



BMC Networking

Table of contents

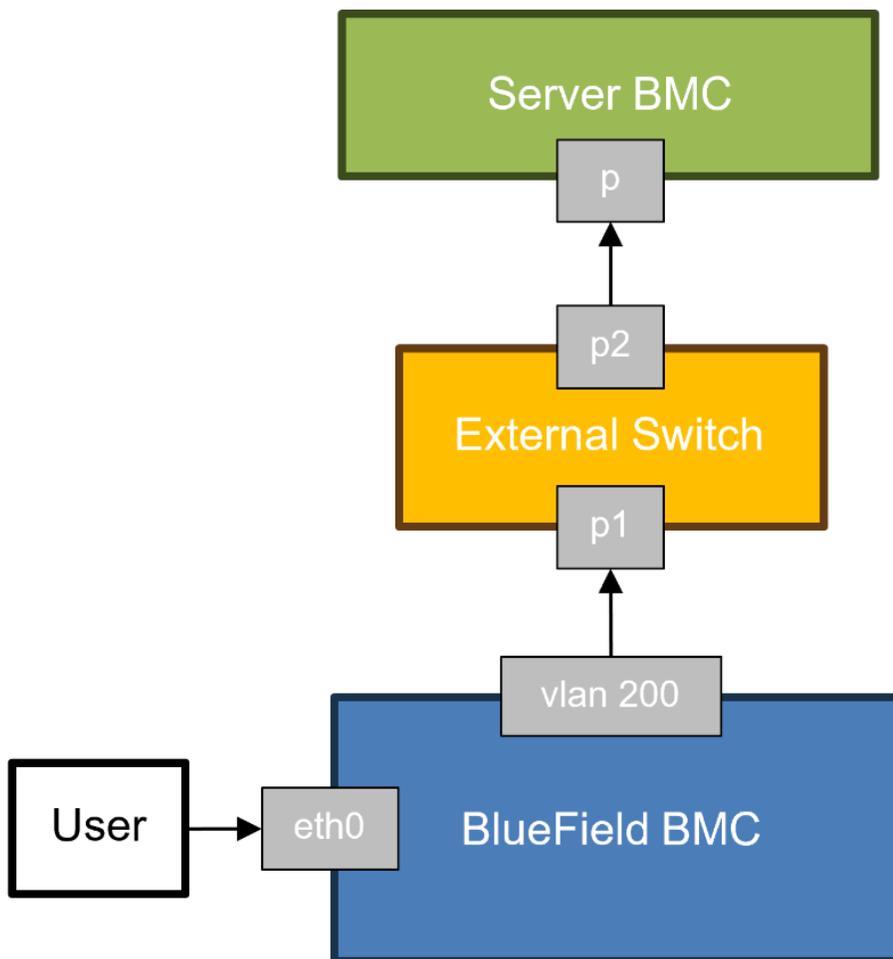
Guest Tunnel	3
Network Protocol Support	6

This section is comprised of the following subpages:

- [Guest Tunnel](#)
- [Network Protocol Support](#)

Guest Tunnel

The NVIDIA® BlueField® BMC is capable of establishing a designated VLAN interface to forward IPMI or HTTPS traffic from an external source to a specific IP address within that VLAN. Users can set a specific IP address for their server BMC, which allows for the management of their server BMC via the BlueField BMC.



To enable this feature, users must configure their network according to the following:

- Assign the remote server's BMC IP address as 192.168.1.10 to enable traffic forwarding
- The VLAN ID for the guest tunnel is 200, thus the external switch linked to the BlueField RJ45 port (OOB) must be set up to accept packets tagged with VLAN 200

The BlueField BMC currently supports the following ports, which act as source ports to accept messages sent by users and forward them to the server BMC within the guest tunnel:

- 8443 – Port on BlueField BMC for accepting HTTPS messages
- 8623 – Port on BlueField BMC for accepting IPMI messages

Info

The guest tunnel is intended solely for debugging purposes. Users should refrain from sending large amounts of network traffic through the guest tunnel, as it may impact the performance of the BlueField BMC.

