Problem determination and troubleshooting

How to perform problem determination actions on the SR780a V3

Lenovo

Problem determination and troubleshooting overview

Perform the following actions to determine the cause of problems on the SR780a V3

- Check the system health status on the XCC2 dashboard or operation panel on the front of the system
- Check the system event log in XCC2
- Check the event log in UEFI
- Check the LEDs on the system
- Check the integrated LCD diagnostics handset

For more information about how to use XCC2, UEFI, or OneCLI to monitor system status and collect logs, refer to the following courses:

- ES51757B – Introducing ThinkSystem tools
- ES52374 – ThinkSystem tools for the ThinkSystem V3 platform
- ES41759C – ThinkSystem problem determination

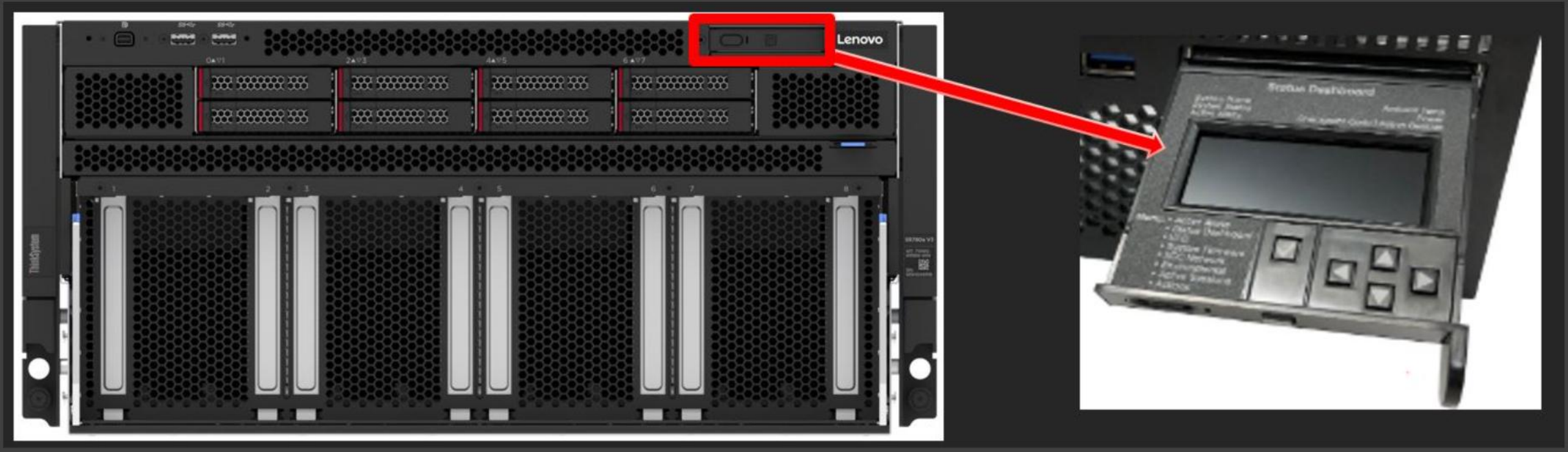
Note: The SR780a V3 does not support the optional external diagnostic panel.

LED descriptions

- Use the LEDs on the front drives, front operator panel, and the rear side of the server for hardware status monitoring and problem determination.
- The front and rear fans do not have status LEDs.
- The system board assembly does not support the internal LED light path feature. As the SR780a V3 does not have top cover, you will not be able to see the system board assembly unless you power off the system and remove the system board tray from the chassis.
- Click HERE to see the location of the SR780a V3 front LCD diagnostic panel.
- For more information about the SR780a V3 LEDs, refer to the *Server components* section of the *ThinkSystem SR780a V3 User Guide* on Lenovo Docs.

LED descriptions

SR780a V3 front LCD diagnostic panel



A demo video explaining how to use the LCD diagnostic panel is available on the course landing page.

GPU problem determination

Use the following tools to monitor or collect service data for GPU problem escalation:

- Use XCC to check the GPU inventory information
- If necessary, use the following command line tools in the OS to collect service data
  - Linux commands
  - NVIDIA driver bundle SMI (system management interface) CLI tool
  - When a failure occurs, if possible, collect all relevant data while the failure condition is still present and before any recovery action is taken
  - When collecting the NVIDIA support bundle, be aware that an AC power cycle will reset everything and the failure condition data will be lost – The NVIDIA_bug_report will show no errors after a power restart

Note:

- If a GPU problem occurs that might require a GPU or GPU baseboard replacement, escalate to the PE level of support for further problem diagnosis.
- The GPU connector and GPU slot are fragile. To avoid damaging either component, do not move a potentially failed GPU to another slot for problem determination.

