Protocol Translation Software Configuration Guide for Cisco 1000 Series Connected Grid Routers (Cisco IOS)

January 2014

OL-31250-01

This chapter provides details about configuring Protocol Translation on the Cisco 1000 Series Connected Grid Router (hereafter referred to as the CGR 1000). Protocol Translation operates within a Supervisory Control and Data Acquisition (SCADA) system.

This chapter includes the following sections:

Information About SCADA

SCADA refers to a control and management system employed in industries such as water management, electric power, and manufacturing. A SCADA system collects data from various types of equipment within the system and forwards that information back to a Control Center for analysis. Generally, individuals located at the Control Center monitor the activity on the SCADA system and intervene when necessary.

The Remote Terminal Unit (RTU) acts as the primary control system within a SCADA system. RTUs are configured to control specific functions within the SCADA system, which can be modified as necessary through a user interface.

Cisco Systems, Inc.
www.cisco.com

Role of the CGR 1000

In the network, the Control Center always serves as the master in the network when communicating with the CGR 1000. The CGR 1000 serves as a proxy master station for the Control Center when it communicates with the RTU.

The CGR 1000 provides Protocol Translation to serve as a SCADA gateway to do the following:

- Receive data from RTUs and relay configuration commands from the Control Center to RTUs.
- Receive configuration commands from the Control Center and relay RTU data to the Control Center.
- Terminate incoming requests from the Control Center, when an RTU is offline.

The CGR 1000 performs Protocol Translation for the following protocols:

- IEC 60870 T101 to/from IEC 60870 T104
- DNP3 serial to DNP3 IP

Key Terms

The following terms are relevant when you configure the T101 and T104 protocol stacks on the CGR 1000:

- Channel–A channel is configured on each CGR 1000 serial port interface to provide a connection to a single RTU for each IP connection to a remote Control Center**.** Each connection transports a single T101 (RTU) or T104 (Control Center) protocol stack.
- Link Address–Refers to the device or station address.
- Link Mode (Balanced and Unbalanced)–Refers to the modes of data transfer.
  - An Unbalanced setting refers to a data transfer initiated from the master.
  - A Balanced setting refers to either a master or slave initiated data transfer.
- Sector–Refers to a single RTU within a remote site.
- Sessions–Represents a single connection to a remote site.

The following terms are relevant when you configure the DNP3 protocol stacks on the CGR 1000:

- Channel–A channel is configured on each CGR 1000 serial port interface to provide a connection to a single RTU for each IP connection to a remote Control Center**.** Each connection transports a single DNP3 serial (RTU) or DNP3 IP (Control Center) protocol stack.
- Link Address–Refers to the device or station address.
- Sessions–Represents a single connection to a remote site.

