

# The Power of IPMI

Werner Fischer  
Technology Specialist  
Thomas-Krenn.AG



2<sup>nd</sup> Open Source Data Center Conference – OSDC 2010  
Nuremberg, 23<sup>rd</sup> and 24<sup>th</sup> June 2010



**Thomas-Krenn.AG**<sup>®</sup>  
Speed is (y)our success



# Agenda



- 1) About Thomas Krenn**
- 2) IPMI basics**
- 3) IPMI details**
- 4) Example: Nagios/Icinga IPMI Plugin**
- 5) Conclusions**



# 1) About Thomas Krenn



- **Server systems, virtualization and accessories**  
**"Made in Germany"**
- **Unique service and support and 24h express delivery**
- **History**
  - 2002: founded by Max Wittenzellner and Thomas Krenn
  - 2005: turned into stock corporation
  - 2008: DIN ISO 9001:2000 certification
  - today:
    - 70 employees – 20 of them being technicians ;-)
    - over 9.000 customers



# Agenda



## 2) IPMI basics

- IPMI main features
- IPMI Illustration
- IPMI Messaging Interfaces
- Channel Privilege Levels
- IPMI Software overview

## 3) IPMI details

## 4) Example: Nagios/Icinga IPMI Plugin

## 5) Conclusions



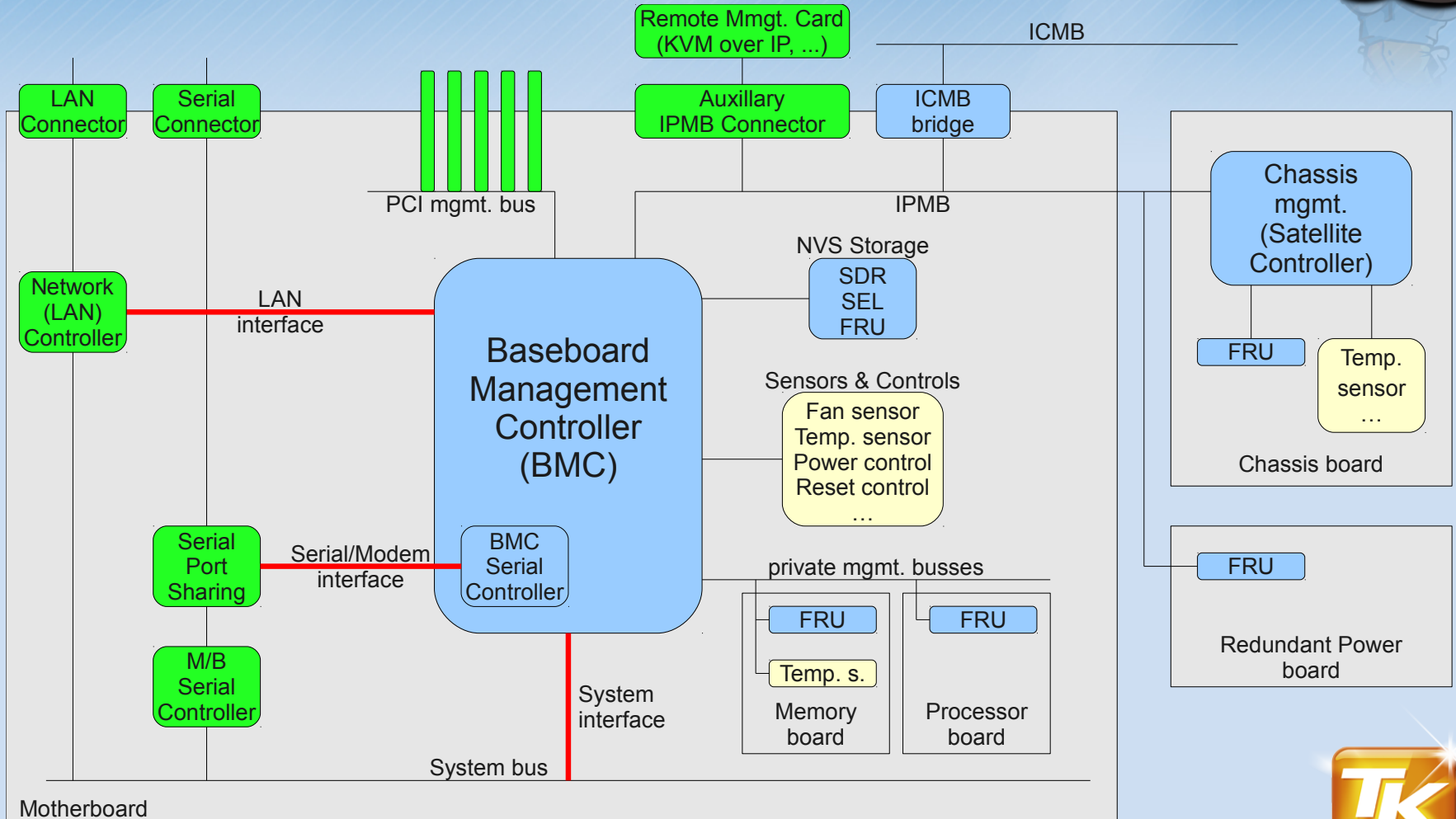
## 2) IPMI basics



- **IPMI = Intelligent Platform Management Interface, developed by Intel, HP, NEC, Dell**
  - 1998: IPMI v1.0
  - 2001: IPMI v1.5
  - 2004: IPMI v2.0
- **IPMI main features:**
  - Monitoring (temperatures, fans, voltages, etc.)
  - Recovery Control (power on/off/reset a server)
  - Logging (System Event Log)
  - Inventory (FRU information)



# 2) IPMI basics



## 2) IPMI basics



- **IPMI Messaging Interfaces (request/response protocol)**
  - System Interfaces
    - local access
    - requires root privileges
  - Serial/Modem Interface
    - access via serial interface or modem
    - requires IPMI user name/password (deactivate auth. NONE)
  - LAN Interface
    - access via network
    - requires IPMI user name/password (deactivate auth. NONE)
  - (ICMB and PCI Management Bus)



## 2) IPMI basics



- **Channel Privilege Levels (for LAN/Serial access)**

