

Troubleshooting uBR Cable Modems Not Coming Online

Document ID: 16510

Contents

Introduction

Before You Begin

- Conventions
- Prerequisites
- Components Used

Troubleshooting Cable Modem State

- Offline State
- Ranging Process – init(r1), init(r2), and init(rc) state
- DHCP – init(d) state
- DHCP – init(i) state
- TOD exchange– init(t) state
- Option file transfer started – init(o) state
- Online, Online(d), Online(pk), Online(pt) state
- Online for Telco Return
- Reject(pk) and Reject(pt) state
- Registration – reject (m) state
- Registration – reject (c) state

Appendix

- Show controller command from CM
- Full Debug Capture on the CM side
- Show controller command from the CMTS
- Timers Explained
- CMTS Sample Config

Related Information

Introduction

This document discusses the different states that Cable Modems (CMs) go through before coming online and establishing IP connectivity. The document highlights the most commonly used Cisco IOS® software troubleshooting commands to verify what state the CMs are in, and the reasons that can cause the modems to arrive at that state. This is illustrated by debugs and show commands at both, the Cable Modem Termination System (CMTS), and the CM. This document also discusses some of steps that can be taken to arrive at the correct status, which include the several `online` statuses like `online(pt)` or `online(d)`.

Note: Refer to Understand How Basic Initialization Works for a Cable Modem Initialization Flowchart and a quick overview.

Before You Begin

Conventions

Refer to Cisco Technical Tips Conventions for more information on document conventions.

Prerequisites

Reader of this document should be familiar with the DOCSIS protocol.

Components Used

This document is not restricted to specific software and hardware versions.

Troubleshooting Cable Modem State

The first and most useful command to use at the CMTS is show cable modem:

```
sydney# show cable modem
```

Interface	Prim Sid	Online State	Timing Offset	Rec Power	QoS	CPE	IP address	MAC address
Cable2/0/U0	4	online(d)	2814	-0.50	6	0	10.1.1.20	0030.96f9.65d9
Cable2/0/U0	5	online(pt)	2290	-0.25	5	0	10.1.1.25	0050.7366.2223
Cable2/0/U0	6	offline	2287	-0.25	2	0	10.1.1.26	0050.7366.2221
Cable2/0/U0	7	online(d)	2815	-0.25	6	0	10.1.1.27	0001.9659.4461

The state field above shows what status the CM is in. The field can have the following values:

CM States (as shown in the CMTS)	Meaning
offline	Cable modem considered offline
init(r1)	Cable modem sent initial ranging
init(r2)	Cable modem is ranging
init(rc)	Cable modem ranging complete
init(d)	Dhcp request received
init(i)	Dhcp reply received; IP address assigned
init(t)	TOD exchange started
init(o)	Option file transfer started
online	Cable modem registered, enabled for data
online(d)	Cable modem registered, but network access for the cable modem is disabled
online(pk)	Cable modem registered, BPI enabled and KEK assigned
online(pt)	Cable modem registered, BPI enabled and TEK assigned
reject(pk)	KEK modem key assignment rejected
reject(pt)	TEK modem key assignment rejected
reject(m)	Cable modem did attempt to register; registration was refused due to bad MIC (Message Integrity Check)

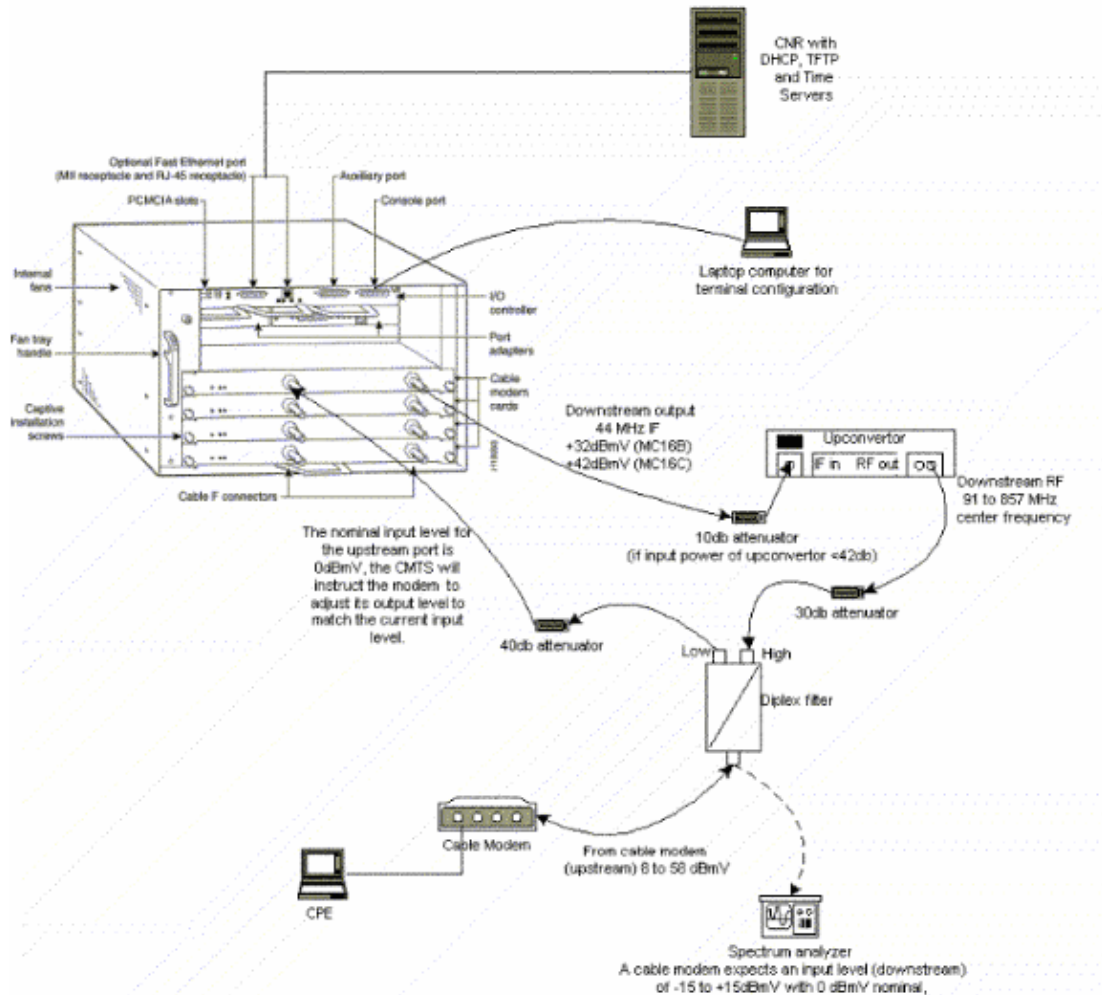
reject(c)	Cable modem did attempt to register; registration was refused due to bad COS (Class of Service)
-----------	---

An equivalent command on the CM side is **show controllers cable-modem 0 mac state** and look at the MAC state field. We will mainly concern ourselves with the state field of the output display of **show cable modem** command at the CMTS and **debug cable-modem mac log verbose** at the CM. Since the output display of the latter command can be quite large, only certain portions where applicable will be shown. A full capture of **debug cable-modem mac log verbose** can be found in the Full Debug Capture on the CM side section at the end of this tech note.

Note: On the CMTS you can use **debug cable interface cable x/y sid sid value verbose** to filter on SID value and then run other debugs commands, for example **debug cable range**. This way the debug output will be limited to the specified SID value and will not impact the CMTS performance.

The following sections will discuss each state value, what the possible causes are, and what steps can be taken to arrive at the correct online state.

Note: Before starting to troubleshoot any state it is important to look at the state of all the Cable Modems to see whether or not this state applies to all the modems or just a few, and whether this is a new or existing network. If it is an existing network, then investigate any recent changes. In most parts of this document it is assumed that the problem affects all Cable Modems and the following lab topology is applicable:



The above setup can be used for troubleshooting purposes, and rules out RF issues, since this setup excludes Cable TV signals.

Note: The uBR7100 has an integrated upconverter so an external upconverter is not required. Refer to Setting the Integrated Upconverter for more information.

Offline State

```
sydney# show cable modem
```

Interface	Prim Sid	Online State	Timing Offset	Rec Power	QoS	CPE	IP address	MAC address
Cable2/0/U0	5	offline	2290	0.00	2	0	10.1.1.25	0050.7366.2223
Cable2/0/U0	6	offline	2811	0.00	2	0	10.1.1.22	0050.7366.1e01
Cable2/0/U0	7	offline	2810	-0.50	2	0	10.1.1.20	0030.96f9.65d9
Cable2/0/U0	8	offline	2810	-0.25	2	0	10.1.1.21	0030.96f9.6605

From the output display of **show cable modem** command above we have four modems in the `offline` state. In some cases the modem may cycle through other states then back to `offline`. The following list gives the most common reasons for a modem not able to achieve quadrature amplitude modulation(QAM) lock:

- The Cable Modem is not connected to the network or it is not turned on
- Weak carrier signal (too much noise)
- Incorrect Downstream Center Frequency
- Incorrect Frequency Specified in the DOCSIS file
- Absence of downstream digital QAM modulated signal
- Incorrect frequency specified in **cable modem change-frequency** on the CMTS router
- Incorrect padding in the MCxx card

Below is cut down output display of **show controllers cable-modem 0** as taken from the Cable Modem (Kuffing) end:

```
kuffing# show controllers cable-modem 0
```

```
BCM Cable interface 0:  
CM unit 0, idb 0x8086C88C, ds 0x8086E460, regaddr = 0x2700000, reset_mask 0x80  
station address 0030.96f9.65d9 default station address 0030.96f9.65d9  
PLD VERSION: 1  
Concatenation: ON Max bytes Q0: 2000 Q1: 2000 Q2: 2000 Q3: 2000
```

```
MAC State is ds_channel_scanning_state, Prev States = 3  
MAC mcfiler 01E02F00 data mcfiler 00000000
```

```
MAC extended header ON  
DS: BCM 3300 Receiver: Chip id = BCM3300  
US: BCM 3300 Transmitter: Chip id = 3300
```

```
Tuner: status=0x00  
Rx: tuner_freq 529776400, symbol_rate 5361000, local_freq 1152000  
    snr_estimate 166(TenthdB), ber_estimate 0, lock_threshold 26000  
    QAM not in lock, FEC not in lock, qam_mode QAM_64 (Annex B)  
Tx: tx_freq 27984000, symbol rate 8 (1280000 sym/sec)  
    power_level: 6.0 dBmV (commanded)  
                7 (gain in US AMP units)  
                63 (BCM3300 attenuation in .4 dB units)  
.....
```

```
!--- Rest of display omitted.
```

From the above we can see that the Signal to Noise ratio estimate is 16.6 dB. Ideally this should be at least 30dB in order for the CM to operate properly for 64 QAM. Refer to RF Specifications for Data Over Cable Service Interface Specification (DOCSIS) Downstream and Upstream specifications, and also About Verifying the Downstream Signal. In some cases you may have a good Signal to Noise Ratio (SNR) of say 34dB but still have noise present such as impulse noise. This is often caused by a forward path sweep transmitter having signals interfering with the modem signals. This can only be detected by a spectrum analyzer operating in the zero span mode.

For more information on investigating noise issues using spectrum analyzer refer to Connecting the Cisco uBR7200 Series Router to the Cable Headend. One indication of impulse noise is the uncorrectable errors seen in the output of **show interfaces cable 2/0 upstream 0** as shown below:

```
sydney# show interfaces cable 2/0 upstream 0

Cable2/0: Upstream 0 is up
  Received 46942 broadcasts, 0 multicasts, 205903 unicasts
  0 discards, 12874 errors, 0 unknown protocol
  252845 packets input, 1 uncorrectable
  12871 noise, 0 microreflections
  Total Modems On This Upstream Channel : 3 (3 active)
  Default MAC scheduler
  Queue[Rng Polls] 0/64, fifo queueing, 0 drops
  Queue[Cont Mslots] 0/104, fifo queueing, 0 drops
  Queue[CIR Grants] 0/64, fair queueing, 0 drops
  Queue[BE Grants] 0/64, fair queueing, 0 drops
  Queue[Grant Shpr] 0/64, calendar queueing, 0 drops
  Reserved slot table currently has 0 CBR entries
  Req IEs 77057520, Req/Data IEs 0
  Init Mtn IEs 1194343, Stn Mtn IEs 117174
  Long Grant IEs 46953, Short Grant IEs 70448
  Avg upstream channel utilization : 1%
  Avg percent contention slots : 96%
  Avg percent initial ranging slots : 4%
  Avg percent minislots lost on late MAPs : 0%
  Total channel bw reserved 0 bps
  CIR admission control not enforced
  Current minislot count : 7192093      Flag: 0
  Scheduled minislot count : 7192182    Flag: 0
```

Note: If the amount of uncorrectable errors is greater than 1 in 10,000 most likely impulse noise present.

The optimal input power level at the CM is **0dBmV**, the receiver has a range of -15dBmV to $+15\text{dBmV}$. This can be measured by the spectrum analyzer. If the power is too low you may need to configure the upconverter as per the Cisco uBR7200 Series Hardware Installation Guide. If the signal is too strong then you may need to add more attenuation at the high frequency port connection. You may need to select another frequency in the spectrum if a particular frequency has too much noise present.

Note: The uBR7100 has an integrated upconverter. Refer to Setting the Integrated Upconverter for more information.



Caution: If the problem is affecting only one or a couple of modems, with several other modems operating correctly, then it's very unlikely the problem is at the upconverter side. Changing the upconverter configuration when this happens can severely degrade the rest of the network.