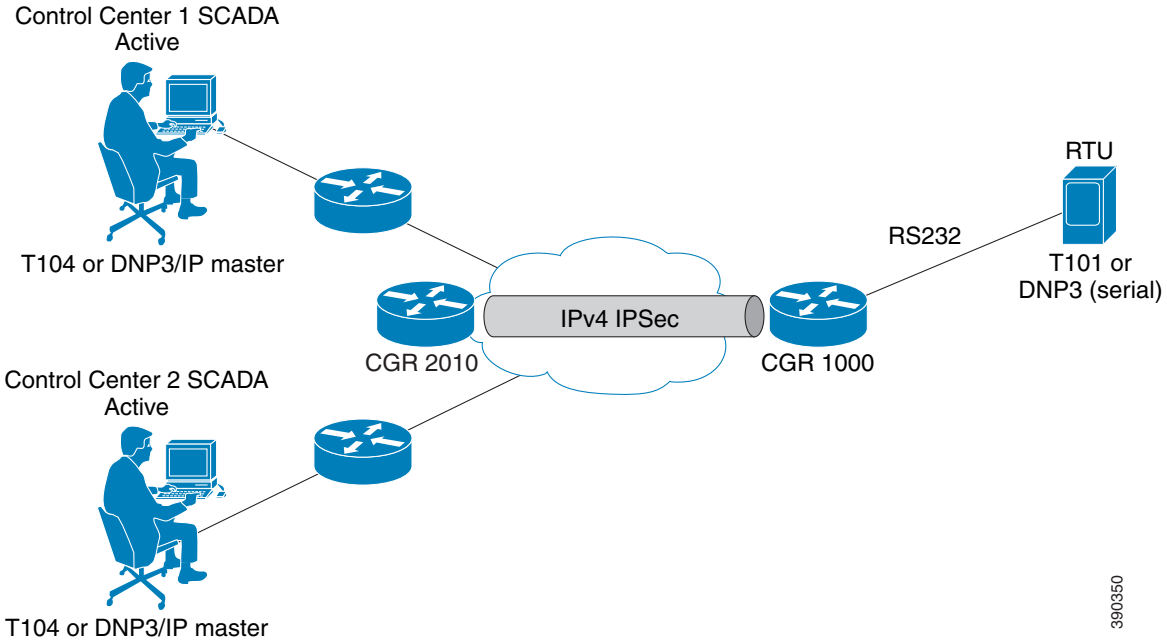
Protocol Translation Application

In the example shown in Figure 1, the CGR 1000 (installed within a secondary substation of the Utility Network) employs Protocol Translation using an IPSec tunnel to provide secure, end-to-end connectivity between Control Centers and RTUs within a SCADA System.

The CGR 1000 connects to the RTU (slave) through an RS232 or RS485 connection. To protect the traffic when forwarded over public infrastructures (for example, cellular), the CGR 1000 forwards SCADA data from the RTU to the Control Center in the SCADA system through an IPSec tunnel (FlexVPN site-to-site or hub and spoke). The IPSec tunnel protects all traffic between the CGR 1000 and

the Head-end aggregation router. SCADA traffic can be inspected through an IPS device (such as the CGR 2010) positioned in the path of the SCADA traffic before it is forwarded to the proper Control Center.

Figure 1 Cisco Connected Grid Routers Providing Connectivity and Security within a SCADA System



Prerequisites

- The ipbasek9 technology package license is required for using the Protocol Translation feature.
- RTUs must be configured and operating in the network.

For each RTU that connects to the CGR 1000, you will need the following information for T101/T104:

- Channel information
  - Channel name
  - Connection type: serial
  - Link transmission procedure setting: unbalanced or balanced
  - Address field of the link (number expressed in octets)
- Session information
  - Session name
  - Size of common address of Application Service Data Unit (ASDU) (number expressed in octets)
  - Cause of transmission (COT) size (number expressed in octets)
  - Information object address (IOA) size (number expressed in octets)
- Sector information
  - Sector name

  - ASDU address (number expressed in octets)

For each RTU that connects to the CGR 1000, you will need the following information for DNP3:

- Channel information
  - Channel name
  - Connection type: serial
  - Link address
- Session information
  - Session name

Guidelines and Limitations

Each channel supports only one session.

Each session supports only one sector.

Default Settings

Parameters	Default
T101/T104	
Role for T101	Master
Role for T104	Slave
DNP3	
Unsolicited Response (DNP3-serial)	Not enabled
Send Unsolicited Message (DNP3-IP)	Enabled

Configuring Protocol Translation

This section includes the following topics:

Note Before making changes to Protocol Translation configuration, stop the Protocol Translation Engine as described in the "Starting and Stopping the Protocol Translation Engine" section on page 14.

Enabling the CGR 1000 Serial Port and SCADA Encapsulation

Before you can enable and configure Protocol Translation on the CGR 1000, you must enable the serial port on the CGR 1000 and enable SCADA encapsulation on that port.

BEFORE YOU BEGIN

Determine availability of the serial port on the CGR 1000.

DETAILED STEPS

	Command	Purpose
Step 1	**configure terminal**	Enters global configuration mode.
Step 2	**interface async** *slot/port*	Enters interface command mode for the serial slot/port. **Note** The slot/port configuration for the serial port can be 1/1 or 1/2.
Step 3	**no shutdown**	Brings up the port, administratively.
Step 4	**encapsulation scada**	Enables encapsulation on the serial port for SCADA protocols.

EXAMPLE

This example shows how to enable serial port 1/1 and enable encapsulation on that port to support SCADA:

```
router# configure terminal
router(config)# interface async 1/1
router (config-if)# no shutdown
router (config-if)# encapsulation scada
```

Configuring T101 and T104 Protocol Stacks

You can configure the T101 and T104 protocol stacks, which allow end-to-end communication between Control Centers (T104) and RTUs (T101) within a SCADA system. If you are using DNP3, see the "Configuring the DNP3 Protocol Stacks" section on page 10.

