syslog

The lib (tries to)implement a RFC5424 compliant syslog interface for ESP8266. syslog messages are send via UDP. Messages are send in the following format:

```
PRI VERSION SP TIMESTAMP SP HOSTNAME SP APP-NAME SP PROCID SP MSGID SP MSG
  PRI:          msg priority: facility * 8 + severity
  TIMESTAMP:    dash (no timestamp) or ISO8601 (2015-12-14T17:26:32Z)
  HOSTNAME:     flashConfig.hostname
  APP-NAME:     tag - (e.g. MQTT, mySQL, REST, ...)
  PROCID:       dash or ESP system tick (µseconds since reboot)
  MSGID:        counter - # syslog messages since reboot
  MSG:          the syslog message
```

The meaning of TIMESTAMP, HOSTNAME, PROCID and MSGID is hardcoded, all others are parameters for the syslog function.

syslog messages are queued on heap until the Wifi stack is fully initialized:

```
Jan  1 00:00:00 192.168.254.82 esp_link 0.126850 1 Reset cause: 4=restart
Jan  1 00:00:00 192.168.254.82 esp_link 0.133970 2 exccause=0 epc1=0x0 epc2=0x0 epc3=0x0 excvaddr=0x0 depc=0x0
Jan  1 00:00:00 192.168.254.82 esp_link 0.151069 3 Flash map 4MB:512/512, manuf 0xC8 chip 0x4016
Jan  1 00:00:00 192.168.254.82 esp_link 0.166935 4 ** esp-link ready
Jan  1 00:00:00 192.168.254.82 esp_link 0.185586 5 initializing MQTT
Jan  1 00:00:00 192.168.254.82 esp_link 0.200681 6 initializing user application
Jan  1 00:00:00 192.168.254.82 esp_link 0.215169 7 waiting for work to do...
Jan  1 00:00:03 192.168.254.82 SYSLOG 3.325626 8 syslogserver: 192.168.254.216:514
Jan  1 00:00:03 192.168.254.82 esp_link 3.336756 9 syslog_init: host: 192.168.254.216, port: 514, lport: 24377, state
Dec 15 11:49:14 192.168.254.82 esp-link 18.037949 10 Accept port 23, conn=3fff5f68, pool slot 0
```

If the remaining heap size reaches a given limit, syslog will add a final obituary and stop further logging until the queue is empty and sufficient heap space is available again.

The module may be controlled by flashconfig variables:

- **syslog_host: host[:port]**

  host is an IP-address or DNS-name. **port** is optional and defaults to 514. DNS-Resolution is done as soon as the Wifi stack is up and running.

- **syslog_minheap: 8192**

  minheap specifies the minimum amount of remaining free heap when queuing up syslog messages. If the remaining heap size is below **minheap**, syslog will insert an obituary message and stop queuing. After processing all queued messages, the logging will be enabled again.

- **syslog_filter: 0..7**

syslog_filter is the minimum severity for sending a syslog message. The filter is applied against the message queue, so any message with a severity numerical higher than **syslog_filter** will be dropped instead of being queued/send.

- **syslog_showtick: 0|1**

  If **syslog_showtick** is set to **1**, syslog will insert an additional timestamp (system tick) as "PROCID" field (before the users real syslog message). The value shown is in seconds, with 1µs resolution since (re)boot or timer overflow.

- **syslog_showdate: 0|1**

  If **syslog_showdate** is set to **1**, syslog will insert the ESPs NTP time into the syslog message. If "realtime_stamp" (NTP 1s ticker) is **NULL**, the time is derived from a pseudo-time based on the absolute value of systemticks.

  Some syslog servers (e.g. Synology) will do crazy things if you set **syslog_showdate** to **1**

The syslog module exports two functions:

```
syslog_init(char *server_name);
syslog(uint8_t facility, uint8_t severity, const char *tag, const char *fmt, ...);
```

syslog_init

usage: `syslog_init(char *server_name);`

syslog_init expects a server name in format "host:port" (see **syslog_host** flashconfig).

If **server_name** is **NULL**, all dynamic allocated memory (buffers, queues, interfaces) are released and the syslog state is set to "SYSLOG_HALTED".

If **server_name** is **""**, syslog state is set to "SYSLOG_HALTED", without clearing the queue.

Otherwise, syslog_init will allocate all required structures (buffers, interfaces) and send all collected syslog messages.

syslog is self-initializing, meaning the syslog_init(server_name) is called on first invocation. The syslog_init function is only for convenience if you have to stop or disable syslog functions.

syslog

usage: `syslog(uint8_t facility, uint8_t severity, const char *tag, const char *fmt, ...);`

- **facility**

  the message facility (see syslog.h, **enum syslog_facility**).

- **severity**

  the message severity (see syslog.h, **enum syslog_severity**)

- **tag**

  user defined tag (e.g. "MQTT", "REST", "UART") to specify where the message belongs to

- ** const char *fmt, ...**

  the desired message, in printf format.

Examples

```
hostname="ems-link02", showtick=0, showdate=0
Syslog message: USER.NOTICE:  - ems-link02 esp_link - 20 syslog_init: host: 192.168.254.216, port: 514, lport: 28271,

hostname="ems-link02", showtick=1, showdate=0
Syslog message: USER.NOTICE:  - ems-link02 esp_link 3.325677 8 syslog_init: host: 192.168.254.216, port: 514, lport:
```

```
hostname="ems-link02", showtick=1, showdate=1, NTP not available
Syslog message: USER.NOTICE:  1970-01-01T00:00:03.325668Z ems-link02 esp_link 3.325668 8 syslog_init: host: 192.168.2

hostname="ems-link02", showtick=1, showdate=1, NTP available
Syslog message: USER.NOTICE:  2015-12-15T11:15:29+00:00 ems-link02 esp_link 182.036860 13 syslog_init: host: 192.168.
```

Notes

- The ESP8266 (NON-OS) needs a delay of **at least 2ms** between consecutive UDP packages. So the syslog throughput is restricted to approx. 500EPS.
- If a syslog message doesn't have the timestamp set (**syslog_showdate** == 0), the syslog *server* will insert *it's own receive timestamp* into the log message.
- If **syslog_showdate** == 1, the syslog *server* MAY replace it's own receive timestamp with the timestamp sent by the syslog client.
- Some syslog servers don't show the fractional seconds of the syslog timestamp
- Setting **syslog_showdate** will send timestamps from 1970 (because of using the internal ticker) until the **SNTP-client** got a valid NTP datagram. Some syslog servers (for example *Synology*) will roll over their database if they get such "old" syslog messages. In fact, you won't see those messages in your current syslog.
- Some servers (e.g. *Synology*) won't show the syslog message if you set **facility** to **SYSLOG_FAC_SYSLOG**.