LLNL IPMI Plans, Random IPMI Ideas, & FreeIPMI Issues

Albert Chu
chu11@llnl.gov

December 6th, 2005

UCRL-PRES-217495

LLNL IPMI Plans

Albert Chu

chu11@llnl.gov

December 6th, 2005

Why FreeIPMI?

- Thunder: 1024 node Linux Itanium2 cluster
  - FreeIPMI part of the Thunder contract.
  - LLNL guided portions of FreeIPMI development
    - Contributed ipmipower, bmc-watchdog, ipmiping, rmcpping, and significant portions of libfreeipmi.
    - HPC Requirements
    - Interface
    - Features

Why FreeIPMI?

- Other IPMI tools didn't meet our needs:
  - OpenIPMI:
    - kernel driver: Didn't work 2 years ago.
    - Tools: ipmicmd required hex code input.
    - library: We were confused back then.
  - Ipmitool: No HPC needs met.
  - Vendor Tools: No HPC, GUI only ...

Current FreeIPMI Use

- FreeIPMI tools employed:
  - bmc-config
    - Config approx. 1050 machines, more on the way
    - Use config file checkout/checkin
  - libfreeipmi – environmentals host monitoring
    - Currently monitor via In-Band
      - Two years ago, couldn't easily integrate out of band.

Current FreeIPMI Use

- FreeIPMI tools employed:
  - ipmipower
    - Run w/ Powerman in Co-Processor mode
      - Tool to manipulate RPC devices from a central location.
    - Power status query 1024 nodes 0.3 seconds
    - Detects non-existent nodes
  - ipmiping, rmcpping
    - General debugging.
    - Very useful early on during integration.

Current FreeIPMI Use

- FreeIPMI tools employed:
  - ipmi-sel, ipmi-sensors
    - General monitoring and debugging.
    - Currently use 'pdsh' to run across cluster.
      - Pdsh is a high-performance, parallel remote shell utility.

IPMI Needs

- IPMI will likely be needed for future clusters.
  - LANL & Sandia will likely need it too.
- FreeIPMI tools need to be updated for IPMI 2.0.
- Need new features to eliminate in-band access.
- Need to HPC-ify remaining tools.
- Need SOL.

IPMI TODO - FreeIPMI

- IPMI 2.0 Support in FreeIPMI
  - CVS Branch: al_ipmi_2_0_branch
  - ipmiping: Done (but not tested)
  - libfreeipmi: IPMI 2.0 support.
    - Packet templates (Done, I think).
    - Packet assembly API (Half done).
    - Crypto and helper functions API (Half done).

IPMI TODO - FreeIPMI

- IPMI 2.0 Support in FreeIPMI
  - bmc-config, ipmi-sel, bmc-info, ipmi-sensors
    - After libfreeipmi API is complete.
      - Zresearch task?
  - ipmipower
    - After libfreeipmi API is complete.
      - Me

IPMI TODO - FreeIPMI

- Libfreeipmi In-Band Driver:
  - Will there be issues with next generation hardware?
  - Support IPMI kernel driver?
    - More support from vendors and Linux community.
      - And help support the kernel driver through testing.
    - FreeIPMI In-Band Driver for BSD, etc.

IPMI TODO - FreeIPMI

- Multihost LAN access
  - Over LAN to one node at a time is useless.
- Host Monitoring:
  - Need to make it work out of band.
  - Will it require work in libfreeipmi?

IPMI TODO - SOL

- Need to support in Conman
  - Performs logging, interactive sessions, shared sessions, and console broadcast.
- Support in Conman?
  - Conman establishes SOL sessions to nodes.
- SOL Daemon?
  - Daemon establishes SOL sessions to nodes.
  - Conman telnet/ssh connects to SOL daemon.

IPMI TODO - SOL

- LLNL will implement IPMI into Conman.
  - We believe it'll be done more quickly.
  - Is this the best approach?
- Multiple approaches are good.
  - Daemon: more support from community?
    - Work around corner cases/bugs more quickly?
  - More scalable?
  - Something vendors want?

IPMI Longer Term Goals

- Vendor Compliance Issues
  - Does ICTS work?
    - On the Ipmitool mailing list, I've already read about IPMI 2.0 non-compliance and different implementations.
  - Needed for security
    - Atleast one vendor I worked with had unchangeable BMC configurations critical for security.
  - For vendors: Good for business
    - Make sure what you have actually works.

IPMI Longer Term Goals

- How to get vendor compliance
  - Enterprise tools vendors distribute?
    - What do vendors/customers want?
      - Easier interfaces/tools?
      - A Web GUI?
      - Compiles on Windows?
  - A common IPMI library used by vendors and open source Linux developers?
- Linux IPMI testsuite?
  - More later

Other Work

- SLAC/Stanford
  - Ganglia support for IPMI host monitoring

Random IPMI Ideas

Albert Chu
chu11@llnl.gov

December 6th, 2005

IPMI Lib

- Everyone has a library of sorts that stuffs bits into a char *buf.
  - Any reason we can't share?
- Start w/ shared header files?
  - Yes, I know I brought this up a year ago, and it died.
  - Should I try again?
    - How to package/distribute with installed libraries.

Linux IPMI test suite

- OpenIPMI has a sanity testsuite.
- Zresearch says they're adding something for FreeIPMI.
- LLNL may do something w/ ipmitool.
- Collect into a “Linux IPMI test suite” project?
  - Bunch of people can pitch in chunks?

