10.5.1.12 Get Intel Management Control Formats

10.5.1.12.1 Get Intel Management Control Command (Intel Command 0x21)

	Bits			
Bytes	**31..24**	**23..16**	**15..08**	**07..00**
00..15	NC-SI Header			
16..19	Manufacturer ID (Intel 0x157)			
20..21	0x20	0x00		

10.5.1.12.2 Get Intel Management Control Response (Intel Command 0x21)

	Bits			
Bytes	**31..24**	**23..16**	**15..08**	**07..00**
00..15	NC-SI Header			
16..19	Response Code		Reason Code	
20..23	Manufacturer ID (Intel 0x157)			
24..26	0x21	0x00	Intel Management Control 1	

Intel Management Control 1 byte is described in Section 10.5.1.11.1.

10.5.1.13 TCO Reset

This command causes the network controller to perform TCO Reset, if Force TCO reset is enabled in the EEPROM.

If the BMC has detected that the operating system is hung and has blocked the Rx/Tx path the Force TCO reset clears the data path (Rx/Tx) of the network controller to enable the BMC to transmit/receive packets through the network controller.

When this command is issued to a channel in a package, it applies only to the specific channel.

After successfully performing the command the network controller considers Force TCO command as an indication that the operating system is hung and clears the DRV_LOAD flag (disable the driver).

10.5.1.13.1 Perform Intel TCO Reset Command (Intel Command 0x22)

	Bits			
Bytes	**31..24**	**23..16**	**15..08**	**07..00**
00..15	NC-SI Header			
16..19	Manufacturer ID (Intel 0x157)			
20	0x22	TCO Mode[1]		

[1] See Section 10.5.2.1.4.

10.5.1.13.2 Perform Intel TCO Reset Response (Intel Command 0x22)

	Bits			
Bytes	**31..24**	**23..16**	**15..08**	**07..00**
00..15	NC-SI Header			
16..19	Response Code		Reason Code	
20..23	Manufacturer ID (Intel 0x157)			
24..26	0x22			

10.5.1.14 Checksum Offloading

This command enables the checksum offloading filters in the network controller.

When enabled, these filters block any packets that did not pass IP, UDP and TCP checksums from being forwarded to the BMC. This feature does not support tunneled IPv4/IPv6 packet inspection.

10.5.1.14.1 Enable Checksum Offloading Command (Intel Command 0x23)

	Bits			
Bytes	**31..24**	**23..16**	**15..08**	**07..00**
00..15	NC-SI Header			
16..19	Manufacturer ID (Intel 0x157)			
20	0x23			

10.5.1.14.2 Enable Checksum Offloading Response (Intel Command 0x23)

	Bits			
Bytes	**31..24**	**23..16**	**15..08**	**07..00**
00..15	NC-SI Header			
16..19	Response Code		Reason Code	
20..23	Manufacturer ID (Intel 0x157)			
24..26	0x23			

10.5.1.14.3 Disable Checksum Offloading Command (Intel Command 0x24)

This command causes the network controller to stop verifying the IP/UDP/TCP checksums.

	Bits			
Bytes	**31..24**	**23..16**	**15..08**	**07..00**
00..15	NC-SI Header			
16..19	Manufacturer ID (Intel 0x157)			
20	0x24			

10.5.1.14.4 Disable Checksum Offloading Response (Intel Command 0x24)

	Bits			
Bytes	**31..24**	**23..16**	**15..08**	**07..00**
00..15	NC-SI Header			
16..19	Response Code		Reason Code	
20..23	Manufacturer ID (Intel 0x157)			
24..26	0x24			

10.5.1.15 LinkSec Support Commands

The following commands can be used by the BMC to control the different aspects of the LinkSec engine.

10.5.1.15.1 Transfer LinkSec Ownership to BMC Command (Intel Command 0x30, Parameter 0x10)

This command causes the 82599 to clear all LinkSec parameters, forcefully release host ownership and grant the ownership to the BMC. The BMC might allow the host to use the BMC's key for traffic by setting the *Host Control – Allow Host Traffic* bit. Activating this command clears all the LinkSec parameters.

	Bits			
Bytes	**31..24**	**23..16**	**15..08**	**07..00**
00..15	NC-SI Header			
16..19	Manufacturer ID (Intel 0x157)			
20..22	0x30	0x10	Host Control	

Table 10-15 LinkSec Host Control

Bit	Description
0	Reserved.
1	Allow Host Traffic: 0b = Host traffic is blocked. 1b = Host traffic is allowed.
2..7	Reserved.

10.5.1.15.2 Transfer LinkSec Ownership to BMC Response (Intel Command 0x30, Parameter 0x10)

	Bits			
Bytes	**31..24**	**23..16**	**15..08**	**07..00**
00..15	NC-SI Header			
16..19	Response Code		Reason Code	
20..23	Manufacturer ID (Intel 0x157)			
24..25	0x30	0x10		

10.5.1.15.3 Transfer LinkSec Ownership to Host Command (Intel Command 0x30, Parameter 0x11)

This command causes the 82599 to clear all LinkSec parameters, release BMC ownership and grant ownership to the host.

In this scenario traffic from/to the MC must be validated by the host's programmed keys. It is recommended that the MC try to establish network communication with a remote station to verify that the host was successful in programming the keys.