Configuring the T101 Protocol Stack

Configure the channel, session, and sector parameters for the T101 protocol stack.

BEFORE YOU BEGIN

Ensure that you have gathered all the required configuration information. See the "Prerequisites" section on page 3.

Enable the serial port and SCADA encapsulation. See the "Enabling the CGR 1000 Serial Port and SCADA Encapsulation" section on page 5.

DETAILED STEPS

	Command	Purpose
Step 1	**configure terminal**	Enters global configuration mode.
Step 2	**scada-gw protocol t101**	Enters configuration mode for the T101 protocol.
Step 3	**channel** *channel_name*	Enters channel configuration mode for the T101 protocol. *channel_name*–Identifies the channel on which the serial port of the CGR 2010 communicates to the RTU. **Note** When the entered channel name does not already exist, the router creates a new channel. Entering the **no** form of this command deletes an existing channel. However, all sessions must be deleted before you can delete a channel.
Step 4	**role master**	Assigns the master role to the T101 protocol channel (default).
Step 5	**link-mode {balanced | unbalanced}**	Configures the link-mode as either balanced or unbalanced. unbalanced–Refers to a data transfer initiated from the master. balanced–Refers to either a master or slave data transfer.
Step 6	**link-addr-size {none | one | two}**	Defines the link address size in octets.
Step 7	**bind-to-interface serial** *slot/port*	Defines the CGR 2010 serial interface on which the system sends its T101 protocol traffic. *slot*–Value of 1. *port*–Value of 1 or 2.
Step 8	**exit**	Ends configuration of the channel and exits channel configuration mode. Saves all settings.
Step 9	**session** *session_name*	Enters session configuration mode and assigns a name to the session.
Step 10	**attach-to-channel** *channel_name*	Attaches the session to the channel. Enter the same channel name that you entered in Step 3. *channel_name*–Identifies the channel.
Step 11	common-addr-size **{one | two | three}**	Defines the common address size in octets.
Step 12	cot size **{one | two | three}**	Defines the cause of transmission such as spontaneous or cyclic data schemes in octets.
Step 13	info-obj-addr-size **{one | two | three}**	Defines the information object element address size in octets.
Step 14	**link-addr-size {one | two | three}**	Defines the link address size in octets.
Step 15	**link-addr** *link_address*	Refers to the link address of the RTU. **Note** The link address entered here must match the value set on the RTU to which the serial port connects. *link_address*–Value of 1 or 2.

	Command	Purpose
Step 16	**exit**	Exits session configuration mode.
Step 17	**sector** *sector_name*	Enters sector configuration mode and assigns a name to the sector for the RTU. *sector_name*–Identifies the sector.
Step 18	**attach-to-session** *session_name*	Attaches the RTU sector to the session. Enter the same session name that you entered in Step 9. *session_name*–Identifies the session.
Step 19	**asdu-addr** *asdu_address*	Refers to the ASDU structure address of the RTU.
Step 20	**exit**	Exits sector configuration mode.
Step 21	**exit**	Exits protocol configuration mode.

EXAMPLE

This example shows how to configure the parameters for the T101 protocol stack for *RTU_10*:

```
router# configure terminal
router(config)# scada-gw protocol t101
router(config-t101)# channel rtu_channel
router(config-t101-channel)# role master
router(config-t101-channel)# link-mode unbalanced
router(config-t101-channel)# link-addr-size one
router(config-t101-channel)# bind-to-interface serial 1/1
router(config-t101-channel)# exit
router(config-t101)# session rtu_session
router(config-t101-session)# attach-to-channel rtu_channel
router(config-t101-session)# common-addr-size two
router(config-t101-session)# cot-size one
router(config-t101-session)# info-obj-addr-size two
router(config-t101-session)# link-addr 3
router(config-t101-session)# exit
router(config-t101)# sector rtu_sector
router(config-t101-sector)# attach-to-session rtu_session
router(config-t101-sector)# asdu-addr 3
router(config-t101-sector)# exit
router(config-t101)# exit
router(config)#
```

Configuring the T104 Protocol Stack

Follow this procedure for each Control Center that you want to connect to over a T104 protocol.

BEFORE YOU BEGIN

Ensure that you have gathered all the required configuration information. See the "Prerequisites" section on page 3.

