Remote Access to Embedded Devices

Status of this Memo

This document is an Internet-Draft and is in full conformance with all provisions of Section 10 of RFC2026 [1].

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF), its areas, and its working groups. Note that other groups may also distribute working documents as Internet-Drafts.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

The list of current Internet-Drafts can be accessed at http://www.ietf.org/ietf/1id-abstracts.txt
The list of Internet-Draft Shadow Directories can be accessed at http://www.ietf.org/shadow.html.

Abstract

The aim of this document is to standardize remote access options to parameters of embedded devices with limited resources. Typically such devices are based on 8-bit or 16-bit microcontrollers with limited memory (64K or less) and a low operating frequency (20 MHz or less). The protocol described in this document uses existing markup formats to specify modifiable parameters of embedded devices and existing protocols to transfer these parameters between clients and servers.

Conventions used in this document

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in RFC-2119 [2].
1. Introduction

Remote access to embedded devices in our homes, factories, and vehicles or to personal mobile electronic appliances becomes reality. The RFC2324 [3] published on April 1st, 1998 was a humorous approach: getting a coffee machine online and needing a Hyper Text Coffee Pot Control Protocol (HTCPCP/1.0) seemed to be funny at that time.

Today RFC2324 does not sound as funny anymore - it is closer to reality today than it was on its publishing date. There is a clear drive in the industry to Internet-enable embedded device like climate control units and appliances - and coffee machines.

There are attempts by several companies to Internet-enable their products Â» with some companies inventing their own proprietary protocols and methods on how to allow remote access via the Internet.

Unfortunately this approach is not user-friendly. Users have to get acquainted with different methods and tools on how to access their devices via email, web or other services Â» and none of them are compatible with each other.

The goal of this document is to find a common ground for remote access functionality Â» from the client viewpoint. No matter what kind of embedded system is connected to the Internet the client should be able to expect some standardized methods for the remote access using common services like email, regular web browsers or minimized web browsers for PDAs or mobile phones,.

This document does not invent any new Internet technologies - it just RECOMMENDS how existing protocols and methods should be used to offer users standardized methods for remote access.
For the scope of this document it does not matter how an embedded device is connected to the Internet. Connection can be directly or via a specialized gateway for embedded devices that might use a simple serial link or other lower cost network or fieldbus to exchange information with the embedded devices. However, this document assumes that there is at least one Internet node that manages one or more embedded device(s) and that offers the standardized protocols and methods described in this document to allow remote access to the embedded device(s). We distinguish between three types of devices:

1) Remote Access Client (RAC): the system that attempts to access an embedded device over the Internet.
2) Remote Access Server (RAS): the system that manages the internet access to one or multiple embedded device and
3) Remote Access Device (RAD): the device that is to be accessed through the Internet.

In some implementations, the RAS might be implemented directly with one RAD. In others, one RAS will be able to handle multiple RADs.

1.1 Terminology and Abbreviations

RAD Æ» Remote Access Device
The embedded device(s) that can be accessed via the Internet.

RAS Æ» Remote Access Server
This is the access point for the Remote Access Client. The RAS manages one or multiple Remote Access Devices and provides the Internet connectivity. The RAS can be part of an embedded device or part of a gateway connecting several Remote Access Devices to the Internet.

RAC Æ» Remote Access Client
A software or hardware client used to provide remote access to a Remote Access Server. This can be a web browser, email client, PDA or any other internet connected device.

RAPDF Æ» Remote Access Parameter Description Format
A format that describes all Remote Access Devices and their configurable parameters connected to a single Remote Access Server.
Remote Access to Embedded Devices

Figure 1 shows a Remote Access Device (RAD 1) that directly implements a Remote Access Server (RAS A). The RAS A can directly serve RAPDF information to a Remote Access Client (RAC).

Figure 2 shows how one Remote Access Server (RAS B) can handle multiple Remote Access Devices (RAD 2-4). The RADs can be connected to the RAS via a local, low-cost serial network or fieldbus. Depending on resources available to RAS and RAD, the RAS MAY retrieve the RAPDF information from a web server instead from the RAD itself, after the appropriate URL was reported to the RAS by each RAD.
2. Remote Access Parameter Description Format (RAPDF)

Every remote access device has certain parameters/variables that we want to be able to read or write through an Internet connection.

All accessible parameters (both, read and write) of a certain device MUST be defined using the Remote Access Parameter Description Format (RAPDF).