Activating this command clears all the LinkSec parameters.

	Bits			
Bytes	**31..24**	**23..16**	**15..08**	**07..00**
00..15	NC-SI Header			
16..19	Manufacturer ID (Intel 0x157)			
20..21	0x30	0x11		

10.5.1.15.4 Transfer LinkSec Ownership to Host Response (Intel Command 0x30, Parameter 0x11)

	Bits			
Bytes	**31..24**	**23..16**	**15..08**	**07..00**
00..15	NC-SI Header			
16..19	Response Code		Reason Code	
20..23	Manufacturer ID (Intel 0x157)			
24..25	0x30	0x11		

10.5.1.15.5 Initialize LinkSec Rx Command (Intel Command 0x30, Parameter 0x12)

This command can be used by the MC to initialize the LinkSec Rx engine. This command should be followed by a Set LinkSec Rx Key command to establish a LinkSec environment.

	Bits			
Bytes	**31..24**	**23..16**	**15..08**	**07..00**
00..15	NC-SI Header			
16..19	Manufacturer ID (Intel 0x157)			
20..23	0x30	0x12	Rx Port Identifier	
24..27	Rx SCI [0..3]			
28..29	Rx SCI [4..5]			

Where:

- **Rx Port Identifier** — the port number by which the NC identifies Rx packets. It is recommended that the MC use 0x0 as the port identifier. Note that the MC should use the same port identifier when performing the key-exchange.
- **Rx SCI** — A 6-byte unique identifier for the LinkSec Tx CA. It is recommended that the MC use its Ethernet MAC address value for this field.

10.5.1.15.6 Initialize LinkSec Rx Response (Intel Command 0x30, Parameter 0x12)

	Bits			
Bytes	**31..24**	**23..16**	**15..08**	**07..00**
00..15	NC-SI Header			
16..19	Response Code		Reason Code	
20..23	Manufacturer ID (Intel 0x157)			
24..25	0x30	0x12		

10.5.1.15.7 Initialize LinkSec Tx Command (Intel Command 0x30, Parameter 0x13)

This command can be used by the MC to initialize the LinkSec Tx engine. This command should be followed by a Set LinkSec Tx Key command to establish a LinkSec environment.

	Bits			
Bytes	**31..24**	**23..16**	**15..08**	**07..00**
00..15	NC-SI Header			
16..19	Manufacturer ID (Intel 0x157)			
20..23	0x30	0x13	Tx Port Identifier	
24..27	Tx SCI [0..3]			
28..31	Tx SCI [4..5]		Reserved	
32..35	Packet Number Threshold			
36	Tx Control			

- **Tx Port Identifier** — For this implementation this field is a don’t care and is automatically set to 0x0.
- **Tx SCI** — A 6-byte unique identifier for the LinkSec Tx CA. It is recommended that the MC use its Ethernet MAC address value for this field.
- **PN Threshold** — When a new key is programmed, the packet number is reset to 0x1. With each Tx packet, The packet number increments by one and is inserted to the packet (to avoid replay attacks). The PN threshold value is the 3 MSBytes of the Tx packet number after which a Key Exchange Required AEN is sent to the MC.

Example: a PN threshold of 0x123456 means that when the PN reaches 0x123456FF a notification is sent. The fourth byte of the PN threshold can be seen as a reserved bit, because it is always treated as 0xFF by the NC.

- **Tx Control**:

Bit	Description
0..4	Reserved.
5	Always Include SCI in Tx: 0b = Do not include SCI in Tx packets. 1b = Include SCI in Tx packets.
6..7	Reserved.

10.5.1.15.8 Initialize LinkSec Tx Response (Intel Command 0x30, Parameter 0x13)

	Bits			
Bytes	**31..24**	**23..16**	**15..08**	**07..00**
00..15	NC-SI Header			
16..19	Response Code		Reason Code	
20..23	Manufacturer ID (Intel 0x157)			
24..25	0x30	0x13		

10.5.1.15.9 Set LinkSec Rx Key Command (Intel Command 0x30, Parameter 0x14)

This command can be used by the MC to set a new LinkSec Rx key. Upon receiving this command the NC must switch to the new Rx key and send the response.

	Bits			
Bytes	**31..24**	**23..16**	**15..08**	**07..00**
00..15	NC-SI Header			
16..19	Manufacturer ID (Intel 0x157)			
20..23	0x30	0x14	Reserved	Rx SA AN
24..27	Rx LinkSec Key MSB	..	..	..
28..31	..	..	..	..
32..35	..	..	..	..
36..39	..	..	..	Rx LinkSec Key LSB

Where:

- **Rx SA AN** — The association number to be used with this key.

- **Rx LinkSec Key** — the 128 bits (16 bytes) key to be used for Rx

10.5.1.15.10 Set LinkSec Rx Key Response (Intel Command 0x30, Parameter 0x14)

	Bits			
Bytes	**31..24**	**23..16**	**15..08**	**07..00**
00..15	NC-SI Header			
16..19	Response Code		Reason Code	
20..23	Manufacturer ID (Intel 0x157)			
24..25	0x30	0x14		

10.5.1.15.11 Set LinkSec Tx Key Command (Intel Command 0x30, Parameter 0x15)

This command can be used by the MC to set a new LinkSec Tx key. Upon receiving this command the NC must switch to the new Tx key and send the response.