Enable the serial port and SCADA encapsulation. See the "Enabling the CGR 1000 Serial Port and SCADA Encapsulation" section on page 5.

DETAILED STEPS

	Command	Purpose
Step 1	**configure terminal**	Enters configuration mode.
Step 2	**scada-gw protocol t104**	Enters configuration mode for the T104 protocol.
Step 3	**channel** *channel_name*	Enters channel configuration mode for the T104 protocol. *channel_name*–Identifies the channel on which the router communicates with the Control Center. **Note** When the entered channel name does not already exist, the router creates a new channel. Entering the **no** form of this command deletes an existing channel. However, all sessions must be deleted before you can delete a channel.
Step 4	**k-value** *value*	Sets the maximum number of outstanding Application Protocol Data Units (APDUs) for the channel. **Note** An APDU incorporates the ASDU and a control header. *value*–Range of values from 1 to 32767. Default value is 12 APDUs.
Step 5	**w-value** *value*	Sets the maximum number of APDUs for the channel. *value*–Range of values from 1 to 32767. Default value is 8 APDUs.
Step 6	**t0-timeout** *value*	Defines the t0-timeout value for connection establishment of the T104 channel.
Step 7	**t1-timeout** *value*	Defines the t1-timeout value for send or test APDUs on the T104 channel.
Step 8	**t2-timeout** *value*	Defines the t2-timeout value for acknowledgements when the router receives no data message. **Note** The t2 value must always be set to a lower value than the t1 value on the T104 channel.
Step 9	**t3-timeout** *value*	Defines the t3-timeout value for sending s-frames in case of a long idle state on the T104 channel. **Note** The t3 value must always be set to a higher value than the t1 value on the T104 channel.
Step 10	**tcp-connection primary local-port** *port_number*	In a configuration where there are redundant Control Centers, sets the value for the primary Control Center as defined on the primary Control Center.
Step 11	**tcp-connection secondary local-port** *port_number*	In a configuration where there are redundant Control Centers, sets the value for the secondary Control Center as defined on the primary Control Center.
Step 12	**exit**	Exits channel configuration mode.

	Command	Purpose
Step 13	**session** *session_name*	Enters session configuration mode and assigns a name to the session. *session_name*–Use the same name that you assigned to the channel in Step 3.
Step 14	**attach-to-channel** *channel_name*	Defines the name of the channel that transports the session traffic.
Step 15	cot size **{one | two | three}**	Defines the cause of transmission (cot), such as spontaneous or cyclic data schemes in octets.
Step 16	**exit**	Exits session configuration mode.
Step 17	**sector** *sector_name*	Enters sector configuration mode and assigns a name to the sector for the Control Center.
Step 18	**attach-to-session** *session_name*	Attaches the Control Center sector to the channel. *session_name*–Use the same name that you assigned to the channel in Step 3.
Step 19	**asdu-addr** *asdu_address*	Refers to the ASDU structure address. Value entered here must match the ASDU value on the RTU. *asdu_address*–Value of 1 or 2.
Step 20	**map-to-sector** *sector_name*	Maps the Control Center (T104) sector to the RTU (T101) sector.
Step 21	Return to Step 1.	Repeat all steps in this section for each Control Center active in the network.

EXAMPLE

This example shows how to configure the parameters for the T104 protocol stack on *Control Center 1* and *Control Center 2,* both of which are configured as *masters*, and how to map the T104 sector to the T101 sector:

To configure Control Center 1 (*cc_master1*), enter the following commands:

```
router# configure terminal
router(config)# scada-gw protocol t104
router(config-t104)# channel cc_master1
router(config-t104-channel)# k-value 12
router(config-t104-channel)# w-value 8
router(config-t104-channel)# t0-timeout 30
router(config-t104-channel)# t1-timeout 15
router(config-t104-channel)# t2-timeout 10
router(config-t104-channel)# t3-timeout 30
router(config-t104-channel)# tcp-connection primary local-port 2050
router(config-t104-channel)# tcp-connection secondary local-port 2051
router(config-t104-channel)# exit
router(config-t104)# session cc_master1
router(config-t104-session)# attach-to-channel cc_master1
router(config-t104-session)# cot-size two
router(config-t104-session)# exit
router(config-t104)# sector cc_master1-sector
router(config-t104-sector)# attach-to-session cc_master1
router(config-t104-sector)# asdu-adr 3
router(config-t104-sector)# map-to-sector rtu_sector
router(config-t104)# exit
router(config)#
```