Querying Guest Tunnel Status

netfunc	cmd	data	Notes
0x3E	0xFD	0x0	Displays the current configuration: <ul style="list-style-type: none"> • 0x01 – Disabled • 0x02 – Enabled

Disabling Guest Tunnel

netfunc	cmd	data	Notes
0x3E	0xFD	0x1	On success, returns: <ul style="list-style-type: none"> • 0x01 – Guest Tunnel is set to Disabled

Enabling Guest Tunnel

netfunc	cmd	data	Notes
0x3E	0xFD	0x2	On success, returns:

netfunc	cmd	data	Notes
			<ul style="list-style-type: none">• 0x02 – Guest Tunnel is set to Enabled

Example for enabling the guest tunnel on BlueField BMC:

```
#bmc> ipmitool raw 0x3e 0xfd 0x2
```

Example for sending the Redfish command to the guest tunnel:

```
#localhost> curl -k -u <bluefield_bmc_username>:  
<bluefield_bmc_password> -H 'content-type: application/json' -X GET  
https://<bluefield_bmc_ip>:8443/redfish/v1/Systems/Bluefield
```

Example for sending the IPMI command to the guest tunnel:

```
#localhost> ipmitool -p 8623 -C 17 -I lanplus -H  
<bluefield_bmc_ip> -U <bluefield_bmc_username> -P  
<bluefield_bmc_password> mc info
```

After receiving the responses from the Redfish and IPMI commands, the content of these responses originates from the server BMC, not the BlueField BMC.

Network Protocol Support

Note

To obtain the BMC MAC address, refer to the board label affixed to the NVIDIA® BlueField® device.

BMC management network interface can be configured using Redfish or IPMI. By default, BMC comes up with the DHCP network configuration.

Network configuration functions:

- Setting DHCP/Static network mode configuration
- Adding/setting IPv4/IPv6 configuration including IP address, gateway, netmask
- Adding DNS servers
- Adding NTP server
- Setting BMC time with NTP server or system RTC

Network Management Redfish Commands

Getting Network Protocol Configuration

```
curl -k -u root:'<password>' -X GET  
https://<bmc_ip>/redfish/v1/Managers/Bluefield_BMC/NetworkProtocol
```

Getting Interface Configuration

```
curl -k -u root:'<password>' -XGET
https://<bmc_ip>/redfish/v1/Managers/Bluefield_BMC/EthernetInterfac
```

Enabling/Disabling Interface

```
curl -k -u root:'<password>' -XPATCH
https://<bmc_ip>/redfish/v1/Managers/Bluefield_BMC/EthernetInterfac
-d '{"InterfaceEnabled": <state>'
```

Where `<state>` can be `true` or `false`.

Note

Disabling the `eth0` interface on the BlueField BMC prevents OOB network functionality on the BMC. This inhibits the ability to execute any Redfish or IPMI commands through the network.

Configuring Static IPv4 Address

```
curl -k -u root:'<password>' -X PATCH
https://<bmc_ip>/redfish/v1/Managers/Bluefield_BMC/EthernetInterfac
-d '{"IPv4StaticAddresses": [{"Address": "
<ip_addr>", "SubnetMask": "<netmask>", "Gateway": "<gw_ip_addr>"}]}'
```

Example:

```
curl -k -u root:'<password>' -X PATCH
https://<bmc_ip>/redfish/v1/Managers/Bluefield_BMC/EthernetInterface
-d '{"IPv4StaticAddresses": [{"Address": "10.7.7.7", "SubnetMask":
"255.255.0.0", "Gateway": "10.7.0.1"}]}'
```

Deleting Static IPv4 Address

```
curl -k -u root:'<password>' -X PATCH
https://<bmc_ip>/redfish/v1/Managers/Bluefield_BMC/EthernetInterface
-d '{"IPv4StaticAddresses": [null]}'
```

Enabling/Disabling IPv4 DHCP

```
curl -k -u root:'<password>' -X PATCH
https://<bmc_ip>/redfish/v1/Managers/Bluefield_BMC/EthernetInterface
-d '{"DHCPv4": {"DHCPEnabled": <state>}}'
```

Where `<state>` can be `true` or `false`.

