioning Properties:			Contains properties and components for service provisioning like MPLS VPNs.
/Provisioning/Engine/			Contains properties for the XML driven provisioning engine.
logLevel	CONFIG	selection	The log Level is the level at which logging is done for this component. These levels are identical to the logging levels defined for JDK1.4 logging package. The levels in descending order are: SEVERE (highest value) WARNING INFO CONFIG FINE FINER FINEST (lowest value).
serviceSchema	service.xsd	string	Specifies the XML schema definition file for defining new services.
/Provisioning/NOM/			Network Object Model for parsing and delta generation of configs.
DocumentBuilderFactory/			This contains the properties for the DOM builder factory.
ignoreComments	true	The valid values are **true** and **false**.	Flag.
ignoreWhiteSpace	false	The valid values are **true** and **false**.	Flag for DOM builder factory.
validation	false	The valid values are **true** and **false**.	Flag for validation of xml files.
catSyntaxFile	catSyntax.xml	string	Contains the XML for Catalyst command syntax.
explicitlyRemoveRouteTargets	false	The valid values are **true** and **false**.	Normally (**false**), the “no ip vrfname” automatically cleans up all its subcommands in IOS. There is no need to clean up each one of the subcommands before taking away the parent command. By setting this value to **true**, ISC explicitly cleans up all router target cubcommands before removing the “ip vrfname”.

Table C-1 DCPL Properties (continued)

iosSyntaxFile	iosSyntax.xml	string	Contains the xml syntax for IOS command.
logLevel	CONFIG	selection	The log Level is the level at which logging is done for this component. These levels are identical to the logging levels defined for JDK1.4 logging package. The levels in descending order are: SEVERE (highest value) WARNING INFO CONFIG FINE FINER FINEST (lowest value).
/Provisioning/PasswordManagement/			User generated Password generation
PasswordFormula/			User generated Password formula generation class
class		string	User generated class file
/Provisioning/ProvDrv/			Contains properties for the XML driven provisioning ProvDrv.
AuditJITUpload	true	The valid values are **true** and **false**.	If the value of this property is set to **false,** the provisioning server does NOT upload a copy of the configuration file from the routers when it processes the Service Request for auditing purpose. Instead, it uses copies of the configuration files that were collected and stored in the Repository earlier. If the value of this property is set to **true**, the provisioning server uploads a copy of the configuration file from the routers when it processes the Service Request for auditing purpose. The default value of this property is **true**.
CleanStagedConfigletWhenForceDeploy	false	The valid values are **true** and **false**.	If this value is **true**, when a service request is force deployed, the staged configlet is removed before provisioning. If this value is the default of **false**, the staged configlet is considered as part of the base configuration during provisioning.
DownloadTemplateToUnmanagedDevice	false	The valid values are **true** and **false**.	If this value is **true**, for an unmanaged device, ISC attempts to download just the template. The configlet generated by the provision is not part of the download. By default, this value is **false** and then there is no attempt to download to an unmanaged device.

Table C-1 DCPL Properties (continued)

MaxNumberOfDevicesPerDownload	100	integer	ISC will try to bundle as much devices as possible during a download attempt. This value set the max number of devices allowed during such an attempt. If the number of devices exceeds this limit, multiple download attempts will take place. You should decrease this limit if the download involves many devices with huge configlets in order to conserve memory usage.
ProvisionJITUpload	true	The valid values are **true** and **false**.	If the value of this property is set to **false**, the provisioning server does NOT upload a copy of the configuration file from the routers when it processes the Service Request for provisioning purpose. Instead, it uses copies of the configuration files that were collected and stored in the Repository earlier. If the value of this property is set to **true**, the provisioning server uploads a copy of the configuration file from the routers when it processes the Service Request for provisioning purpose.
ProvisioningBatchSize	10	integer, 0-2147483647	Provisioning Driver divides the requested Service Requests into batches while performing the deployment. This parameter specifies the number of Service Requests that will be processed as a batch.
SaveConfigletsFromAllSRs	true	The valid values are **true** and **false**.	If the value of this property is set to true, for each device in a SR, the provisioning server will save the configlet contributed from all SRs that are processed in the same provisioning run. If the value is set to false, only the configlet contributed by the current SR is saved for this device in this SR even though this same device may be in multiple SRs that are processed by the same provisioning run.
logLevel	CONFIG	selection	The log Level is the level at which logging is done for this component. These levels are identical to the logging levels defined for JDK1.4 logging package. The levels in descending order are: SEVERE (highest value) WARNING INFO CONFIG FINE FINER FINEST (lowest value).
/Provisioning/Service/			Contains different services and their properties.

Table C-1 DCPL Properties (continued)

TE/			Traffic Engineering Provisioning Service related properties section.
enableLogging	true	The valid values are **true** and **false**.	When the value is the default of **true**, debugging of logging is enabled for this service. When the value is **false**, debugging of logging is not enabled for this service.
platform/			Used by ProvDrv
CISCO_ROUTER/			Used by ProvDrv
serviceBladeClass	com.cisco.vpnsc.prov.te.ServiceBlade.TeServiceBlade	string	Identifies ServiceBlade class name for ProvDrv.
sendAuditEvent	true	The valid values are **true** and **false**.	Set **true** to enable sending audit event for this service.
Uds/			User defined services.
platform/			Service platform
CISCO_ROUTER/			Cisco router
serviceBladeClass	com.cisco.vpnsc.prov.uds.UdsServiceBlade	string	Uds Service Blade.
l2vpn/			MPLS Layer 2 VPN Provisioning.
DownloadWeights/			Specifies the download weights for different devices in an L2VPN service request. The higher the weight, the sooner we download to that device. By default the weights are set to 0, so that all devices get downloaded at the same time during service deployment.
weightForCE	0	integer	Download weight for CE devices.
weightForPE	0	integer	Download weight assigned to PE devices.
weightForPE_CLE	0	integer	download weight for PE_CLE devices.
platform/			Contains properties for L2VPN for different platforms.
CATOS/			Service blade parameters for CATOS.
serviceBladeClass	com.cisco.vpnsc.prov.l2vpn.L2VPNServiceBlade	string	ServiceBladeClass location.
CISCO_ROUTER/			
iosXRConfigType	XML		Config type for IOS XR devices for MPLS service blade

Table C-1 DCPL Properties (continued)

serviceBladeClass	com.cisco.vpnsc.prov.l2vpn.L2VPNServiceBlade	string	ServiceBladeClass location.
dataFileSchema	l2vpnData.xsd	string	Layer 2 VPN Data File schema.
logLevel	CONFIG	selection	The log Level is the level at which logging is done for this component. These levels are identical to the logging levels defined for JDK1.4 logging package. The levels in descending order are: SEVERE (highest value) WARNING INFO CONFIG FINE FINER FINEST (lowest value).
parseConfigAfterProvisioning	false	The valid values are **true** and **false**.	This property controls the parsing of the configuration file after the provisioning is completed in order to make sure that device inventory is in sync with network.
saveDebugData	true	The valid values are **true** and **false**.	If this property is set to **true**, whenever an SR is provisioned, the uploaded config files and input XML data are saved to a temporary directory for debugging purposes.
sendAuditEvent	true	The valid values are **true** and **false**.	Set true to enable sending audit event for this service.
serviceFile	l2vpnService.xml	string	Layer 2 VPN Service definition file.
logLevel/	SEVERE	selection	The log Level is the level at which logging is done for this component. These levels are identical to the logging levels defined for JDK1.4 logging package. The levels in descending order are: SEVERE (highest value) WARNING INFO CONFIG FINE FINER FINEST (lowest value).
mpls/			Contains properties for MPLS/BGP Layer 3 VPN service.
DownloadWeights/			Specifies the download weights for different devices in an MPLS-VPN service request. The higher the weight, the sooner we download to that device. By default the weights are set to 0, so that all devices get downloaded at the same time during service deployment.
weightForCE	0	integer	Download weight for CE devices.
weightForMVRFCE	0	integer	Download weight for MVRFCE. The higher the weight the sooner we download to this device while deploying a service request.