To confirm that the CM has not been able to achieve QAM lock turn on **debug cable-modem mac log verbose** you should see output similar to the following:

```
5w0d: 3084365.172 CMAC_LOG_STATE_CHANGE
```

```
ds_channel_scannie
```


frequency value under this option. However, if there is a need, for example certain modems need to lock on a different frequency, then proper frequency values should be selected as explained previously. The following debugs illustrate this with the CM locking on initially at 453MHz and then at 535.25MHz which was specified in the DOCSIS configuration file, thus causing the modem to reset and cycle through this process indefinitely:

```

4d00h: 345773.916 CMAC_LOG_WILL_SEARCH_SAVED_DS_FREQUENCY      453000000
4d00h: 345774.956 CMAC_LOG_UCD_MSG_RCVD                      1
4d00h: 345775.788 CMAC_LOG_DS_64QAM_LOCK_ACQUIRED            453000000

4d00h: 345775.792 CMAC_LOG_DS_CHANNEL_SCAN_COMPLETED
4d00h: 345775.794 CMAC_LOG_STATE_CHANGE                      wait_ucd_state
4d00h: 345776.946 CMAC_LOG_UCD_MSG_RCVD                      1
4d00h: 345778.960 CMAC_LOG_UCD_MSG_RCVD                      1
4d00h: 345778.962 CMAC_LOG_ALL_UCDS_FOUND
4d00h: 345778.966 CMAC_LOG_STATE_CHANGE                      wait_map_state
4d00h: 345778.968 CMAC_LOG_FOUND_US_CHANNEL                1
4d00h: 345780.996 CMAC_LOG_UCD_MSG_RCVD                      1
4d00h: 345781.000 CMAC_LOG_UCD_NEW_US_FREQUENCY            27984000
4d00h: 345781.004 CMAC_LOG_SLOT_SIZE_CHANGED                8
4d00h: 345781.084 CMAC_LOG_UCD_UPDATED
4d00h: 345781.210 CMAC_LOG_MAP_MSG_RCVD
4d00h: 345781.212 CMAC_LOG_INITIAL_RANGING_MINISLOTS        40
4d00h: 345781.216 CMAC_LOG_STATE_CHANGE                      ranging_l_state
4d00h: 345781.220 CMAC_LOG_RANGING_OFFSET_SET_TO           9610
4d00h: 345781.222 CMAC_LOG_POWER_LEVEL_IS                 22.0 dBmV (comma)
4d00h: 345781.226 CMAC_LOG_STARTING_RANGING
4d00h: 345781.228 CMAC_LOG_RANGING_BACKOFF_SET              0
4d00h: 345781.232 CMAC_LOG_RNG_REQ_QUEUED                  0
4d00h: 345781.272 CMAC_LOG_RNG_REQ_TRANSMITTED
4d00h: 345781.280 CMAC_LOG_RNG_RSP_MSG_RCVD
4d00h: 345781.282 CMAC_LOG_RNG_RSP_SID_ASSIGNED            3
4d00h: 345781.284 CMAC_LOG_ADJUST_RANGING_OFFSET            2288
4d00h: 345781.288 CMAC_LOG_RANGING_OFFSET_SET_TO           11898
4d00h: 345781.292 CMAC_LOG_ADJUST_TX_POWER                 7
4d00h: 345781.294 CMAC_LOG_POWER_LEVEL_IS                 24.0 dBmV (comma)
4d00h: 345781.298 CMAC_LOG_STATE_CHANGE                      ranging_2_state
4d00h: 345781.302 CMAC_LOG_RNG_REQ_QUEUED                  3
4d00h: 345782.298 CMAC_LOG_RNG_REQ_TRANSMITTED
4d00h: 345782.300 CMAC_LOG_RNG_RSP_MSG_RCVD
4d00h: 345782.304 CMAC_LOG_RANGING_SUCCESS
4d00h: 345782.316 CMAC_LOG_STATE_CHANGE                      dhcp_state
4d00h: 345782.450 CMAC_LOG_DHCP_ASSIGNED_IP_ADDRESS          10.1.1.25
4d00h: 345782.452 CMAC_LOG_DHCP_TFTP_SERVER_ADDRESS          172.17.110.136
4d00h: 345782.456 CMAC_LOG_DHCP_TOD_SERVER_ADDRESS          172.17.110.136
4d00h: 345782.460 CMAC_LOG_DHCP_SET_GATEWAY_ADDRESS
4d00h: 345782.464 CMAC_LOG_DHCP_TZ_OFFSET                    0
4d00h: 345782.466 CMAC_LOG_DHCP_CONFIG_FILE_NAME            frequency.cm
4d00h: 345782.470 CMAC_LOG_DHCP_ERROR_ACQUIRING_SEC_SVR_ADDR
4d00h: 345782.474 CMAC_LOG_DHCP_COMPLETE
4d00h: 345782.598 CMAC_LOG_STATE_CHANGE                      establish_tod_state
4d00h: 345782.606 CMAC_LOG_TOD_REQUEST_SENT
4d00h: 345782.620 CMAC_LOG_TOD_REPLY_RECEIVED              3178880491
4d00h: 345782.628 CMAC_LOG_TOD_COMPLETE
4d00h: 345782.630 CMAC_LOG_STATE_CHANGE                      security_associate_state
4d00h: 345782.634 CMAC_LOG_SECURITY_BYPASSED
4d00h: 345782.636 CMAC_LOG_STATE_CHANGE                      configuration_file
4d00h: 345782.640 CMAC_LOG_LOADING_CONFIG_FILE              frequency.cm
4d00h: %LINEPROTO-5-UPDOWN: Line protocol on Interface cable-modem0, changed state to up
4d00h: 345783.678 CMAC_LOG_CONFIG_FILE_PROCESS_COMPLETE
4d00h: 345783.682 CMAC_LOG_DS_FREQ_OVERRIDE                535250000
4d00h: 345783.686 CMAC_LOG_STATE_CHANGE                    reset_hardware_state
4d00h: 345784.048 CMAC_LOG_STATE_CHANGE                      wait_for_link_up_state
4d00h: 345784.052 CMAC_LOG_DRIVER_INIT_IDB_RESET            0x082A5226
4d00h: 345784.054 CMAC_LOG_LINK_DOWN

```

```

4d00h: 345784.056 CMAC_LOG_LINK_UP
4d00h: 345784.062 CMAC_LOG_STATE_CHANGE ds_channel_scanning_state
4d00h: 345785.198 CMAC_LOG_DS_NO_QAM_FEC_LOCK 535250000
4d00h: 345785.212 CMAC_LOG_DS_TUNER_KEEPALIVE
4d00h: 345787.018 CMAC_LOG_UCD_MSG_RCVD 1
4d00h: 345787.022 CMAC_LOG_DS_64QAM_LOCK_ACQUIRED 453000000

```

Note: Frequency override.

Incorrect frequency specified in **cable modem change–frequency** on the CMTS router can also cause the CM to switch frequencies and, if the frequency configured on the CMTS is not chosen carefully then similar result to the above will be seen. The **cable modem change–frequency** command on the CMTS is also optional and is typically left out by default.

After a downstream channel has been acquired, the next task is to locate a suitable upstream channel. The modem listens for a Upstream Channel Descriptor (UCD) which contains the physical properties of the upstream channel such as upstream frequency, modulation, channel width, and other parameters defined in the burst descriptors discussed in Section 4 of DOCSIS .

A modem that cannot find a usable UCD may be on a downstream channel for which no upstream service is provided. This is likely to be a headend misconfiguration. The **show controllers cable** command is a good place to start. Another possible reason a modem may not find a usable UCD is that its hardware or MAC may not support the parameters in the burst descriptors. This is likely to be either a headend misconfiguration or a less than DOCSIS compliant modem.

Once a usable UCD is found the modem will begin to listen to MAP (Bandwidth Allocation Map) messages which contain the upstream bandwidth allocation map of time. A section of time is mapped out into mini–slots and assigned to individual modems. There are also regions in the MAP for broadcast, contention based initial maintenance (or broadcast) ranging. It is these regions of the MAP that the modem must send its initial ranging requests until the CMTS responds with a ranging response (RNG–RSP).

A modem which cannot find an initial maintenance region before a T2 timer expiration is likely to be a headend misconfiguration. One should also check the insertion–interval for the cable interface on the CMTS. Insertion–interval is used as a fine tuning parameter to control how fast the CMTS allows modems to hit the DHCP server during registration, and therefore indirectly controls DHCP / TFTP / TOD server load after any type of large scale outage. It directly controls the length of time to recover the network.



Caution: Incorrect settings of insertion–interval will cause hours and hours of modems being offline, while the provisioning server has zero load. The best value for insertion–interval is **automatic**.

Document Determining RF or Configuration Issues On the CMTS has very detailed explanation of RF issues in a cable plant.

Ranging Process – init(r1), init(r2), and init(rc) state

At this stage, the CM begins a ranging process to calculate the necessary transmit power level to reach the CMTS at its desired input power level. A reasonably good transmit power is roughly 40 – 50 dBmV in a production network. Other hardware may vary. Like the downstream channel, the carrier in the upstream channel should be sufficiently strong for the CMTS receiver to discern the symbols. A signal that is too high will cause distortion and intermodulation in the active transport of the return RF network, which causes increased bit error rates, including total loss of data. This will be due to clipping of the signal.

The CM sends a ranging request (RNG–REQ) message to the CMTS and waits for a ranging response (RNG–RSP) message or a T3 timer expiration. If a T3 timeout occurs, the retry count increments. If the retry

count is less than the maximum number of retries, the modem transmits another RNG-REQ at a higher power level. This ranging process occurs in the initial maintenance or broadcast regions of the MAP because the CMTS has not assigned the modem a service identifier (SID) for unicast transmissions in the MAP. Thus, broadcast ranging is contention based and subject to collisions. To compensate for this the modems have a ranging backoff algorithm to calculate a random backoff time between RNG-REQ transmissions. This can be configured using **cable upstream range-backoff** command. When the transmit power has reached a sufficient level for the CMTS, it will respond to the RNG-REQ with a RNG-RSP containing a temporary SID. This SID will be used to identify unicast transmission regions in the MAP for unicast ranging.

Below output shows CM with SID 6 in **init(r1)** state indicating the CM cannot get past the initial ranging stage:

```
sydney#show cable modem
Interface   Prim Online   Timing Rec   QoS CPE IP address   MAC address
           Sid  State      Offset Power
Cable2/0/U0 5  offline   2287   0.00  2   0   10.1.1.25   0050.7366.2223
Cable2/0/U0 6  init(r1) 2813   12.00  2   0   10.1.1.22   0050.7366.1e01
Cable2/0/U0 7  offline   2810   0.25  2   0   10.1.1.20   0030.96f9.65d9
```

The debug below shows how the CM fails to complete the ranging process and resetting after a T3 timer expiration and number of retries exceeded. Note the **CMAC_LOG_ADJUST_TX_POWER** messages coming from the CMTS asking the CM to adjust its power:

```
1w3d: 871160.618 CMAC_LOG_STATE_CHANGE          ranging_1_state
1w3d: 871160.618 CMAC_LOG_RANGING_OFFSET_SET_TO        9610

1w3d: 871160.622 CMAC_LOG_POWER_LEVEL_IS          19.0 dBmV (comman)
1w3d: 871160.622 CMAC_LOG_STARTING_RANGING
1w3d: 871160.622 CMAC_LOG_RANGING_BACKOFF_SET      0
1w3d: 871160.622 CMAC_LOG_RNG_REQ_QUEUED          0
1w3d: 871160.678 CMAC_LOG_RNG_REQ_TRANSMITTED
1w3d: 871160.682 CMAC_LOG_RNG_RSP_MSG_RCVD
1w3d: 871160.682 CMAC_LOG_RNG_RSP_SID_ASSIGNED      6
1w3d: 871160.682 CMAC_LOG_ADJUST_RANGING_OFFSET    2813
1w3d: 871160.682 CMAC_LOG_RANGING_OFFSET_SET_TO    12423
1w3d: 871160.686 CMAC_LOG_ADJUST_TX_POWER          -48
1w3d: 871160.686 CMAC_LOG_STATE_CHANGE          ranging_2_state
1w3d: 871160.686 CMAC_LOG_RNG_REQ_QUEUED          6
1w3d: 871161.690 CMAC_LOG_RNG_REQ_TRANSMITTED
1w3d: 871161.690 CMAC_LOG_RNG_RSP_MSG_RCVD
1w3d: 871161.694 CMAC_LOG_ADJUST_TX_POWER          -36
1w3d: 871161.694 CMAC_LOG_RANGING_CONTINUE
1w3d: 871162.698 CMAC_LOG_RNG_REQ_TRANSMITTED
1w3d: 871162.898 CMAC_LOG_T3_TIMER
1w3d: 871163.734 CMAC_LOG_RNG_REQ_TRANSMITTED
1w3d: 871163.934 CMAC_LOG_T3_TIMER
1w3d: 871164.766 CMAC_LOG_RNG_REQ_TRANSMITTED
1w3d: 871164.966 CMAC_LOG_T3_TIMER
131.CABLEMODEM.CISCO: 1w3d: %UBR900-3-RESET_T3_RETRIES_EXHAUSTED: R03.0 Ranging
1w3d: 871164.966 CMAC_LOG_RESET_T3_RETRIES_EXHAUSTED
1w3d: 871164.966 CMAC_LOG_STATE_CHANGE          reset_interface_state
1w3d: 871164.966 CMAC_LOG_STATE_CHANGE          reset_hardware_state
```

Note: `init(r1)` is `ranging_1_state` and `init(r2)` is `ranging_2_state` You can get an indication of the transmit power on the CM by displaying the following command:

```
Staryn# show controllers cable-modem 0
```

```
BCM Cable interface 0:
CM unit 0, idb 0x2010AC, ds 0x86213E0, regaddr = 0x800000, reset_mask 0x80
station address 0050.7366.2223 default station address 0050.7366.2223
PLD VERSION: 32
```

```

MAC State is wait_for_link_up_state, Prev States = 2
MAC mcfilter 00000000 data mcfilter 00000000

MAC extended header ON
DS: BCM 3116 Receiver: Chip id = 2
US: BCM 3037 Transmitter: Chip id = 30AC

Tuner: status=0x00
Rx: tuner_freq 0, symbol_rate 5055932, local_freq 11520000
    snr_estimate 30640, ber_estimate 0, lock_threshold 26000
    QAM not in lock, FEC not in lock, qam_mode QAM_64
Tx: tx_freq 27984000, power_level 0x20 (8.0 dBmV), symbol_rate 8 (1280000 sym/s)

```

If a modem cannot proceed out of ranging state, the likely cause is an insufficient transmit power level. In the setup above transmit power can be adjusted by adjusting attenuation at the low frequency port. Increased attenuation will result in increased transmit power levels. Roughly 20 – 30 dB of attenuation is a good place to start. After initial ranging init(r1) the modem proceeds onto init(r2) which is where the modem must configure the transmit timing offset and power level to ensure that transmissions from the modem are received at the correct time and are at an acceptable input power level at the CMTS receiver. This is performed through a conversation of unicast RNG–REQ and RNG–RSP messages. The RNG–RSP messages contain power and timing offset corrections the modem must make. The modem continues to transmit RNG–REQ and perform adjustments per RNG–RSP until the RNG–RSP message indicates ranging success or ranging complete by reaching the init(rc) state. If a modem cannot proceed out of init (r2) the transmit power needs to be refined. Below is an output display of a CM in init(r2) state.

```
sydney# show cable modem
```

Interface	Prim Sid	Online State	Timing Offset	Rec Power	QoS	CPE	IP address	MAC address
Cable2/0/U0	5	init(r2)	2289	*4.00	2	0	10.1.1.25	0050.7366.2223
Cable2/0/U0	6	online	2811	-0.25	5	0	10.1.1.22	0050.7366.1e01
Cable2/0/U0	7	online	2811	-0.50	5	0	10.1.1.20	0030.96f9.65d9

Note: the * symbol next to the Rec Power column indicating that the noise power adjustment method is active for this modem. If you see an ! this means the modem has reached its maximum transmit power.