Configuring Static DNS Server IPv4 and IPv6

```
curl -k -u root:'<password>' -X PATCH
https://<bmc_ip>/redfish/v1/Managers/Bluefield_BMC/EthernetInterface
-d '{"StaticNameServers": ["<dns_ip>"]}'
```

Configuring Static IPv6 Address

```
curl -k -u root:'<password>' -X PATCH
https://<bmc_ip>/redfish/v1/Managers/Bluefield_BMC/EthernetInterface
-d '{"IPv6StaticAddresses": [{"Address": "<ip>", "PrefixLength":
<len>}]}'
```

Example:

```
curl -k -u root:'<password>' -X PATCH
https://<bmc_ip>/redfish/v1/Managers/Bluefield_BMC/EthernetInterface
-d '{"IPv6StaticAddresses": [{"Address":
"fe80::3eec:efff:fe3b:e02f", "PrefixLength": 64}]}'
```

Enabling/Disabling IPv6 DHCP

```
curl -k -u root:'<password>' -X PATCH
https://<bmc_ip>/redfish/v1/Managers/Bluefield_BMC/EthernetInterface
-d '{"DHCPv6": {"OperatingMode": "<state>"}}'
```

Where `<state>` can be:

- `Enabled` – DHCPv6 is enabled for this interface
- `Disabled` – DHCPv6 is disabled for this interface

```
curl -k -u root:'<password>' -X PATCH
https://<bmc_ip>/redfish/v1/Managers/Bluefield_BMC/EthernetInterface
```

```
-d '{"StatelessAddressAutoConfig": {"IPv6AutoConfigEnabled": "  
<state>"}}'
```

Where `<state>` can be:

- `true` – Indicate IPv6 stateless address autoconfiguration (SLAAC) is enabled for this interface
- `false` – Indicate IPv6 stateless address autoconfiguration is disabled for this interface

Enabling/Disabling NTP

```
curl -k -u root:'<password>' -X PATCH  
https://<bmc_ip>/redfish/v1/Managers/Bluefield_BMC/NetworkProtocol  
-d '{"NTP": {"ProtocolEnabled": <state>}}'
```

Where `<state>` can be `true` or `false`.

Configuring Static NTP Server IP

```
curl -k -u root:'<password>' -X PATCH  
https://<bmc_ip>/redfish/v1/Managers/Bluefield_BMC/NetworkProtocol  
-d '{"NTP": {"NTPServers": ["<ntp_server_ip>"]}}'
```

Network Management IPMI Commands

The following subsections list the available network IPMI commands.

Configuring IPv4 Mode

The following command sets LAN channel 1 IP config mode to static or DHCP which corresponds to network interface `eth0`.

```
ipmitool lan set 1 ipsrc <mode>
```

Where `<mode>` can be `static` or `dhcp`.

Configuring IPv6 Mode

The following command sets LAN channel 1 IP config mode to static or DHCP which corresponds to network interface `eth0`.

```
ipmitool lan6 set 1 rtr_cfg <mode>
```

Where `<mode>` can be `static` or `dynamic`. `both` is unsupported.

Adding IPv4 Address

The following commands add IPv4 address, default gateway, and netmask to the network interface `eth0`.

- IP address:

```
ipmitool lan set 1 ipaddr <ip-address>
```

- Default gateway:

```
ipmitool lan set 1 defgw ipaddr <ip-address>
```

- Netmask:

```
ipmitool lan set 1 netmask <netmask>
```

Note

IPMI supports only a single static IP address. If multiple static IP addresses are configured on the system, the new netmask will be applied to only one of them.