Table C-1 DCPL Properties (continued)

weightForPE	0	integer	Download weight assigned to PE devices.
weightForPE_CLE	0	integer	Download weight for PE_CLE devices.
platform/			Platform related classes.
CATOS/			Service blade parameters for CATOS.
serviceBladeClass	com.cisco.vpnsc.prov.mpls.MplsServiceBlade	string	ServiceBladeClass location.
CISCO_ROUTER			IOS.
iosXRConfigType	XML		Config type for IOS XR devices for MPLS service blade
serviceBladeClass	com.cisco.vpnsc.prov.mpls.MplsServiceBlade	string	ServiceBladeClass location
allowDuplicateIpAddressForPPPoATM	false	The valid values are **true** and **false**.	Provision PPPoATM by allowing duplicate IP addresses for MPLS Service Requests. Ignore duplicate IP address on Loopback and Multilink interfaces.
allowOverwriteManualAssignedAddress	false	The valid values are **true** and **false**.	Allow manually-assigned IP address in Service Request overwrite the pre-existing interface IP address. **False** means if an MPLS service request tries to provision a manually-assigned IP address to an interface that already has a different IP address on it, ISC detects that and reports the error. **True** means ISC allows the new IP address to overwrite the existing IP address.
allowShared VLAN Modification	false	The valid values are **true** and **false**.	For residential services, if the flag is on, **true**, shared VLAN attributes are available for modify in edit mode. If the flag is off, **false**, attributes are in read only mode.
auditIpAddressViaUnnumbered	false	The valid values are **true** and **false**.	When the value is the default of **false**, the auditor only looks for the IP address of a provisioned interface. When the value is **true**, the auditor tries to match the IP address of the unnumbered interface, if one exists.
auditMaxrouteThreshold	true	The valid values are **true** and **false**.	This property controls whether an audit will be run on the Max Route Threshold for a Service Request. This is needed to maintain backward compatibility.
auditPartialCommands	false	The valid values are **true** and **false**.	This property is set for the autodiscovered systems containing a superset of the commands that ISC supports.

Table C-1 DCPL Properties (continued)

dataFileSchema	l3vpnData.xsd	string	Specifies the schema for the data XML file for MPLS/BGP layer3 VPNs.
excludeNoKeepaliveConfigOnPort Channel	false	The valid values are **true** and **false**.	Exclude the no keepalive command on the port channel trunk port.
forceRemoveNonBroadcastStatic RouteOnPE	false	The valid values are **true** and **false**.	The default value is **false**. When the value is set to **true**, ISC removes the non-broadcast type static route command that has a pre-existing long syntax, even if the command was not provisioned by ISC. The non-broadcast type static route command is removed from a PE router prior to provisioning. Long syntax contains both an outgoing interface name and a next hop IP address.
ignoreLoopbackWhileRemovingVRF	false	The valid values are **true** and **false**.	Remove a VRF, even when some Loopback interfaces are still pointing to it.
ignoreMajorInterfaceCheck	false	The valid values are **true** and **false**.	This property controls the check for a proper major interface name in an unmanaged CE. If set to **true**, ISC bypasses the check for a proper major interface name. Note: This will work only for UnmanagedCE devices
ignoreStatusMessagesForUnmanaged CEs	false	The valid values are **true** and **false**.	If set to **true**, this property prevents the generation of status messages for unmanaged CEs
logLevel	CONFIG	selection	The log Level is the level at which logging is done for this component. These levels are identical to the logging levels defined for JDK1.4 logging package. The levels in descending order are: SEVERE (highest value) WARNING INFO CONFIG FINE FINER FINEST (lowest value).
parseConfigAfterProvisioning	false	The valid values are **true** and **false**.	This property controls the parsing of the configuration file after the provisioning is completed in order to make sure that device inventory is in sync with network.
passAuditForNonBroadcastStatic RouteOnPE	false	The valid values are **true** and **false**.	When this property is set to **true**, the ISC auditor does not generate an error message if the static route was found with a different format (such as, a PE interface name instead of a CE IP address).

Table C-1 DCPL Properties (continued)

passIpAddressAuditWhenNoAddress Detected	false	The valid values are **true** and **false**.	Pass the IP address command auditing if uploaded router config does not contain an IP address. This is to prevent the audit failure from appended template blob overwriting the provisioned IP address command.
reapplyIpAddress	false	The valid values are **true** and **false**.	Re-apply the same IP address to the interface when decommission a service request. This option is only applicable to manually-assigned IP addresses. It does not work for automatically-assigned IP addresses. When this property is in effect, the interface negate command will not be generated.
removeSubInterface	true	The valid values are **true** and **false**.	Removing the ISC generated subinterface commands in decommission service requests.
routeMapDeletedAfterLastLink Deletion	true	The valid values are **true** and **false**.	If this property is set to **true**, the route map configuration is automatically removed from the device after the last link is deleted. If **false**, the route map configuration is left as it is in the device.
saveDebugData	true	The valid values are **true** and **false**.	If this property is set to **true**, whenever an SR is provisioned, the uploaded config files and input XML data are saved to a temporary directory for debugging purposes.
sendAuditEvent	true	The valid values are **true** and **false**.	Set true to enable sending audit event for this service.
serviceFile	l3vpnService.xml	string	Specifies the XML file containing the service definition for MPLS/BGP layer3 VPNs. The schema for this file is specified by Provisioning.Engine.serviceSchema
skipIpAddressValidationOn UnmanagedCE	false	The valid values are **true** and **false**.	When the value is **false**, the IP addresses between a PE and an unmanaged CE are validated to ensure they are in the same subnetwork and valid host addresses. When the value is **true**, this validation is bypassed.
useNextHopAddressForStaticRoutes	false	The valid values are **true** and **false**.	For Static Routes, use local router outbound interface or IP address of the next hop to reach the destination network.

Table C-1 DCPL Properties (continued)

useOnlyExtraCEloopbackForGrey AccessList	false	The valid values are **true** and **false**.	With Extra CE loopback, the user can select this option to add only the loopback address instead of the interface ip address and extra CE loopback.
shared/			Properties shared by MPLS VPN, L2VPN and VPLS.
FeatureQuery/			ISC components that check if certain features are available for certain devices based on their software version and platform information.
enableValidation	true	The valid values are **true** and **false**.	If enabled, FeatureQuery will check if the features are available based on the feature matrix and device OS version (IOS Version or PIX Version). If disable it will assume that all features are available on all platforms (should be used for testing only).
IosXrVersionFilesDir		string	Path to IOS XR version XML files.
actionTakenOnUNIVlanList	prune	string	Action taken when switch port **allowed vlan** cmd is absent for ERS service.
leaveSystemMTUUnset	false	The valid values are **true** and **false**.	If this property is set as **true**: U-PE system MTU is not set as default, or set as value given by user; N-PE SVI MTU is set as 9216 for VPLS(EWS and ERS) and L2VPN(EWS). If this property is set as **false**: U-PE system MTU is set as minimum value 1522, or set as value given by user; N-PE SVI MTU is not set as default, or set as value given by user.
overwriteInterfaceDescription	true	The valid values are **true** and **false**.	By default, ISC generates a **description** subcommand for all the physical interfaces it provisioned. Set this property to false if this behavior is not desirable. This property does not apply to logical interfaces or other CLI objects that have a **description** subcommand (Example: crypto map entries, gre Interfaces, and so on).
transferUNIDescToVlanName	false	The valid values are **true** and **false**.	Controls provisioning of the VLAN name on the PE-POP. If set to **true**, the VLAN name is assigned from the description for the UNI. If set to the default of **false**, no VLAN name is assigned.