	Bits			
Bytes	**31..24**	**23..16**	**15..08**	**07..00**
00..15	NC-SI Header			
16..19	Manufacturer ID (Intel 0x157)			
20..23	0x30	0x15	Reserved	Tx SA AN
24..27	Tx LinkSec Key MSB	..	..	..
28..31	..	..	..	..
32..35	..	..	..	..
36..39	..	..	..	Tx LinkSec Key LSB

Where:

- **Tx SA AN** — The association number to be used with this key.
- **Tx LinkSec Key** — the 128 bits (16 bytes) key to be used for Tx

10.5.1.15.12 Set LinkSec Tx Key Response (Intel Command 0x30, Parameter 0x15)

	Bits			
Bytes	**31..24**	**23..16**	**15..08**	**07..00**
00..15	NC-SI Header			
16..19	Response Code		Reason Code	
20..23	Manufacturer ID (Intel 0x157)			
24..25	0x30	0x15		

10.5.1.15.13 Enable Network Tx Encryption Command (Intel Command 0x30, Parameter 0x16)

This command can be used by the MC to re-enable encryption of outgoing pass-through packets.

After this command is issued and until a response is received, the state of any outgoing packets is undetermined.

By default network Tx encryption is enabled.

	Bits			
Bytes	**31..24**	**23..16**	**15..08**	**07..00**
00..15	NC-SI Header			
16..19	Manufacturer ID (Intel 0x157)			
20..21	0x30	0x16		

10.5.1.15.14 Enable Network Tx Encryption Response (Intel Command 0x30, Parameter 0x16)

Following sending this response the NC must stop encrypting outgoing pass-through packets.

	Bits			
Bytes	**31..24**	**23..16**	**15..08**	**07..00**
00..15	NC-SI Header			
16..19	Response Code		Reason Code	
20..23	Manufacturer ID (Intel 0x157)			
24..25	0x30	0x16		

10.5.1.15.15 Disable Network Tx Encryption Command (Intel Command 0x30, Parameter 0x17)

This command can be used by the MC to disable encryption of outgoing pass-through packets.

After this command is issued and until a response is received, the state of any outgoing packets is undetermined.

	Bits			
Bytes	**31..24**	**23..16**	**15..08**	**07..00**
00..15	NC-SI Header			
16..19	Manufacturer ID (Intel 0x157)			
20..21	0x30	0x17		

10.5.1.15.16 Disable Network Tx Encryption Response (Intel Command 0x30, Parameter 0x17)

Following sending this response the NC must start encrypting outgoing pass-through packets.

	Bits			
Bytes	**31..24**	**23..16**	**15..08**	**07..00**
00..15	NC-SI Header			
16..19	Response Code		Reason Code	
20..23	Manufacturer ID (Intel 0x157)			
24..25	0x30	0x17		

10.5.1.15.17 Enable Network Rx Decryption Command (Intel Command 0x30, Parameter 0x18)

This command can be used by the MC to re-enable decryption of incoming pass-through packets. This causes the NC to execute LinkSec offload and to post the frames to the MC (or host) only if the LinkSec operation succeeds.

After this command is issued and until a response is received, the state of any incoming packets is undetermined.

By default network Rx decryption is disabled.

	Bits			
Bytes	**31..24**	**23..16**	**15..08**	**07..00**
00..15	NC-SI Header			
16..19	Manufacturer ID (Intel 0x157)			
20..21	0x30	0x18		

10.5.1.15.18 Enable Network Rx Decryption Response (Intel Command 0x30, Parameter 0x18)

Following sending this response the NC must begin decrypting incoming pass-through packets.

	Bits			
Bytes	31..24	23..16	15..08	07..00
00..15	NC-SI Header			
16..19	Response Code		Reason Code	
20..23	Manufacturer ID (Intel 0x157)			
24..25	0x30	0x18		

10.5.1.15.19 Disable Network Rx Decryption Command (Intel Command 0x30, Parameter 0x19)

This command can be used by the MC to disable decryption of incoming pass-through packets.

After this command is issued and until a response is received, the state of any incoming packets is undetermined.

	Bits			
Bytes	31..24	23..16	15..08	07..00
00..15	NC-SI Header			
16..19	Manufacturer ID (Intel 0x157)			
20..21	0x30	0x19		

10.5.1.15.20 Disable Network Rx Decryption Response (Intel Command 0x30, Parameter 0x19)

Following sending this response the NC must stop decrypting incoming pass-through packets.

	Bits			
Bytes	31..24	23..16	15..08	07..00
00..15	NC-SI Header			
16..19	Response Code		Reason Code	
20..23	Manufacturer ID (Intel 0x157)			
24..25	0x30	0x19		

10.5.1.15.21 Get LinkSec Parameters format (Intel Command 0x31)

The following commands can be used by the MC to retrieve the different LinkSec parameters.

These commands responses are valid only if the BMC owns the LinkSec.

10.5.1.15.22 Get LinkSec Rx Parameters Command (Intel Command 0x31, Parameter 0x01)

	Bits			
Bytes	**31..24**	**23..16**	**15..08**	**07..00**
00..15	NC-SI Header			
16..19	Manufacturer ID (Intel 0x157)			
20..22	0x31	0x01		

10.5.1.15.23 Get LinkSec Rx Parameters Response (Intel Command 0x31, Parameter 0x01)

This command enables the MC to retrieve the currently configured set of Rx LinkSec parameters.