Privilege Level	Description
Callback	Lowest Privilege Level. Allows only initiating a callback.
User	Allows only IPMI 'begin' commands (query sensors). Changing the BMC configuration, writing data to the BMC, executing power on/off or reset commands is prohibited.
Operator	Allows nearly all IPMI commands. Only changes of out-of-band interfaces are prohibited.
Administrator	Allows all IPMI commands.

- **use privilege level 'User' for monitoring purposes**





## 2) IPMI basics



- **IPMI Software overview**

	<b>ipmitool</b>	<b>ipmiutil</b>	<b>freeipmi</b>	<b>OpenIPMI (openipmish)</b>
<i>OS Support</i>	Linux, BSD, Solaris, (Windows)	Linux, BSD, Solaris, Windows, EFI	Linux, BSD, Solaris, (Windows)	Linux
<i>Target Market</i>	admins, developers, oem's	admins, developers, oem's	HPC, universities, admins, ...	(Kernel-driver)
<i>Included in Linux Distros</i>	Debian, Gentoo, RedHat, SLES, Ubuntu (univ.)	Gentoo	Debian (Squeeze), Gentoo, RedHat (exp.) Ubuntu (univ.),	Debian, Gentoo, RedHat, SLES, Ubuntu



# Agenda



## 3) IPMI details

- IPMI example configuration of a LAN interface
- Remote Control
- Sensors
  - Sensor Classes
  - Sensor Types
  - SDR (Sensor Data Record) Types
  - example query with ipmitool/freeipmi
- System Event Log (SEL)
- Platform Event Filtering (PEF)
- Serial over LAN (SOL)
- Field Replaceable Unit (FRU) data

## 4) Example: Nagios/Icinga IPMI Plugin

## 5) Conclusions



### 3) IPMI details



- **IPMI example configuration of a LAN interface**

```
[root@testserver ~]# ipmitool lan print 1
Set in Progress           : Set Complete
Auth Type Support         : NONE MD5 PASSWORD
Auth Type Enable         : Callback :
                          : User      : MD5
                          : Operator :
                          : Admin   : MD5
                          : OEM     :
IP Address Source        : Static Address
IP Address                : 192.168.1.211
Subnet Mask               : 255.255.255.0
MAC Address               : 00:0e:0c:ea:92:a2
[...]
```