Table C-1 DCPL Properties (continued)

useSRDescriptionToGenerateDebug Data	false	The valid values are **true** and **false**.	This property is used to generate more intuitive debug data for easy fixing of issues.
staging/			
platform/			Platform related classes.
CATOS/			Service blade parameters for CATOS.
serviceBladeClass	com.cisco.vpnsc.prov.staging.StagingServiceBlade	string	ServiceBladeClass location.
CISCO_ROUTER/			IOS.
serviceBladeClass	com.cisco.vpnsc.prov.staging.StagingServiceBlade	string	ServiceBladeClass location.
logLevel	CONFIG	selection	The log Level is the level at which logging is done for this component. These levels are identical to the logging levels defined for JDK1.4 logging package. The levels in descending order are: SEVERE (highest value) WARNING INFO CONFIG FINE FINER FINEST (lowest value).
parseConfigAfterProvisioning	false	The valid values are **true** and **false**.	This property controls the parsing of the configuration file after the provisioning is completed to make sure that device inventory is in sync with network.
saveDebugData	true	The valid values are **true** and **false**.	If this property is set to **true**, whenever an SR is provisioned, the uploaded config files and input XML data are saved to a temporary directory for debugging purposes.
sendAuditEvent	true	The valid values are **true** and **false**.	Set **true** to enable sending audit event for this service.
serviceFile	stagingService.xml	string	Specifies the XML file containing the service definition for staging service. The schema for this file is specified by Provisioning.Engine.serviceSchema.
vpls/			Contains properties for Virtual Private LAN Service.

Table C-1 DCPL Properties (continued)

DownloadWeights/			Specifies the download weights for different devices in an MPLS VPN service request. The higher the weight, the sooner we download to that device. By default the weights are set to 0, so that all devices get downloaded at the same time during service deployment.
weightForCE	0	integer	Download weight for CE devices.
weightForPE	0	integer	Download weight assigned to PE devices.
weightForPE_CLE	0	integer	Download weight for PE_CLE devices.
dataFileSchema	vplsData.xsd	string	Specifies the schema for the data XML file for VPLS.
logLevel	CONFIG	selection	The log Level is the level at which logging is done for this component. These levels are identical to the logging levels defined for JDK1.4 logging package. The levels in descending order are: SEVERE (highest value) WARNING INFO CONFIG FINE FINER FINEST (lowest value).
parseConfigAfterProvisioning	false	The valid values are **true** and **false**.	This property controls the parsing of the configuration file after the provisioning is completed to make sure that device inventory is in sync with network.
platform/			Platform related classes.
CATOS/			Service blade parameters for CATOS.
serviceBladeClass	com.cisco.vpnsc.prov.vpls.VplsServiceBlade	string	ServiceBladeClass location.
CISCO_ROUTER/			IOS.
serviceBladeClass	com.cisco.vpnsc.prov.vpls.VplsServiceBlade	string	ServiceBladeClass location.
saveDebugData	true	The valid values are **true** and **false**.	If this property is set to **true**, whenever an SR is provisioned, the uploaded config files and input XML data are saved to a temporary directory for debugging purposes.
sendAuditEvent	true	The valid values are **true** and **false**.	Set **true** to enable sending audit event for this service.

Table C-1 DCPL Properties (continued)

serviceFile	vplsService.xml	string	Specifies the XML file containing the service definition for VPLS. The schema for this file is specified by Provisioning.Engine.serviceSchema.
SLA Properties:			Service Level Agreement. This component deals with creating SAA probes between different devices and to collect/aggregate the data corresponding to those probes, in order to provide different SLA reports.
/SLA/copyRunningToStartup	true	The valid values are **true** and **false**.	If **true** and if showInRunningConfig is **true** - the running configuration will be copied to startup after the router SA Agent configuration has been changed.
/SLA/daysToKeepDailyStats	365	integer, 30-3650 days	Specifies how many days should the SLA database keep the daily statistics. Specifying a low number keeps the database small but you will not be able to access daily reports beyond this period.
/SLA/daysToKeepHourlyStats	60	integer, 7-1000 days	Specifies how many days should the SLA database keep the hourly statistics. Specifying a low number keeps the database small but you will not be able to access hourly reports beyond this period.
/SLA/logLevel	CONFIG	selection	The log Level is the level at which logging is done for this component. These levels are identical to the logging levels defined for JDK1.4 logging package. The levels in descending order are: SEVERE (highest value) WARNING INFO CONFIG FINE FINER FINEST (lowest value).
/SLA/rowAgeOut	3600	integer, 0-2073600 seconds	The time after which a probe is completely removed after its life is over. In seconds.
/SLA/showInRunningConfig	true	The valid values are **true** and **false**.	If true, the configured SLAs appear in the router's running configuration.
SYSTEM Properties:			The properties common to all sub-systems in ISC can be found under this component. Most of the values here are set at the time of installation.
/SYSTEM/app_dir	<*vpnsc_home*>	string	Location of the ISC installation.
/SYSTEM/ciscoURL	http://www.cisco.com	string	The Cisco URL.
/SYSTEM/databaseServer	<*db_server*>	string	The database server fully qualified name.

Table C-1 DCPL Properties (continued)

/SYSTEM/email/			Properties related to e-mails sent out by ISC.
from	<*mailfrom*>	string	The **from** field in the e-mail header of the mails sent out by ISC.
smtpHost	<*mailhost*>	string	The server using which e-mail messages from ISC should be sent out.
/SYSTEM/fullyManaged/			Properties related to e-mails sent out by ISC in case of fully managed devices.
auditableCommandsFileLocation		string	This property specifies the full path to the file containing the list of prefixes of auditable commands used in the Fully Managed feature.
enforcementAuditScript		string	Script to be invoked when failure of enforcement audit is detected.
externalEventsEmailRecipients	<*mailto*>	string	The comma or space separated list of email addresses to which notification should be sent out when receiving a config-change event originated outside ISC.
/SYSTEM/license/			Properties related to ISC Licensing.
emailRecipients	<*mailto*>	string	The comma separated list of e-mail addresses to which the License Threshold e-mails should be sent out.
refreshInterval	1	integer, 1-24 hours	License refresh interval in hours.
threshold	90	integer, 1-100%	VPN and ACTIVATION Threshold in percent for e-mail notification.
/SYSTEM/masterServer	<*master_server*>	string	The master server fully qualified name.
/SYSTEM/maxTaskLimit	500	integer	maxTaskLimit.
/SYSTEM/role	master	string	The possible value is: master.
/SYSTEM/tibco/			TIBCO related properties.
port	<*tibco_port*>	integer	The port on which TIBCO Rendezvous listens for events.
prefix	cisco.vpnsc.	string	Prefix for all TIBCO messages originating from ISC.
rva-http-port	<*rva_http_port*>	integer	The http port for TIBCO Rendezvous agent web interface.
rva-port	<*rva_port*>	integer	The port on which TIBCO Rendezvous agent listens for events.
/SYSTEM/tmpdir	<*vpnsc_tmp*>	string	Location for temporary files.

