



## **Electric Imp: Blessing Overview** card

last updated: october 2013

more information on blessing available at  
<http://electricimp.com/docs/manufacturing/>

# Blessing in context



Develop model firmware “gumball” in the IDE

**model:** gumball



**model:** gumball  
**build:** 129

Promote a “gumball” build to Ops Console



Confirm and deploy chosen build



*Blessed* devices automatically download the latest production firmware of its model – model: gumball, build: 129 – when the end user connects the device to the Internet



Repeat “develop > promote > deploy” process any time

# What you need to bless devices



model: gumball

- Upon blessing, the device’s unique ID is assigned to a model e.g., model: gumball, in the Electric Imp database. When the end user connects to the device to the Internet it will run the latest deployed build of that model

For blessing devices to a model, you will need:

<b>Model factory firmware</b>	<b>Factory imp(s)</b>	<b>Production line</b>	<b>HTTP callbacks*</b>
<ul style="list-style-type: none"><li>• Factory firmware runs on each device on the production line to test and bless the device.</li></ul>	<ul style="list-style-type: none"><li>• A factory imp runs factory firmware and is inserted into each device.</li><li>• Factory imps can be created from any imp, so you can have spares for the line.</li></ul>	<ul style="list-style-type: none"><li>• To bless devices on the production line, the factory imps will need access to the Internet via WiFi and each device will need to be powered on during blessing.</li></ul>	<ul style="list-style-type: none"><li>• Webhooks can be used to track device test results and blessing.</li><li>• More information at <a href="http://electricimp.com/docs/manufacturing/webhooks/">http://electricimp.com/docs/manufacturing/webhooks/</a></li></ul>

\* Optional

## Quick overview of how-to bless

- 1 Develop factory firmware**
  - Develop factory firmware in the IDE and deploy via the Ops Console to factory imps
  - Factory firmware can test devices and, upon pass, bless the device
  - Test pass/fail is shown by a solid green or red LED, respectively, on the imp
  - You will need to use the blessing API in the factory firmware

---
- 2 Designate factory imp(s)**
  - Assigning an imp as a factory imp can be done via the Ops Console
  - A minimum of 1 factory imp is needed per production line, but more can be made for higher throughput, multiple lines, spares, etc.
  - The factory imp only needs to be Blinked-up once

---
- 3 Run factory firmware**
  - Insert the factory imp into each device on the production line
  - The factory imp will boot and run the factory firmware
  - Pass or fail will be indicated by solid green or red on the imp's LED, respectively
  - The imp is then removed and used to bless the next device on the production line

---
- 4 View model activity (on-going)**
  - Track the number of blessed devices per model in the Ops Console

---
- 5 Ship devices; end-user BlinkUp**
  - When the end user connects the device to the Internet using BlinkUp in your app, it will automatically download and run the model firmware

## 1 Develop factory firmware

*The factory firmware is responsible for testing and blessing your new devices. It's run by inserting the factory imp into each new device on the production line.*

*Factory firmware should perform any desired diagnostic tests on the device's hardware – checking peripherals can be accessed, and so on. After a successful test, it blesses the device via the blessing API, which permanently associates the unique ID of the device with your model ID – and hence ensures it will automatically load the correct firmware from then onwards.*

*Develop your factory firmware using the IDE and deploy to factory imps via the Ops Console.*

### **test**

*"customer defined" [firmware only]*

Runs diagnostic test on your device's hardware.

