Logging the Internet of Things - Connected power plants demand a new paradigm

Rocco Gagliardi
Defense Department, scip AG
roga@scip.ch
https://www.scip.ch

Marc Ruef (Editor)
Research Department, scip AG
maru@scip.ch
https://www.scip.ch

Abstract: The IoT changes the log paradigm. The actual infrastructure will be heavily impacted. Some mix of products are emerging. The solution design must be carefully assessed.

Keywords: Cisco, Dashboard, Framework, Graylog, HTTP, IBM, Internet of Things, IoT, Joke, Kibana

1. Preface

This paper was written in 2016 as part of a research project at scip AG, Switzerland. It was initially published online at *https://www.scip.ch/en/?labs.20160804* and is available in English and German. Providing our clients with innovative research for the information technology of the future is an essential part of our company culture.

2. Introduction

The *log* is born to inform humans about the status of a specific system, just *in case of a problem*, as record of what happened. Some times – a good example: `xen_netfront: xennet: skb rides the rocket: 21 slots` – the message sounds like a joke*: is the human-human communication.

In the wonderful world of *SIEM* (Security Information and Event Management), the plethora of messages generated by different (part of) systems was interpreted and correlated: processed or – better – pre-processed by machine for the end-human-user.

In the era of the *IoT* [1] (Internet of Things), we will assist in a basic paradigm change: the message generated by a machine will be used by another machine, not by a human.

3. The Network is the Computer

I'm not sure what John Cage had in mind with the statement *The Network is the computer*, but I don't think he meant IoT. The IoT refer to (a huge number of) interconnected devices that runs with very little human intervention. We install them in our houses, we wear them during our training, part of them are with us during trips and part remains at home, informing us about the *location of our goldfish or the status of our fridge* [2].

In the industry, the IoT is used to monitor the status of very complex installations, using thousands of sensors with millions of samples and *help saving lot of money* [3]. Data generated by these devices (sensors, actuators, etc.) can be used to predict the system behavior and to improve future versions.

4. Classic vs IoT log

How changes the traditional log compared with the IoT?

Key	Classic Log	IoT Log
Avg number of logs/[s]	low	high
Avg length of log [bytes]	high	low
Validity timerange	minutes – days	micro seconds – seconds
Log content	State transfer	Telemetry
Log main scope	History	Precog
Communication streams	Machine ⇒ Human	Machine ⇒ Machine

As use case, take a power plant monitoring. Some of the advantages to constantly monitor the operating parameters of a complex machine such, for example, a gas turbine:

- Control of operations within the parameters established by the manufacturer
- Identification of possible problems before they lead to failure

Possible consequences:

- The manufacturer reduces warranty costs expected by operational mistakes
- The insurance premiums are lower because the more accurate models
- The operator can properly plan their maintenance and better comply with their SLA

Extend the idea to other parts of the system, and you will quickly have lot of sensors to monitor:

Parameter	Value
Sensors	Analog: 30.000, Digital: 20.000
Sampling rate	Analog: 50ms, Digital: 1s
Data type	Analog: float, Digital: short-int
History	1 – 3 years

The necessary infrastructure to handle such a data-flow coming from so many different sources, diverges from the classical architecture, and must assure:

- **Storage capability**: Not only space, but also query flexibility and access speed.
- **Very low latency**: The signal validity life is trending toward microseconds; not only the network must assure optimal performance, but also all other components.
- **High computational power**: Large amount of data require CPU power to be interpreted and correlated; form landscape recognition to median of PH values at different temperatures, high speed computers are a must.

5. Emerging solutions

Alongside the classic logging technologies, new specific solutions for the telemetry are being consolidated. No particular standard is adopted, since the data are just *k*→*v* tuples. Regarding the communication protocol, almost everyone is based on IP. On top of IP, the choice is mainly between MQTT, XMPP and CoAP.

In the storage area, for telemetry, the choice goes to dedicated databases (Graphite/InfluxDB/Hadoop) and specific software to display charts or build dashboards. To mix new features in old solutions, some parser/extractors may be used to extract performance data and push it to telemetry databases while maintaining the old processes in place.

5.1. Communication fundamentals

The key component is the IP protocol, especially the "new" version 6. The battery for many sensors, especially those designed for the IEEE802.15.4, must last for months; so, the device is not online all the time, cannot communicate at high speed or – sometimes – is just out of range. This kind on networks are known as LLNs (Low-power and Lossy Networks).

- *MQTT/MQTT-S* is a publish/subscribe messaging protocol designed for lightweight M2M communications originally developed by IBM
- *XMPP* (eXtensible Messaging and Presence Protocol) has its roots in instant messaging and is a contender for mass scale management of consumer goods.
- *CoAP* (Constrained Application Protocol) over UDP is used for resource constrained, low-power sensors and devices connected via lossy networks, especially when there are a high number of sensors and devices within the network.