Table C-1 DCPL Properties (continued)

Scheduler Properties:			Scheduler reads the task repository and schedules tasks on every minute boundary. Each scheduled task is passed to Task manager for execution.
/Scheduler/logLevel	CONFIG	selection	The log Level indicates the level at which logging is done for this component. These levels are identical to the logging levels defined for JDK1.4 logging package. The levels in descending order are: SEVERE (highest value) WARNING INFO CONFIG FINE FINER FINEST (lowest value).
/Scheduler/syncInterval	5	integer, 0-10 minutes	When scheduler starts up for the first time, it reads all the scheduling information from the task repository. After that, it depends on the events generated by task repository for receiving changes to the scheduling information. It can also periodically synchronize with the task repository by re-reading it at regular intervals. This property specifies, in minutes, that interval. If the value for the interval is 0, scheduler will not synchronize with the task repository and only depends on the events.
Services Properties:			Common services.
/Services/Common/			
allowForcePurge	true	The valid values are **true** and **false**.	With the default value of **true**, you can force purge a Service Request. If the value is **false**, you cannot force purge a Service Request.
disallowVlan1	true	The valid values are **true** and **false**.	This prevents allocating VLAN ID 1 for services configured by ISC. This is applicable for both auto allocation of VLAN from VLAN resource pool and manual allocation. Set this property to **true** to block ISC from deploying services with VLAN ID 1
pseudoWireVlanMode	false	The valid values are **true** and **false**.	This property is effective only for IOS XR L2VPN services. The default is **false**. When set to **true**, this configures pseudowire transport mode to VLANs.
SnmpService Properties:			The Snmp Service package provides APIs to perform SNMP get() and set() operations.
/SnmpService/misc			Advanced settings.

Table C-1 DCPL Properties (continued)

enableDebug	false	The valid values are **true** and **false**.	Enables the AdventNet SNMP stack debug messages. Messages are written to the TaskLogs directory in files stdout and stderr. Warning: These log files grow quickly and are NOT managed by the ISC logger. Requires WatchDog restart.
rcvPktBuffSize	96	integer, 64-512	Buffer size in K bytes, for SNMP stack receive buffer.
/SnmpService/defaultSNMPVersion	1	integer, 1-2	The default SNMP version used to connect to Cisco router. Used if the SNMP version is not specified per router. Valid Values: SNMPv1/SNMPv2c - 1 SNMPv3 - 2.
/SnmpService/defaultSecurityLevel	3	integer, 1-3	The default security level used to connect to Cisco router. Used if the security level is not specified per router. Values: authentication no encryption - 1 authentication encryption - 2 no authentication no encryption - 3.
/SnmpService/logLevel	CONFIG	selection	The log Level is the level at which logging is done for this component. These levels are identical to the logging levels defined for JDK1.4 logging package. The levels in descending order are: SEVERE (highest value) WARNING INFO CONFIG FINE FINER FINEST (lowest value).
/SnmpService/maxTaskDuration	5	integer, 1-30	Maximum duration in minutes for collecting device interface information. A longer duration is required for devices with large numbers of interfaces. This period must be longer than 2^(retries+1) * timeout.
/SnmpService/retries	3	integer, 0-10	The number of retries to be used by the SNMP protocol.
/SnmpService/timeout	5	integer, 0-300 seconds	Timeout value to be used by the SNMP protocol. Unit: seconds
TE Properties:			Traffic Engineering Management (TEM) Properties
/TE/Deployment			Control the operation of TEM Provisioning
maxCacheSize	60	integer, >0	Maximum cache size.
oneDeviceEachTimeThreshold	500	integer, >0	When the total number of tunnels to be provisioned exceeds this threshold number, provision one device at a time.

Table C-1 DCPL Properties (continued)

partialConfigAudit	false	The valid values are **true** and **false**.	When the value is the default of **false**, the config audit is not limited. When the value is set to **true**, only a partial config audit (audit of only the PENDING tunnels) occurs for primary and backup tunnel deployment.
tunnelMplsIp	true	The valid values are **true** and **false**.	When the value is the default of **true**, this indicates to deploy the **mpls ip** command when provisioning TE primary tunnels, which enables MPLS IP switching on the router. When the value is set to **false**, this indicates not to deploy the **mpls ip** command when provisioning TE primary tunnels.
/TE/repository			TEM Repository-related Properties
checkPermissionEnabled	false	The valid values are **true** and **false**.	This property enables or disables Role-Based Access Control (RBAC) checking during particular TEM operations, such as topology population, discovery, and service deployment. When the value is the default of **false**, RBAC permission checking is not enabled. When the values is set to **true**, RBAC permission checking is enabled and performance degrades.
TE Topology Properties:			TEM Topology-related Properties
/TE Topology/TrafficData			Color Control for Traffic Data Displays
Green	0-25	integer, 0-100 (percentage)	Topology representations for a link performance utilization range, specified as a percentage (default: 0-25), are displayed in the color green.
Orange	51-75	integer, 0-100 (percentage)	Topology representations for a link performance utilization range, specified as a percentage (default: 51-75), are displayed in the color orange.
Red	76-100	integer, 0-100 (percentage)	Topology representations for a link performance utilization range, specified as a percentage (default: 76-100), are displayed in the color red. Greater than 100% is also displayed in red.
Yellow	26-50	integer, 0-100 (percentage)	Topology representations for a link performance utilization range, specified as a percentage (default: 26-50), are displayed in the color yellow.

Table C-1 DCPL Properties (continued)

TaskManager Properties:			Task manager executes tasks that are scheduled by scheduler. Task execution consists of executing different actions that comprise the task. Task manager manages the dependencies between these actions.
/TaskManager/CollectConfig			The Collect Config task uploads the running configuration.
logLevel	CONFIG	selection	The log Level indicates the level at which logging is done for this component. These levels are identical to the logging levels defined for JDK1.4 logging package. The levels in descending order are: SEVERE (highest value) WARNING INFO CONFIG FINE FINER FINEST (lowest value).
/TaskManager/logLevel	CONFIG	selection	The log Level indicates the level at which logging is done for this component. These levels are identical to the logging levels defined for JDK1.4 logging package. The levels in descending order are: SEVERE (highest value) WARNING INFO CONFIG FINE FINER FINEST (lowest value).
VpnInvServer Properties:			Corba Server for VpnInvServer IDL backward compatibility.
/VpnInvServer/logLevel	SEVERE	selection	The log Level is the level at which logging is done for this component. These levels are identical to the logging levels defined for JDK1.4 logging package. The levels in descending order are: SEVERE (highest value) WARNING INFO CONFIG FINE FINER FINEST (lowest value).
aagent Properties:			AAgent component related defines.
/aagent/defaultVersion	3.6.3	string	The default 3k firmware version for AAgent.
/aagent/directories/			Various directories for aagent.
dmd	<*vpnsc_home*>/resources/AAgent/DMDFiles	string	File path and name.
input	<*vpnsc_home*>/resources/java/classes/common/AAgent/com/cisco/vpnscagent	string	File path and name.