### 3) IPMI details



- **Remote Control**

```
[user@adminpc ~]$ ipmitool -I lan -H 192.168.1.211 \  
                    -U admin -P relation power status  
Chassis Power is off  
[user@adminpc ~]$
```

```
[user@adminpc ~]$ ipmitool -I lan -H 192.168.1.211 \  
                    -U admin -P relation power on  
Chassis Power Control: Up/On  
[user@adminpc ~]$
```

```
[user@adminpc ~]$ ipmitool -I lan -H 192.168.1.211 \  
                    -U admin -P relation power status  
Chassis Power is on  
[user@adminpc ~]$
```



### 3) IPMI details



- **Sensors: Sensor Classes (1/2)**

Discrete	Threshold
<p>multiple states possible:</p> <ul style="list-style-type: none"><li>• up to 15 states</li><li>• each state is reflected by a bit</li><li>• multiple state bits can active at a time</li></ul>	<p>changes event status on analog reading comparison to threshold values</p>
<p>can provide:</p> <ul style="list-style-type: none"><li>• generic states (Table 42-2) or</li><li>• sensor-specific states (Table 42-3)</li></ul>	<p>provides:</p> <ul style="list-style-type: none"><li>• analog reading of the sensor and</li><li>• discr. threshold comparison status bit (generic state, Table 42-2)</li></ul>
<p>other classes similar to discrete:</p> <ul style="list-style-type: none"><li>• Digital: term often used for discrete sensors with two possible states</li><li>• OEM: discrete sensor where the meaning of the states (offsets) are OEM defined</li></ul>	



# 3) IPMI details



- **Sensors: Sensor Classes (2/2)**

## Discrete

```
[root@test ~]# ipmitool sdr get "PS2 Status"
Sensor ID          : PS2 Status (0x71)
Entity ID         : 10.2 (Power Supply)
Sensor Type (Discrete): Power Supply
States Asserted   : Power Supply
                  [Presence detected]
                  [Power Supply AC lost]
Assertion Events  : Power Supply
                  [Presence detected]
                  [Power Supply AC lost]
Assertions Enabled : Power Supply
                  [Presence detected]
                  [Failure detected]
                  [Predictive failure]
                  [Power Supply AC lost]
[...]
Deassertions Enabled : Power Supply
[...]
```

## Threshold

```
[root@test ~]# ipmitool sdr get "Fan 1"
Sensor ID          : Fan 1 (0x50)
Entity ID         : 29.1 (Fan Device)
Sensor Type (Analog) : Fan
Sensor Reading     : 5719 (+/- 0) RPM
Status            : ok
Nominal Reading    : 6708.000
Normal Minimum     : 2451.000
Normal Maximum     : 10965.000
Lower critical     : 1720.000
Lower non-critical : 1978.000
Positive Hysteresis : 86.000
Negative Hysteresis : 86.000
Minimum sensor range : Unspecified
Maximum sensor range : Unspecified
Event Message Control : Per-threshold
Readable Thresholds : lcr lnc
Settable Thresholds : lcr lnc
Threshold Read Mask : lcr lnc
Assertion Events    :
Assertions Enabled  : lnc- lcr-
Deassertions Enabled : lnc- lcr-
```



# 3) IPMI details



- **Sensors: Sensor Types**

## Sensor Types (Table 42-3)

```
[root@testserver ~]# ipmitool sdr type list
Sensor Types:
  Temperature
  Current
  Physical Security
  Processor
  Power Unit
  Other
  Drive Slot / Bay
  System Firmwares
  Watchdog
  Critical Interrupt
  Module / Board
  Add-in Card
  Chip Set
  Cable / Interconnect
  System Boot Initiated
  OS Boot
  Slot / Connector
  Watchdog
  Entity Presence
  LAN
  Battery
  Version Change
  Voltage
  Fan
  Platform Security
  Power Supply
  Cooling Device
  Memory
  POST Memory Resize
  Event Logging Disabled
  System Event
  Button
  Microcontroller
  Chassis
  Other FRU
  Terminator
  Boot Error
  OS Critical Stop
  System ACPI Power State
  Platform Alert
  Monitor ASIC
  Management Subsystem Health
  Session Audit
  FRU State
```



### 3) IPMI details



- **Sensors: SDR (Sensor Data Record) Types**

Name (Chapter 43)	SDR Type
Full Sensor Record	01h
Compact Sensor Record	02h
Event-Only Record	03h
Entity Association Record	08h
Device-relative Entity Association Record	09h
Reserved Records	0Ah:0Fh
Generic Device Locator Record	10h
FRU Device Locator Record	11h
Management Controller Device Locator Record	12h
Management Controller Confirmation Record	13h
BMC Message Channel Info Record	14h
OEM Record	C0h