	Bits			
Bytes	**31..24**	**23..16**	**15..08**	**07..00**
00..15	NC-SI Header			
16..19	Response Code		Reason Code	
20..23	Manufacturer ID (Intel 0x157)			
24..27	0x31	0x01	Reserved	
28..31	LinkSec Owner Status	LinkSec Host Control Status	Rx Port Identifier	
32..35	SCI [0..3]			
36..39	SCI [4..5]		Reserved	Rx SA AN
40..43	Rx SA Packet Number			

Where:

Table 10-16 LinkSec Owner Status

Value	Description
0x0	Host is LinkSec owner.
0x1	BMC is LinkSec owner.

Table 10-17 LinkSec Host Control Status

Bit	Description
0	Reserved.
1	Allow Host Traffic: 0b = Host traffic is blocked. 1b = Host traffic is allowed.
2..7	Reserved.

- **Rx Port Identifier** — The Rx Port identifier
- **Rx SCI** — The Rx SCI identifier.
- **Rx SA AN** — The association number associated with the active SA (for which the last valid Rx LinkSec packet was received).
- **Rx SA Packet Number** — Is the last packet number, as read from the last valid Rx LinkSec packet.

10.5.1.15.24 Get LinkSec Tx Parameters Command (Intel Command 0x31, Parameter 0x02)

This command enables the MC to retrieve the currently configured set of Tx LinkSec parameter.

	Bits			
Bytes	**31..24**	**23..16**	**15..08**	**07..00**
00..15	NC-SI Header			
16..19	Manufacturer ID (Intel 0x157)			
20..22	0x31	0x02		

10.5.1.15.25 Get LinkSec Tx Parameters Response (Intel Command 0x31, Parameter 0x02)

	Bits			
Bytes	**31..24**	**23..16**	**15..08**	**07..00**
00..15	NC-SI Header			
16..19	Response Code		Reason Code	
20..23	Manufacturer ID (Intel 0x157)			
24..27	0x31	0x2	Reserved	
28..31	LinkSec Owner Status	LinkSec Host Control Status	Tx Port Identifier	
32..35	SCI [0..3]			
36..39	SCI [4..5]		Reserved	Tx SA AN
40..43	Tx SA Packet Number			
44.47	Packet Number Threshold			
48	Tx Control Status			

Where:

Table 10-18 LinkSec Owner Status

Value	Description
0x0	Host is LinkSec owner.
0x1	BMC is LinkSec owner.

Table 10-19 LinkSec Host Control Status

Bit	Description
0	Reserved.
1	Allow Host Traffic: 0b = Host traffic is blocked. 1b = Host traffic is allowed.
2..7	Reserved.

- **Tx Port Identifier** — Reserved to 0x0 for this implementation.
- **Tx SCI** — The Rx SCI identifier.
- **Tx SA AN** — The association number currently used for the active SA.
- **Tx SA Packet Number** — Is the last packet number, as read from the last valid Rx LinkSec packet.

- **Packet Number Threshold:**

Table 10-20 Tx Control Status:

Bit	Description
0..4	Reserved.
5	Include SCI: 0b = Do not include SCI in Tx packets. 1b = Include SCI in Tx packets.
6..7	Reserved.

10.5.1.16 LinkSec AEN (Intel AEN 0x80)

The following is the AEN that can be sent by the NC following a LinkSec event.

This AEN must be enabled using the NC-SI AEN Enable command, using bit 16 (0x10000) of the AEN enable mask.

	Bits			
Bytes	**31..24**	**23..16**	**15..08**	**07..00**
00..15	NC-SI AEN Header			
20..23	Reserved			0x80
24..27	Reserved			LinkSec Event Cause

Where:

LinkSec Event Cause has the following format:

Bit #	Description
0	Host requested ownership.
1	Host released ownership.
2	Tx Key Packet Number (PN) threshold met.
3..7	Reserved.

10.5.2 SMBus Programming

This section describes the SMBus transactions supported in Advanced Pass Through (APT) mode.

10.5.2.1 Write SMBus Transactions (BMC → the 82599)

The following table lists the different SMBus write transactions supported by the 82599.

TCO Command	Transaction	Command		Fragmentation	Section
Transmit Packet	Block Write	First: Middle: Last:	0x84 0x04 0x44	Multiple	10.5.2.1.1
Transmit Packet	Block Write	Single:	0xC4	Single	10.5.2.1.1
Receive Enable	Block Write	Single:	0xCA	Single	10.5.2.1.3
Management Control	Block Write	Single:	0xC1	Single	10.5.2.1.5
Update MNG RCV filter parameters	Block Write	Single:	0xCC	Single	10.5.2.1.6
Force TCO	Block Write	Single:	0xCF	Single	10.5.2.1.4
Request Status	Block Write	Single:	0xDD	Single	10.5.2.1.2
Update LinkSec parameters	Block Write	Single:	0xC9	Single	10.5.2.1.7

10.5.2.1.1 Transmit Packet Command

The Transmit Packet command behavior is detailed in section 3.2.5. The Transmit Packet fragments have the following format:

Function	Command	Byte Count	Data 1	...	Data N
Transmit first fragment	0x84	N	Packet data MSB	...	Packet data LSB
Transmit middle fragment	0x04				
Transmit last fragment	0x44				
Transmit single fragment	0xC4				

The payload length is limited to the maximum payload length set in the EEPROM.