Table C-1 DCPL Properties (continued)

working	<*vpnsc_home*>/resources/java/archives	string	File path and name.
cfr Properties:			The Command Flow Runner component. This currently runs within the Tomcat server (in the ISC web application) and is responsible for running MPLSOAM troubleshooting workflows.
/cfr/Diagnostics/			
disableTunnelDiagnostics	false	The valid values are **true** and **false**.	Set to **true** to disable tunnel diagnostics, in order to avoid errors when running MDE across networks with non-Cisco devices in the tunnel LSPs.
/cfr/LogHandler	com.cisco.mgmt.workflow.util.IscLogHandler		Set the CFR to use a custom handler for logging. The handler should log to a separate file and format the log messages using the java.util.logging.SimpleFormatter instead of the ISC default XML formatting.
/cfr/logLevel	INFO		The level of logging information the Command Flow Runner will log (it will log from the set level upwards). The logging levels are as defined in the java.util.logging package.
lockmanager Properties:			Component that handles device locking. When different jobs (such as provisioning) try to update the config on the device, they obtain **software** locks so that two different jobs do not update the config at the same time. LockManager provides a way to obtain and later release such software locks.
/lockmanager/collectConfigLock	false	The valid values are **true** and **false**.	Determines if a software lock is to be applied to the devices in the CollectConfig task. If **true**, a software lock is applied to all devices prior to executing the CollectConfig operation, and is released upon completion of the CollectConfig operation. Note that a software lock is not applied to the optional device attributes and interfaces operations. This flag is read by the CollectConfig task upon execution.

Table C-1 DCPL Properties (continued)

/lockmanager/lockTimeoutInHours	8	integer, 1-168 hours	Timeout in hours for a lock held by a lock holder. If the lock holder does not free a lock within this time the lockmanager will automatically release the device lock.
/lockmanager/logLevel	SEVERE	selection	The log Level is the level at which logging is done for this component. These levels are identical to the logging levels defined for JDK1.4 logging package. The levels in descending order are: SEVERE (highest value) WARNING INFO CONFIG FINE FINER FINEST (lowest value).
/lockmanager/queueServicingInterval	100	integer, 10-2000 milliseconds	How often in milliseconds to service pending lock requests. A lower value decreases the average time it takes to get a lock at the expense of CPU processing overhead.
nbi Properties:			Northbound API (Nbi) component related defines.
/nbi/BackwardCompatible			Path for execQuery requests.
RecordNumber	false	The valid values are **true** and **false**.	For execQuery requests, the number embedded in the output class name include **Record** for the default, **false**, or **Record#1** for **true**.
/nbi/CompositeDir	<*vpnsc_home*>/ resources/java/ xml/com/cisco/ vpnsc/repository/ meta/xml/ composite	string	Path to composite XML files. Do not change it or the composite meta XML files will not be backed up.
/nbi/CustomerReportMetaDir	<*vpnsc_home*>/ resources/java/ xml/com/cisco/ vpnsc/repository/ meta/xml	string	Path to user defined report meta XML files. Do not change it or the report meta XML files will not be backed up.
/nbi/Formatter	com.cisco.vpnsc. nbi.io.NbiSimple Formatter	string	File path and name.
/nbi/Logger	com.cisco.vpnsc. nbi.util.NbiVpns cLogger	string	File path and name.
/nbi/MetaCheckInterval	300000	string	Set the time for next meta check to happen.

Table C-1 DCPL Properties (continued)

/nbi/MetaDir	<*vpnsc_home*>/resources/java/xml/com/cisco/vpnsc/repository/meta/xml	string	Path to meta XML files. Do not change it or the meta XML will not be backed up.
/nbi/ProvidedReportMetaDir	<*vpnsc_home*>/resources/java/xml/com/cisco/vpnsc/repository/meta/xml	string	Path to ISC provided report meta XML files. Do not change it or the report meta xml files will not be backed up.
/nbi/Reader	com.cisco.vpnsc.nbi.io.NbiSoapReader	string	File path and name.
/nbi/RequestParserMgr	com.cisco.vpnsc.nbi.parser.NbiRequestParserMgr	string	File path and name.
/nbi/SSLfilepath	<*vpnsc_home*>/bin/client.keystore	string	Path to client.keystore file for NBI SSL connections.
/nbi/SessionTimeout	1200000	string	Amount of time the session is valid. A session is the socket connection between the client and the NBI server through the Tomcat server.
/nbi/TransactionParser	com.cisco.vpnsc.nbi.parser.NbiWsdlParser	string	File path and name.
/nbi/Validation	true	The valid values are **true** and **false**.	Variable to enable validation of incoming Nbi API XML attributes.
/nbi/WaitTimeout	1200	integer	The time in seconds to wait for a Service Request to deploy.
/nbi/Writer/			
SoapEncapsulation	false	The valid values are **true** and **false**.	SoapEncapsulation.
/nbi/Writer	com.cisco.vpnsc.nbi.io.NbiSoapWriter	string	File path and name.
/nbi/logHandler	com.cisco.vpnsc.nbi.util.VpnscLogHandler	string	Custom log handler for nbi. This handler allows NBI to use alternate formatter from default one used by rest of ISC. In this case, NBI defaults to using SimpleFormatter which dumps simple output as opposed to XML output.

Table C-1 DCPL Properties (continued)

/nbi/logLevel	WARNING	selection	The log Level is the level at which logging is done for this component. These levels are identical to the logging levels defined for JDK1.4 logging pack age. The levels in descending order are: SEVERE (highest value) WARNING INFO CONFIG FINE FINER FINEST (lowest value).
notification Properties:			Event notification related defines.
/notification/Logger	com.cisco.vpnsc.nbi.util.NbiVpnscLogger	string	File path and name.
/notification/clientEnabled	false	The valid values are **true** and **false**.	Set to true for enabling the example event receiving servlet.
/notification/clientHost	<*master_server*>	string	TIBCO event client host.
/notification/clientMethod	/notification/servlet eventListener	string	TIBCO event client method.
/notification/clientPort	<*http_port*>	string	TIBCO event client port.
/notification/clientRegFile	<*vpnsc_home*>/resources/nbi/notification clientReg.txt	string	Client TIBCO event registration file name.
/notification/logFormatter	java.util.logging.SimpleFormatter	string	File path and name.
/notification/logHandler	com.cisco.vpnsc.nbi.util.VpnscLogHandler	string	Custom log handler.
/notification/logLevel	WARNING	selection	The log Level is the level at which logging is done for this component. These levels are identical to the logging levels defined for JDK1.4 logging package. The levels in descending order are: SEVERE (highest value) WARNING INFO CONFIG FINE FINER FINEST (lowest value).
/notification/password	cisco	string	Both username and password are same as the ones used for GUI login.
/notification/remotePassword		string	User password for remote system authentication, if required, for example, when LDAP is in use.
/notification/remoteUsername		string	Username for remote system authentication, if required, for example, when LDAP is in use.