### 3) IPMI details



- **Sensors: example query with ipmitool**
  - use 'ipmitool sdr type [...]' for sensor type
  - use 'ipmitool sdr elist [...]' for SDR type

```
[root@testserver ~]# ipmitool sdr type Other
PS1 +12V Power      | 7Ch | ok | 10.1 | 80 Watts
PS2 +12V Power      | 7Dh | ok | 10.2 | 104 Watts
[root@testserver ~]# ipmitool sdr type Other -v
Sensor ID           : PS1 +12V Power (0x7c)
Entity ID           : 10.1 (Power Supply)
Sensor Type (Analog) : Other
Sensor Reading      : 80 (+/- 6) Watts
Status              : ok
Nominal Reading     : 372.000
Normal Minimum      : 100.000
Normal Maximum      : 744.000
Upper critical      : 840.000
Upper non-critical  : 792.000
[...]
```



### 3) IPMI details



- **Sensors: example query with freeipmi**

```
[root@testserver ~]# ipmimonitoring
Record_ID | Sensor Name | Sensor Group | Monitoring Status |
Sensor Units | Sensor Reading
[...]
17 | Fan 5 | Fan | Nominal | RPM | 9052.000000
18 | Fan 6 | Fan | Nominal | RPM | 8060.000000
19 | PS1 AC Current | Current | Nominal | A | 0.124000
20 | PS2 AC Current | Current | Nominal | A | 0.992000
[...]
33 | Power Redundancy | Power Unit | Critical | N/A | 'Redundancy
Lost' 'Non-redundant:Sufficient Resources from Redundant'
34 | BMC Watchdog | Watchdog 2 | Nominal | N/A | 'OK'
35 | Platform Secu V | Platform Security Violation Attempt |
Nominal | N/A | 'OK'
36 | Physical Scrtty | Physical Security | Critical | N/A |
'General Chassis Intrusion'
```



# 3) IPMI details



- **System Event Log (SEL)**
  - stored in non-volatile storage

```
[root@testserver ~]# ipmitool sel elist
40 | 06/21/2010 | 14:29:29 | Power Supply PS1 Status | Power Supply AC lost | Asserted
54 | 06/21/2010 | 14:29:29 | Power Unit Power Redundancy | Fully Redundant
68 | 06/21/2010 | 14:29:29 | Power Unit Power Redundancy | Redundancy Lost
7c | 06/21/2010 | 14:29:29 | Power Unit Power Redundancy | Non-Redundant: Sufficient from Redundant
[...]
2fc | 06/21/2010 | 15:20:32 | Physical Security Physical Scrtcy | General Chassis intrusion | Asserted
```

```
[root@testserver ~]# ipmitool sel elist
Power Supply PS1 Status | Power Supply AC lost | Asserted
Power Unit Power Redundancy | Fully Redundant
Power Unit Power Redundancy | Redundancy Lost
Power Unit Power Redundancy | Non-Redundant: Sufficient from Redundant
[...]
Physical Security Physical Scrtcy | General Chassis intrusion | Asserted
```



## 3) IPMI details



- **Platform Event Filtering (PEF)**
  - BMC takes selected actions on event messages
  - actions can be:
    - system power off
    - system reset
    - generating SNMP trap
  - tools: freeipmi (pef-config), ipmiutil

```
[root@testserver ~]# pef-config --info
PEF version:                2.0
Alert action support:       Yes
Power down action support:  Yes
Power reset action support: Yes
Power cycle action support: Yes
OEM action support:         Yes
Diagnostic interrupt action support: Yes
[...]
```



# 3) IPMI details



- **Serial over LAN (SOL) (1/2)**
  - allows text-based access to
    - BIOS
    - GRUB
    - operating systems
      - Linux serial console
      - Windows debugger  
(<http://support.microsoft.com/kb/151981/>)
  - SOL is implemented as a payload type in RMCP+