If the overall packet length is bigger than 1536 bytes, the packet is silently discarded by the 82599.

10.5.2.1.2 Request Status Command

The BMC can initiate a request to read the 82599 manageability status by sending this command.

When it receives this command, the 82599 initiates a notification to the BMC (when it is ready with the status), and then the BMC is able to read the status, by issuing a Read Status command (see section 10.5.2.2.3). Request Status Command format:

Function	Command	Byte Count	Data 1
Request status	0xDD	1	0

10.5.2.1.3 Receive Enable Command

The Receive Enable command is a single fragment command that is used to configure the 82599.

This command has two formats: short, 1-byte legacy format (providing backward compatibility with previous components) and long, 14-byte advanced format (allowing greater configuration capabilities).

Note: If the Receive Enable command is short and thus does not include all the parameters, then the parameters are taken from most recent previous configuration (either the most recent long Receive Enable command in which the particular value was set, or the EEPROM if there was no such previous long Receive Enable command).

Func.	Cmd	Byte Count	Data 1	Data 2	...	Data 7	Data 8	...	Data 11	Data 12	Data 13	Data 14
Legacy receive enable	0xCA	1	Receive control byte	-	...	-	-	...	-	-	-	-
Advanced receive enable		14 0x0E		MAC addr. MSB		MAC addr. LSB	IP addr. MSB		IP addr. LSB	BMC SMBus addr.	Interf. data byte	Alert value byte

While...

- **Receive control byte** (data byte 1) has the following format:

Field	Bit(s)	Description
RCV_EN	0	Receive TCO Enable. 0b = Disable Receive TCO packets. Rx Packets are not directed to BMC and Auto ARP response is not enabled. 1b = Enable Receive TCO packets. Setting this bit enables all manageability receive filtering operation. The enable of the specific filtering is done through loading the Receive Enable 1 word in the EEPROM, or through special configuration command (see Section 10.5.2.1.6).
RCV_ALL	1	Receive All Enable. When set to 1b, all LAN packets received over the wire that passed L2 filtering are forwarded to the BMC. This flag is meaningful only if the RCV_EN bit is set as well.
EN_STA	2	Enable Status reporting when set to 1b.
EN_ARP_RES	3	Enable ARP Response. 0b = Disable. The 82599 treats ARP packets as any other packet. These packets are forwarded to BMC if it passes other (non-ARP) filtering. 1b = Enable. The 82599 automatically responds to all received ARP requests that match its IP address. *Note:* Setting this bit doesn't change the Rx filtering settings. Appropriate Rx filtering to enable ARP request packets to reach the manageability unit should be set by the BMC or by the EEPROM. The BMC IP address is provided as part of the Receive Enable message (bytes 8-11). If short version of the command is used the 82599 uses IP address configured in the most recent long version of the command in which the EN_ARP_RES bit was set. If no such previous long command exists, then the 82599 uses the IP address configured in the EEPROM as ARP response IPv4 address in pass-through LAN configuration structure. If *CBDM* bit is set the 82599 uses the BMC dedicated Ethernet MAC address in ARP response packets. If the *CBDM* bit is not set, BMC uses the host Ethernet MAC address. Setting this bit requires appropriate assertion of bits RCV_EN and RCV_ALL. Otherwise, the command aborts with no processing.
NM	5:4	Notification Method. Defines the notification method that the 82599 uses. 00b = SMBus alert 01b = Asynchronous notify 10b = Direct receive 11b = Not supported. *Note:* In dual SMBus address mode, both SMBus addresses must be configured to the same notification method.
Reserved	6	Reserved.
CBDM	7	Configure BMC dedicated Ethernet MAC address. *Note:* This bit should be 0b when the RCV_EN bit (bit 0) is not set. 0b = The 82599 shares the same Ethernet MAC address for manageability and host defined in the EEPROM LAN Core 0/1 Modules in the EEPROM. 1b = The 82599 uses a dedicated Ethernet MAC address. The BMC Ethernet MAC address is set in bytes 2-7 in this command. If short version of the command is used, the 82599 uses the Ethernet MAC address configured in the most recent long version of the command in which the *CBDM* bit was set. If no such previous long command exists, then the 82599 uses the Ethernet MAC address configured in the MMAL and MMAH fields in the EEPROM. When the dedicated Ethernet MAC address feature is activated, the 82599 uses the following registers for Rx filtering. The BMC should not modify the following registers: MNG Decision Filter – MDEF7 (and its corresponding bit MANC2H[7]) MNG Ethernet MAC Address 3 – MMAL3 and MMAH3 (and its corresponding bit MFVAL[3]).

- MNG Ethernet MAC address (data bytes 2-7)

Ignored if CBDM bit is not set. This Ethernet MAC address is used for configuration of the dedicated Ethernet MAC address. In addition, it is used in the ARP response packet, when EN_ARP_RES bit is set. This Ethernet MAC address continues to be used when the *CBDM* bit is set in subsequent short versions of this command.