Table C-1 DCPL Properties (continued)

/notification/username	admin	string	Both username and password are s same as the ones used for GUI login.
pal Properties:			The PAL Device interaction component. This runs within the Tomcat server and is responsible for running device interaction for the CFR to run the OAM troubleshooting workflows.
/pal/failureScenario			The system parameter that represents the current failure scenario. For use with the Canned Response mechanism for testing.
/pal/logHandler	com.cisco.mgmt.workflow.util.IscLogHandler		Set the PAL to use a custom handler for logging. The handler should log to a separate file and will format the log messages using the java.util.logging.SimpleFormatter instead of the ISC default XML formatting.
/pal/logLevel	INFO		The log Level is the level at which logging is done for this component. These levels are identical to the logging levels defined for JDK1.4 logging package. The levels in descending order are: SEVERE (highest value) WARNING INFO CONFIG FINE FINER FINEST (lowest value).
/pal/responseDir	/vob/ntg/dev/resources/pal/testnetwork		The base directory where the failure scenarios are held. Used by the canned response mechanism and transport for failure scenario testing.
repository Properties:			The component for Database related properties.
/repository/Concurrency/			To setup properties for re-try loop to avoid deadlock
NOICE_FACTOR	500	integer	Add random noise to each process that is being retried.
NO_OF_RETRIES	3	integer	Number of retries before throwing deadlock exception.
TIME_BASE	2	integer	The base number to calculate the wait time. For example, a value of 2 for this property and 3 retries means, the process will be retried every 2^0, 2^1, and 2^2 seconds.
/repository/IPAddressPool/			IP Address Pool Constants.
AGE_TIME	1440	integer	The Aging interval for released IP Address, in minutes. The default is 24 hours (1440 minutes).

Table C-1 DCPL Properties (continued)

RecoverIPAddrSleepInterval	60	integer, 10 - 144000 minutes	The time in minutes for recovering Aged IP addresses recovery service to wait between recovery cycles. The default is 60 minutes. Changing this value initiates the recovery process.
releaseAndReuseAgedAddresses	true	The valid values are **true** and **false**.	The default value is **false**. When the value is set to **true**, the user wants a manual allocation of the address in the aged address to succeed. When the value is set to **true**, the address is released from the Aged Pool and moved to the Allocated pool when manually allocated.
/repository/common			Repository common constants.
MCAST_SUBSUME_ALL_SRS	true	The valid values are **true** and **false**.	This property set at **true** indicates that the user wants all the MPLS VPN links of a VPN to be subsumed when Multicast is enabled for that VPN.
releaseAndReuseAgedAddresses	true	The valid values are **true** and **false**.	The default value is **false**. When the value is set to **true**, the address will be released from the Aged Pool and moved to the Allocated pool when manually allocated.
/repository/deviceConfig/		null	Configuration file related properties.
maxVersions	10	integer, 1-50	Maximum number of configuration files to be stored per device in the repository before older versions automatically get purged.
/repository/mlshare/			Share directory for both MPLS and L2VPN.
logLevel	SEVERE	selection	The log Level is the level at which logging is done for this component. These levels are identical to the logging levels defined for JDK1.4 logging package. The levels in descending order are: SEVERE (highest value) WARNING INFO CONFIG FINE FINER FINEST (lowest value).
/repository/persistence/			Properties for database.
Versions	5	integer	The number of maximum versions for a Versioning Persistent Objects.
catalog	directory	string	Catalog.
driver	*<db_driver>*	string	The class name for the driver.
initialConnections	1	integer, 1-20	Number of initial connections.
location	*<repository_home>*	string	The directory containing the repository.db and repository.log files.
password	sql	string	Password for opening a DB connection.

Table C-1 DCPL Properties (continued)

schema	DBA	string	Schema.
slaurl	jdbc:sybase:Tds:<*local_db_server*>:<*db_port_sla*>/?JCONNECT_VERSION=5&serviceName=sla	string	The url for opening a JDBC connection to the SLA database.
url	<*db_url*>	string	The url for opening a JDBC connection.
username	dba	string	User id to open a db connection.
/repository/rbac/			The component for RBAC User Access Model, user Authentication.
cache/isEternal	false	The valid values are **true** and **false**.	Specifies whether the elements in the RBAC cache are eternal, never expire. The value **true** indicates the elements in the cache are eternal and never expire. The default value **false** indicates the elements in the cache can expire.
cache/maxElementsInMemory	5000	integer, 1000 to 10000	Specifies the maximum number of elements in cache memory. Default: 5000.
overflowToDisk	false	The valid values are **true** and **false**.	Specifies whether to use disk to store cache.
cache/timeToIdleSeconds	120	integer, 60 to 1800 seconds	Specifies the default number of seconds for an element to live in cache from its last accessed or modified date. Default: 120 seconds.
cache/timeToLiveSeconds	300	integer, 100 to 3600 seconds	Specifies the default number of seconds for an element to live in cache from its creation date. Default: 300 seconds.
/repository/rbac/checkCreatorPermission Enabled	true	The valid values are **true** and **false**.	The creator of objects can give the permissions of Modify or Delete to others. If this flag is false, enable RBAC permission checkin.
/repository/rbac/checkPermissionEnabled	true	The valid values are **true** and **false**.	The creator of objects can give the permissions of Modify or Delete to others. If this flag is false, enable RBAC permission checkin.

Table C-1 DCPL Properties (continued)

/repository/rbac/enableAutologin	true	The valid values are **true** and **false**.	The property controls whether user may store login information in form of cookies on the computer from which the user connects. If enabled, automatic login, based on the cookie information is permitted. Also user is presented with a screen in which he or she can elect to store login information on the local user's computer. With this property set to false no autologin or options associated with it are available.
/repository/rbac/logLevel	CONFIG	selection	The log Level is the level at which logging is done for this component. These levels are identical to the logging levels defined for JDK1.4 logging package. The levels in descending order are: SEVERE (highest value) WARNING INFO CONFIG FINE FINER FINEST (lowest value).
/repository/rbac/partialQueryResult Expected	true	The valid values are **true** and **false**.	When checking Permission on a list of Persistent Objects, and the current user does not the specified permission to all the objects in the result list, partial results will be returned if this flag is true; Insufficient Permission exception will be generated if the flag is **false**.
/repository/rbac/webSessionTimeoutSec	1800	integer, 1 - 2,147,483,647	Timeout of inactive web client session in seconds. Default is 30 minutes.
/repository/ual/			User Access/Audit Log
cleanUALogs	true	The valid values are **true** and **false**.	Indicates whether to let the system automatically clean up UAL log entries based on ual.maxAgeInDays.
maxAgeInDays	30	integer	Maximum age of the User Access/Audit Logs in days after which the UALog Cleanup Service will delete them. if 0 then UALogs deletion is disabled even if cleanUALogs is set to true.
watchdog Properties:			All the servers in ISC are launched and managed by the Watchdog.
/watchdog/criticalServers		string	If any of these servers enters the disabled state, then it would mean that the system is NOT healthy. If this value is null/empty then every single server is critical.
/watchdog/diskspace/			Contains properties related to disk space monitoring.
dirsToMonitor		string	The directories (and ultimately the disks that contain them) to be monitored.

Table C-1 DCPL Properties (continued)

disksToMonitor		string	The disks to be monitored for space constraints.
emailRecipients	<*mailto*>	string	The comma separated list of e-mail addresses to which the disk space related e-mails should be sent out.
highWatermark	<*highwater*>	string	High watermark for the directories (disks) being monitored. The value should be a number followed by a < (for percent) or m or M (for Mbytes). These values should correspond to the available/free space on the disk. If the available disk space stabilizes above this value (after falling below the low watermark), an e-mail is sent to the addresses specified in the property watchdog.diskspace.emailRecipients.
lowWatermark	<*lowwater*>	string	Low watermark for the directories (disks) being monitored. The value should be be a number followed by a % (for percent) or m or M (for Mbytes). These values should correspond to the available/free space on the disk. If the available disk space falls below this value, an e-mail is sent to the addresses specified in the property watchdog.diskspace.emailRecipients.
sleepInterval	60000	integer, 30000-300000 milliseconds	Time between two status checks for disk space limits in milliseconds.
/watchdog/group/			Group.
database_users	scheduler httpd	string	The servers that access database.
/watchdog/groups	database_users	string	The space separated list of different groups in the system.
/watchdog/heartbeat/			Properties related to watchdog heartbeat mechanism are specified here.
period	120000	integer, 30000-86400000 milliseconds	The minimum time between each heartbeat request in milliseconds.
period_poller	60000	integer, 30000-86400000 milliseconds	The minimum time between each heartbeat request for dbpoller and nspoller in milliseconds.
sendEvents	false	The valid values are **true** and **false**.	If set to true, watchdog sends out TIBCO events every time a heartbeat succeeds or fails. If set to false, no such events will be sent.