To configure Control Center 2 (*cc_master2*), enter the following commands:

```
router(config)# scada-gw protocol t104
router(config-t104)# channel cc_master2
router(config-t104-channel)# k-value 12
router(config-t104-channel)# w-value 8
router(config-t104-channel)# t0-timeout 30
router(config-t104-channel)# t1-timeout 15
router(config-t104-channel)# t2-timeout 10
router(config-t104-channel)# t3-timeout 30
router(config-t104-channel)# tcp-connection primary local-port 2060
router(config-t104-channel)# tcp-connection secondary local-port 2061
router(config-t104-channel)# exit
router(config-t104)# session cc_master2
router(config-t104-session)# attach-to-channel cc_master2
router(config-t104-session)# cot-size two
router(config-t104-session)# exit
router(config-t104)# sector cc_master2-sector
router(config-t104-sector)# attach-to-session cc_master2
router(config-t104-sector)# asdu-adr 3
router(config-t104-sector)# map-to-sector rtu_sector
router(config-t104-sector)# exit
router(config-t104)# exit
router(config)#
```

Configuring the DNP3 Protocol Stacks

You can configure the DNP3 serial and DNP3 IP protocol stacks, which allow end-to-end communication between Control Centers and RTUs within a SCADA system.

Configuring DNP3 Serial

Configure the channel and session parameters for the DNP serial communication with an RTU.

BEFORE YOU BEGIN

Ensure that you have gathered all the required configuration information. See the "Prerequisites" section on page 3.

Enable the serial port and SCADA encapsulation. See the "Enabling the CGR 1000 Serial Port and SCADA Encapsulation" section on page 5.

DETAILED STEPS

	Command	Purpose
Step 1	**configure terminal**	Enters global configuration mode.
Step 2	**scada-gw protocol dnp3-serial**	Enters configuration mode for the DNP3 serial protocol.

	Command	Purpose
Step 3	**channel** *channel_name*	Enters channel configuration mode for the DNP3 serial protocol. *channel_name*–Identifies the channel on which the CGR 2010 serial port communicates to the RTU. **Note** When the entered channel name does not already exist, the router creates a new channel. Entering the **no** form of this command deletes an existing channel. However, all sessions must be deleted before you can delete a channel.
Step 4	**bind-to-interface async** *slot/port*	Defines the CGR 2010 async interface on which the system sends its DNP3 protocol traffic. *slot*–Value of 1. *port*–Value of 1 or 2.
Step 5	**link-addr source** *source_address*	Refers to the link address of the master. *source_address*–Range of values from 1 to 65535.
Step 6	**unsolicited-response enable**	(Optional) Allows unsolicited responses. Entering the **no** form of this command disables unsolicited responses. The default is disabled.
Step 7	**exit**	Ends configuration of the channel and exits channel configuration mode. Saves all settings.
Step 8	**session** *session_name*	Enters session configuration mode and assigns a name to the session. **Note** When the entered session name does not already exist, the router creates a new session. Entering the **no** form of this command deletes an existing session.
Step 9	**attach-to-channel** *channel_name*	Attaches the session to the channel. **Note** Enter the same channel name that you entered in Step 3. *channel_name*–Identifies the channel.
Step 10	**link-addr dest** *destination_address*	Refers to the link address of the slave. *destination_address*–Range of values from 1 to 65535.
Step 11	**exit**	Exits session configuration mode.
Step 12	**exit**	Exits protocol configuration mode.

EXAMPLE

This example shows how to configure the parameters for the DPN3-serial protocol stack:

```
router# configure terminal
router(config)# scada-gw protocol dnp3-serial
router(config-dnp3s)# channel rtu_channel
router(config-dnp3s-channel)# bind-to-interface async 1/1
router(config-dnp3s-channel)# link-addr source 3
router(config-dnp3s-channel)# unsolicited-response enable
```

```
router(config-dnp3s-channel)# exit
router(config-dnp3s)# session rtu_session
router(config-dnp3s-session)# attach-to-channel rtu_channel
router(config-dnp3s-session)# link-addr dest 3
router(config-dnp3s-session)# exit
router(config-dnp3s)# exit
router(config)#
```

Configuring DNP3 IP

Follow the steps below for the Control Center that you want to connect to over DNP3 IP. For redundancy, you can create multiple connections that share the same session configuration under the same session.

