

Call Button / HTTP – System Description

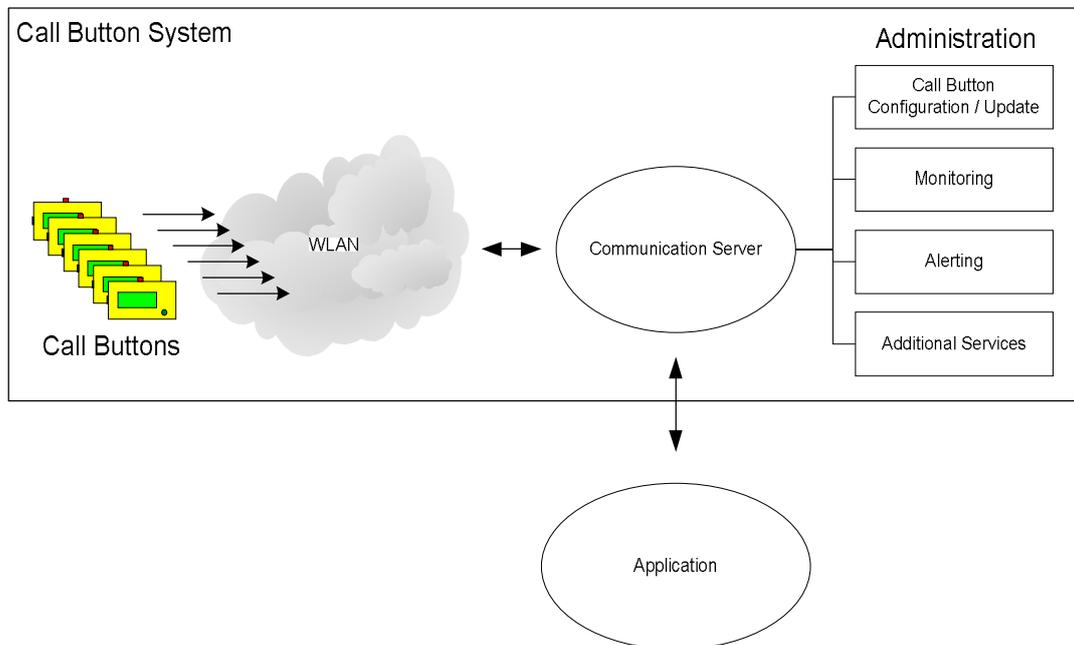
Detlef Reil, 04/13/2004, applies to Call Button release 040413, V1.52

Software System

Up to now, for the communication between the Call Buttons and the application, *Rabus Server* was as the central communication server:

Rabus communication and administration server

Every Call Button event is communicated to *Rabus Server*, which buffers the Call Button status and forwards it to the application. The application communicates with *Rabus Server* over a proprietary protocol (Rabus protocol). The Call Buttons are virtually reachable for the application at any time – display changes from the application are buffered by Rabus and forwarded to the Call Buttons during the frequent wakeup events.

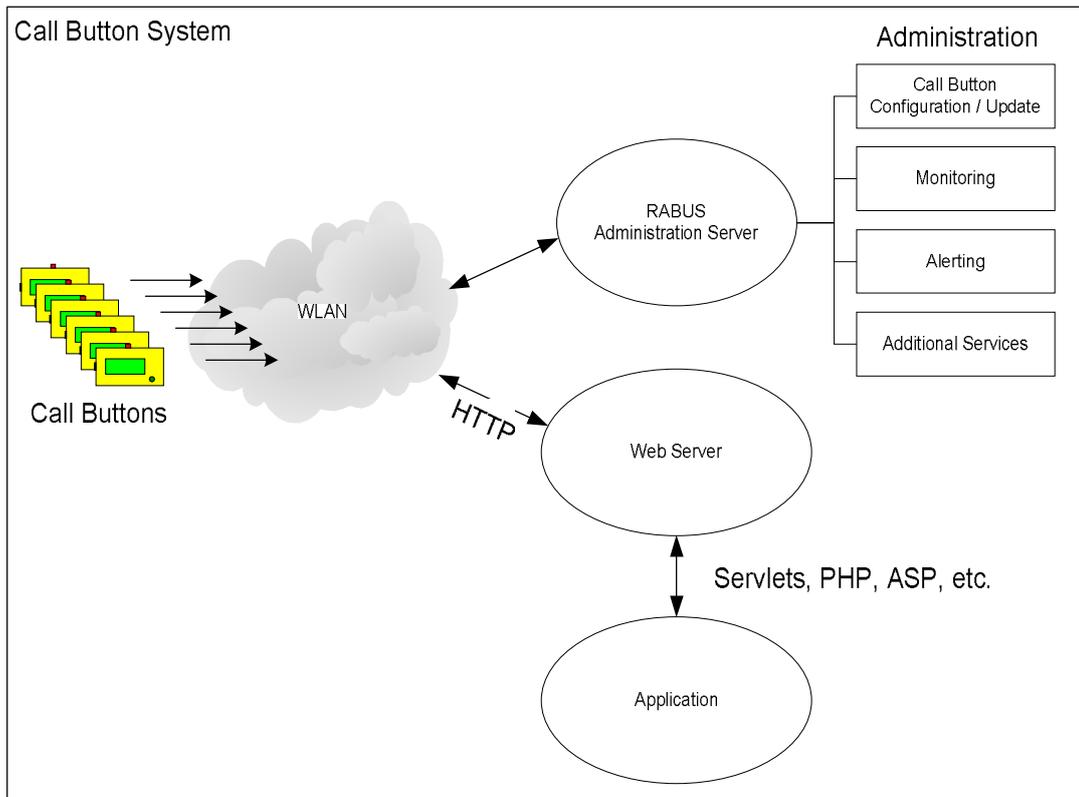


With the new version the Call Buttons can be alternatively configured to communicate directly with the application over the HTTP protocol:

HTTP over web server and *Rabus* as pure administration server

A Call Button event (e.g. button press) establishes a direct HTTP connection to the web server and requests a page with a configurable URL which is on reception shown on the display.