2.1 RAPDF Outline

RAPDF is based on HTML 4.0 [4] and obeys the following rules:

1.) The header section MUST include the META tag

   <META name="type" content="RAPDF 1.0">

2.) The body section MUST include AT LEAST ONE HTML FORM. The name of the form and the action identifies exactly one RAD (Remote Access Device). For example:

   <FORM name="Remote Access to MyHomeDevice" action="mailto:remote@mail.com?subject=RAPDF" method="post">
   ...
   </FORM>

   Additionally, it is RECOMMENDED to use the <LABEL> tag to ensure proper display of the forms' name:

   <LABEL for="ra_my_form">MyHomeDevice</LABEL>
   <FORM name="Remote Access to MyHomeDevice" action="mailto:remote@mail.com?subject=RAPDF" method="post" id="ra_my_form">
   ...
   </FORM>

3.) For every parameter of the embedded device the form contains a dedicated field The tags LABEL and ID MUST be used to associate a name with the corresponding parameter:

   <LABEL for="ra_temp_lr">Living Room Temperature</LABEL>
   <INPUT name="temp_living_room" type="text" id="ra_temp_lr">

   <INPUT name="light_living" type="checkbox" value="Living Room" id="ra_light_lr">
   <LABEL for="ra_light_lr">Living Room Lights</LABEL>
4.) The usage of formatting tags such as paragraphs, tables, inline graphics or tables has no effect on the automatic interpretation of an RAPDF. Devices or processing tools that work automatically on a RAPDF MAY ignore all formatting tags (especially if implemented on medium to low performance systems).

5.) It is recommended to use FIELDSET and LEGEND where appropriate. See the Appendix for examples.

6.) The `readonly` tag is supported for inputs and marks parameters of the embedded device that cannot be changed. Example:

   `<LABEL for="ra_temp">Current Temperature</LABEL>
   <INPUT name="temp" type="text" id="ra_temp_lr" readonly>`

Note that there is no particular language specified for an RAPDF. The description in the `<LABEL>` section can be provided in any language.

The primary identifier to match a parameter to the RAPDF is the `name` of each `<FORM>`, `<INPUT>` or `<SELECT>` element. These variable names MUST remain the same in any language, as these would be most likely hard-coded into the firmware of the RAD.

Also note that any other HTML tags for formatting or links may be used outside or within a FORM. However, each FORM MUST be associated with a RAD.

2.2 Storing the RAPDF

Manufacturers of Remote Access Devices MUST provide the RAPDF information along with their devices.

As code and data space in many embedded devices is limited, it is recommended to provide the RAPDF information on HTTP servers [5] on the Internet. This way an embedded device does not need to store the...
entire RAPDF information locally. Storing the URL indicating where to find the RAPDF is sufficient for such cases.

If the embedded device integrates a Remote Access Server (RAS) directly and its CPU performance and disk storage capacity is sufficient in size, the RAPDF MAY be stored and made available by the device itself.

3. Protocols and Formats Used

Which protocols are used to exchange the RAPDF information between the RAD, RAS and RAC is not specified in this document. Depending on the application a variation of existing email or web protocols MAY be used. Also depending on the application additional security protocols should be used to prohibit unauthorized access.

The information flow and formats used are:

Information sent from the RAS to the RAC MUST be in RAPDF. This ensures support of a wide variety of existing and future clients.

Information sent from the RAC to the RAS MUST be the regular POST reply used for HTML forms (see [4] and [5]).

Information flow between RAD and RAS is application and manufacturer specific. Either the RAD provides the RAS with the entire RAPDF or it provides the RAS with information where to get the RAPDF (for example by giving it a URL to the RAPDF info).

4. General RAPDF Usage Example

To illustrate how Remote Access MAY be implemented in accordance to this document, here is an example:

In an automated home, one RAS provides access to several RAD, such as thermostats, light controls, etc. The RAS is configured to recognize a list of authorized users.

Using a RAC, an authorized user requests the RAPDF from the RAS. The RAS assembles the RAPDF information by either getting the data from the RAD themselves or if they are minimal implementations, retrieves the RAPDF information from the manufacturer’s web page (using a URL reported by the RAD).

Once the RAPDF is assembled and the current data values / settings inserted, the RAS transmits the RAPDF to the RAC who requested it.
If the user requests changes to the data values / settings, the new values are transmitted back to the RAS who is in charge of applying them.

Security Considerations

This document does not address the security issue, as it focuses on the data format exchanged between the RAC and RAS, not on the protocols used to exchange the data. Depending on the security level required by the application, an appropriate secure transmission protocol SHOULD be used.

References

1 Bradner, S., "The Internet Standards Process -- Revision 3", BCP 9, RFC 2026, October 1996.


3 L. Masinter, "Hyper Text Coffee Pot Control Protocol (HTCPCP/1.0)"
   RFC2324, April 1998.

4 Dave Ragget, Arnaud Le Hors, Ian Jacobs, "HTML 4.01",

5 R. Fielding, J. Gettys, J. Mogul, H. Frystyk, L. Masinter, P. Leach, T. Berners-Lee, "Hypertext Transfer Protocol -- HTTP/1.1",
   RFC2616, June 1999.
Appendix A Æ» RAPDF Examples

NOTE: Additional examples are available at:
http://www.embeddedinternetworking.com/remoteaccess/

A.1 RAPDF Example with minimal formatting

<HTML>
<HEAD>
<meta http-equiv="Content-Type" content="text/html; charset=windows-1252">
<META name="type" content="RAPDF 1.0">
<TITLE>Remote Access to Home Control Center</TITLE>
</HEAD>

