EPSON

111-56-URM-017

TrueOrder™ KDS

Remote Management

About this Guide
This guide contains instructions for an organization with multiple TrueOrder KDS sites to set up centralized control over all sites using a single remote server. That server can provide firmware updates, parser updates, and even logo updates to every TrueOrder KDS configured to access it, reducing the need to arrange multiple site visits.

Form: F-730-028 v1.00

Table of Contents

1. <u>Introduction</u>

The remote management functionality in TrueOrder exists to allow easier centralized control over multi-site operations. Instead of having to send a technician to every site to push out the updates there, some updates can be performed from a central location and then pulled to each site when needed.

Note that all updates are only pulled when the TrueOrder device powers on; we minimize the interruptions to regular operations by insuring that the updates are only performed when the device has already been powered off and interruptions are already happening.

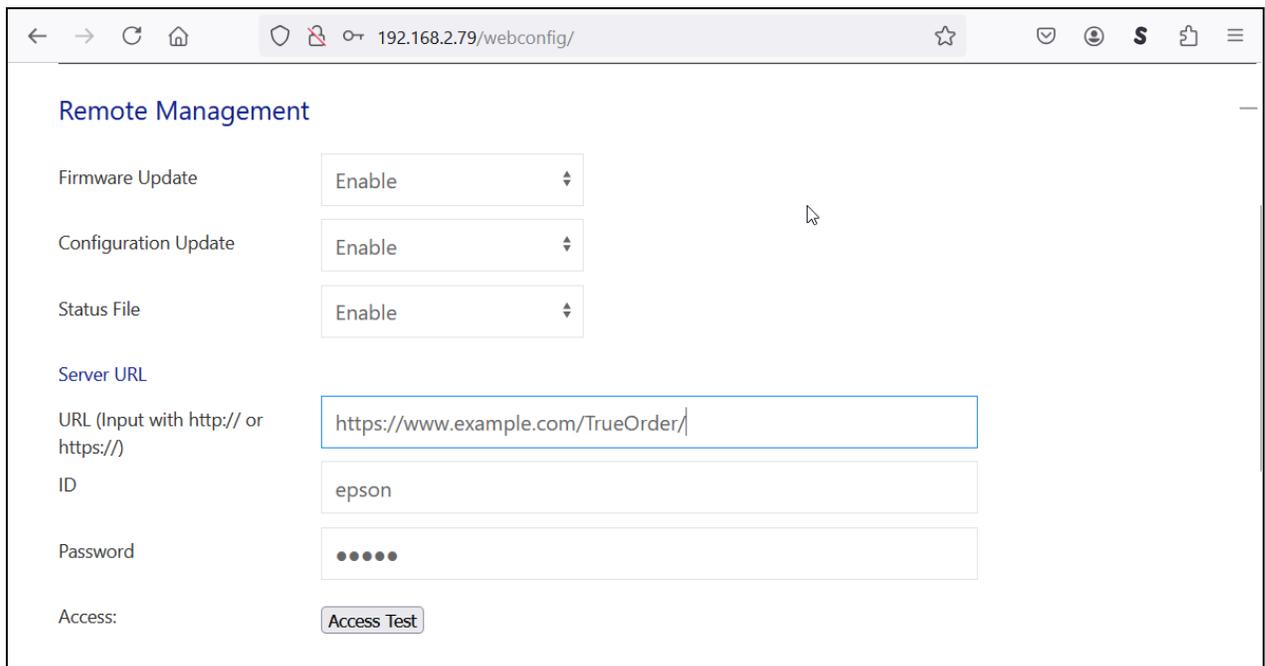# 2. Server Requirements and Initial Setup

2.1. Server Side

The central requirement for running remote management is to have a web server with a storage capacity of a few gigabytes, set up to be accessible to all TrueOrder devices at all locations. Any web server should work.

The web server needs to serve the contents of a single directory per device architecture, but does not need to provide any listings of that directory. All URLs will be hardcoded filenames, programmed names (constant per device but different between each device) or names referred to within one of the files.

If you want to implement the status file functionality as well, then the directory needs to contain a second directory named 'statusfiles', and this directory needs to be writable by the server.

2.2. Client Side

Configuring the TrueOrder side of things is done via the /webconfig/ URL:

The settings on the page are relatively straightforward:

- Firmware Update – enable the ability to update the TrueOrder firmware remotely
- Configuration Update – enable the ability to update the TrueOrder parser and Logo fields remotely
- Status File – enable the ability to transmit some status information to the server (If none of the three settings above are enabled, the rest of the fields will be disabled and uneditable)
- Server URL – the base URL for all server accesses from this device
- ID/Password – the authentication information used to access the server
- Access Test – used to test that the server is accessible from the TrueOrder device

This configuration has to be performed on every TrueOrder device in the system individually, normally with all of them being set to point to the same Server URL.
(This is not a requirement if there are good reasons to have different URLs, such as staged rollouts.)

3. <u>Server File Layout and Handling</u>

3.1. Model-Specific Layout

The server URL entered on the device is the parent URL, and it will always contain one or more directories based on the model names to be supported. The model name will be a four-character description such as 'mtic', 'ls89' or 'eloi'. Because different models can have different CPUs, they need different firmware and even different parsers, so they are kept in separate directories on the server.