Table C-1 DCPL Properties (continued)

startDelay	5000	integer, 0-60000 milliseconds	Time to wait before making the first heartbeat request in milliseconds.
timeout	3000	integer, 1000-600000 milliseconds	The period of time before which response for heartbeat request should be received by the watchdog, in milliseconds.
wds/			Heartbeat properties for intra-watchdog communication.
delay	5000	integer, 1000-60000 milliseconds	The period in between heartbeats. (from master watchdog to slave watchdog and vice-versa) in milliseconds.
initDelay	1000	integer, 1000-5000 milliseconds	The initial period of time for which the heartbeat thread waits before trying for a heartbeat after a watchdog registers with the MasterWatchdog, in milliseconds.
masterReconnectAttemptDelay	2000	integer, 100-60000 milliseconds	The sleep time between two successive attempts by a slave watchdog to reconnect to master watchdog, in milliseconds.
maxAllowedMisses	3	integer	The maximum number of consecutive misses that a watchdog should miss for the master to consider it inactive or unregistered.
maxAttemptsForMasterReconnect	500	integer	Once the slave watchdog loses connection with the master, it will try this many times to try and establish the connection. If it cannot re-establish a connection with the master even after making these many attempts, it shuts itself down. Between attempts, it sleeps **watchdog.heartbeat.wds.masterReconnectAttemptDelay** time. The value for this property should be specified in milliseconds. A value of 0 indicates that the slave watchdog has no upper limit on the number of reconnect attempts.
/watchdog/java/			Java.
flags	-XX:+UseAltSigs	string	Any other flags to be passed on to **java.**
vmtype	-server	string	The flag to be passed on to java (-server or -client).

Table C-1 DCPL Properties (continued)

/watchdog/logLevel	FINEST	selection	The log Level is the level at which logging is done for this component. These levels are identical to the logging levels defined for JDK1.4 logging package. The levels in descending order are: SEVERE (highest value) WARNING INFO CONFIG FINE FINER FINEST (lowest value).
/watchdog/server/	httpd nspoller dbpoller dispatcher worker scheduler lockmanager cornerstonebridge	string	Server.
cnsserver/			Monitors CNS events from IE2100 boxes. Communication between client and server is completely handled using TIBCO events.
heartbeat/			Heartbeat related properties.
startDelay	10000	integer, 0-60000 milliseconds	Time to wait before making the first heartbeat request in milliseconds.
timeout	3000	integer, 1000-600000 milliseconds	The period of time before which response for heartbeat request should be received by the watchdog, in milliseconds.
java/			Java attributes for this server.
flags		string	Any additional java flags specific to this server. If the value is changed, watchdog restart is required for the new value to take effect.
class	com.cisco.vpnsc.watchdog.servers.WDCnsServer	string	Heartbeat Handler - Checks for valid TIBCO Connection.
cmd	java com.cisco.vpnsc.cns.CnsServer	string	Implementation to monitor CNS events from IE2100 boxes.
dependencies	dbpoller	string	Dependencies.
logLevel	CONFIG	selection	The log Level is the level at which logging is done for this component. These levels are identical to the logging levels defined for JDK1.4 logging package. The levels in descending order are: SEVERE (highest value) WARNING INFO CONFIG FINE FINER FINEST (lowest value).

Table C-1 DCPL Properties (continued)

dbpoller/			This server keeps polling the database to see if it is functional.
class	com.cisco.vpnsc.watchdog.servers.WDDatabase	string	Name of class responsible for getting heartbeats.
connectionextend	5	integer, 1-15	For Oracle RAC failover, increase this value to make sure the failover happens before dbpoller stops.
logLevel	CONFIG	selection	The log Level is the level at which logging is done for this component. These levels are identical to the logging levels defined for JDK1.4 logging package. The levels in descending order are: SEVERE (highest value) WARNING INFO CONFIG FINE FINER FINEST (lowest value).
select	select id from vpnsc_host	string	SQL select statement to issue when pinging the database.
discovery/			Handles various ISC Discovery workflow related tasks.
class	com.cisco.vpnsc.discovery.engine.server.DiscoveryServer	string	Heartbeat Handler.
cmd	java com.cisco.vpnsc.discovery.engine.server.DiscoveryImpl	string	Implementation of the Discovery work interface.
dependencies	dbpoller	string	dependencies
heartbeat/			Heartbeat related properties.
startDelay	10000	integer, 0-60000 milliseconds	Time to wait before making the first heartbeat request in milliseconds.
timeout	3000	integer, 1000-60000 milliseconds	The period of time before which response for heartbeat request should be received by the watchdog, in milliseconds. To discover large networks with a complex topology, we recommend you reset this property to 180000 milliseconds (3 minutes).
java/			Java attributes for this server

Table C-1 DCPL Properties (continued)

flags		string	Any additional java flags specific to this server. If the value is changed, watchdog restart is required for the new value to take effect. To discover large networks with a complex topology, we recommend you reset this property to -Xmx3072m -XX:PermSize=256m -XX:MaxPermSize=512m.
logLevel	CONFIG	selection	The log Level is the level at which logging is done for this component. These levels are identical to the logging levels defined for JDK1.4 logging package. The levels in descending order are: SEVERE (highest value) WARNING INFO CONFIG FINE FINER FINEST (lowest value).
dispatcher/			Dispatcher service of the Distribution framework.
app_args	Dispatcher com.cisco.vpnsc.dist.vpnsc.VpnscDispatcherImpl	string	Args to the class that starts this service.
class	com.cisco.vpnsc.watchdog.servers.WDDispatcher	string	The class that proxies this service for the watchdog.
cmd	java com.cisco.vpnsc.watchdog.ext.ServiceLauncherImpl	string	Command to start the server.
dependencies	dbpoller nspoller	string	The other services that this service depends on Heartbeat related properties.
heartbeat/			
startDelay	45000	integer, 0-60000 milliseconds	Time to wait before making the first heartbeat request in milliseconds.
timeout	3000	integer, 1000-60000 milliseconds	The period of time before which response for heartbeat request should be received by the watchdog, in milliseconds.
java/			Java attributes for this server
flags		string	Any additional java flags specific to this server. If the value is changed, watchdog restart is required for the new value to take effect.