BEFORE YOU BEGIN

Ensure that you have gathered all the required configuration information. See the "Prerequisites" section on page 3.

Enable the serial port and SCADA encapsulation. See the "Enabling the CGR 1000 Serial Port and SCADA Encapsulation" section on page 5.

DETAILED STEPS

	Command	Purpose
Step 1	**configure terminal**	Enters configuration mode.
Step 2	**scada-gw protocol dnp3-ip**	Enters configuration mode for the DNP-IP protocol.
Step 3	**channel** *channel_name*	Enters channel configuration mode for the DNP-IP protocol. *channel_name*–Identifies the channel on which the router communicates with the Control Center. **Note** When the entered channel name does not already exist, the router creates a new channel. Entering the **no** form of this command deletes an existing channel. However, all sessions must be deleted before you can delete a channel.
Step 4	**link-addr dest** *destination_address*	Refers to the link address of the master. *destination_address*–Range of values from 1 to 65535.
Step 5	**send-unsolicited-msg enable**	(Optional) Allow unsolicited messages. The default is enabled.

	Command	Purpose
Step 6	**tcp-connection local-port** [**default** | *local_port*] **remote-ip** [**any** | *remote_ip* | *remote_subnet*]	Configures the local port number and remote IP address for the TCP connection: • **default**–20000. • *local_port*–Range of values from 2000 to 65535. • any–Any remote hosts 0.0.0.0/0 • *remote_ip*–Single host: A.B.C.D • *remote_subnet*–Subnet: A.B.C.D/LEN **Note** Every <local-port, remote-ip> tuple must be unique per channel. If *remote_subnet* is specified, when two channels have the same local ports, the remote subnets cannot overlap each other.
Step 7	**exit**	Exits channel configuration mode.
Step 8	**session** *session_name*	Enters session configuration mode and assigns a name to the session. **Note** When the entered session name does not already exist, the router creates a new session. Entering the **no** form of this command deletes an existing session.
Step 9	**attach-to-channel** *channel_name*	Attaches the session to the channel. Enter the same channel name that you entered in Step 3. *channel_name*–Identifies the channel.
Step 10	**link-addr source** *source_address*	Refers to the link address of the slave. *source_address*–Value of 1-65535.
Step 11	map-to-session *session_name*	Maps the dnp3-ip session to an existing dnp3-serial session. **Note** One dnp3-ip session can be mapped to only one dnp3-serial session.
Step 12	**exit**	Exits session configuration mode.
Step 13	**exit**	Exits protocol configuration mode.

EXAMPLE

This example shows how to configure the DNP3 IP parameters:

```
router# configure terminal
router(config)# scada-gw protocol dnp3-ip
router(config-dnp3n)# channel cc_channel
router(config-dnp3n-channel)# link-addr dest 3
router(config-dnp3n-channel)# tcp-connection local-port default remote-ip any
router(config-dnp3n-channel)# exit
router(config-dnp3n)# session cc_session
router(config-dnp3n-session)# attach-to-channel cc_channel
router(config-dnp3n-session)# link-addr source 3
router(config-dnp3n-session)# map-to-session rtu_session
router(config-dnp3n)# exit
router(config)# exit
```

Starting and Stopping the Protocol Translation Engine

BEFORE YOU BEGIN

After configuring the SCADA protocols on the CGR 1000, start the Protocol Translation Engine. Before making any configuration changes to Protocol Translation, stop the Protocol Translation Engine.

DETAILED STEPS

	Command	Purpose
Step 1	**configure terminal**	Enters global configuration mode.
Step 2	**scada-gw {enable | disable}**	Starts or stops the Protocol Translation Engine on the CGR 1000.

EXAMPLE

```
router# configure terminal
router(config)# scada-gw enable
```

Verifying Configuration

Command	Purpose
show running-config	Shows the configuration of the router including those features that are active and their settings.
show scada database	Shows the data points fetched from DNP3-Serial/T101 slave RTUs and sent to DNP3-Serial/T101 masters on the CGR 1000.
show scada statistics	Shows statistics for the SCADA gateway, including the number of messages sent and received, timeouts, and errors.
show scada tcp	Shows the TCP connection established on the CGR 1000 T104/DNP3-IP slave side.