On the CMTS:

```
sydney# conf t
```

```

Enter configuration commands, one per line. End with CNTL/Z.
sydney(config)#access-list 101 permit ip host 10.1.1.10 host 172.17.110.136
sydney(config)#access-list 101 permit ip host 172.17.110.136 host 10.1.1.10
sydney(config)#^Z

```

where **10.1.1.10** is ip address of Cable interface on the CMTS
and **172.17.110.136** is ip address of DHCP server

```
sydney# debug list 101
```

```
sydney# debug ip packet detail
```

```

IP packet debugging is on
for access list: 101
(detailed)
sydney#

```

```
2w5d: IP: s=10.1.1.10 (local), d=172.17.110.136 (Ethernet1/0), len 604, sending
```

```
2w5d: UDP src=67, dst=67
```

```
2w5d: IP: s=172.17.110.136 (Ethernet1/0), d=10.1.1.10, len 328, rcvd 4
```

```
2w5d:      UDP src=67, dst=67
```

You can also use **debug ip udp** if this is a test or lab router:

```
sydney# debug ip udp
```

```
2w5d: UDP: rcvd src=0.0.0.0(68), dst=255.255.255.255(67), length=584
2w5d: UDP: sent src=10.1.1.10(67), dst=172.17.110.136(67), length=604
2w5d: UDP: rcvd src=172.17.110.136(67), dst=10.1.1.10(67), length=308
2w5d: UDP: sent src=0.0.0.0(67), dst=255.255.255.255(68), length=328
2w5d: UDP: rcvd src=0.0.0.0(68), dst=255.255.255.255(67), length=584
2w5d: UDP: sent src=10.1.1.10(67), dst=172.17.110.136(67), length=604
2w5d: UDP: rcvd src=172.17.110.136(67), dst=10.1.1.10(67), length=308
2w5d: UDP: sent src=0.0.0.0(67), dst=255.255.255.255(68), length=328
```



Caution: Running **debug ip udp** command on a Universal Broadband Router (uBR) cannot be used in

conjunction with an access list because this may cause the uBR to halt the system in order to keep up with the debugging. In this case, all the modems may lose sync, and debugging will be useless. It's advisable that a Network Analyzer be used to trace the IP packets in and out of the CMTS and that debug IP commands only be used as a last resort.

Note: The above access list is configured globally and has no effect on the IP operation. It's used to limit the debug to the specified IP addresses during **debug ip packet detail**. Make sure you run **debug list 101** first.

If no packets are seen through debug messages, check the configuration of the **cable helper-address** statement on the cable interface to which this modem is attached. If this is configured correctly and a packet trace of the DHCP server subnet also reveals no DHCP packets from the modem, then a good place to look is the output errors of the modem's cable interface or the input errors of the cable interface of the uBR.

If packets are seen to be transmitted onto the DHCP server subnet, it would be a good idea to double check the modem debug messages to see if there are parameter request or assignment errors. This would be the stage of troubleshooting where one should investigate the routing between the modem and the DHCP server. It would also be advisable to double-check the DHCP server configuration and the DHCP logs.

Below is a sample debug taken at the CM by running **debug cable-modem mac log verbose** command:

```
1w3d: 865015.920 CMAC_LOG_RANGING_SUCCESS
1w3d: 865015.920 CMAC_LOG_STATE_CHANGE                                dhcp_state
1w3d: 865053.580 CMAC_LOG_RNG_REQ_TRANSMITTED
1w3d: 865053.584 CMAC_LOG_RNG_RSP_MSG_RCVD
1w3d: 865055.924 CMAC_LOG_WATCHDOG_TIMER
131.CABLEMODEM.CISCO: 1w3d: %UBR900-3-RESET_DHCP_WATCHDOG_EXPIRED:
Cable Interface Reset due to DHCP watchdog timer expiration
1w3d: 865055.924 CMAC_LOG_RESET_DHCP_WATCHDOG_EXPIRED
1w3d: 865055.924 CMAC_LOG_STATE_CHANGE                                reset_interface_state
1w3d: 865055.924 CMAC_LOG_DHCP_PROCESS_KILLED
1w3d: 865055.924 CMAC_LOG_STATE_CHANGE                                reset_hardware_state
```

As can be seen above the DHCP process failed and the Cable Modem was reset.

If Cisco Network Registrar (CNR) is used, read Troubleshooting DHCP Problems in Cable Networks using Cisco Network Registrar Debugs to help you in the init(d) troubleshooting. This document contains very detail information on how to use the CNR debugs.

DHCP – init(d) state

The next stage after successful ranging is acquiring network configuration via DHCP. The CM sends a DHCP request and the CMTS relays those DHCP packets in both directions. Below is a an output display of **show cable modem** showing a modem with SID 7 in init(d), which indicates that the DHCP request was received from the Cable Modem:

```
sydney# show cable modem
```

Interface	Prim Sid	Online State	Timing Offset	Rec Power	QoS	CPE	IP address	MAC address
Cable2/0/U0	7	init(d)	2811	0.25	2	0	10.1.1.20	0030.96f9.65d9
Cable2/0/U0	8	online	2813	0.25	3	0	10.1.1.21	0030.96f9.6605
Cable2/0/U0	9	online	2812	-0.75	3	0	10.1.1.22	0050.7366.1e01

Note: The Cable Modem cycles through init(r1) to init(d) indefinitely. Possible causes as follows:

- Missing **cable helper-address** *ip address* command in the CMTS or incorrect *ip address*
- IP Connectivity issue from the CMTS to the DHCP server
- DHCP server down
- Wrong default gateway configured at the DHCP server
- Low transmit power at the CM or low upstream SNR, refer to RF Specifications.
- DHCP server overload
- DHCP server is out of IP addresses
- Reserved IP address for modem is inside wrong scope, see Understanding IP Address Management in Network Registrar GUI User's Guide.

Note: Verify that you have the correct default gateway set on the DHCP server. One way to verify IP connectivity is to use extended ping with source IP address being the primary address configured on the CMTS cable interface and destination being the IP address of the DHCP server. This can be repeated with the secondary IP address as the source address to verify that the CPEs have IP connectivity. See CMTS Sample Config.

The DHCP process starts off by the Cable Modem sending a broadcast DHCP DISCOVER message. If a DHCP server responds to the DISCOVER with an OFFER, the modem may choose to send a REQUEST for the offered configuration. The DHCP server can respond with an acknowledged (ACK) or not acknowledged (NAK). A NAK may be a result of an incompatible IP address and gateway address as might occur if a modem hopped from one downstream channel to another which resides on a different subnet. When the modem seeks renewal of the lease, the IP address and the gateway address of the DHCP REQUEST message will be different network numbers and the DHCP server will refuse the REQUEST with a NAK. These situations are rare and the modem will simply release the lease and start over with a DHCP DISCOVER message.

Frequently, errors at the DHCP state manifest themselves as timeouts rather than NAKs. The order of DHCP messages should be DISCOVER, OFFER, REQUEST, ACK. If the modem is transmitting a DISCOVER with no OFFER response from the DHCP server, turn on IP debugging on the CMTS. This can be done with the following steps:

DHCP – init(i) state

Once a reply to the DHCP request has been received and an IP address assigned to the Cable Modem the next the **show cable modem** gives is `init(i)`:

```
sydney# show cable modem
```

Interface	Prim Sid	Online State	Timing Offset	Rec Power	QoS	CPE	IP address	MAC address
Cable2/0/U0	7	init(i)	2815	-0.25	2	0	10.1.1.20	0030.96f9.65d9
Cable2/0/U0	8	online	2813	0.25	3	0	10.1.1.21	0030.96f9.6605
Cable2/0/U0	9	online	2812	0.50	3	0	10.1.1.22	0050.7366.1e01

From the above the Cable Modem with **SID 7** never gets beyond state `init(i)`. Repetitive **show cable modem** displays will usually show the Cable Modem cycling between `init(r1)`, `init(r2)`, `init(rc)`, `init(d)` and `init(i)` indefinitely.

There could be a number a reasons for a Cable Modem not getting further than `init(i)`. Here is a list of the most common ones:

- Incorrect or invalid DOCSIS file specified in the DHCP server
- TFTP server issues, for example incorrect ip address, TFTP server unreachable
- Problems getting TOD or Timing Offset
- Incorrect Router setting in the DHCP configuration

Since the Cable Modem has reached as far as `init(i)` we know that it has gone as far as obtaining an IP address. This can be clearly shown in the output display of the output of **debug cable-modem mac log verbose** at the Cable Modem below:

```

3d20h: 334402.548 CMAC_LOG_RANGING_SUCCESS
3d20h: 334402.548 CMAC_LOG_STATE_CHANGE                               dhcp_state
3d20h: 334415.492 CMAC_LOG_DHCP_ASSIGNED_IP_ADDRESS                10.1.1.20

!--- IP address Assigned to CM.

3d20h: 334415.492 CMAC_LOG_DHCP_TFTP_SERVER_ADDRESS                  172.17.110.136
3d20h: 334415.492 CMAC_LOG_DHCP_TOD_SERVER_ADDRESS                  172.17.110.136
3d20h: 334415.492 CMAC_LOG_DHCP_SET_GATEWAY_ADDRESS
3d20h: 334415.492 CMAC_LOG_DHCP_TZ_OFFSET                           0
3d20h: 334415.496 CMAC_LOG_DHCP_CONFIG_FILE_NAME                   nofile

!--- DOCSIS file CM is trying to load.

3d20h: 334415.496 CMAC_LOG_DHCP_ERROR_ACQUIRING_SEC_SVR_ADDR
3d20h: 334415.496 CMAC_LOG_DHCP_ERROR_ACQUIRING_LOG_ADDRESS
3d20h: 334415.496 CMAC_LOG_DHCP_COMPLETE
3d20h: 334415.508 CMAC_LOG_STATE_CHANGE                               establish_tod_state
3d20h: 334415.512 CMAC_LOG_TOD_REQUEST_SENT                        172.17.110.136
3d20h: 334415.524 CMAC_LOG_TOD_REPLY_RECEIVED                      3178343318
3d20h: 334415.524 CMAC_LOG_TOD_COMPLETE
3d20h: 334415.528 CMAC_LOG_STATE_CHANGE                               security_association_state
3d20h: 334415.528 CMAC_LOG_SECURITY_BYPASSED
3d20h: 334415.528 CMAC_LOG_STATE_CHANGE                             configuration_file
3d20h: 334415.528 CMAC_LOG_LOADING_CONFIG_FILE                     nofile

!--- DOCSIS file name.

133.CABLEMODEM.CISCO: 3d20h: %LINEPROTO-5-UPDOWN: Line protocol on Interface cap
3d20h: 334416.544 CMAC_LOG_CONFIG_FILE_TFTP_FAILED                   -1
3d20h: 334416.548 CMAC_LOG_CONFIG_FILE_PROCESS_COMPLETE
3d20h: 334416.548 CMAC_LOG_RESET_CONFIG_FILE_READ_FAILED

```

Similarly, TFTP server issues would give similar errors resulting in the CM resetting and cycling through the same process indefinitely:

```

3d21h: 336136.520 CMAC_LOG_STATE_CHANGE                               dhcp_state
3d21h: 336149.404 CMAC_LOG_DHCP_ASSIGNED_IP_ADDRESS                10.1.1.20

```

```

3d21h: 336149.404 CMAC_LOG_DHCP_TFTP_SERVER_ADDRESS          172.17.110.100

!--- Incorrect TFTP Server address.

3d21h: 336149.404 CMAC_LOG_DHCP_TOD_SERVER_ADDRESS          172.17.110.136
3d21h: 336149.404 CMAC_LOG_DHCP_SET_GATEWAY_ADDRESS
3d21h: 336149.404 CMAC_LOG_DHCP_TZ_OFFSET                    0
3d21h: 336149.408 CMAC_LOG_DHCP_CONFIG_FILE_NAME            platinum.cm
3d21h: 336149.408 CMAC_LOG_DHCP_ERROR_ACQUIRING_SEC_SVR_ADDR
3d21h: 336149.408 CMAC_LOG_DHCP_ERROR_ACQUIRING_LOG_ADDRESS
3d21h: 336149.408 CMAC_LOG_DHCP_COMPLETE
3d21h: 336149.420 CMAC_LOG_STATE_CHANGE                      establish_tod_state
3d21h: 336149.424 CMAC_LOG_TOD_REQUEST_SENT                172.17.110.136
3d21h: 336149.436 CMAC_LOG_TOD_REPLY_RECEIVED              3178345052
3d21h: 336149.436 CMAC_LOG_TOD_COMPLETE
3d21h: 336149.440 CMAC_LOG_STATE_CHANGE                      security_association_state
3d21h: 336149.440 CMAC_LOG_SECURITY_BYPASSED
3d21h: 336149.440 CMAC_LOG_STATE_CHANGE                      configuration_file
3d21h: 336149.440 CMAC_LOG_LOADING_CONFIG_FILE              platinum.cm
133.CABLEMODEM.CISCO: 3d21h: %LINEPROTO-5-UPDOWN: Line protocol on Interface cap
3d21h: 336163.252 CMAC_LOG_RNG_REQ_TRANSMITTED
3d21h: 336163.252 CMAC_LOG_RNG_RSP_MSG_RCVD
3d21h: 336165.448 CMAC_LOG_CONFIG_FILE_TFTP_FAILED          -1

!--- TFTP process failing.

3d21h: 336165.448 CMAC_LOG_CONFIG_FILE_PROCESS_COMPLETE
3d21h: 336165.452 CMAC_LOG_RESET_CONFIG_FILE_READ_FAILED
3d21h: 336165.452 CMAC_LOG_STATE_CHANGE                      reset_interface_state

```

A way to test the TFTP server is by trying to download a small file (like the DOCSIS config file) into the flash card of the CMTS. This is done by using the **copy tftp flash** command. Notice that in the output below there was an error trying to open the file named platinum.cm. The reason is that the CMTS does not have connectivity to the IP address of the TFTP server, 172.17.110.100, since it is fake.

```

sydney# copy tftp flash

Address or name of remote host []? 172.17.110.100

Source filename []? platinum.cm

Destination filename [platinum.cm]?

Accessing tftp://172.17.110.100/platinum.cm...

%Error opening tftp://172.17.110.100/platinum.cm (Permission denied)

sydney#

```

Here it is necessary to check the connectivity to the TFTP server.