If testPass is true, the device is blessed and the imp's LED is turned on solid green for success,

If testPass is false, the device is not blessed and the imp's LED is turned solid red for failure

### **server.bless\***

(bool testPass, function callback)

*API call from factory firmware*

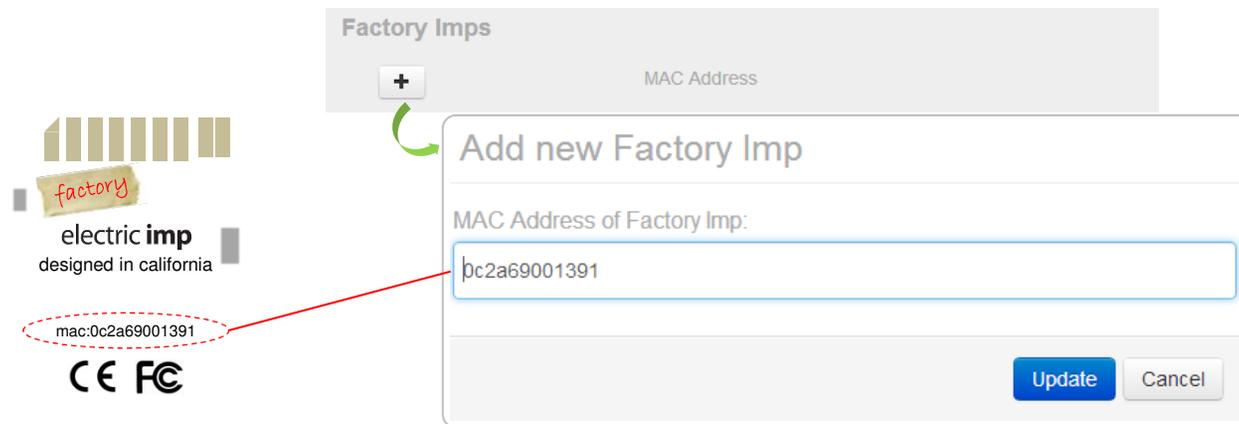
If testPass is true, permanently enrolls a devices as part of a specified model so that when the end-user connects the device via BlinkUp the device will download the latest deployed model build from your account. The LED is turned on solid green and the callback is called after the blessing has been completed.

If testPass is false, turns the imp LED on solid red. No blessing is performed and the callback is not called.

*Factory firmware does not run agent code. You can use the function `agent.send()` in the device firmware to post test results to a webhook.*

## 2 Designate factory imp

To test and run your factory firmware you will need to designate it to a factory imp. You can create factory imps via the Ops Console by entering the mac address of the imp (located on the back of the imp card). The factory imp must be BlinkedUp to your account via the Electric Imp before it can be assigned as a factory imp.