# 3) IPMI details



- **Serial over LAN (SOL) (2/2)**

```
wfischer@wfischer-t410-ubuntu: ~  
File Edit View Terminal Help  
wfischer@wfischer-t410-ubuntu:~$ ipmitool -I lanplus -H 192.168.1.211 -o intelplus -U admin -P relation sol activate  
Main Advanced Security Server Management Boot Options >  
System BIOS  
Version          S5000.86B.10.00.0094  
Build Date       10/13/2008  
Processor  
Intel(R) Xeon(R) CPU          5110 @ 1.6 GHz  
Core Frequency   1.6 GHz  
Count            2  
Memory  
Size             8 GB  
Quiet Boot       [Disabled]  
POST Error Pause [Disabled]  
System Date      [Mon 06/21/2010]  
System Time      [14:17:31]  
^|System Time has  
+|configurable fields for  
+|Hours, Minutes, and  
*|Seconds.  
*|Hours are in 24-hour  
*|format.  
*|Use [Enter] or [Tab] key  
*|to select the next field.  
*|Use [+] or [-] key to  
*|  
*|><   Select Screen  
*|^v    Select Item  
*|+/-   Change Value  
*|Enter Select Field  
*|F1    General Help  
*|F9    Optimized Defaults  
*|F10   Save and Exit  
*|ESC   Exit  
v|
```



# 3) IPMI details



- **Field Replaceable Unit (FRU) data**

```
[root@testserver ~]# ipmitool fru print
FRU Device Description : Builtin FRU Device (ID 0)
  Chassis Type          : Rack Mount Chassis
  Chassis Serial        : 9000041568
  Chassis Extra         : SR2500LX
  Board Mfg Date        : Sat Apr 28 11:37:00 2007
  Board Mfg             : Intel
  Board Product         : S5000PAL0
  Board Serial          : BZAU71700054
  Board Part Number     : D13607-805
  Product Manufacturer  : Intel
  Product Name          : S5000PAL
  Product Part Number   : .....
  Product Serial        : .....
[... ]
FRU Device Description : Pwr Supply 1 FRU (ID 2)
  Product Manufacturer  : DELTA
  Product Name          : DPS-750EBA
  Product Part Number   : D20850-006
  Product Version       : 02
  Product Serial        : DLD0712004978
```



# Agenda



- 1) About Thomas Krenn
- 2) IPMI basics
- 3) IPMI details
- 4) Example: Nagios/Icinga IPMI Plugin
- 5) Conclusions








## 4) Example: Nagios/Icinga IPMI Plugin



- **how the Nagios IPMI Sensor Monitoring Plugin works**
  - it's a shell script (Bash)
  - it uses ipmitool, gawk
  - you can use the plugin with every IPMI-compatible server
  - it follows the *Nagios plug-in development guidelines*
  - clear illustration within the Nagios web interface

Host ↑↓	Service ↑↓	Status ↑↓	Last Check ↑↓	Duration ↑↓	Attempt ↑↓	Status Information
<a href="#">examplehost</a>	<a href="#">IPMI Fans</a> 	OK	2009-11-02 16:06:33	0d 0h 6m 12s	1/4	Fan Status: OK
	<a href="#">IPMI Power Supply</a>	OK	2009-11-02 16:06:24	0d 0h 32m 21s	1/4	Power Supply Status: OK
	<a href="#">IPMI Physical Security</a>	WARNING	2009-11-02 16:06:14	0d 0h 40m 32s	4/4	Physical Security Status: Warning [Intrusion = Inc]
	<a href="#">IPMI Temperature</a> 	OK	2009-11-02 16:06:02	0d 0h 31m 43s	1/4	Temperature Status: OK
	<a href="#">IPMI Voltage</a> 	OK	2009-11-02 16:05:48	0d 0h 31m 57s	1/4	Voltage Status: OK



# 5) Conclusions



- **IPMI has multiple very useful features**
  - **IPMI is built into most modern Servers**
  - **IPMI has rarely been used until now**
- **so start using the Power of IPMI today ;-)**

- **Further information**

- [http://download.intel.com/design/servers/ipmi/IPMI2\\_0E4\\_Markup\\_061209.pdf](http://download.intel.com/design/servers/ipmi/IPMI2_0E4_Markup_061209.pdf)
- [http://www.thomas-krenn.com/de/wiki/IPMI\\_Grundlagen](http://www.thomas-krenn.com/de/wiki/IPMI_Grundlagen)
- <http://www.thomas-krenn.com/ipmi-plugin>
- <http://lists.thomas-krenn.com>