Getting IPv4 Config

The following command gets IPv4 network config for channel 1 which corresponds to the network interface `eth0`.

```
ipmitool lan print 1
```

Setting IPv6 Address

The following command adds IPv6 address to the network interface `eth0`.

```
ipmitool lan6 set 1 nlock static_addr 0 enable <ipv6-address> 64
```

Getting IPv6 Config

The following command gets IPv6 network config for channel 1 which corresponds to the network interface `eth0`.

```
ipmitool lan6 print 1
```

Getting DNS Server

```
ipmitool raw 0x32 0x6B
```

Output:

```
0b 31 30 2e 31 35 2e 31 32 2e 36 37
```

This output corresponds to `10.15.12.67`.

Adding DNS Server

```
ipmitool raw 0x32 0x6C 0x0b 0x31 0x30 0x2e 0x31 0x35 0x2e 0x31  
0x32 0x2e 0x36 0x37
```

Output:

```
0x0b 0x31 0x30 0x2e 0x31 0x35 0x2e 0x31 0x32 0x2e 0x36 0x37
```

This output corresponds to `10.15.12.67`.

Getting NTP Server

```
ipmitool raw 0x32 0xA7
```

Output:

```
01 11 31 2e 69 6e 2e 70 6f 6f 6c 2e 6e 74 70 2e 6f 72 67
```

Where:

- `01` – NTP status enable/disable
- `11` – NTP server length
- `31 2e 69 6e 2e 70 6f 6f 6c 2e 6e 74 70 2e 6f 72 67` – NTP server address byte stream which corresponds to `1.in.pool.ntp.org`

Adding NTP Server

```
ipmitool raw 0x32 0xA8 0x01 0x31 0x2e 0x69 0x6e 0x2e 0x70 0x6f  
0x6f 0x6c 0x2e 0x6e 0x74 0x70 0x2e 0x6f 0x72 0x67
```

Where:

- `31 2e 69 6e 2e 70 6f 6f 6c 2e 6e 74 70 2e 6f 72 67` – NTP server address byte stream which corresponds to `1.in.pool.ntp.org`

Enabling NTP Time Sync

The following command enables time sync to NTP server.

```
ipmitool raw 0x32 0xA8 0x02 0x01
```

Where:

- `0x01` – enable NTP

Disabling NTP Time Sync

The following command disables time sync to NTP server.

```
ipmitool raw 0x32 0xA8 0x02 0x00
```

Where:

- `0x00` – disable NTP

Configuring Router IPv6 Mode

The following command sets router mode to static or DHCP.

```
ipmitool lan6 set 1 rtr_cfg <mode>
```

Where `<mode>` can be:

- static
- Dynamic

(i) Note

Configuring static mode also requires setting the static router IP and static router MAC address.

(i) Note

Router prefix can only be 0.

Configuring IPv6 Static Router IP

The following command sets the IPv6 address for the static router.

```
ipmitool raw 0x0c 0x01 0x01 0x41 <ip-hex>
```

Where:

- `<ip-hex>` – the IP address

Configuring IPv6 Static Router MAC

The following command sets the MAC address for the static router.

```
ipmitool raw 0x0c 0x01 0x01 0x42 <mac-hex>
```

Where:

- `<mac-hex>` – the IP MAC address

Notice

This document is provided for information purposes only and shall not be regarded as a warranty of a certain functionality, condition, or quality of a product. NVIDIA Corporation ("NVIDIA") makes no representations or warranties, expressed or implied, as to the accuracy or completeness of the information contained in this document and assumes no responsibility for any errors contained herein. NVIDIA shall have no liability for the consequences or use of such information or for any infringement of patents or other rights of third parties that may result from its use. This document is not a commitment to develop, release, or deliver any Material (defined below), code, or functionality.

NVIDIA reserves the right to make corrections, modifications, enhancements, improvements, and any other changes to this document, at any time without notice.