Problems getting Time of Day (TOD) or Timing Offset would also result in modem not achieving online status:

```

3d21h: 338322.500 CMAC_LOG_STATE_CHANGE                      dhcp_state
3d21h: 338334.260 CMAC_LOG_RNG_REQ_TRANSMITTED
3d21h: 338334.260 CMAC_LOG_RNG_RSP_MSG_RCVD
3d21h: 338335.424 CMAC_LOG_DHCP_ASSIGNED_IP_ADDRESS          10.1.1.20
3d21h: 338335.424 CMAC_LOG_DHCP_TFTP_SERVER_ADDRESS          172.17.110.136
3d21h: 338335.424 CMAC_LOG_DHCP_ERROR_ACQUIRING_TOD_ADDRESS
3d21h: 338335.424 CMAC_LOG_DHCP_SET_GATEWAY_ADDRESS
3d21h: 338335.424 CMAC_LOG_DHCP_ERROR_ACQUIRING_TZ_OFFSET
3d21h: 338335.424 CMAC_LOG_DHCP_CONFIG_FILE_NAME            platinum.cm
3d21h: 338335.428 CMAC_LOG_DHCP_ERROR_ACQUIRING_SEC_SVR_ADDR

```

```

3d21h: 338335.428 CMAC_LOG_DHCP_ERROR_ACQUIRING_LOG_ADDRESS
3d21h: 338335.428 CMAC_LOG_DHCP_COMPLETE
3d21h: 338335.428 CMAC_LOG_RESET_DHCP_FAILED
3d21h: 338335.432 CMAC_LOG_STATE_CHANGE reset_interface_state
3d21h: 338335.432 CMAC_LOG_STATE_CHANGE reset_hardware_state
3d21h: 338336.016 CMAC_LOG_STATE_CHANGE wait_for_link_up_state

```

Note: Prior to Cisco IOS Software Release version 12.1(1) TOD needed to be specified in the DHCP server in order for the Cable Modem to go online. However, after Cisco IOS Software Release version 12.1(1) TOD is not required but the Cable Modem still needs to get the timing offset, as shown in the following debugs:

```

344374.528 CMAC_LOG_STATE_CHANGE dhcp_state
344377.292 CMAC_LOG_RNG_REQ_TRANSMITTED
344377.292 CMAC_LOG_RNG_RSP_MSG_RCVD
344387.412 CMAC_LOG_DHCP_ASSIGNED_IP_ADDRESS 10.1.1.20
344387.412 CMAC_LOG_DHCP_TFTP_SERVER_ADDRESS 172.17.110.136
344387.412 CMAC_LOG_DHCP_TOD_SERVER_ADDRESS 172.17.110.136

```

!--- TOD server IP address obtained.

```

344387.412 CMAC_LOG_DHCP_SET_GATEWAY_ADDRESS
344387.412 CMAC_LOG_DHCP_ERROR_ACQUIRING_TZ_OFFSET

```

!--- Timing offset not specified in DHCP server.

```

344387.412 CMAC_LOG_DHCP_CONFIG_FILE_NAME platinum.cm
344387.412 CMAC_LOG_DHCP_ERROR_ACQUIRING_SEC_SVR_ADDR
344387.412 CMAC_LOG_DHCP_ERROR_ACQUIRING_LOG_ADDRESS
344387.412 CMAC_LOG_DHCP_COMPLETE
344387.412 CMAC_LOG_RESET_DHCP_FAILED
344387.412 CMAC_LOG_STATE_CHANGE reset_interface_state

```

!--- Modem resetting.

In the debugs below we have **no time-server** specified but we have a timing offset configured in the DHCP server hence the Cable Modem going online:

```

3d23h: 345297.516 CMAC_LOG_DHCP_ASSIGNED_IP_ADDRESS 10.1.1.20
3d23h: 345297.516 CMAC_LOG_DHCP_TFTP_SERVER_ADDRESS 172.17.110.136
3d23h: 345297.516 CMAC_LOG_DHCP_ERROR_ACQUIRING_TOD_ADDRESS
3d23h: 345297.516 CMAC_LOG_DHCP_SET_GATEWAY_ADDRESS
3d23h: 345297.516 CMAC_LOG_DHCP_TZ_OFFSET
03d23h: 345297.516 CMAC_LOG_DHCP_CONFIG_FILE_NAME platinum.c
3d23h: 345297.520 CMAC_LOG_DHCP_ERROR_ACQUIRING_SEC_SVR_ADDR
3d23h: 345297.520 CMAC_LOG_DHCP_ERROR_ACQUIRING_LOG_ADDRESS
3d23h: 345297.520 CMAC_LOG_DHCP_COMPLETE
3d23h: 345297.532 CMAC_LOG_STATE_CHANGE establish_tod_state
3d23h: 345297.532 CMAC_LOG_TOD_NOT_REQUESTED_NO_TIME_ADDR
3d23h: 345297.532 CMAC_LOG_STATE_CHANGE security_association_state
3d23h: 345297.536 CMAC_LOG_SECURITY_BYPASSED
3d23h: 345297.536 CMAC_LOG_STATE_CHANGE configuration_file
3d23h: 345297.536 CMAC_LOG_LOADING_CONFIG_FILE platinum.cm
3d23h: 345297.568 CMAC_LOG_CONFIG_FILE_PROCESS_COMPLETE
3d23h: 345297.568 CMAC_LOG_STATE_CHANGE registration_state
3d23h: 345297.592 CMAC_LOG_REG_RSP_MSG_RCVD
3d23h: 345297.592 CMAC_LOG_COS_ASSIGNED_SID 1/7
3d23h: 345297.596 CMAC_LOG_RNG_REQ_QUEUED 7
3d23h: 345297.596 CMAC_LOG_REGISTRATION_OK
3d23h: 345297.596 CMAC_LOG_STATE_CHANGE establish_privacy_state
3d23h: 345297.596 CMAC_LOG_PRIVACY_NOT_CONFIGURED
3d23h: 345297.596 CMAC_LOG_STATE_CHANGE maintenance_state
133.CABLEMODEM.CISCO: 3d23h: %LINEPROTO-5-UPDOWN: Line protocol on Interface changed state

```

For a comprehensive list of what DHCP options are required and which ones are optional please refer to DHCP and the DOCSIS Configuration File for Cable Modems (DOCSIS 1.0) tech note.

Note: Note: A common mistake to make when using CNR as a DHCP server is to select NTP server under Servers option in the Policy configuration menu. Instead, time–offset and time–server should be selected under Bootp Compatible option. For further information on configuring CNR refer to Configuring DHCP in the CNR documentation.

Not including a Router option setting in the DHCP server or specifying an invalid IP address in the Router option field will also result in modem not getting beyond init(i) state, as can be seen from **debug cable–modem mac log verbose** below:

```
1d16h: 146585.940 CMAC_LOG_CONFIG_FILE_TFTP_FAILED -
1d16h: 146585.940 CMAC_LOG_CONFIG_FILE_PROCESS_COMPLETE
1d16h: 146585.944 CMAC_LOG_RESET_CONFIG_FILE_READ_FAILED
1d16h: 146585.944 CMAC_LOG_STATE_CHANGE reset_interface_state
1d16h: 146585.944 CMAC_LOG_STATE_CHANGE reset_hardware_state
```

Note: An invalid DOCSIS configuration file, especially one with Maximum Upstream Transmit Burst set to 255 in the Class of Service in DOCSIS CPE Configurator, can prevent the modem from proceeding any further than init(i). This is typically seen with early DOCSIS specifications which set this value in mini–slot units. The recommended value is 1600 or 1800 bytes.

TOD exchange– init(t) state

After a modem has acquired its network parameters it must request the time of day from a Time Of Day (TOD) server. TOD uses a UTC timestamp (seconds from January 1, 1970). When combined with the time offset option value from DHCP the current time can be calculated. The time is used for syslog and event log timestamps.

Below we have Cable Modems with SID 1 and 2 in init(t). Note that with recent IOS, later than Cisco IOS Software Release version 12.1(1) the Cable Modem will still come online even though the TOD exchange failed, see the debugs output following the **show cable modem** command below:

```
sydney# show cable mode

Interface   Prim Online   Timing Rec   QoS CPE IP address   MAC address
           Sid  State
Cable2/0/U0 1  init(t)    2808   0.00   2   0   10.1.1.20    0030.96f9.65d9
Cable2/0/U0 2  init(t)    2809   0.25   2   0   10.1.1.21    0030.96f9.6605
Cable2/0/U0 3  init(i)    2810  -0.25   2   0   10.1.1.22    0050.7366.1e01
2d01h: 177933.712 CMAC_LOG_STATE_CHANGE dhcp_state
2d01h: 177933.716 CMAC_LOG_RNG_REQ_TRANSMITTED
2d01h: 177933.716 CMAC_LOG_RNG_RSP_MSG_RCVD
2d01h: 177946.596 CMAC_LOG_DHCP_ASSIGNED_IP_ADDRESS 10.1.1.20
2d01h: 177946.596 CMAC_LOG_DHCP_TFTP_SERVER_ADDRESS 172.17.110.136
2d01h: 177946.596 CMAC_LOG_DHCP_TOD_SERVER_ADDRESS 172.17.110.130
2d01h: 177946.596 CMAC_LOG_DHCP_SET_GATEWAY_ADDRESS
2d01h: 177946.596 CMAC_LOG_DHCP_TZ_OFFSET 0
2d01h: 177946.600 CMAC_LOG_DHCP_CONFIG_FILE_NAME platinum.cm
2d01h: 177946.600 CMAC_LOG_DHCP_ERROR_ACQUIRING_SEC_SVR_ADDR
2d01h: 177946.600 CMAC_LOG_DHCP_ERROR_ACQUIRING_LOG_ADDRESS
2d01h: 177946.600 CMAC_LOG_DHCP_COMPLETE
2d01h: 177946.612 CMAC_LOG_STATE_CHANGE establish_tod_state
2d01h: 177946.716 CMAC_LOG_RNG_REQ_TRANSMITTED
2d01h: 177946.716 CMAC_LOG_RNG_RSP_MSG_RCVD
133.CABLEMODEM.CISCO: 2d01h: %LINEPROTO-5-UPDOWN: Line protocol on Interface cap
2d01h: 177947.716 CMAC_LOG_RNG_REQ_TRANSMITTED
2d01h: 177947.716 CMAC_LOG_RNG_RSP_MSG_RCVD
2d01h: 177948.616 CMAC_LOG_TOD_REQUEST_SENT 172.17.110.130
```

```

2d01h: 177948.716 CMAC_LOG_RNG_REQ_TRANSMITTED
2d01h: 177954.616 CMAC_LOG_TOD_REQUEST_SENT 172.17.110.130
2d01h: 177954.716 CMAC_LOG_RNG_REQ_TRANSMITTED
2d01h: 177954.716 CMAC_LOG_RNG_RSP_MSG_RCVD
2d01h: 177960.616 CMAC_LOG_TOD_REQUEST_SENT 172.17.110.130
2d01h: 177960.712 CMAC_LOG_RNG_REQ_TRANSMITTED
2d01h: 177960.716 CMAC_LOG_RNG_RSP_MSG_RCVD
2d01h: 177961.716 CMAC_LOG_RNG_REQ_TRANSMITTED

```

131.CABLEMODEM.CISCO: 2d01h: %UBR900-3-TOD_FAILED_TIMER_EXPIRED:TOD failed, but Cable Interface proceeding to operational state

```

2d01h: 177986.616 CMAC_LOG_TOD_WATCHDOG_EXPIRED
2d01h: 177986.616 CMAC_LOG_STATE_CHANGE security_association_state
2d01h: 177986.616 CMAC_LOG_SECURITY_BYPASSED
2d01h: 177986.616 CMAC_LOG_STATE_CHANGE configuration_file
2d01h: 177986.620 CMAC_LOG_LOADING_CONFIG_FILE platinum.cm
2d01h: 177986.644 CMAC_LOG_CONFIG_FILE_PROCESS_COMPLETE
2d01h: 177986.644 CMAC_LOG_STATE_CHANGE registration_state
2d01h: 177986.644 CMAC_LOG_REG_REQ_MSG_QUEUED
2d01h: 177986.648 CMAC_LOG_REG_REQ_TRANSMITTED
2d01h: 177986.652 CMAC_LOG_REG_RSP_MSG_RCVD
2d01h: 177986.652 CMAC_LOG_COS_ASSIGNED_SID 1/1
2d01h: 177986.656 CMAC_LOG_RNG_REQ_QUEUED 1
2d01h: 177986.656 CMAC_LOG_REGISTRATION_OK