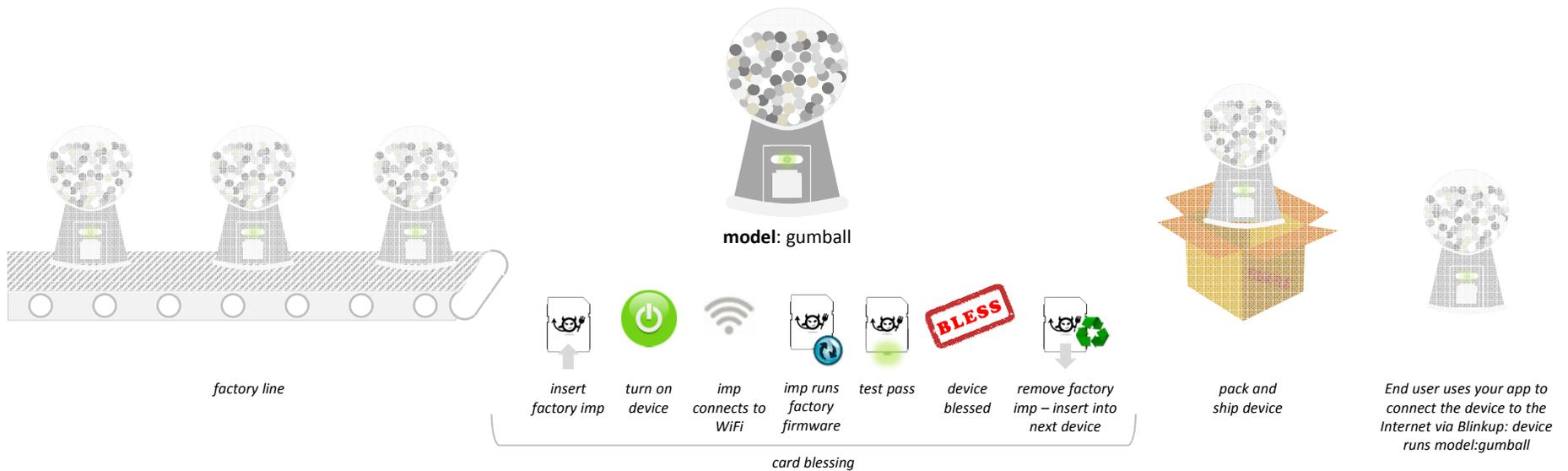


### Notes:

- Any imp card can be used as a factory imp
- You should mark the card so you do not switch it with another card accidentally
- A minimum of one factory imp is required, but you may designate and use as many as you like
- Factory imps can be re-designated to a new model and new factory firmware and used again after a production run
- To connect your factory imp(s) to the Internet, insert it in any imp-enabled device and use BlinkUp via the Electric Imp app. This needs to be done only once; the factory imp will continue to use these network details until you Blinkup again with a different network.

### 3 Run factory firmware: test and bless

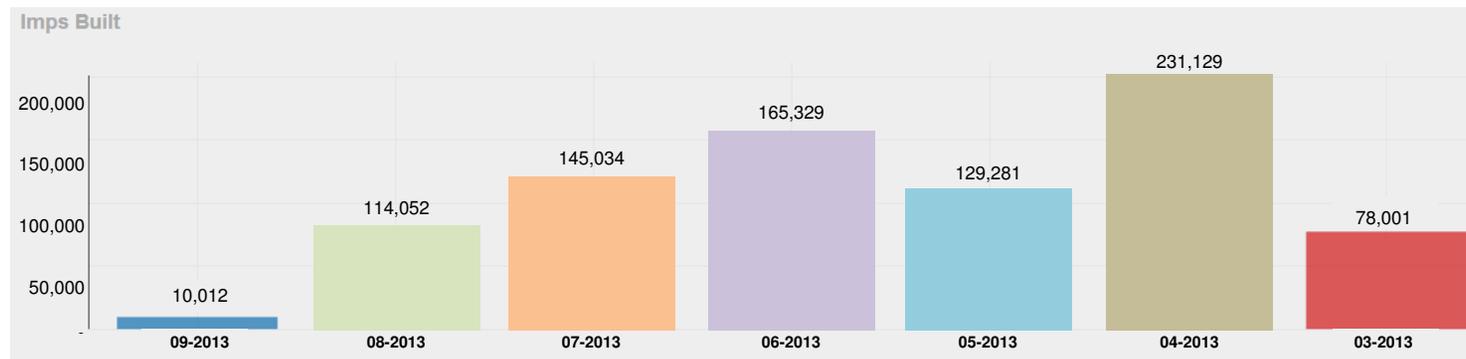
*On the production line, a factory imp is inserted into each device. The factory imp automatically connects and runs the factory firmware. The LED in the factory imp can indicate test pass/fail (solid green for pass and solid red for fail). After a successful test, the device will be blessed to its designated model.*



\* The LED in the factory imp will turn red if it does not pass the test

## 4 View model activity

To track blessed devices, you can view the Ops Console Beta. This tool allows you to view the number of devices that are built and activated. Additionally, the Ops Console allows you to deploy new model builds at any time if you have firmware updates.



[built] = the number of devices that were successfully blessed on the production line via a factory imp

You can also use webhooks to track blessed devices which can be added in the Ops Console. Learn more about factory webhooks at: <http://electricimp.com/docs/manufacturing/webhooks/>