This example shows the output from the **show scada tcp** and **show scada statistics** commands:

```
router# show scada tcp
DNP3 network channel [test]: 4 max simultaneous connections

conn: local-ip: 3.3.3.21        local-port 20000        remote-ip 3.3.3.15
data-socket 1

Total:
  1 current client connections
  0 total closed connections

router# show scada statistics
DNP3 network Channel [test]:
  5 messages sent, 2 messages received
  0 timeouts, 0 aborts, 0 rejections
```

```
  2 protocol errors, 2 link errors, 0 address errors

DNP3 serial Channel [test]:
  152 messages sent, 152 messages received
  1 timeouts, 0 aborts, 0 rejections
  0 protocol errors, 0 link errors, 0 address errors
```

Debug Commands

This section lists the following debug commands:

- SCADA DNP3-IP Debug Commands
- SCADA DNP3-Serial Debug Commands
- SCADA Driver Debug Commands
- SCADA Function Level Debug Commands
- SCADA Protocol Layer Debug Commands
- SCADA T101 Trace Debug Commands
- SCADA T104 Trace Debug Commands
- SCADA Protocol TCP Level Debug Commands

Table 1 SCADA DNP3-IP Debug Commands

Command	Purpose
debug scada dnp3n application	DNP3-IP application trace
debug scada dnp3n datalink	DNP3-IP datalink trace
debug scada dnp3n event	DNP3-IP event trace
debug scada dnp3n physical	DNP3-IP physical trace
debug scada dnp3n transport	DNP3-IP transport trace

Table 2 SCADA DNP3-Serial Debug Commands

Command	Purpose
debug scada dnp3s application	DNP3-Serial application trace
debug scada dnp3s datalink	DNP3-Serial datalink trace
debug scada dnp3s event	DNP3-Serial event trace
debug scada dnp3s physical	DNP3-Serial physical trace
debug scada dnp3s transport	DNP3-Serial transport trace

Table 3 SCADA Driver Debug Commands

Command	Purpose
debug scada driver event	Driver event trace
debug scada driver packet	Driver packet trace

Table 4 SCADA Function Level Debug Commands

Command	Purpose
debug scada function config	Configuration trace
debug scada function control	Control trace
debug scada function file	File trace
debug scada function freeze	Freeze trace
debug scada function physical	Physical trace
debug scada function poll	Poll trace
debug scada function stack	Stack trace
debug scada function umode	Umode trace

Table 5 SCADA Protocol Layer Debug Commands

Command	Purpose
debug scada layer application	Application Layer
debug scada layer network-physical	Network Physical Layer
debug scada layer serial-physical	Serial Physical Layer

Table 6 SCADA T101 Trace Debug Commands

Command	Purpose
debug scada t101 application	T101 application trace
debug scada t101 datalink	T101 datalink trace
debug scada t101 event	T101 event trace
debug scada t101 physical	T101 physical trace
debug scada t101 transport	T101 transport trace

Table 7 SCADA T104 Trace Debug Commands

Command	Purpose
debug scada t104 application	T104 application trace
debug scada t104 datalink	T104 datalink trace
debug scada t104 event	T104 event trace
debug scada t104 physical	T104 physical trace
debug scada t104 transport	T104 transport trace

Table 8 SCADA Protocol TCP Level Debug Commands

Command	Purpose
debug scada tcp event	TCP event trace
debug scada tcp packet	TCP packet trace