- Server URL configured on TrueOrder: https://www.example.com/TrueOrder/

- Server directory for MicroTouch: https://www.example.com/TrueOrder/mtic/
  - Information file: https://www.example.com/TrueOrder/mtic/information.json
  - Firmware file: https://www.example.com/TrueOrder/mtic/fw-update_3.20mtic.zip
  - Parser file: https://www.example.com/TrueOrder/mtic/defaultparsermtic_3.0_arm64.ipk
  - Logos file: https://www.example.com/TrueOrder/mtic/logos.xml
  - Status directory: https://www.example.com/TrueOrder/mtic/statusfiles/
    - Stats: https://www.example.com/TrueOrder/mtic/statusfiles/mtic.001e06427e2e.json (These files are generated by the devices, not the administrator.)

- Server directory for Logic Controls: https://www.example.com/TrueOrder/ls89/
  - … (same as above)

- Server directory for ELO: https://www.example.com/TrueOrder/eloi/
  - … (same as above)

Everything else will be under one of those subdirectories.

3.2. Update Information

The most important file under this directory has a hardcoded name 'information.json'.

This file has the following format:
```
{
   "RemoteManagement" : {
      "Revision" : "2024-08-13",
      "Firmware" : "fw-update_3.20mtic.zip",
      "Logos" : "logos20240812.xml",
      "Parser" : "defaultparsermtic_3.0_arm64.ipk"
   }
}
```

This file will need to be managed manually on the server end. It consists of a revision string indicating a general collective version number of the files on the server, and three filenames to indicate the files the server has available. Note that the server may have other files on it, so this file indicates which of the other files are needed. This allows the system to get away with only having one hardcoded filename and there is no need to provide directory listings because all other names are in this file. Also previous versions of firmware can be safely archived on the server, since only the filenames mentioned here will be used.

Also note that if the revision string changes, **all** other files will be checked for changes, though the firmware file will be checked separately against the installed version as well. This revision string is explicitly not tied to any other firmware version, and can be whatever you want. Putting in the last modified date for the file would work.

Another useful note: a missing parameter or blank string for any of Firmware, Logos, or Parser will be treated as 'do not change' rather than 'remove'. So removing a parameter can be used to tell the TrueOrder device to not even download the file, even if the Revision string has changed.

3.3. Firmware Files

New firmware for a device will be provided as a ZIP archive. This is the same archive that would be used for a manual firmware update done on-site. The file name should not be changed; it will contain both the model name and the version number in the file, and these will be checked.

To update the firmware:
- The supplied file has to be put in the model name directory
- The information.json file needs to be modified to include the full filename under "Firmware"
- The information.json file needs to be modified to change the "Revision" string

Each device will then update itself at its next power cycle if 'Firmware Update' is selected.

3.4. Parser Files

The parser files are provided as .ipk package files. The file name should not be changed; it will contain both the model name in the file, and this will be checked.

To update the parser:
- The supplied file has to be put in the model name directory
- The information.json file needs to be modified to include the full filename under "Parser"
- The information.json file needs to be modified to change the "Revision" string

Each device will then update itself at its next power cycle if 'Configuration Update' is selected.

3.5. Logo Files

The logo file is not provided by Epson, but instead is generated locally. In the TrueOrder web configurator, on the page where logos can be uploaded, there is also a button for export that will create an XML file containing an encoded version of the logo files.

Unlike the firmware and parser files, the filename is not particularly important. That said, it is useful to change the filename each time, because the filename will be saved and returned in any status file, so keeping the name unique with each change can assist in tracking which devices have updated their files already.

To update the logos:
- The generated file has to be put in the model name directory
- The information.json file needs to be modified to include the full filename under "Logos"
- The information.json file needs to be modified to change the "Revision" string

Each device will then update itself at its next power cycle if 'Configuration Update' is selected.

3.6. Status Files

The status files are always PUT into the 'statusfiles' directory under the model name directory, and they always have a filename of the form xxxx.mmmmmmmmmmmm.json, where 'xxxx' is the model name and 'mmmmmmmmmmmm' is the device MAC address in lowercase hexadecimal format.

Note that this is a 'PUT', not a 'POST', and since the filename is always the same it will by default simply overwrite the previous file. If you wish to have tracking or a database on the server side this will require coding on the server to determine when a new file is added.

The file format is a simple JSON file:

```
{
    "fwVersion" : "3.20",
    "modelCode" : "mtic",
    "parser" : "defaultparsermtic_3.0_arm64",
    "nvramTotal" : "14379MB",
    "nvramFree" : "9679MB",
    "ramTotal" : "1865MB",
    "ramFree" : "218MB",
    "LogoFile" : "logos20240812.xml"
}
```

Most of the entries are fairly self-explanatory:
- fwVersion – current installed version of firmware
- modelCode – four-character model name for this device
- parser – most recently installed parser name (this will not include the .ipk file extension)
- nvramTotal – total size of non-volatile memory
- nvramFree – current available size of non-volatile memory
- ramTotal – total size of board RAM
- ramFree – current available RAM
- LogoFile – most recently installed logo file (or missing if no logos were installed by remote update)

3.7. Operation Sequence

The basic sequence of actions on the TrueOrder side is fairly direct.
- If none of the enable flags are set, exit
- Download the information.json file
- If there is no Revision string, exit
- If the Revision string is different from the last one:
  - If 'Configuration Update' is enabled, there's a valid "Parser" field in information.json, and it contains the correct model code, download and install the Parser file
  - If 'Configuration Update' is enabled, and there's a valid "Logos" field in information.json, download and install the Parser file
  - If 'Firmware Update' is enabled, there's a valid "Firmware" in information.json, it contains the correct model code, and it contains a different version, download and install the Firmware file.
- If 'Status File' is enabled, upload the status file to the server

Note that because a full firmware installation will restart with the updated remote management application, the information.json file will be downloaded a second time before the status file is sent.