```

!--- Modem online.

```

2d01h: 177986.656 CMAC_LOG_STATE_CHANGE establish_privacy_state
2d01h: 177986.656 CMAC_LOG_PRIVACY_NOT_CONFIGURED
2d01h: 177986.656 CMAC_LOG_STATE_CHANGE maintenance_state
2d01h: 177988.716 CMAC_LOG_RNG_REQ_TRANSMITTED

```

Below is debug captured from a Cable Modem running Cisco IOS Software Release version 12.0(7)T showing the modem resetting due to TOD timer expiration. The modem in this case never achieves online status.

```

18:31:23: 66683.974 CMAC_LOG_STATE_CHANGE dhcp_state
18:31:24: 66684.110 CMAC_LOG_DHCP_ASSIGNED_IP_ADDRESS 10.1.1.25
18:31:24: 66684.114 CMAC_LOG_DHCP_TFTP_SERVER_ADDRESS 172.17.110.136
18:31:24: 66684.118 CMAC_LOG_DHCP_TOD_SERVER_ADDRESS 172.17.110.130
! Deliberate wrong IP Address
18:31:24: 66684.122 CMAC_LOG_DHCP_SET_GATEWAY_ADDRESS
18:31:24: 66684.124 CMAC_LOG_DHCP_TZ_OFFSET 0
18:31:24: 66684.128 CMAC_LOG_DHCP_CONFIG_FILE_NAME platinum.cm
18:31:24: 66684.132 CMAC_LOG_DHCP_ERROR_ACQUIRING_SEC_SVR_ADDR
18:31:24: 66684.136 CMAC_LOG_DHCP_COMPLETE
18:31:24: 66684.260 CMAC_LOG_STATE_CHANGE establish_tod_state
18:31:24: 66684.268 CMAC_LOG_TOD_REQUEST_SENT
18:31:25: %LINEPROTO-5-UPDOWN: Line protocol on Interface cable-modem0, changed state to up
18:31:29: 66689.952 CMAC_LOG_RNG_REQ_TRANSMITTED
18:31:29: 66689.956 CMAC_LOG_RNG_RSP_MSG_RCVD
18:32:04: 66724.266 CMAC_LOG_WATCHDOG_TIMER
18:32:04: %UBR900-3-RESET_TOD_WATCHDOG_EXPIRED: Cable Interface Reset due to TOD watchdog
18:32:04: 66724.272 CMAC_LOG_RESET_TOD_WATCHDOG_EXPIRED
18:32:04: 66724.274 CMAC_LOG_STATE_CHANGE reset_interface

```

!--- Modem resetting.

Time of day errors almost always point to a DHCP misconfiguration. Possible misconfigurations that can result in TOD errors are gateway address misconfigurations or the wrong TOD server address. Make sure you can ping the time-server to rule out IP connectivity issues and also make sure the time-server is available.

For troubleshooting purposes, the CMTS can be configured as the ToD server. The commands are:

```

sydney# conf t

Enter configuration commands, one per line.  End with CNTL/Z.

sydney(config)# cable time-server

sydney(config)# service udp-small-servers max-servers 25

```

Some of the commands that can be used to debug ToD problems when the CMTS is configured as the ToD are **show cable clock**, **show controllers clock-reference**.

Option file transfer started – init(o) state

The main configuration and administration interface to the cable modem is the configuration file downloaded from the provisioning server. This configuration file contains:

- Downstream channel and upstream channel identification and characteristics
- Class of Service settings
- Baseline Privacy settings
- General operational settings
- Network management information
- Software upgrade fields
- Filters
- Vendor specific settings

A Cable Modem stuck in init(o) state usually indicates the Cable Modem has started or is ready to download the configuration file but was unsuccessful due to the following possible reasons:

- Incorrect, corrupt (for example: ASCII instead of binary), or missing DOCSIS configuration file
- Unable to reach the TFTP server, either is unavailable, too busy or no IP connectivity
- Invalid or missing Configuration Parameters in DOCSIS file
- Wrong file permissions on the TFTP server

Note: You may not always see init(o), instead you might see init(i) and then cycling through from init(r1) to init(i). A more accurate state can be derived by displaying the output of **show controller cable-modem 0 mac state**. Here is a cut down display:

```

kuffing# show controller cable-modem 0 mac state

MAC State:                configuration_file_state
Ranging SID:              4
Registered:               FALSE
Privacy Established:      FALSE

```

The **debug cable-modem mac log verbose** following the **show cable modem** command below won't tell you if it's a configuration file being corrupt or TFTP server failed. The debugs point to both of them.

```

sydney# show cable modem

Interface  Prim Online  Timing Rec   QoS CPE IP address  MAC address
          Sid  State      Offset Power
Cable2/0/U0 1  init(o)  2812   0.00  2  0  10.1.1.21  0030.96f9.6605
Cable2/0/U0 2  init(o)  2814   0.50  2  0  10.1.1.22  0050.7366.1e01

w3d: 880748.992 CMAC_LOG_STATE_CHANGE                dhcp_state
1w3d: 880751.652 CMAC_LOG_RNG_REQ_TRANSMITTED
1w3d: 880751.656 CMAC_LOG_RNG_RSP_MSG_RCVD

```

```

1w3d: 880761.876 CMAC_LOG_DHCP_ASSIGNED_IP_ADDRESS 10.1.1.20
1w3d: 880761.876 CMAC_LOG_DHCP_TFTP_SERVER_ADDRESS 172.17.110.136
1w3d: 880761.876 CMAC_LOG_DHCP_TOD_SERVER_ADDRESS 172.17.110.136
1w3d: 880761.876 CMAC_LOG_DHCP_SET_GATEWAY_ADDRESS
1w3d: 880761.876 CMAC_LOG_DHCP_TZ_OFFSET 0
1w3d: 880761.880 CMAC_LOG_DHCP_CONFIG_FILE_NAME data.cm

```

!--- Corrupt configuration file.

```

1w3d: 880761.880 CMAC_LOG_DHCP_ERROR_ACQUIRING_SEC_SVR_ADDR
1w3d: 880761.880 CMAC_LOG_DHCP_ERROR_ACQUIRING_LOG_ADDRESS
1w3d: 880761.880 CMAC_LOG_DHCP_COMPLETE
1w3d: 880761.892 CMAC_LOG_STATE_CHANGE establish_tod_state
1w3d: 880761.896 CMAC_LOG_TOD_REQUEST_SENT 172.17.110.136
1w3d: 880761.904 CMAC_LOG_TOD_REPLY_RECEIVED 3180091733
1w3d: 880761.908 CMAC_LOG_TOD_COMPLETE
1w3d: 880761.908 CMAC_LOG_STATE_CHANGE security_association_state
1w3d: 880761.908 CMAC_LOG_SECURITY_BYPASSED
1w3d: 880761.912 CMAC_LOG_STATE_CHANGE configuration_file_state
1w3d: 880761.912 CMAC_LOG_LOADING_CONFIG_FILE data.cm
1w3d: 880762.652 CMAC_LOG_RNG_REQ_TRANSMITTED
1w3d: 880762.652 CMAC_LOG_RNG_RSP_MSG_RCVD
133.CABLEMODEM.CISCO: 1w3d: %LINEPROTO-5-UPDOWN: Line protocol on Interface cable-modem0,
changed state to up
1w3d: 880762.928 CMAC_LOG_CONFIG_FILE_TFTP_FAILED -1
1w3d: 880762.932 CMAC_LOG_CONFIG_FILE_PROCESS_COMPLETE
1w3d: 880762.932 CMAC_LOG_RESET_CONFIG_FILE_READ_FAILED
1w3d: 880762.932 CMAC_LOG_STATE_CHANGE reset_interface_state
1w3d: 880762.932 CMAC_LOG_STATE_CHANGE reset_hardware_state

```

An example of invalid Configuration Parameters in the DOCSIS CPE Configurator is invalid or missing Vendor ID or Vendor Specific Information. The result is similar to the above debugs in addition to the following messages:

```

133.CABLEMODEM.CISCO: 00:13:07: %LINEPROTO-5-UPDOWN: Line protocol on Interface cable-modem0
changed state to up

00:13:08: 788.004 CMAC_LOG_CONFIG_FILE_CISCO_BAD_TYPE 155
00:13:08: 788.004 CMAC_LOG_CONFIG_FILE_CISCO_BAD_TYPE 115
00:13:08: 788.004 CMAC_LOG_CONFIG_FILE_CISCO_BAD_TYPE 116
00:13:08: 788.004 CMAC_LOG_CONFIG_FILE_CISCO_BAD_ATTR_MAX LENG128
00:13:08: 788.008 CMAC_LOG_CONFIG_FILE_PROCESS_COMPLETE
00:13:08: 788.008 CMAC_LOG_RESET_CONFIG_FILE_READ_FAILED

```

Online, Online(d), Online(pk), Online(pt) state

sydney#show cable modem

Interface	Prim Sid	Online State	Timing Offset	Rec Power	QoS	CPE	IP address	MAC address
Cable2/0/U0	4	online	2810	-0.75	6	0	10.1.1.20	0030.96f9.65d9
Cable2/0/U0	5	online(pt)	2290	0.25	5	0	10.1.1.25	0050.7366.2223
Cable2/0/U0	7	online(d)	2815	0.00	6	0	10.1.1.27	0001.9659.4461

With the exception of online(d), online, online(pk) and online(pt) indicate that the CM has achieved online status and is able to transmit and receive data. Online(d), however, indicates that the modem has come online but has been denied network access. This is typically caused by disabling Network Access option under Radio Frequency info in the DOCSIS CPE Configurator. The default for Network Access is enabled. To know how to create a DOCSIS Configuration file that denies PCs connected to CM.

This can be clearly seen from the display of **show cable modem** above and the **debug cable-modem mac log verbose**:

```

04:11:34: 15094.700 CMAC_LOG_STATE_CHANGE dhcp_state

04:11:46: 15106.392 CMAC_LOG_RNG_REQ_TRANSMITTED
04:11:46: 15106.396 CMAC_LOG_RNG_RSP_MSG_RCVD
04:11:47: 15107.620 CMAC_LOG_DHCP_ASSIGNED_IP_ADDRESS 10.1.1.20
04:11:47: 15107.620 CMAC_LOG_DHCP_TFTP_SERVER_ADDRESS 172.17.110.136
04:11:47: 15107.620 CMAC_LOG_DHCP_TOD_SERVER_ADDRESS 172.17.110.136
04:11:47: 15107.620 CMAC_LOG_DHCP_SET_GATEWAY_ADDRESS
04:11:47: 15107.620 CMAC_LOG_DHCP_TZ_OFFSET 0
04:11:47: 15107.624 CMAC_LOG_DHCP_CONFIG_FILE_NAME noaccess.cm

!--- Network Access disabled.

04:11:47: 15107.624 CMAC_LOG_DHCP_ERROR_ACQUIRING_SEC_SVR_ADDR
04:11:47: 15107.624 CMAC_LOG_DHCP_ERROR_ACQUIRING_LOG_ADDRESS
04:11:47: 15107.624 CMAC_LOG_DHCP_COMPLETE
04:11:47: 15107.636 CMAC_LOG_STATE_CHANGE establish_tod_state
04:11:47: 15107.640 CMAC_LOG_TOD_REQUEST_SENT 172.17.110.136
04:11:47: 15107.648 CMAC_LOG_TOD_REPLY_RECEIVED 3179226080
04:11:47: 15107.652 CMAC_LOG_TOD_COMPLETE
04:11:47: 15107.652 CMAC_LOG_STATE_CHANGE security_association_stat
04:11:47: 15107.652 CMAC_LOG_SECURITY_BYPASSED
04:11:47: 15107.652 CMAC_LOG_STATE_CHANGE configuration_file_state
04:11:47: 15107.652 CMAC_LOG_LOADING_CONFIG_FILE noaccess.c
133.CABLEMODEM.CISCO: 04:11:48: %LINEPROTO-5-UPDOWN: Line protocol on Interface cable-modem
changed state to up
04:11:48: 15108.672 CMAC_LOG_CONFIG_FILE_PROCESS_COMPLETE
04:11:48: 15108.672 CMAC_LOG_STATE_CHANGE registration_state
04:11:48: 15108.672 CMAC_LOG_REG_REQ_MSG_QUEUED
04:11:48: 15108.676 CMAC_LOG_REG_REQ_TRANSMITTED
04:11:48: 15108.680 CMAC_LOG_REG_RSP_MSG_RCVD
04:11:48: 15108.680 CMAC_LOG_COS_ASSIGNED_SID 1/4
04:11:48: 15108.684 CMAC_LOG_RNG_REQ_QUEUED 4
04:11:48: 15108.684 CMAC_LOG_NETWORK_ACCESS_DENIED
04:11:48: 15108.684 CMAC_LOG_REGISTRATION_OK
04:11:48: 15108.684 CMAC_LOG_STATE_CHANGE establish_privacy_state
04:11:48: 15108.684 CMAC_LOG_PRIVACY_NOT_CONFIGURED
04:11:48: 15108.684 CMAC_LOG_STATE_CHANGE maintenance_state
04:11:49: 15109.392 CMAC_LOG_RNG_REQ_TRANSMITTED

```

Another way to check is by examining the output of **show controllers cable-modem 0 mac state** on the Cable Modem.

(Beginning of display has been omitted)

```

Config File:
  Network Access:          FALSE

!--- Network Access denied.

Maximum CPEs:              3
Baseline Privacy:
  Auth. Wait Timeout:      10
  Reauth. Wait Timeout:    10
  Auth. Grace Time:        600
  Op. Wait Timeout:        1
  Retry Wait Timeout:      1
  TEK Grace Time:          600
  Auth. Reject Wait Time:  60
COS 1:
  Assigned SID:            4
  Max Downstream Rate:     10000000
  Max Upstream Rate:       1024000
  Upstream Priority:        7
  Min Upstream Rate:       0

```

```

Max Upstream Burst:    0
Privacy Enable:       FALSE

```

(The rest of the display has been omitted.)