- MNG IP address (data bytes 8-11)

Ignored if EN_ARP_RES bit is not set. This IP address is used to filter ARP request packets. This IP address continues to be used when EN_ARP_RES is set in subsequent short versions of this command.

- Asynchronous notification SMBus address (data byte 12)

This address is used for the asynchronous notification SMBus transaction and for direct receive.

- Interface data (data byte 13)

Interface data byte to be used in asynchronous notification.

- Alert data (data byte 14).

Alert value data byte to be used in the asynchronous notification.

10.5.2.1.4 Force TCO Command

This command causes the 82599 to perform a TCO reset, if Force TCO reset is enabled in word Common Firmware Parameters in the EEPROM. The Force TCO reset clears the data path (Rx/Tx) of the 82599 to enable the BMC to transmit/receive packets through the 82599.

Note: In single address mode, both ports are reset when the command is issued. In dual address mode, Force TCO reset is asserted only to the port related to the SMB address the command was issued to.

The 82599 considers the Force TCO command as an indication that the operating system is hung and clears the DRV_LOAD flag.

Force TCO Reset command format:

Function	Command	Byte Count	Data 1
Force TCO reset	0xCF	1	TCO mode

TCO mode is listed in the following table:

Field	Bit(s)	Description
DO_TCO_RST	0	Do TCO reset. 0b = Do nothing. 1b = Perform TCO reset.
Reserved	1	Reserved, set to 0b.

Field	Bit(s)	Description
Firmware Reset[1]	2	Reset manageability and re-load manageability related EEPROM words 0b = Do nothing. 1b = Issue firmware reset to manageability. *Note:* Setting this bit generates a one time firmware reset event. Following a firmware reset, management related data from the EEPROM is loaded.
Reserved	7:3	Reserved, (Set to 0x00).

[1] Before initiating a Firmware Reset command, disable TCO receive via the Receive Enable command, set RCV_EN to 0b, and then wait for 200 milliseconds before initiating the Firmware Reset command. In addition, the BMC should not transmit during this period.

10.5.2.1.5 Management Control

This command is used to set generic manageability parameters. The parameters are listed in the following table. The command is 0xC1, which states that it is a management control command. The first data byte is the parameter number and the data afterwards (length and content) are parameter specific as listed in the table.

Note: If in the update configuration, the parameter that the BMC sets is not supported by the 82599, the 82599 does not NACK the transaction. After the transaction ends, the 82599 discards the data and asserts a transaction abort status (see Section 3.2.5.2).

Following is the format of the Management Control command:

Function	Command	Byte Count	Data 1	Data 2	...	Data N
Management Control	0xC1	N	Parameter Number (PN#)	Parameter Dependent		

This table lists the different parameters and their content:

Parameter	PN#	Parameter Data
Keep PHY Link Up	0x00	A single byte parameter — Data 2: Bit 0 Programming of the MMNGC.MNG_VETO bit. Bit [7:1] Reserved.

10.5.2.1.6 Update MNG RCV Filter Parameters

This command is used to set the manageability receive filters parameters. The parameters are listed in the following table. The command is 0xCC, which states that it is a parameter update. The first data byte is the parameter number and the data afterwards (length and content) are parameter specific as listed in the table.

Note: If in the update configuration, the parameter that the BMC sets is not supported by the 82599, the 82599 does not NACK the transaction. After the transaction ends, the 82599 discards the data and asserts a transaction abort status (see Section 3.2.5.2).

Detailed description of receive filtering capabilities and configuration is described in Section 10.3.

The format of the update MNG RCV filter parameters is listed in the following table:

Function	Command	Byte Count	Data 1	Data 2	...	Data N
Update MNG RCV Filter Parameters	0xCC	N	Parameter Number (PN#)	Parameter Dependent		

The following table lists the different parameters and their contents**:**

Parameter	PN#	Parameter Data
Filters Enable	0x1	Defines generic filters configuration. The structure of this parameter is 4 bytes as the MANC Value LSB and MANC Value MSB loaded from the EEPROM. *Note:* General filter enable is in the Receive Enable command, which enable receive filtering. This parameter specifies which filters should be enabled. ARP filtering and dedicated Ethernet MAC address can also be enabled through the Receive Enable command (see Section 10.5.2.1.3).
MNG2HOST configuration	0xA	This parameter defines which manageability packets are directed to the host memory as well. Data 2:5 = MNG2H register setting (Data 2 is the MSB).
Fail-Over configuration	0xB	Fail-Over Structure Configuration (see Section 10.2.2.2.4). The bytes of this parameter are loaded to the fail-over configuration register. Data 2:5 = Fail-over configuration register (Data 2 is the MSB).
Flex Filter 0 Enable MASK and Length	0x10	Flex Filter 0 Mask. Data 2:17 = MASK. Bit 0 in data 2 is the first bit of the MASK Data 18:19 = Reserved. Should be zero. Data 20 = Flexible Filter length (must be >= 2).
Flex Filter 0 Data	0x11	Data 2 – Group of flex filter's bytes: 0x0 = bytes 0-29. 0x1 = bytes 30-59. 0x2 = bytes 60-89. 0x3 = bytes 90-119. 0x4 = bytes 120-127. Data 3:32 = Flex filter data bytes. Data 3 is LSB. Group's length is not mandatory 30 bytes; it can vary according to filter's length and must NOT be padded by zeros.
Flex Filter 1 Enable MASK and Length	0x20	Same as parameter 0x10 but for filter 1.
Flex Filter 1 Data	0x21	Same as parameter 0x11 but for filter 1
Flex Filter 2 Enable MASK and Length	0x30	Same as parameter 0x10 but for filter 2.
Flex Filter 2 Data	0x31	Same as parameter 0x11 but for filter 2.
Flex Filter 3 Enable MASK and Length	0x40	Same as parameter 0x10 but for filter 3.
Flex Filter 3 Data	0x41	Same as parameter 0x11 but for filter 3.