Table C-1 DCPL Properties (continued)

logLevel	CONFIG	selection	The log Level is the level at which logging is done for this component. These levels are identical to the logging levels defined for JDK1.4 logging package. The levels in descending order are: SEVERE (highest value) WARNING INFO CONFIG FINE FINER FINEST (lowest value).
httpd/		httpd	httpd
class	com.cisco.vpnsc.watchdog.servers.WDHttpd	string	Class.
cmd	<*vpnsc_home*>/bin/tomcat.sh start fg	string	The command to start httpd on this host.
dependencies	dbpoller	string	Dependencies.
heartbeat/			Heartbeat.
port	<*http_port*>	integer	The port on which httpd should run.
startDelay	45000	integer, 0-60000 milliseconds	Time to wait before making the first heartbeat request in milliseconds.
timeout	10000	integer, 1000-600000 milliseconds	The period of time before which response for heartbeat request should be received by the watchdog, in milliseconds.
url	http://localhost:<*http_port*>/isc/about.htm	string	url
logLevel	CONFIG	selection	The log Level is the level at which logging is done for this component. These levels are identical to the logging levels defined for JDK1.4 logging package. The levels in descending order are: SEVERE (highest value) WARNING INFO CONFIG FINE FINER FINEST (lowest value).
lockmanager/			Component that handles locking.
class	com.cisco.vpnsc.watchdog.servers.WDLockManager	string	Class that keeps track of lockmanager heartbeats.
cmd	java com.cisco.vpnsc.lockmanager.LockManagerImpl	string	Command that starts up the lockmanager.
dependencies	nspoller	string	Lock Manager depends on the NS.

Table C-1 DCPL Properties (continued)

heartbeat/			Heartbeat related properties.
startDelay	10000	integer, 0-60000 milliseconds	Time to wait before making the first heartbeat request in milliseconds.
timeout	3000	integer, 1000-600000 seconds	The period of time before which response for heartbeat request should be received by the watchdog, in milliseconds.
java/			Java attributes for this server.
flags		string	Any additional java flags specific to this server. If the value is changed, watchdog restart is required for the new value to take effect.
logLevel	CONFIG	selection	The log Level is the level at which logging is done for this component. These levels are identical to the logging levels defined for JDK1.4 logging package. The levels in descending order are: SEVERE (highest value) WARNING INFO CONFIG FINE FINER FINEST (lowest value).
maxQuickDieCount	3	integer	The maximum number of times a server can die consecutively without having a successful heartbeat. If this number is exceeded, the server is marked as disabled.
nspoller/			This server polls the NameServer to see if it is running.
class	com.cisco.vpnsc.watchdog.servers.WDNameServer	string	Class.
logLevel	CONFIG	selection	The log Level is the level at which logging is done for this component. These levels are identical to the logging levels defined for JDK1.4 logging package. The levels in descending order are: SEVERE (highest value) WARNING INFO CONFIG FINE FINER FINEST (lowest value).
rgserver/			TEM server for the TEM tunnel generation algorithm.
heartbeat/			
rgport		string	The port on which rgserver should run.
startDelay	45000	integer, 0-60000 milliseconds	Time to wait before making the first heartbeat request in milliseconds.

Table C-1 DCPL Properties (continued)

timeout	3000	integer, 1000-600000 milliseconds	The period of time before which response for heartbeat request should be received by the watchdog, in milliseconds.
class	com.cisco.vpnsc.watchdog.servers.WDRGServer	string	Class.
cmd	rgserver.sh	string	Command to start the rgserver.
dependencies	httpd	string	Servers that must be functioning for this server to function normally.
logLevel	CONFIG	selection	The log Level is the level at which logging is done for this component. These levels are identical to the logging levels defined for JDK1.4 logging package. The levels in descending order are: SEVERE (highest value) WARNING INFO CONFIG FINE FINER FINEST (lowest value).
scheduler/			Scheduler.
class	com.cisco.vpnsc.watchdog.servers.WDScheduler	string	Class.
cmd	java com.cisco.vpnsc.scheduler.Scheduler	string	Command to start the scheduler.
dependencies	dbpoller worker	string	Dependencies.
heartbeat/			Heartbeat related properties.
startDelay	30000	integer, 0-60000 milliseconds	Time to wait before making the first heartbeat request in milliseconds.
timeout	3000	integer, 1000-600000 milliseconds	The period of time before which response for heartbeat request should be received by the watchdog, in milliseconds.
java/			Java attributes for this server.
flags		string	Any additional java flags specific to this server. If the value is changed, watchdog restart is required for the new value to take effect.
logLevel	CONFIG	selection	The log Level is the level at which logging is done for this component. These levels are identical to the logging levels defined for JDK1.4 logging package. The levels in descending order are: SEVERE (highest value) WARNING INFO CONFIG FINE FINER FINEST (lowest value).

Table C-1 DCPL Properties (continued)

startTimeout	240000	integer, 5000-600000	The timeout for the initial heartbeat response. The first heartbeat should happen within this time.
worker/			Worker service of the distribution framework.
app_args	Worker com.cisco.vpnsc.dist.WorkerImpl, com.cisco.vpnsc.sla.sql.SlaMaintenanceService, com.cisco.vpnsc.repository.ual.UALCleanupServiceImpl, com.cisco.vpnsc.license.LicenseSynchronize, com.cisco.vpnsc.cleanup.TaskLogCleanupService, com.cisco.vpnsc.cleanup.TempFileCleanupService, com.cisco.vpnsc.cleanup.RuntimeTaskCleanupService”	string	Arguments to the class specified in the **cmd** property.
class	com.cisco.vpnsc.watchdog.servers.WDWorker	string	The server class that proxies Worker service for the watchdog.
cmd	java com.cisco.vpnsc.watchdog.ext.ServiceLauncherImpl	string	Command to start the worker.
dependencies	nspoller	string	Servers that have to be functioning for this server to function normally.
heartbeat/			Heartbeat related properties.
startDelay	45000	integer, 0-60000 milliseconds	Time to wait before making the first heartbeat request in milliseconds.
timeout	3000	integer, 1000-600000 milliseconds	The period of time before which response for heartbeat request should be received by the watchdog, in milliseconds.
java/			Java attributes for this server.

Table C-1 DCPL Properties (continued)

flags	-Xmx512m -Xbootclasspath/p:<*vpnsc_home*>/thirdparty/jar/AdventNetSnmp3_3.2.jar:<*vpnsc_home*>/thirdparty/jar/cryptix32.jar -Dcom.cisco.insmbu.templatemgr.backend.PropFile=<*vpnsc_home*>/resources/templatesystem/Template.properties	string	Any additional java flags specific to this server. If the value is changed, watchdog restart is required for the new value to take effect.
logLevel	CONFIG	selection	The log Level is the level at which logging is done for this component. These levels are identical to the logging levels defined for JDK1.4 logging package. The levels in descending order are: SEVERE (highest value) WARNING INFO CONFIG FINE FINER FINEST (lowest value).
/watchdog/serverStatus/			The properties related to the server status monitoring function provided by the watchdog are specified here.
emailRecipients	<*mailtoRestart*>	string	Comma separated list of e-mail addresses to which notices about server state changes should be e-mailed
stableTime	60000	integer, 20000-300000 milliseconds	Time in milliseconds that has to pass before a server's status can be considered stable (for the purpose of sending out a server status e-mail notification).
/watchdog/servers	httpd nspoller dbpoller dispatcher worker scheduler lockmanager cornerstonebridge	string	Server.
/watchdog/waitDelay	3000	integer, 20000-300000 milliseconds	The time period for which the wait() calls in watchdog wait, before checking the wait condition, in milliseconds.
xml Properties:			The component for XML-based properties.

Table C-1 DCPL Properties (continued)

/xml/ValidatorRule			
filepath	<*vpnsc_home*>/resources/java/classes/common/com/cisco/vpnsc/util/validator/xml	string	Validator rules file path and name.
/xml/queries/			Properties for RepQueryLoader.
filepath	<*vpnsc_home*>/resources/java/xml/com/cisco/vpnsc/repository/Queries.xml	string	File path and name.