Customer should obtain the latest relevant information before placing orders and should verify that such information is current and complete.

NVIDIA products are sold subject to the NVIDIA standard terms and conditions of sale supplied at the time of order acknowledgement, unless otherwise agreed in an individual sales agreement signed by authorized representatives of NVIDIA and customer ("Terms of Sale"). NVIDIA hereby expressly objects to applying any customer general terms and conditions with regards to the purchase of the NVIDIA product referenced in this document. No contractual obligations are formed either directly or indirectly by this document.

NVIDIA products are not designed, authorized, or warranted to be suitable for use in medical, military, aircraft, space, or life support equipment, nor in applications where failure or malfunction of the NVIDIA product can reasonably be expected to result in personal injury, death, or property or environmental damage. NVIDIA accepts no liability for inclusion and/or use of NVIDIA products in such equipment or applications and therefore such inclusion and/or use is at customer's own risk.

NVIDIA makes no representation or warranty that products based on this document will be suitable for any specified use. Testing of all parameters of each product is not necessarily performed by NVIDIA. It is customer's sole responsibility to evaluate and determine the applicability of any information contained in this document, ensure the product is suitable and fit for the application planned by customer, and perform the necessary testing for the application in order to avoid a default of the application or the product. Weaknesses in customer's product designs may affect the quality and reliability of the NVIDIA product and may result in additional or different conditions and/or requirements beyond those contained in this document. NVIDIA accepts no liability related to any default, damage, costs, or problem which may be based on or attributable to: (i) the use of the NVIDIA product in any manner that is contrary to this document or (ii) customer product designs.

No license, either expressed or implied, is granted under any NVIDIA patent right, copyright, or other NVIDIA intellectual property right under this document. Information published by NVIDIA regarding third-party products or services does not constitute a license from NVIDIA to use such products or services or a warranty or endorsement thereof. Use of such information may require a license from a third party under the patents or other intellectual property rights of the third party, or a license from NVIDIA under the patents or other intellectual property rights of NVIDIA.

Reproduction of information in this document is permissible only if approved in advance by NVIDIA in writing, reproduced without alteration and in full compliance with all applicable export laws and regulations, and accompanied by all associated conditions, limitations, and notices.

THIS DOCUMENT AND ALL NVIDIA DESIGN SPECIFICATIONS, REFERENCE BOARDS, FILES, DRAWINGS, DIAGNOSTICS, LISTS, AND OTHER DOCUMENTS (TOGETHER AND SEPARATELY, "MATERIALS") ARE BEING PROVIDED "AS IS." NVIDIA MAKES NO WARRANTIES, EXPRESSED, IMPLIED, STATUTORY, OR OTHERWISE WITH RESPECT TO THE MATERIALS, AND EXPRESSLY DISCLAIMS ALL IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY, AND FITNESS FOR A PARTICULAR PURPOSE. TO THE EXTENT NOT PROHIBITED BY LAW, IN NO EVENT WILL NVIDIA BE LIABLE FOR ANY DAMAGES, INCLUDING WITHOUT LIMITATION ANY DIRECT, INDIRECT, SPECIAL, INCIDENTAL, PUNITIVE, OR CONSEQUENTIAL DAMAGES, HOWEVER CAUSED AND REGARDLESS OF THE THEORY OF LIABILITY, ARISING OUT OF

ANY USE OF THIS DOCUMENT, EVEN IF NVIDIA HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES. Notwithstanding any damages that customer might incur for any reason whatsoever, NVIDIA's aggregate and cumulative liability towards customer for the products described herein shall be limited in accordance with the Terms of Sale for the product.

Trademarks

NVIDIA and the NVIDIA logo are trademarks and/or registered trademarks of NVIDIA Corporation in the U.S. and other countries. Other company and product names may be trademarks of the respective companies with which they are associated.

Copyright 2025. PDF Generated on 12/15/2025