Parameter	PN#	Parameter Data
Filters Valid	0x60	4 bytes to determine which of the the 82599 filter registers contain valid data. Loaded into the MFVAL0 and MFVAL1 registers. Should be updated after the contents of a filter register are updated. Data 2 = MSB of MFVAL ... Data 5 is the LSB
Decision Filters	0x61	5 bytes to load the Manageability Decision Filters (MDEF). Data 2 = Decision filter number. Data 3 = MSB of MDEF register for this decision filter ... Data 6 is the LSB.
VLAN Filters	0x62	3 bytes to load the VLAN tag filters (MAVTV). Data 2 = VLAN filter number. Data 3 = MSB of VLAN filter. Data 4 = LSB of VLAN filter.
Flex Ports Filters	0x63	3 bytes to load the manageability flex port filters (MFUTP). Data 2 = Flex port filter number. Data 3 = MSB of flex port filter. Data 4 = LSB of flex port filter.
IPv4 Filters	0x64	5 bytes to load the IPv4 address filter (MIPAF, DW 15:12). Data 2 = IPv4 address filter number (0-3). Data 3 = MSB of IPv4 address filter … Data 6 is the LSB.
IPv6 Filters	0x65	17 bytes to load IPv6 address filter (MIPAF). Data 2 = IPv6 address filter number (0-3). Data 3 = MSB of IPv6 address filter … Data 18 is the LSB.
MAC Filters	0x66	7 bytes to load Ethernet MAC address filters (MMAL, MMAH). Data 2 = Ethernet MAC address filters pair number (0-3). Data 3 = MSB of Ethernet MAC address … Data 8 is the LSB.
Ethertype Filters	0x67	6 bytes to load Ethertype filters (MTQF). Data 2 = METF filter index (valid values are 0..3). Data 3 = MSB of METF ... Data 6 is the LSB.
Extended Decision Filter	0x68	10 bytes to load the extended decision filters (MDEF_EXT & MDEF). Data 2 = MDEF filter index (valid values are 0..6). Data 3 = MSB of MDEF_EXT (DecisionFilter1) ... Data 6 is the LSB. Data 7 = MSB of MDEF (DecisionFilter0) ... Data 10 is the LSB. The command must overwrite any previously stored value. *Note:* Previous Decision Filter command (0x61) is still supported. For legacy reasons — If previous Decision Filter command (0x61) is called — it should set the MDEF as provided and set the extended Decision Filter (MDEF_EXT) to 0x0.

10.5.2.1.7 Update LinkSec Parameters

This command is used to set the manageability LinkSec parameters. The parameters are listed in the following table. The first data byte is the parameter number and the data afterwards (length and content) are parameter specific as listed in the table.

This is the format of the Update LinkSec parameters command:

Function	Command	Byte Count	Data 1	Data 2	...	Data N
Update LinkSec Filter Parameters	0xC9	N	Parameter Number (PN#)	Parameter Dependent		

The following table lists the different parameters and their contents**:**

Parameter	PN#	Parameter Data
Transfer LinkSec ownership to BMC	0x10	Data 2: Host Control: Bit 0 = Reserved. Bit 1 = Allow host traffic (0b – blocked, 1b – allowed). Bit 2...31 = Reserved.
Transfer LinkSec ownership to Host	0x11	No data needed.
Initialize LinkSec Rx	0x12	Data 2: Rx Port Identifier (MSB) ... Data 3: (LSB). Rx Port Identifier – the port number by which the 82599 identifies Rx packets. It is recommended that the BMC use 0x0 as the port identifier. *Note:* The BMC should use the same port identifier when performing the key-exchange. Data 4 : Rx MAC SecY (MSB) … Data 9: (LSB).
Initialize LinkSec Tx	0x13	Data 2: Tx Port Identifier (MSB) ... Data 3: (LSB) — must be set to zero. Data 4: Tx SCI (MSB) … Data 7: Tx SCI (LSB). Tx SCI – A 6-byte unique identifier for the LinkSec Tx CA. It is recommended that the BMC use its Ethernet MAC address value for this field. Data 8: Reserved. Data 9: Reserved. Data 10: Packet Number Threshold (MSB) … Data 12: (LSB). PN Threshold – When a new key is programmed, the packet number is reset to 0x1. With each Tx packet, The packet number is incremented by one and inserted to the packet (to avoid replay attacks). The packet number threshold value is 3 MSBytes of the Tx Packet number after which a Key Exchange Required AEN is sent to the BMC. Example: a PN threshold of 0x123456 means that when the packet number reaches 0x12345600 a notification is sent. Data 22: Tx Control — See Table 10-21.
Set LinkSec Rx Key	0x14	Data 2: Reserved. Data 3: Rx SA AN (The association number to be used with this key). Data 4: Rx LinkSec Key (MSB) … Data 19: (LSB) — (16 bytes key to be used).
Set LinkSec Tx Key	0x15	Data 3: Tx SA AN (The association number to be used with this key). Data 4: Tx LinkSec Key (MSB) … Data 19: (LSB) — (16 bytes key to be used).