In frequent intervals, the Call Buttons connect to Rabus for administration and surveillance purposes. Rabus can then initiate configuration changes or firmware updates or raise an alarm for the administrator when the system does not operate properly (Batterie empty, Call Button does not respond,...)



In case *Rabus Server* is not available, HTTP operation can proceed for a certain time (e.g. 10 days) before the Call Button is disabled. Thus, in order to obtain a high-availability system, only the (standard) web server must be secure. If the Rabus Server breaks down, there should be enough time for reactivation.

A continuous operation without *Rabus* is not possible.

Installation

- Web server and web application on the application server PC
- Rabus Server on the administration server PC (see *Rabus Server Handbook*)
- Administration tools (*RabusAdmin*, *RabusMonitor* (see *Rabus Server Handbook*), *FBConfig* (see *FBConfig Handbook*) on the administration PC

After installation, Rabus Server must be configured using RabusAdmin. Choose the middle option of the „Application Server Settings“ („Direct HTTP connection“).

New Call Buttons must be attached to the administration PC with a serial RS232 cable and configured with FBConfig. For the HTTP communication the following parameters must be configured in addition to the usual LAN and WEP settings:

- Property page HTTP: Activate HTTP protocol, enter URL
- Property page Network: Configure HTTP server IP in addition to the administration server IP: Host IP to web server, port (default: 80), router to web server, if applicable

The first button press after a configuration by FBConfig initiates a transaction to *Rabus Server* and sends the new parameter values. Subsequent button events will initiate a communication to the web server.

Communication Cycle

On a user event (button press, barcode scan, auto call, HTTP wakeup event) a TCP connection to the web server is established and a HTTP GET is sent with the configured URL. On reception, the HTTP response will be parsed and the page body will be displayed when an OK status (2XX) is contained. On a different status the Call Button will show message 16 and the HTTP status, e.g. 404 for „Resource not found“. Subsequently, the TCP connection will be finished.

Independantly from the HTTP communication to the web server, the Call Button frequently connects to *Rabus Server* in order to report the own status. The interval can be configured in *Rabus Server* to a time between 1 and 255 minutes. Thus, every Call Button will communicate at least every 4½ hours to *Rabus Server*.

In case *Rabus Server* is not reachable, an appropriate error message is displayed. HTTP operation is possible for several days anyway. During TCP connection establishment „please wait... No admin connection, xx days left“ is shown.

If there is still no connection to *Rabus Server* within 10 days, the Call Button is disabled for further HTTP transactions. Any subsequent button press will try to connect to *Rabus Server* rather than to the web server. As soon as *Rabus* is back in operation the button is re-enabled and the next button event will communicate with the HTTP server.

In order to force a connection to *Rabus Server*, the button must be held down for at least two seconds until the release information screens are showing up. After releasing the button a *Rabus* transaction is initiated.

HTTP Implementation on the Call Buttons

Because of the restrictions of the optimized hardware, only the most necessary features of the HTTP protocol have been implemented. By programming an application at the server-side using tools like

- Java Servlets
- PHP or Perl
- ASP
- Coldfusion
- etc.

the full Call Button functionality can be used similarly to a *Rabus* application.

The session state of a client request must be saved within the application on the server side – the Call Button is not able to store cookies, to manage dynamic URLs or to maintain persistent HTTP connections. However, the Call Button can include its ID (Name) in the URL, so that the application can associate the session with the saved current state. HTTP, release 1.0 is used for communication.

URL

The URL is configured on the Call Button and may consist of up to 128 characters (relative path allowed). The following wildcards will be replaced by the Call Button before sending the request:

- \$N by the configured Call Button name
- \$\$ by the scanned barcode
- \$E by the type of the event (1=button press, 2=barcode scanned, 3=HTTP wakeup, 4=auto call)
- \$\$ by the dollar character (\$)

Example:

```
/homepage.php?name=$N&event=$E&scan=$S
```

Will be replaced by

```
/homepage.php?name=BUTTON0001&event=2&scan=1234567890
```

HTTP Headers in response

The following HTTP headers are processed when present in the web server response:

Wakeup: 20m
or
Wakeup: 10s

HTTP wakeup time. The Call Button will initiate an HTTP transaction after that time.
Allowed values: 1 to 255 minutes or 5 to 255 seconds

Lamp: 1

Lamp status to be set until the next event. Allowed values:

0 – off
1 – flashing ½ Hz
2 – flashing 1 Hz
3 – flashing 2 Hz
4 – flashing 3 Hz
5 – flashing 5 Hz
7 – constant on

Body

The text body must be in plain text (US-ASCII), the display has 4 x 20 characters. HTML tags are not supported, meaning that the tags are displayed as plain text when present.