Configuration Example

The following example shows how to configure the serial port interface for SCADA, configure T101 and T104 protocol stacks, and start the Protocol Translation Engine on the CGR 1000:

```
router# configure terminal
router(config)# interface async 1/1
router (config-if)# no shutdown
router (config-if)# encapsulation scada
router (config-if)# exit
router(config)# scada-gw protocol t101
router(config-t101)# channel rtu_channel
router(config-t101-channel)# role master
router(config-t101-channel)# link-mode unbalanced
router(config-t101-channel)# link-addr-size one
router(config-t101-channel)# bind-to-interface serial 1/1
router(config-t101-channel)# exit
router(config-t101)# session rtu_session
router(config-t101-session)# attach-to-channel rtu_channel
router(config-t101-session)# common-addr-size two
router(config-t101-session)# cot-size one
router(config-t101-session)# info-obj-addr-size two
router(config-t101-session)# link-addr 3
router(config-t101-session)# exit
router(config-t101)# sector rtu_sector
router(config-t101-sector)# attach-to-session rtu_session
router(config-t101-sector)# asdu-addr 3
router(config-t101-sector)# exit
router(config-t101)# exit
router(config)# scada-gw protocol t104
router(config-t104)# channel cc_master1
router(config-t104-channel)# k-value 12
router(config-t104-channel)# w-value 8
router(config-t104-channel)# t0-timeout 30
router(config-t104-channel)# t1-timeout 15
router(config-t104-channel)# t2-timeout 10
```

```
router(config-t104-channel)# t3-timeout 30
router(config-t104-channel)# tcp-connection primary local-port 2050
router(config-t104-channel)# tcp-connection secondary local-port 2051
router(config-t104-channel)# exit
router(config-t104)# session cc_master1
router(config-t104-session)# attach-to-channel cc_master1
router(config-t104-session)# cot-size two
router(config-t104-session)# exit
router(config-t104)# sector cc_master1-sector
router(config-t104-sector)# attach-to-session cc_master1
router(config-t104-sector)# asdu-adr 3
router(config-t104-sector)# map-to-sector rtu_sector
router(config-t104)# exit
router(config)# scada-gw protocol t104
router(config-t104)# channel cc_master2
router(config-t104-channel)# k-value 12
router(config-t104-channel)# w-value 8
router(config-t104-channel)# t0-timeout 30
router(config-t104-channel)# t1-timeout 15
router(config-t104-channel)# t2-timeout 10
router(config-t104-channel)# t3-timeout 30
router(config-t104-channel)# tcp-connection primary local-port 2060
router(config-t104-channel)# tcp-connection secondary local-port 2061
router(config-t104-channel)# exit
router(config-t104)# session cc_master2
router(config-t104-session)# attach-to-channel cc_master2
router(config-t104-session)# cot-size two
router(config-t104-session)# exit
router(config-t104)# sector cc_master2-sector
router(config-t104-sector)# attach-to-session cc_master2
router(config-t104-sector)# asdu-adr 3
router(config-t104-sector)# map-to-sector rtu_sector
router(config-t104-sector)# exit
router(config-t104)# exit
router(config)# scada-gw enable
```

This example configures end-to-end communication between Control Centers and RTUs within a SCADA system using the DNP3 protocol stacks and starts the Protocol Translation Engine on the CGR 1000:

```
router# configure terminal
router(config)# interface async 1/1
router (config-if)# no shutdown
router (config-if)# encapsulation scada
router (config-if)# exit
router(config)# scada-gw protocol dnp3-serial
router(config-dnp3s)# channel rtu_channel
router(config-dnp3s-channel)# bind-to-interface async 1/1
router(config-dnp3s-channel)# link-addr source 3
router(config-dnp3s-channel)# unsolicited-response enable
router(config-dnp3s-channel)# exit
router(config-dnp3s)# session rtu_session
router(config-dnp3s-session)# attach-to-channel rtu_channel
router(config-dnp3s-session)# link-addr dest 3
router(config-dnp3s-session)# exit
router(config-dnp3s)# exit
router(config)# scada-gw protocol dnp3-ip
router(config-dnp3n)# channel cc_channel
router(config-dnp3n-channel)# link-addr dest 3
router(config-dnp3n-channel)# tcp-connection local-port default remote-ip any
router(config-dnp3n-channel)# exit
router(config-dnp3n)# session cc_session
router(config-dnp3n-session)# attach-to-channel cc_channel
router(config-dnp3n-session)# link-addr source 3
```

```
router(config-dnp3n-session)# map-to-session rtu_session
router(config-dnp3n)# exit
router(config)# exit
router(config)# scada-gw enable
```

Obtaining Documentation and Submitting a Service Request

For information on obtaining documentation, using the Cisco Bug Search Tool (BST), submitting a service request, and gathering additional information, see *What's New in Cisco Product Documentation* at: http://www.cisco.com/en/US/docs/general/whatsnew/whatsnew.html.

Subscribe to *What's New in Cisco Product Documentation*, which lists all new and revised Cisco technical documentation, as an RSS feed and deliver content directly to your desktop using a reader application. The RSS feeds are a free service.