Following, a very short list of key points for each protocol.

Protocol	Pro	Contra
MQTT	Pushed by IBM. Subscribed services (Many2Many). Two way communication over unreliable nets. NAT is not critical. QoS in place (Fire-and-forget, At-least-once and Exactly-once).	Low power, but not for extremely constrained devices. Normally "online" all the time (addressed in MQTT-SN). Long topic names, impractical for 802.15.4 (addressed in MQTT-SN).
CoAP	Pushed by CISCO. Primarly a One2One protocol. Resource discovery. Interoperate with HTTP/REST.	Sensor is typically a server, so NAT must be designed carefully. Since UDP, no SSL/TLS. DTLS can be used.
XMPP	Pushed by CISCO. Real-time. Massive scalability. Security.	Not been practical over LLNs. Need for an XML parser.

5.2. Products by phases

Following, a summary of products with references; refer to the schema for the interconnections between components. Please remember that the same process can be implemented in very different manners: just use the product most familiar to you.

Phase	Components	Key to consider
Collect	*Fluent Bit* [4], *Collectd* [5], *Telegraf* [6], *Beats* [7], *rsyslog* [8]	System type. Framework already present.
Queue	*Redis* [9], *MemCached* [10], *RabbitMQ* [11]	Routing customization. Performance. Delivery assurance.
Parse	*Fluentd* [12], *Logstash* [13], *rsyslog* [14], *Graylog* [15]	Input / Output. Message parsing plugins.
Store	*InfluxDB* [16], *Prometeus* [17], *Hadoop* [18], *Elasticsearch* [19], *RDD* [20]	Speed. Query language. Granularity.
Visualise	*Kibana* [21], *Graylog* [22], *Chronograf* [23], *Grafana* [24], *Thruk* [25], *Cacti* [26]	Authentication / Authorization. Visualitation types. Query language / Transformation functions. Dashboard customization.
Act	*Graylog* [27], *Kapacitor* [28]	Triggering capabilities. Query language / Transformation functions. Storage. Integration.

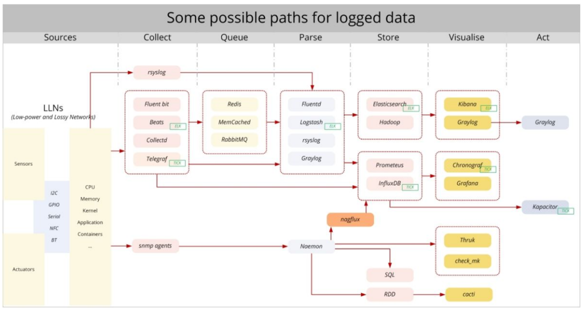

Figure: Some possible paths for logged data

6. Summary

Logging and using the enormous amount of data generated by the upcoming IoT infrastructure will be challenging for the whole IT infrastructure: for the network, for the CPUs, and for the software development. A lot of solutions are popping out, all with pros und contras. Depending on what are the primary goals of the project, one (mix) may be better as another.

7. Footnote

*) This "joke" appeared in our XEN infrastructure just after an upgrade, and wasn't funny. For more goto *Kernel Line Tracing: Linux perf Rides the Rocket* [29].

8. External Links

[1] http://postscapes.com/what-exactly-is-the-internet-of-things-infographic/
[2] http://www.makeuseof.com/tag/internet-things-10-useful-products-must-try-2016/
[3] http://www.gereports.com/big-data-industrial-internet-can-help-southwest-save-100-million-fuel/
[4] http://fluentbit.io/
[5] http://collectd.org/
[6] https://influxdata.com/time-series-platform/telegraf/
[7] https://www.elastic.co/products/beats
[8] http://www.rsyslog.com/
[9] http://redis.io/
[10] http://www.memcached.org/
[11] http://www.rabbitmq.com/
[12] http://redis.io/
[13] https://www.elastic.co/products/logstash
[14] http://www.rsyslog.com/
[15] https://www.graylog.org/
[16] https://influxdata.com/
[17] https://prometheus.io/
[18] https://hadoop.apache.org/
[19] https://www.elastic.co/products/elasticsearch
[20] http://oss.oetiker.ch/rrdtool/
[21] https://www.elastic.co/products/kibana
[22] https://www.graylog.org/
[23] https://influxdata.com/time-series-platform/chronograf/
[24] https://grafana.org/
[25] http://www.thruk.org/
[26] http://cacti.net/
[27] https://www.graylog.org/
[28] https://influxdata.com/time-series-platform/kapacitor/
[29] http://www.brendangregg.com/blog/2014-09-11/perf-kernel-line-tracing.html