Monitoring GPU status with XCC

GPU inventory and firmware information is available in XCC. Servicers can also find GPU sensor information in the XCC logs.
Click the following links to see screenshots:

- H100 GPU inventory list in XCC
- H100 GPU details in XCC
- B200 GPU inventory list in XCC
- B200 GPU details in XCC
- GPU inventory examples in the XCC FFDC mini log

Monitoring GPU status with Linux commands

In a Linux environment, use the following commands to collect GPU-related information or the system log and then export the command line output as a txt file for further analysis:

- Collecting GPU information: `dmesg | grep <amd or nvidia> > <folder/filename>`
  Example: `dmesg | grep amd > /tmp/dmesg.txt`

```
root@monaco:/# dmesg | grep amd > /tmp/dmesg.txt
```

```
root@monaco:/# cat /tmp/dmesg.txt
[    0.000000] Linux version 5.15.0-105-generic (buildd@lcy02-amd64-007) (gcc (Ubuntu 11.4.0-1ubuntu1~22.0
4) 11.4.0, GNU ld (GNU Binutils for Ubuntu) 2.38) #115-Ubuntu SMP Mon Apr 15 09:52:04 UTC 2024 (Ubuntu 5.1
5.0-105.115-generic 5.15.148)
[    1.918249] perf/amd_iommu: Detected AMD IOMMU #0 (2 banks, 4 counters/bank).
[    1.918270] perf/amd_iommu: Detected AMD IOMMU #1 (2 banks, 4 counters/bank).
[    1.918293] perf/amd_iommu: Detected AMD IOMMU #2 (2 banks, 4 counters/bank).
[    1.918316] perf/amd_iommu: Detected AMD IOMMU #3 (2 banks, 4 counters/bank).
[    1.918338] perf/amd_iommu: Detected AMD IOMMU #4 (2 banks, 4 counters/bank).
[    1.918359] perf/amd_iommu: Detected AMD IOMMU #5 (2 banks, 4 counters/bank).
[    1.918383] perf/amd_iommu: Detected AMD IOMMU #6 (2 banks, 4 counters/bank).
[    1.918408] perf/amd_iommu: Detected AMD IOMMU #7 (2 banks, 4 counters/bank).
[    3.717962] amdkcl: loading out-of-tree module taints kernel.
[    3.717962] amdkcl: loading out-of-tree module taints kernel.
[    3.718017] amdkcl: module verification failed: signature and/or required key missing - tainting kernel
[    4.055041] [drm] amdgpu kernel modesetting enabled.
[    4.056679] [drm] amdgpu version: 6.7.0
[    4.063171] amdgpu: Virtual CRAT table created for CPU
[    4.072045] amdgpu: Topology: Add CPU node
[    4.282580] amdgpu: PeerDirect support was initialized successfully
[    4.350903] amdgpu 0000:45:00.0: amdgpu: Fetched VBIOS from ROM BAR
[    4.351626] amdgpu: ATOM BIOS: 113-MI35RIOV-001
[    4.352467] amdgpu 0000:45:00.0: [drm:vcn_v4_0_3_early_init [amdgpu]] VCN decode is enabled in VM mode
[    4.353928] amdgpu 0000:45:00.0: [drm:jpeg_v4_0_3_early_init [amdgpu]] JPEG decode is enabled in VM mod
```

- Collecting error messages in the system log folder:
  `demsg | grep -i error /var/log/<system log folder name depends on the OS> > <output folder/filename>`
  Example: `demsg | grep -i error /var/log/syslog > /tmp/amdsyslog.txt`

Lenovo

Monitoring GPU status with SMI commands

System Management Interface (SMI) is a command line utility bundle for GPU drivers. SMI commands can be used to manage and monitor GPU devices. NVIDIA GPUs support their own version of SMI.
Click the following links to see examples of SMI command outputs:

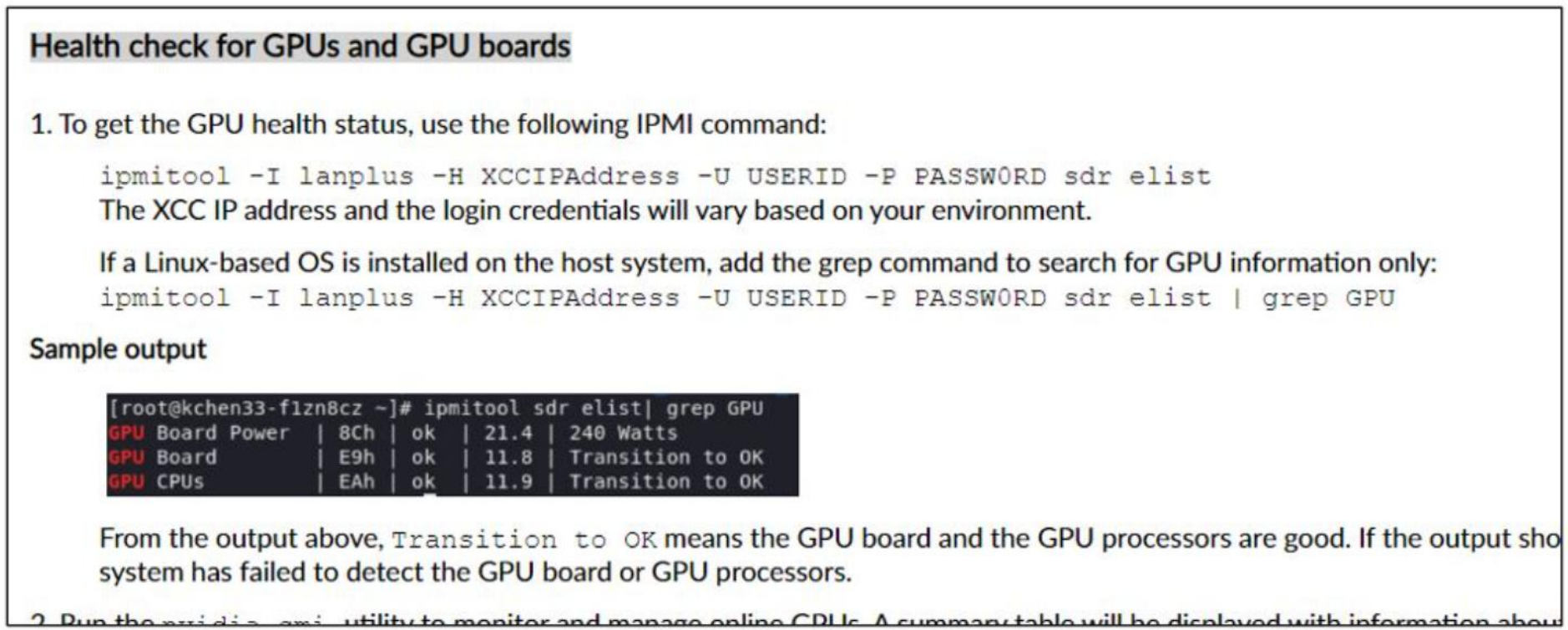
- `nvidia-smi` (to display NVIDIA GPUs installed in the system)
- `nvidia-smi -L` (to display NVIDIA GPUs installed in the system with UUID)
- `nvidia-smi -q --id=1 -f <output file name>` (to export NVIDIA GPU inventory information)
- `nvidia-smi pci --getErrorCounters` (to display NVIDIA GPUs error counters data)

For more information about NVIDIA SMI information, refer to the following websites:

- NVIDIA SMI

NVIDIA GPU and GPU board problem determination

The problem determination steps for NVIDIA GPUs are the same as those used with the previous generation of GPUs. If necessary, collect XCC service data or use the `nvidia-bug-report.sh` command to collect a GPU bug report for problem escalation.
Refer to the following article for complete instructions of how to monitor NVIDIA GPU status and how to collect NVIDIA GPU logs: https://support.lenovo.com/tw/en/solutions/ht512069



Health check for GPUs and GPU boards

1. To get the GPU health status, use the following IPMI command:

```
ipmitool -I lanplus -H XCCIPAddress -U USERID -P PASSW0RD sdr elist
```

The XCC IP address and the login credentials will vary based on your environment.

If a Linux-based OS is installed on the host system, add the grep command to search for GPU information only:

```
ipmitool -I lanplus -H XCCIPAddress -U USERID -P PASSW0RD sdr elist | grep GPU
```

Sample output

```
[root@kchen33-f1zn8cz ~]# ipmitool sdr elist| grep GPU
GPU Board Power  | 8Ch | ok  | 21.4 | 240 Watts
GPU Board        | E9h | ok  | 11.8 | Transition to OK
GPU CPUs         | EAh | ok  | 11.9 | Transition to OK
```

From the output above, `Transition to OK` means the GPU board and the GPU processors are good. If the output sho
system has failed to detect the GPU board or GPU processors.

2. Run the nvidia-smi utility to monitor and manage online GPUs. A summary table will be displayed with information abou