Parameter	PN#	Parameter Data
Enable LinkSec Network Tx encryption	0x16	No data needed.
Disable LinkSec Network Tx encryption	0x17	No data needed.

Table 10-21 Tx Control

Bit	Description
0..4	Reserved.
5	Always Include SCI in Tx: 0b = Do not include SCI in Tx packets. 1b = Include SCI in Tx packets.
6..7	Reserved.

10.5.2.2 Read SMBus Transactions (the 82599 to BMC)

The following table lists the different SMBus read transactions supported by the 82599. All the read transactions are compatible with SMBus Read Block Protocol format.

TCO Command	Transaction	Command	Op-Code		Fragmentation	Section
Receive TCO Packet	Block Read	0xC0 or 0xD0	First: Middle: Last[1]	0x90 0x10 0x50	Multiple	10.5.2.2.1
Read Receive Enable configuration	Block Read	0xDA	Single:	0xDA	Single	10.5.2.2.7
Read the 82599 Status	Block Read	0xC0 or 0xD0 or 0xDE	Single:	0xDD	Single	10.5.2.2.3
Read Management parameters	Block Read	0xD1	Single:	0xD1	Single	10.5.2.2.5
Read MNG RCV filter parameters	Block Read	0xCD	Single:	0xCD	Single	10.5.2.2.6
Get system Ethernet MAC Address	Block Read	0xD4	Single	0xD4	Single	10.5.2.2.4
Read LinkSec parameters	Block Read	0xD9	Single	0xD9	Single	10.5.2.2.8

[1] Last fragment of the receive TCO packet is the packet status.

Note: The 82599 responds to one of the commands 0xC0/0xD0 within the time defined in the SMBus notification timeout and flags word in the EEPROM (see Section 6.4.4.3.)

0xC0/0xD0 commands are used for more than one payload. If the BMC issues these read commands, and the 82599 has no pending data to transfer, it always returns as default opcode 0xDD with the 82599 status, and does not NACK the transaction.

If an SMBus Quick Read command is received, it is handled as a Read the 82599 Status command (See Section 10.5.2.2.3 for details).

10.5.2.2.1 Receive TCO LAN Packet Transaction

The BMC uses this command to read the packet received on the LAN and its status. When the 82599 has a packet to deliver to the BMC, it asserts the SMBus notification, for the BMC to read the data (or direct receive). Upon receiving notification of the arrival of LAN receive packet, the BMC should begin issuing a Receive TCO packet command using the block read protocol. The packet can be delivered in more than one SMBus fragment (at least two — one for the packet, and the other one for the status), and the BMC should follow the *F* and *L* bit.

The opcode can have these values:

- 0x90 — First fragment.
- 0x10 — Middle fragment.
- 0x50 — Packet status (last fragment) as described in Section 10.5.2.2.2.

If the external BMC does not finish reading the entire packet within a timeout period since the packet has arrived, the packet is silently discarded. The timeout period is set according to the SMBus notification timeout EEPROM parameter (see Section 6.4.4.3)

Function	Command
Receive TCO packet	0xC0 or 0xD0

Data returned from the 82599:

Function	Byte Count	Data 1 (Op-Code)	Data 2	...	Data N
Receive TCO First Fragment	N	90	Packet Data Byte	...	Packet Data Byte
Receive TCO Middle Fragment		10			
Receive TCO Last Fragment		50			

10.5.2.2.2 Receive TCO LAN Status Payload Transaction

This transaction is the last transaction that the 82599 issues when a packet that was received from the LAN is transferred to the BMC. The transaction contains the status of the received packet. The format of the status transaction is as follows:

Function	Byte Count	Data 1 (Op-Code)	Data 2 – Data 17 (Status data)
Receive TCO Long Status	17 (0x11)	0x50	See Table 10-22. For more details on the specific bit fields see Section 7.1.6.

Table 10-22 Receive TCO Last Fragment Status Data Content

Name	Bit(s)	Description
Packet Length	13:0	Packet length including CRC, only 14 LSB bits.
Reserved	24:14	Reserved.
CRC	25	CRC stripped indication.
Reserved	28:26	Reserved.
VEXT	29	Additional VLAN present in packet.
Reserved	33:30	Reserved.
Reserved	34	Reserved.
LAN	35	LAN number.
Reserved	63:36	Reserved.
Reserved	71:64	Reserved.
Status	79:72	See Table 10-23.
Reserved	87:80	Reserved.
MNG status	127:88	See Table 10-24. This field should be ignored if Receive TCO is not enabled.

Table 10-23 Status Info

Field	Bit(s)	Description
Reserved	7:4	Reserved.
IPCS	3	IPv4 Checksum Calculated on Packet.
L4CS	2	L4 Checksum Calculated on Packet.