Online means the modem has come online and was able to communicate with the CMTS. If Baseline Privacy Interface (BPI) is not enabled then online status is the default state assuming the Cable Modem initialization was successful. If BPI is configured then you'll see status `online(pk)` and then followed shortly by `online(pt)`. Here is a debug output display taken on the CM side with the **debug cable-modem mac log verbose** showing only the registration part:

```

5d03h: 445197.804 CMAC_LOG_STATE_CHANGE                registration_state
5d03h: 445197.804 CMAC_LOG_REG_REQ_MSG_QUEUED
5d03h: 445197.812 CMAC_LOG_REG_REQ_TRANSMITTED
5d03h: 445197.816 CMAC_LOG_REG_RSP_MSG_RCVD
5d03h: 445197.816 CMAC_LOG_COS_ASSIGNED_SID            1/4
5d03h: 445197.816 CMAC_LOG_RNG_REQ_QUEUED              4
5d03h: 445197.816 CMAC_LOG_REGISTRATION_OK
5d03h: 445197.816 CMAC_LOG_STATE_CHANGE                establish_privacy_state
5d03h: 445197.820 CMAC_LOG_PRIVACY_FSM_STATE_CHANGE
  machine: KEK, event/state: EVENT_1_PROVISIONED/STATE_A_START, new state: STATE_B_AUTH_WAIT
5d03h: 445197.828 CMAC_LOG BPKM_REQ_TRANSMITTED
5d03h: 445197.848 CMAC_LOG BPKM_RSP_MSG_RCVD
5d03h: 445197.848 CMAC_LOG_PRIVACY_FSM_STATE_CHANGE
  machine: KEK, event/state: EVENT_3_AUTH_REPLY/STATE_B_AUTH_WAIT, new state: STATE_C_AUTHO
5d03h: 445198.524 CMAC_LOG_PRIVACY_FSM_STATE_CHANGE
  machine: TEK, event/state: EVENT_2_AUTHORIZED/STATE_A_START, new state: STATE_B_OP_WAIT
5d03h: 445198.536 CMAC_LOG_RNG_REQ_TRANSMITTED
5d03h: 445198.536 CMAC_LOG_RNG_RSP_MSG_RCVD
5d03h: 445198.536 CMAC_LOG BPKM_REQ_TRANSMITTED
5d03h: 445198.536 CMAC_LOG BPKM_RSP_MSG_RCVD
5d03h: 445198.540 CMAC_LOG_PRIVACY_FSM_STATE_CHANGE
  machine: TEK, event/state: EVENT_8_KEY_REPLY/STATE_B_OP_WAIT, new state: STATE_D_OPERATIO
5d03h: 445198.548 CMAC_LOG_PRIVACY_INSTALLED_KEY_FOR_SID 4
5d03h: 445198.548 CMAC_LOG_PRIVACY_ESTABLISHED
5d03h: 445198.552 CMAC_LOG_STATE_CHANGE                maintenance_state
5d03h: 445201.484 CMAC_LOG_RNG_REQ_TRANSMITTED
5d03h: 445201.484 CMAC_LOG_RNG_RSP_MSG_RCVD

```

If there is a problem with BPI in general you'll see `reject(pk)` meaning we could not get through the key authentication stage. This is covered in the `reject(pk)` and `reject(pt)` section.

Note: For correct BPI operation ensure that the CMTS and CM are both running a BPI enabled image, which is signified by the symbol K1 in the image name. Also ensure that the field **Baseline Privacy Enable** is set to 1 under the Class of Service option in the DOCSIS CPE Configurator. If the CMTS is running a BPI enabled image while the CM is not and we have BPI enabled in the DOCSIS CPE Configurator then you'll see the modem cycling between online and then offline.

Online for Telco Return

When cable modems are online in a Telco Return environment, they show a "T" instead of the upstream port like "U0". The output below shows this situation

```

ubr7223# show cable modem

Interface    Prim Online    Timing    Rec    QoS    CPE IP address    MAC address
            Sid    State    Offset    Power
Cable2/0/T  94    online    0        0.00    3      2    10.10.169.151    0020.4066.1
Cable2/0/T  95    online    0        0.00    3      1    10.10.168.18     0020.4061.

```

Cable2/0/T	96	online	0	0.00	3	1	10.10.169.240	0020.4066.1
Cable2/0/U0	97	online	307	0.25	4	1	10.10.168.108	0020.4002.1
Cable2/0/T	98	online	0	0.00	3	1	10.10.169.245	0020.4003.1
Cable2/0/U0	99	online	332	0.25	4	0	10.10.168.110	0020.400b.1
Cable2/0/U0	100	online	277	0.25	4	1	10.10.169.114	0020.4002.1
Cable2/0/T	101	online	0	0.00	3	1	10.10.169.175	0020.4066.1

The output above shows the cable modems in online state in a mixed environment. Notice that cable modems with SID 97, 99, and 100 use port upstream 0 while the rest of the cable modems use telco return for the upstream path. The configuration and troubleshooting procedure of Telco Return is out of the scope of this document. Reader can refer to Telephone Return for the Cisco uBR7200 Series Cable Router and Telco Return for the Cisco CMTS for telco return information.

Reject(pk) and Reject(pt) state

Below is a display output of show cable modem on the CMTS router:

```
sydney# show cable modem
Interface Prim Online Timing Rec QoS CPE IP address MAC address
          Sid State Offset Power
Cable2/0/U0 1 offline 2811 0.00 2 0 10.1.1.27 0001.9659.4461
Cable2/0/U0 2 reject(pk) 2812 0.00 6 0 10.1.1.20 0030.96f9.65d9
Cable2/0/U0 3 online 2287 0.00 5 0 10.1.1.25 0050.7366.2223
```

```
01:58:51: %UBR7200-5-UNAUTHSIDTIMEOUT: CMTS deleted BPI unauthorized Cable Modem 0030.96f9
```

In most cases where there is a problem with the BPI configuration you'll see a `reject(pk)`. This state is typically caused by the following:

- Corrupt public key by the CM in the auth request. Refer to sample debug cable privacy for proper sequence of events.
- Presence of `cable privacy authenticate-modem` configuration command on the CMTS router but no Radius server present.
- Improperly configured Radius server.
- Improperly configured Radius server.

`Reject(pt)` is typically caused by invalid TEK or traffic encryption key.

For more information see Baseline Privacy Interface Specification .

```
sydney# debug cable privacy
02:32:08: CMTS Received AUTH REQ.
02:32:08: Created a new CM key for 0030.96f9.65d9.
02:32:08: CMTS generated AUTH_KEY.
02:32:08: Input : 70D158F106B0B75
02:32:08: Public Key:
02:32:08: 0x0000: 30 68 02 61 00 DA BA 93 3C E5 41 7C 20 2C D1 87
02:32:08: 0x0010: 3B 93 56 E1 35 7A FC 5E B7 E1 72 BA E6 A7 71 91
02:32:08: 0x0020: F4 68 CB 86 A8 18 FB A9 B4 DD 5F 21 B3 6A BE CE
02:32:08: 0x0030: 6A BE E1 32 A8 67 9A 34 E2 33 4A A4 0F 8C DB BD
02:32:08: 0x0040: D0 BB DE 54 39 05 B0 E0 F7 19 29 20 8C F9 3A 69
02:32:08: 0x0050: E4 51 C6 89 FB 8A 8E C6 01 22 02 34 C5 1F 87 F6
02:32:08: 0x0060: A3 1C 7E 67 9B 02 03 01 00 01
02:32:08: RSA public Key subject:
02:32:08: 0x0000: 30 7C 30 0D 06 09 2A 86 48 86 F7 0D 01 01 01 05
02:32:08: 0x0010: 00 03 6B 00 30 68 02 61 00 DA BA 93 3C E5 41 7C
02:32:08: 0x0020: 20 2C D1 87 3B 93 56 E1 35 7A FC 5E B7 E1 72 BA
02:32:08: 0x0030: E6 A7 71 91 F4 68 CB 86 A8 18 FB A9 B4 DD 5F 21
```

```

02:32:08: 0x0040: B3 6A BE CE 6A BE E1 32 A8 67 9A 34 E2 33 4A A4
02:32:08: 0x0050: 0F 8C DB BD D0 BB DE 54 39 05 B0 E0 F7 19 29 20
02:32:08: 0x0060: 8C F9 3A 69 E4 51 C6 89 FB 8A 8E C6 01 22 02 34
02:32:08: 0x0070: C5 1F 87 F6 A3 1C 7E 67 9B 02 03 01 00 01
02:32:08: RSA encryption result = 0
02:32:08: RSA encrypted output:
02:32:08: 0x0000: B6 CA 09 93 BF 2C 05 66 9D C5 AF 67 0F 64 2E 31
02:32:08: 0x0010: 67 E4 2A EA 82 3E F7 63 8F 01 73 10 14 4A 24 ED
02:32:08: 0x0020: 65 8F 59 D8 23 BC F3 A8 48 7D 1A 08 09 BF A3 A8
02:32:08: 0x0030: D6 D2 5B C4 A7 36 C4 A9 28 F0 6C 5D A1 3B 92 A2
02:32:08: 0x0040: BC 99 CC 1F C9 74 F9 FA 76 83 ED D5 26 B4 92 EE
02:32:08: 0x0050: DD EA 50 81 C6 29 43 4F 73 DA 56 C2 29 AF 05 53
02:32:08: CMTS sent AUTH response.
02:32:08: CMTS Received TEK REQ.
02:32:08: Created a new key for SID 2.
02:32:08: CMTS sent KEY response.

```

Below is a sample debug output on the CM when we have authorization failure:

```

6d02h: 527617.480 CMAC_LOG_CONFIG_FILE_PROCESS_COMPLETE
6d02h: 527617.480 CMAC_LOG_STATE_CHANGE registration_state
6d02h: 527617.484 CMAC_LOG_REG_REQ_MSG_QUEUED
6d02h: 527617.488 CMAC_LOG_REG_REQ_TRANSMITTED
6d02h: 527617.492 CMAC_LOG_REG_RSP_MSG_RCVD
6d02h: 527617.492 CMAC_LOG_COS_ASSIGNED_SID 1/2
6d02h: 527617.492 CMAC_LOG_RNG_REQ_QUEUED 2
6d02h: 527617.492 CMAC_LOG_REGISTRATION_OK
6d02h: 527617.496 CMAC_LOG_STATE_CHANGE establish_privacy_state
6d02h: 527617.496 CMAC_LOG_PRIVACY_FSM_STATE_CHANGE
machine: KEK, event/state: EVENT_1_PROVISIONED/STATE_A_START, new state: STATE_B_AUTH_WAI
6d02h: 527617.504 CMAC_LOG BPKM_REQ_TRANSMITTED
6d02h: 527617.504 CMAC_LOG BPKM_RSP_MSG_RCVD
6d02h: 527617.508 CMAC_LOG_PRIVACY_FSM_STATE_CHANGE
machine: KEK, event/state: EVENT_2_AUTH_REJECT/STATE_B_AUTH_WAIT, new state: STATE_E_AUTH
129.CABLEMODEM.CISCO: 6d02h: %CMBPKM-1-AUTHREJECT: Authorization request rejected by CMTS:
Unauthorized CM
6d02h: 527618.588 CMAC_LOG_RNG_REQ_TRANSMITTED
6d02h: 527618.592 CMAC_LOG_RNG_RSP_MSG_RCVD

```

Similarly a **debug cable privacy** on the CMTS router would give the following errors:

```

02:47:00: CMTS Received AUTH REQ.

02:47:00: Sending KEK REJECT.
02:47:05: %UBR7200-5-UNAUTHSIDTIMEOUT: CMTS deleted BPI unauthorized Cable Modem 0030.96f9

```

Note: The CM keeps cycling from reject(pk) to init(r1) indefinitely.

Another possible error that can be encountered is that, due to encryption export restrictions, some vendor modems may require the following command on the CMTS router in the interface configuration:

```
sydney(config-if)# cable privacy 40-bit-des
```

Registration – reject (m) state

After configuration, the modem sends a registration request (REG-REQ) with a required subset of the configuration settings as well as the CM and CMTS message integrity checks (MIC). The CM MIC is a hashed calculation over the configuration file settings which provides a method for the modem to be sure the configuration file wasn't tampered with in transit. The CMTS MIC is much the same thing except it also includes a setting for a **cable shared-secret** authentication string. This shared secret is known by the CMTS

and ensures that only authorized modems will be allowed to register with the CMTS.

```
sydney# show cable modem
```

Interface	Prim Sid	Online State	Timing Offset	Rec Power	QoS	CPE	IP address	MAC address
Cable2/0/U0	1	reject(m)	2807	0.00	2	0	10.1.1.20	0030.96f9.65d9
Cable2/0/U0	2	online	2284	-0.50	5	0	10.1.1.25	0050.7366.2223
Cable2/0/U0	3	offline	18669	0.25	2	0	10.1.1.26	0050.7366.2221

```
01:17:59: %UBR7200-5-AUTHFAIL: Authorization failed for Cable Modem 0030.96f9.60
01:18:21: %UBR7200-5-AUTHFAIL: Authorization failed for Cable Modem 0030.96f9.60
```

The above output shows that Cable Modem with SID 1 is in reject(m) state. This is caused by bad Message Integrity Check (MIC) typically caused by:

- Mismatch between cable shared-secret configured under the cable interface and CMTS Authentication value under Miscellaneous option in the DOCSIS CPE Configurator. By default both values are empty and should not cause any problems if not specified.
- Corrupt configuration file (DOCSIS file).