Ipmitool + Ipmipower

- Some early discussions with Ipmitool Author

```
> ipmitool -I lan -H host -U user
  -P passwd chassis status
System Power         : on
Power Overload       : false
...
```

Ipmitool + Ipmipower

```
> ipmipower -h host[0-10] -u user
  -p passwd –outputtype=hostrange –
  stat
```

```
host[0-8]: on
host[9-10]: off
```

- Command line interface is similar

Ipmitool + Ipmipower

- Ipmitool
  - Wide breadth of IPMI command support
  - No HPC stuff
- Ipmipower
  - Only power control commands
  - HPC: Scales, multihost comm., output aggregation
- ???

FreeIPMI Issues

Albert Chu

chu11@llnl.gov

December 6th, 2005

CVS - Tagging

- Send out e-mail to alert developers before a release tag.
  - I often need to tweak the RPM spec file just a bit.
    - i.e. Add a new document that's been added since the last time I edited the RPM spec file.

CVS - Branching

- Do we need stable and devel branches?
  - “wait for the next release” policy is not working.

Fish/Scheme

- From the GNU webpage:
  - "more people know C, so more people will find it easy to read and modify the program if it is written in C."
- FreeIPMI 0.1.3: Scheme “wrapped” C code.
- FreeIPMI 0.2.0: Scheme now implements significant portions of the fish tools
  - ipmi-sel, bmc-info, bmc-config, ipmi-sensors

Fish/Scheme

- From the mailing list:
  - Users don't know how to debug Scheme.
    - Can't do basic printf debugging.
    - Trail of wrapper functions can get confusing.
  - Users don't know Scheme and can't contribute.
- My opinion is its a big “barrier to entry”
  - i.e. what is the “int argc, char **argv” equivalent?

Fish/Scheme

- Do people use Fish? Or do they use the individual command line tools?
- Is Scheme really the best way to implement these tools?

Libfreeipmi – IPMI 2.0 Session

- IPMI 1.5: Individual LAN packets loosely independent of each other.
  - Need to "carry around" session id from session setup for each LAN packet.
  - Need to "carry around" password for each packet.
  - Only need to manage sequence numbers in packets.

Libfreeipmi – IPMI 2.0 Session

- IPMI 2.0: Additional data to “carry around” and manage than before.
  - Authentication, Integrity, Confidentiality algorithm
  - Generated Keys
    - Dependent on algorithms selected during session setup.
  - Session setup: packet fields are hashed dependent on data from previous packets.

Libfreeipmi – IPMI 2.0 Session

- Idea: Create a IPMI session context?
  - ipmi_session_seq_num_increment(&ctx);
  - ipmi_session_set(&ctx, IPMI_RAKP_2, obj);
  - ipmi_session_set(&ctx, IPMI_RAKP_4, obj);
  - assemble_ipmi_lan_pkt(&ctx, ...);
- This will keeps packet assembly functions independent of UDM?

Libfreeipmi - fiid_template_t

```
typedef struct fiid_field
{
  u_int32_t len;
  char key[FIID_FIELD_MAX];
} fiid_field_t;


typedef fiid_field_t const
 fiid_template_t[];
```

Libfreeipmi - fiid_template_t

- Policy: The templates never deviate from the IPMI spec.
  - Convenience templates may be created as needed.

Libfreeipmi - fiid_template_t

- IPMI 1.5
  - Fields in the spec are required, with a few exceptions.
    - Auth-Code isn't required if authtype == none.
    - Chassis Identify interval (byte #1) is optional.
    - Get Chassis Status Response optionally returns a Front Panel Status.
  - To my knowledge all fields are a fixed length.
    - i.e. Usernames/Passwords are zero padded.

Libfreeipmi - fiid_template_t

- IPMI 2.0
  - Many more fields are optional
    - OEM fields dependent on payload type.
    - Payload padding may not be necessary in some cases.
    - No integrity data if:
      - authtype == NONE
      - you're in the session setup
    - Chassis Identify
      - Optional byte #2 is dependent on optional byte #1 being sent.

Libfreeipmi - fiid_template_t

- IPMI 2.0
  - Many fields are variable length:
    - Padding length dependent on payload size.
    - Payload length dependent on encryption type
      - i.e. Should an initialization vector be sent?
    - Authentication/Integrity length depends on hash algorithm
      - HMAC-SHA1 – 20 byte digest
      - HMAC-MD5 or MD5 – 16 byte digest
      - HMAC-SHA1-96 – 12 byte digest
    - Passwords can be padded to 16 or 20 bytes.

Libfreeipmi - fiid_template_t

- The old template style won't work.
  - Should a field be set? Should it be sent?
    - It's a lot more complicated now.
  - How long is a field?
    - Short term temporary solution:

```
{256, "auth_code"},
{32,  "auth_code_len"}, /* XXX */
```

Libfreeipmi - fiid_template_t

- Idea 1:

```
struct fiid_obj
{
    void *buffer;
    u_int32_t buflen;
    fiid_field_info_t *info;
    u_int32_t infolen;
};
```

Libfreeipmi - fiid_template_t

- Idea 2:

```
struct fiid_obj
{
    void **data;
    fiid_field_info_t *info;
    u_int32_t infolen;
};
```

Libfreeipmi - fiid_template_t

- Idea 3:

```
struct X_template_t
{
   fiid_template_t template;
   u_int32_t auth_code_len;
   /* And extra stuff we want */
};
```

Libfreeipmi – Convenience Code

- Libfreeipmi is getting BIG
  - A significant amount is “convenience code”:
    - ipmi-sensors code
    - ipmi-ping code
  - Move into other libs?
  - Restructure some code?
- Split Lib?

Multihost LAN

- How to do it?
  - ipmipower “polling loop” model
    - like OpenIPMI model
  - pdsh “window of threads” model
  - other?