<BODY>

<LABEL for="ra_form_name">Remote Access to Home Control Center</LABEL>
<FORM name="Remote Access to MyDevice" action="remote@mail.com?subject=RAPDF%20Command%20Set" method="post" id="ra_form_name">

<P>
<LABEL for="ra_temp_lr">Living Room Temperature</LABEL>
<INPUT name="temp_living_room" type="text" id="ra_temp_lr"><br>
<LABEL for="ra_temp_mb">Master Bedroom Temperature</LABEL>
<INPUT name="temp_master_bedroom" type="text" id="ra_temp_mb"><br>
<LABEL for="ra_temp_gr">Guest Room Temperature</LABEL>
<INPUT name="temp_guest_bedroom" type="text" id="ra_temp_gr"><br>
</P>

<P>
<LABEL for="ra_light_lr">Living Room Lights</LABEL><br>
<INPUT name="light_living" type="checkbox" value="Living Room" id="ra_light_lr">
<LABEL for="ra_light_lr">Living Room Lights</LABEL><br>
<LABEL for="ra_light_ki">Kitchen Lights</LABEL><br>
<INPUT name="light_kitchen" type="checkbox" value="Kitchen" id="ra_light_ki">
<LABEL for="ra_light_ki">Kitchen Lights</LABEL><br>
<LABEL for="ra_light_ba">Bathroom Lights</LABEL><br>
<INPUT name="light_bath" type="checkbox" value="Bathroom" id="ra_light_ba">
<LABEL for="ra_light_ba">Bathroom Lights</LABEL><br>
</P>

<P>
<LABEL for="ra_safety">Safety Shut-off</LABEL><br>
<SELECT name="safety_shutoff" id="ra_safety">
 <OPTION>None</OPTION>
 <OPTION>Water Heater</OPTION>
 <OPTION>Furnace</OPTION>
</SELECT>
</P>
</FORM>
</BODY>
A.2 RAPDF Example with FIELDSET and LEGEND

<html>
  <head>
    <meta http-equiv="Content-Type" content="text/html; charset=windows-1252">
    <meta name="type" content="RAPDF 1.0">
    <title>Remote Access to Home Control Center</title>
  </head>
  <body>
    <label for="ra_form_name">Remote Access to Home Control Center</label>
    <form name="Remote Access to MyDevice" action="mailto:ra@mail.com?subject=RAPDF%20Command%20Set"
          method="post" id="ra_form_name">
      <fieldset>
        <legend>Climate Control Settings</legend>
        <label for="ra_temp_lr">Living Room Temperature</label>
        <input name="temp_living_room" type="text" id="ra_temp_lr"><br>
        <label for="ra_temp_mb">Master Bedroom Temperature</label>
        <input name="temp_master_bedroom" type="text" id="ra_temp_mb"><br>
        <input type="submit" value="Send Settings"> <input type="reset" value="Reset Values">
      </fieldset>
    </form>
  </body>
</html>
<LABEL for="ra_temp_gr">Guest Room Temperature</LABEL><br>

<FIELDSET><LEGEND>Light Controls</LEGEND>
  <INPUT name="light_living" type="checkbox" value="Living Room"
  id="ra_light_lr">
  <LABEL for="ra_light_lr">Living Room Lights</LABEL><br>
  <INPUT name="light_kitchen" type="checkbox" value="Kitchen"
  id="ra_light_ki">
  <LABEL for="ra_light_ki">Kitchen Lights</LABEL><br>
  <INPUT name="light_bath" type="checkbox" value="Bathroom"
  id="ra_light_ba">
  <LABEL for="ra_light_ba">Bathroom Lights</LABEL><br>
</FIELDSET>

<FIELDSET><LEGEND>Safety Shut Off</LEGEND>
  <LABEL for="ra_safety">Select the device to shut off</LABEL><br>
  <SELECT name="safety_shutoff" id="ra_safety">
    <OPTION>None</OPTION>
    <OPTION>Water Heater</OPTION>
    <OPTION>Furnace</OPTION>
    <OPTION>Stove</OPTION>
  </SELECT>
</FIELDSET>

<FIELDSET><LEGEND>Timing Information</LEGEND>
  <INPUT name="time_method" type="radio" value="Once">
  <LABEL for="ra_time_now">Activate settings now</LABEL><br>
  <INPUT name="time_method" type="radio" value="Daily">
  <LABEL for="ra_time_daily">Activate settings every 24 hours</LABEL><br>
  <INPUT name="time_method" type="radio" value="Weekly">
  <LABEL for="ra_time_weekly">Activate settings every 7 days</LABEL><br>
</FIELDSET>

</FORM>
</BODY>
</HTML>
Author’s Addresses

Olaf Pfeiffer  
Embedded Systems Academy  
50 Airport Parkway  
San Jose, California 95110  
Email: opfeiffer@esacademy.com

Dr. Paul (Pawel) Lukowicz  
ETH (Swiss Federal Institute of Technology) Zurich  
Wearable Computing Lab  
Gloriastasse 35  
CH-8092 Zurich  
Email: lukowicz@ife.ee.ethz.ch