Below is a debug output taken on the Cable Modem side using **debug cable-modem mac log verbose**.

```
00:32:08: 1928.816 CMAC_LOG_STATE_CHANGE establish_tod_e
00:32:08: 1928.820 CMAC_LOG_TOD_REQUEST_SENT 172.17.110.136
00:32:08: 1928.828 CMAC_LOG_TOD_REPLY_RECEIVED 3179139839
00:32:08: 1928.832 CMAC_LOG_TOD_COMPLETE
00:32:08: 1928.832 CMAC_LOG_STATE_CHANGE security_association_stat
00:32:08: 1928.832 CMAC_LOG_SECURITY_BYPASSED
00:32:08: 1928.832 CMAC_LOG_STATE_CHANGE configuration_e
00:32:08: 1928.832 CMAC_LOG_LOADING_CONFIG_FILE platinum.cm
00:32:09: 1929.708 CMAC_LOG_RNG_REQ_TRANSMITTED
00:32:09: 1929.712 CMAC_LOG_RNG_RSP_MSG_RCVD
133.CABLEMODEM.CISCO: 00:32:09: %LINEPROTO-5-UPDOWN: Line protocol on Interface
00:32:09: 1929.852 CMAC_LOG_CONFIG_FILE_PROCESS_COMPLETE
00:32:09: 1929.856 CMAC_LOG_STATE_CHANGE registration_state
00:32:09: 1929.856 CMAC_LOG_REG_REQ_MSG_QUEUED
00:32:09: 1929.860 CMAC_LOG_REG_REQ_TRANSMITTED
00:32:09: 1929.864 CMAC_LOG_REG_RSP_MSG_RCVD
00:32:09: 1929.864 CMAC_LOG_RESET_AUTHENTICATION_FAILURE
00:32:09: 1929.868 CMAC_LOG_STATE_CHANGE reset_interface_state
00:32:09: 1929.868 CMAC_LOG_STATE_CHANGE reset_hardware_state
```

To rectify the problem ensure you have a valid configuration file and an identical value under CMTS Authentication to what's configured in **cable shared-secret line** under the cable interface.

Registration – reject (c) state

```
sydney# show cable modem
```

Interface	Prim Sid	Online State	Timing Offset	Rec Power	QoS	CPE	IP address	MAC address
Cable2/0/U0	1	offline	2807	-0.25	2	0	10.1.1.20	0030.96f9.65d9
Cable2/0/U0	2	online	2284	-0.25	5	0	10.1.1.25	0050.7366.2223
Cable2/0/U0	3	reject(c)	2286	-0.25	2	0	10.1.1.26	0050.7366.2221

```
20:35:59: %UBR7200-5-CLASSFAIL: Registration failed for Cable Modem 0050.7366.2Q
```

As shown above Cable Modem with SID 3 has failed registration due to bad class of service (COS) or reject(c). Typically this is caused by:

- CMTS router is unable or unwilling to grant a particular requested COS
- Misconfigured parameter(s) in Class of Service option in DOCSIS CPE Configurator, for example having two classes of service with the same ID.

Below is **debug cable-modem mac log verbose** taken on the CM side showing failure due to bad COS:

```

1w3d: 885643.820 CMAC_LOG_STATE_CHANGE registration_state
1w3d: 885643.820 CMAC_LOG_REG_REQ_MSG_QUEUED
1w3d: 885643.824 CMAC_LOG_REG_REQ_TRANSMITTED
1w3d: 885643.828 CMAC_LOG_REG_RSP_MSG_RCVD
1w3d: 885643.828 CMAC_LOG_SERVICE_NOT_AVAILABLE 0x01,0x01,0x01
1w3d: 885643.828 CMAC_LOG_RESET_SERVICE_NOT_AVAILABLE
1w3d: 885643.828 CMAC_LOG_STATE_CHANGE reset_interface_state
1w3d: 885643.832 CMAC_LOG_STATE_CHANGE reset_hardware_state
1w3d: 885644.416 CMAC_LOG_STATE_CHANGE wait_for_link_up_state
1w3d: 885644.420 CMAC_LOG_DRIVER_INIT_IDB_RESET 0x8039E23C
1w3d: 885644.420 CMAC_LOG_LINK_DOWN
1w3d: 885644.420 CMAC_LOG_LINK_UP
1w3d: 885644.420 CMAC_LOG_STATE_CHANGE ds_channel_scanning_state
133.CABLEMODEM.CISCO: 1w3d: %LINEPROTO-5-UPDOWN: Line protocol on Interface cable-modem0,
changed state to down
1w3d: 885645.528 CMAC_LOG_UCD_MSG_RCVD 1
1w3d: 885646.828 CMAC_LOG_DS_64QAM_LOCK_ACQUIRED 453000000

```

Similarly, **debug cable registration** on the CMTS router gives the following message:

```

sydney# debug cable registration

CMTS registration debugging is on

sydney#

1d04h: %UBR7200-5-CLASSFAIL: Registration failed for Cable Modem 0001.9659.4461
on interface Cable2/0/U0:

Bad/Missing Class of Service Config in REG-REQ

```

Note how the modem eventually resets and starts all over again.

Appendix

Show controller command from CM

```

kuffing# show controllers cable-modem 0 mac state

MAC State: maintenance_state
Ranging SID: 1
Registered: TRUE
Privacy Established: TRUE
MIB Values:
  Mac Resets: 0
  Sync lost: 0
  Invalid Maps: 0
  Invalid UCDS: 0
  Invalid Rng Rsp: 0
  Invalid Reg Rsp: 0
  T1 Timeouts: 0
  T2 Timeouts: 0
  T3 Timeouts: 0
  T4 Timeouts: 0
  Range Aborts: 0

```


Preamble Value Offset: 896
FEC Error Correction: 5
FEC Codeword Info Bytes: 34
Scrambler Seed: 338
Maximum Burst Size: 0
Guard Time Size: 48
Last Codeword Length: 1
Scrambler on/off: 1

Burst Descriptor 2:

Interval Usage Code: 4
Modulation Type: 1
Differential Encoding: 2
Preamble Length: 128
Preamble Value Offset: 896
FEC Error Correction: 5
FEC Codeword Info Bytes: 34
Scrambler Seed: 338
Maximum Burst Size: 0
Guard Time Size: 48
Last Codeword Length: 1
Scrambler on/off: 1

Burst Descriptor 3:

Interval Usage Code: 5
Modulation Type: 1
Differential Encoding: 2
Preamble Length: 72
Preamble Value Offset: 944
FEC Error Correction: 5
FEC Codeword Info Bytes: 75
Scrambler Seed: 338
Maximum Burst Size: 6
Guard Time Size: 8
Last Codeword Length: 1
Scrambler on/off: 1

Burst Descriptor 4:

Interval Usage Code: 6
Modulation Type: 1
Differential Encoding: 2
Preamble Length: 80
Preamble Value Offset: 936
FEC Error Correction: 8
FEC Codeword Info Bytes: 220
Scrambler Seed: 338
Maximum Burst Size: 0
Guard Time Size: 8
Last Codeword Length: 1
Scrambler on/off: 1

Config File:

Network Access: TRUE
Maximum CPEs: 3
Baseline Privacy:
Auth. Wait Timeout: 10
Reauth. Wait Timeout: 10
Auth. Grace Time: 600
Op. Wait Timeout: 1
Retry Wait Timeout: 1
TEK Grace Time: 600
Auth. Reject Wait Time: 60
COS 1:
Assigned SID: 1
Max Downstream Rate: 10000000
Max Upstream Rate: 1024000

```
Upstream Priority:      6
Min Upstream Rate:     0
Max Upstream Burst:    0
Privacy Enable:        TRUE

Ranging Backoff Start: 0 (at initial ranging)
Ranging Backoff End:   3 (at initial ranging)
Data Backoff Start:    0 (at initial ranging)
Data Backoff End:      4 (at initial ranging)
```

```
IP Address:            10.1.1.20
Net Mask:              255.255.255.0
TFTP Server IP Address: 172.17.110.136
Time Server IP Address: 172.17.110.136
Config File Name:      privacy.cm
Time Zone Offset:      0
Log Server IP Address: 0.0.0.0
```

```
Drop Ack Enabled:     TRUE
```

Mac Sid Status

```
Max Sids: 4  Sids In Use: 1
Mac Sid 0:
  Sid: 1  State: 2
Mac Sid 1:
  Sid: 0  State: 1
Mac Sid 2:
  Sid: 0  State: 1
Mac Sid 3:
  Sid: 0  State: 1
Test sid queue:      0
kuffing#
```

Full Debug Capture on the CM side

```
kuffing# debug cable mac log verbose
```

```
1w0d: 606764.132 CMAC_LOG_LINK_UP
1w0d: 606764.132 CMAC_LOG_STATE_CHANGE ds_channel_scanning_state
1w0d: 606764.136 CMAC_LOG_WILL_SEARCH_DS_FREQUENCY_BAND 99/805790200/997799800/600030
1w0d: 606764.136 CMAC_LOG_WILL_SEARCH_DS_FREQUENCY_BAND 98/601780000/799789900/600030
1w0d: 606764.136 CMAC_LOG_WILL_SEARCH_DS_FREQUENCY_BAND 97/403770100/595779700/600030
1w0d: 606764.140 CMAC_LOG_WILL_SEARCH_DS_FREQUENCY_BAND 96/73753600/115755700/600030
1w0d: 606764.140 CMAC_LOG_WILL_SEARCH_DS_FREQUENCY_BAND 95/217760800/397769800/600030
1w0d: 606764.140 CMAC_LOG_WILL_SEARCH_DS_FREQUENCY_BAND 94/121756000/169758400/600030
1w0d: 606764.144 CMAC_LOG_WILL_SEARCH_DS_FREQUENCY_BAND 93/175758700/211760500/600030
1w0d: 606764.144 CMAC_LOG_WILL_SEARCH_DS_FREQUENCY_BAND 92/79753900/85754200/600030
1w0d: 606764.148 CMAC_LOG_WILL_SEARCH_DS_FREQUENCY_BAND 91/55752700/67753300/600030
1w0d: 606764.148 CMAC_LOG_WILL_SEARCH_DS_FREQUENCY_BAND 90/177000000/213000000/600000
1w0d: 606764.148 CMAC_LOG_WILL_SEARCH_DS_FREQUENCY_BAND 89/219000000/225000000/600000
1w0d: 606764.152 CMAC_LOG_WILL_SEARCH_DS_FREQUENCY_BAND 88/141000000/171000000/600000
1w0d: 606764.152 CMAC_LOG_WILL_SEARCH_DS_FREQUENCY_BAND 87/135012500/135012500/600000
1w0d: 606764.152 CMAC_LOG_WILL_SEARCH_DS_FREQUENCY_BAND 86/123012500/129012500/600000
1w0d: 606764.156 CMAC_LOG_WILL_SEARCH_DS_FREQUENCY_BAND 85/405000000/447000000/600000
1w0d: 606764.156 CMAC_LOG_WILL_SEARCH_DS_FREQUENCY_BAND 84/339012500/399012500/600000
1w0d: 606764.160 CMAC_LOG_WILL_SEARCH_DS_FREQUENCY_BAND 83/333025000/333025000/600000
1w0d: 606764.160 CMAC_LOG_WILL_SEARCH_DS_FREQUENCY_BAND 82/231012500/327012500/600000
1w0d: 606764.160 CMAC_LOG_WILL_SEARCH_DS_FREQUENCY_BAND 81/111025000/117025000/600000
1w0d: 606764.164 CMAC_LOG_WILL_SEARCH_DS_FREQUENCY_BAND 80/93000000/105000000/600000
1w0d: 606764.164 CMAC_LOG_WILL_SEARCH_DS_FREQUENCY_BAND 79/453000000/855000000/600000
1w0d: 606764.164 CMAC_LOG_WILL_SEARCH_SAVED_DS_FREQUENCY 453000000
1w0d: 606765.416 CMAC_LOG_UCD_MSG_RCVD 1
131.CABLEMODEM.CISCO: 1w0d: %LINK-3-UPDOWN: Interface cable-modem0, changed state to up
1w0d: 606766.576 CMAC_LOG_DS_64QAM_LOCK_ACQUIRED 453000000
```

```

1w0d: 606766.576 CMAC_LOG_DS_CHANNEL_SCAN_COMPLETED
1w0d: 606766.576 CMAC_LOG_STATE_CHANGE wait_ucd_state
1w0d: 606767.416 CMAC_LOG_UCD_MSG_RCVD 1
1w0d: 606769.416 CMAC_LOG_UCD_MSG_RCVD 1
1w0d: 606769.416 CMAC_LOG_ALL_UCDS_FOUND
1w0d: 606769.416 CMAC_LOG_STATE_CHANGE wait_map_state
1w0d: 606769.420 CMAC_LOG_FOUND_US_CHANNEL 1
1w0d: 606771.416 CMAC_LOG_UCD_MSG_RCVD 1
1w0d: 606771.416 CMAC_LOG_UCD_NEW_US_FREQUENCY 27984000
1w0d: 606771.416 CMAC_LOG_SLOT_SIZE_CHANGED 8
1w0d: 606771.436 CMAC_LOG_UCD_UPDATED
1w0d: 606771.452 CMAC_LOG_MAP_MSG_RCVD
1w0d: 606771.452 CMAC_LOG_INITIAL_RANGING_MINISLOTS 41
1w0d: 606771.452 CMAC_LOG_STATE_CHANGE ranging_1_state
1w0d: 606771.452 CMAC_LOG_RANGING_OFFSET_SET_TO 9610
1w0d: 606771.456 CMAC_LOG_POWER_LEVEL_IS 20.0 dBmV (commanded)
1w0d: 606771.456 CMAC_LOG_STARTING_RANGING
1w0d: 606771.456 CMAC_LOG_RANGING_BACKOFF_SET 0
1w0d: 606771.456 CMAC_LOG_RNG_REQ_QUEUED 0
1w0d: 606771.512 CMAC_LOG_RNG_REQ_TRANSMITTED
1w0d: 606771.516 CMAC_LOG_RNG_RSP_MSG_RCVD
1w0d: 606771.516 CMAC_LOG_RNG_RSP_SID_ASSIGNED 1
1w0d: 606771.516 CMAC_LOG_ADJUST_RANGING_OFFSET 2810
1w0d: 606771.516 CMAC_LOG_RANGING_OFFSET_SET_TO 12420
1w0d: 606771.516 CMAC_LOG_ADJUST_TX_POWER 17
1w0d: 606771.520 CMAC_LOG_STATE_CHANGE ranging_2_state
1w0d: 606771.520 CMAC_LOG_RNG_REQ_QUEUED 1
1w0d: 606772.524 CMAC_LOG_RNG_REQ_TRANSMITTED
1w0d: 606772.524 CMAC_LOG_RNG_RSP_MSG_RCVD
1w0d: 606772.524 CMAC_LOG_RANGING_SUCCESS
1w0d: 606772.524 CMAC_LOG_STATE_CHANGE dhcp_state
1w0d: 606773.564 CMAC_LOG_RNG_REQ_TRANSMITTED
1w0d: 606773.564 CMAC_LOG_RNG_RSP_MSG_RCVD
1w0d: 606775.560 CMAC_LOG_RNG_REQ_TRANSMITTED
1w0d: 606775.564 CMAC_LOG_RNG_RSP_MSG_RCVD
1w0d: 606778.560 CMAC_LOG_RNG_REQ_TRANSMITTED
1w0d: 606778.564 CMAC_LOG_RNG_RSP_MSG_RCVD
1w0d: 606780.564 CMAC_LOG_RNG_REQ_TRANSMITTED
1w0d: 606780.564 CMAC_LOG_RNG_RSP_MSG_RCVD
1w0d: 606782.560 CMAC_LOG_RNG_REQ_TRANSMITTED
1w0d: 606782.564 CMAC_LOG_RNG_RSP_MSG_RCVD
1w0d: 606785.408 CMAC_LOG_DHCP_ASSIGNED_IP_ADDRESS 10.1.1.20
1w0d: 606785.408 CMAC_LOG_DHCP_TFTP_SERVER_ADDRESS 172.17.110.136
1w0d: 606785.408 CMAC_LOG_DHCP_TOD_SERVER_ADDRESS 172.17.110.136
1w0d: 606785.408 CMAC_LOG_DHCP_SET_GATEWAY_ADDRESS
1w0d: 606785.408 CMAC_LOG_DHCP_TZ_OFFSET 0
1w0d: 606785.412 CMAC_LOG_DHCP_CONFIG_FILE_NAME privacy.cm
1w0d: 606785.412 CMAC_LOG_DHCP_ERROR_ACQUIRING_SEC_SVR_ADDR
1w0d: 606785.412 CMAC_LOG_DHCP_ERROR_ACQUIRING_LOG_ADDRESS
1w0d: 606785.412 CMAC_LOG_DHCP_COMPLETE
1w0d: 606785.424 CMAC_LOG_STATE_CHANGE establish_tod_state
1w0d: 606785.428 CMAC_LOG_TOD_REQUEST_SENT 172.17.110.136
1w0d: 606785.440 CMAC_LOG_TOD_REPLY_RECEIVED 3179817738
1w0d: 606785.440 CMAC_LOG_TOD_COMPLETE
1w0d: 606785.440 CMAC_LOG_STATE_CHANGE security_association_state
1w0d: 606785.444 CMAC_LOG_SECURITY_BYPASSED
1w0d: 606785.444 CMAC_LOG_STATE_CHANGE configuration_file_state
1w0d: 606785.444 CMAC_LOG_LOADING_CONFIG_FILE privacy.cm
1w0d: 606785.560 CMAC_LOG_RNG_REQ_TRANSMITTED
1w0d: 606785.564 CMAC_LOG_RNG_RSP_MSG_RCVD
133.CABLEMODEM.CISCO: 1w0d: %LINEPROTO-5-UPDOWN: Line protocol on Interface cable-modem0,
changed state to up
1w0d: 606786.460 CMAC_LOG_CONFIG_FILE_PROCESS_COMPLETE
1w0d: 606786.460 CMAC_LOG_STATE_CHANGE registration_state
1w0d: 606786.464 CMAC_LOG_REG_REQ_MSG_QUEUED
1w0d: 606786.468 CMAC_LOG_REG_REQ_TRANSMITTED

```

```

1w0d: 606786.472 CMAC_LOG_REG_RSP_MSG_RCVD
1w0d: 606786.472 CMAC_LOG_COS_ASSIGNED_SID 1/1
1w0d: 606786.472 CMAC_LOG_RNG_REQ_QUEUED 1
1w0d: 606786.472 CMAC_LOG_REGISTRATION_OK
1w0d: 606786.476 CMAC_LOG_STATE_CHANGE establish_privacy_state
1w0d: 606786.476 CMAC_LOG_PRIVACY_FSM_STATE_CHANGE machine: KEK, event/state:
EVENT_1_PROVISIONED/STATE_A_START, new state: STATE_B_AUTH_WAIT
1w0d: 606786.480 CMAC_LOG_BPKM_REQ_TRANSMITTED
1w0d: 606786.496 CMAC_LOG_BPKM_RSP_MSG_RCVD
1w0d: 606786.496 CMAC_LOG_PRIVACY_FSM_STATE_CHANGE machine: KEK, event/state:
EVENT_3_AUTH_REPLY/STATE_B_AUTH_WAIT, new state: STATE_C_AUTHORIZED
1w0d: 606787.176 CMAC_LOG_PRIVACY_FSM_STATE_CHANGE machine: TEK, event/state:
EVENT_2_AUTHORIZED/STATE_A_START, new state: STATE_B_OP_WAIT
1w0d: 606787.184 CMAC_LOG_BPKM_REQ_TRANSMITTED
1w0d: 606787.188 CMAC_LOG_BPKM_RSP_MSG_RCVD
1w0d: 606787.192 CMAC_LOG_PRIVACY_FSM_STATE_CHANGE machine: TEK, event/state:
EVENT_8_KEY_REPLY/STATE_B_OP_WAIT, new state: STATE_D_OPERATIONAL
1w0d: 606787.200 CMAC_LOG_PRIVACY_INSTALLED_KEY_FOR_SID 1
1w0d: 606787.200 CMAC_LOG_PRIVACY_ESTABLISHED
1w0d: 606787.204 CMAC_LOG_STATE_CHANGE maintenance_state
1w0d: 606787.560 CMAC_LOG_RNG_REQ_TRANSMITTED

```

Show controller command from the CMTS

```
sydney# show controllers cable 2/0
```

```

Interface Cable2/0
Hardware is MC16B
BCM3210 revision=0x56B0
idb 0x619705D8 MAC regs 0x3D100000 PLX regs 0x3D000000
rx ring entries 1024 tx ring entries 128 MAP tx ring entries 128
Rx ring 0x4B0607C0 shadow 0x6198DDF8 head 272
Tx ring 0x4B062800 shadow 0x6198EE68 head 127 tail 127 count 0
MAP Tx ring 0x4B062C40 shadow 0x6198F2D8 head 33 tail 33 count 0

MAP timer sourced from slot 2

throttled 0 enabled 0 disabled 0
Rx: spurious 769 framing_err 0 hcs_err 1 no_buffer 0 short_pkt 0
    no_enqueue 0 no_enp 0 miss_count 0 latency 8
    invalid_sid 0 invalid_mac 0 bad_ext_hdr_pdu 0 concat 0 bad-concat 0
Tx: full 0 drop 0 stuck 0 latency 0
MTx: full 0 drop 0 stuck 0 latency 9
Slots 132642 NoUWCollNoEngy 2 FECorHCS 1 HCS 1
Req 1547992064 ReqColl 0 ReqNoise 14211 ReqNoEnergy 1547905820
ReqData 0 ReqDataColl 0 ReqDataNoise 0 ReqDataNoEnergy 0
Rng 89613 RngColl 0 RngNoise 255
FECBlks 248575 UnCorFECBlks 2 CorFECBlks 0
MAP FIFO overflow 0, Rx FIFO overflow 0, No rx buf 0
DS FIFO overflow 0, US FIFO overflow 0, US stuck 0
Bandwidth Requests= 0x11961
Piggyback Requests= 0xECC1
Ranging Requests= 0x15D15
Timing Offset = 0x0
Bad bandwidth Requests= 0x0
No MAP buffer= 0x0
Cable2/0 Downstream is up
    Frequency not set, Channel Width 6 MHz, 64-QAM, Symbol Rate 5.056941 Msps
    FEC ITU-T J.83 Annex B, R/S Interleave I=32, J=4
    Downstream channel ID: 0
Cable2/0 Upstream 0 is up
    Frequency 27.984 MHz, Channel Width 1.600 MHz, QPSK Symbol Rate 1.280 Msps
    Spectrum Group is overridden
    SNR 29.8280 dB
    Nominal Input Power Level 0 dBmV, Tx Timing Offset 2815

```

Ranging Backoff automatic (Start 0, End 3)
 Ranging Insertion Interval automatic (60 ms)
 Tx Backoff Start 0, Tx Backoff End 4
 Modulation Profile Group 1
 Concatenation is enabled
 part_id=0x3137, rev_id=0x03, rev2_id=0xFF
 nb_agc_thr=0x0000, nb_agc_nom=0x0000
 Range Load Reg Size=0x58
 Request Load Reg Size=0x0E
 Minislot Size in number of Timebase Ticks is = 8
 Minislot Size in Symbols = 64
 Bandwidth Requests = 0x11969
 Piggyback Requests = 0xECC8
 Invalid BW Requests= 0x0
 Minislots Requested= 0x1C13EF
 Minislots Granted = 0x1C13EF
 Minislot Size in Bytes = 16
 Map Advance (Dynamic) : 2454 usecs
 UCD Count = 40287

Timers Explained

T1	10 sec	The time to wait for a usable UCD
T2	12 sec	The time to wait for an initial maintenance interval for broadcast ranging
T3	200 msec	The time to wait for a RNG-RSP during ranging.
T4	30 sec	The time to wait for a station maintenance interval to perform station maintenance ranging.
T6	6 sec	The time to wait for a REG-RSP during registration.

CMTS Sample Config

```

sydney# wr t

Building configuration...
Current configuration:
!
version 12.1
service timestamps debug uptime
service timestamps log uptime
no service password-encryption
!
hostname sydney
!
boot system flash ubr7200-ik1s-mz_121-2_T.bin
no logging buffered
enable password cisco
!
no cable qos permission create
no cable qos permission update
cable qos permission modems
!
!
!
!
ip subnet-zero
  
```

```
no ip domain-lookup
!
!
!
!
!
interface FastEthernet0/0
  no ip address
  shutdown
  half-duplex
!
interface Ethernet1/0
  ip address 172.17.110.139 255.255.255.224
!
interface Ethernet1/1
  no ip address
  shutdown
!
interface Ethernet1/2
  no ip address
  shutdown
!
interface Ethernet1/3
  no ip address
  shutdown
!
interface Ethernet1/4
  no ip address
  shutdown
!
interface Ethernet1/5
  no ip address
  shutdown
!
interface Ethernet1/6
  no ip address
  shutdown
!
interface Ethernet1/7
  no ip address
  shutdown
!
interface Cable2/0
  ip address 10.10.1.1 255.255.255.0 secondary
  ip address 10.1.1.10 255.255.255.0
  no keepalive
  cable downstream annex B
  cable downstream modulation 64qam
  cable downstream interleave-depth 32
  cable upstream 0 frequency 28000000
  cable upstream 0 power-level 0
  no cable upstream 0 shutdown
  cable upstream 1 shutdown
  cable upstream 2 shutdown
  cable upstream 3 shutdown
  cable upstream 4 shutdown
  cable upstream 5 shutdown
  cable dhcp-giaddr policy
  cable helper-address 172.17.110.136
!
interface Cable3/0
  no ip address
  no keepalive
  shutdown
  cable downstream annex B
  cable downstream modulation 64qam
```

```
cable downstream interleave-depth 32
cable upstream 0 shutdown
cable upstream 1 shutdown
cable upstream 2 shutdown
cable upstream 3 shutdown
cable upstream 4 shutdown
cable upstream 5 shutdown
!
ip classless
ip route 0.0.0.0 0.0.0.0 172.17.110.129
no ip http server
!
!
line con 0
  exec-timeout 0 0
  transport input none
line aux 0
line vty 0
  exec-timeout 0 0
  password cisco
  login
line vty 1 4
  password cisco
  login
!
end
```

sydney# **show version**

```
Cisco Internetwork Operating System Software
IOS (tm) 7200 Software (UBR7200-IK1S-M), Version 12.1(2)T,  RELEASE SOFTWARE (fc1)
Copyright (c) 1986-2000 by cisco Systems, Inc.
Compiled Tue 16-May-00 13:36 by ccai
Image text-base: 0x60008900, data-base: 0x613E8000
```

```
ROM: System Bootstrap, Version 11.1(10) [dschwart 10], RELEASE SOFTWARE (fc1)
BOOTFLASH: 7200 Software (UBR7200-BOOT-M), Version 12.0(10)SC,
EARLY DEPLOYMENT RELEASE SOFTWARE (fc1)
```

```
sydney uptime is 1 day, 4 hours, 31 minutes
System returned to ROM by reload
System image file is "slot0:ubr7200-ik1s-mz_121-2_T.bin"
```

```
cisco uBR7223 (NPE150) processor (revision B) with 57344K/8192K bytes of memory.
Processor board ID SAB0249006T
R4700 CPU at 150Mhz, Implementation 33, Rev 1.0, 512KB L2 Cache
3 slot midplane, Version 1.0
```

```
Last reset from power-on
Bridging software.
```

```
X.25 software, Version 3.0.0.
8 Ethernet/IEEE 802.3 interface(s)
1 FastEthernet/IEEE 802.3 interface(s)
2 Cable Modem network interface(s)
125K bytes of non-volatile configuration memory.
1024K bytes of packet SRAM memory.
```

```
20480K bytes of Flash PCMCIA card at slot 0 (Sector size 128K).
4096K bytes of Flash internal SIMM (Sector size 256K).
Configuration register is 0x2102
```

Related Information

- **Building DOCSIS 1.0 Configuration Files Using Cisco DOCSIS Configurator** (registered customers only)
 - **Technical Support – Cisco Systems**
-

[Contacts & Feedback](#) | [Help](#) | [Site Map](#)

© 2010 – 2011 Cisco Systems, Inc. All rights reserved. [Terms & Conditions](#) | [Privacy Statement](#) | [Cookie Policy](#) | [Trademarks of Cisco Systems, Inc.](#)

Updated: Jan 12, 2006

Document ID: 16510