### Add new Webhook

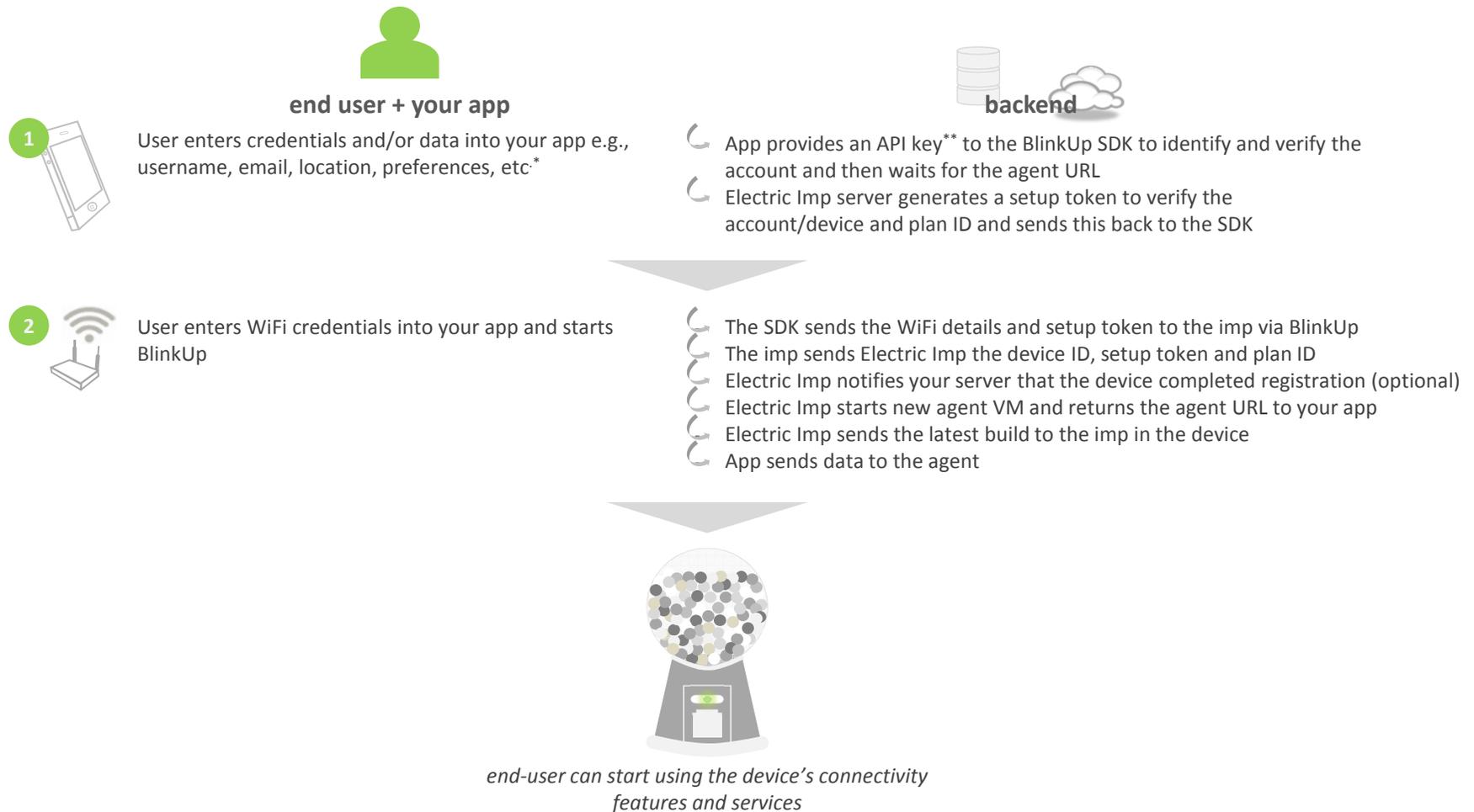
Url:

application/x-www-form-urlencoded

enrollment

## 5 End-user BlinkUp

*The device will automatically download the latest deployed build of its model firmware when the end user connects the device to the Internet with BlinkUp in your app.*



\* personally identifiable information (PII) stored on your server(s); PII will not be stored on Electric Imp servers

\*\* Electric Imp will provide an API key when the SDK is sent to you that will be associated to your Electric Imp account

## Resources

*You can find more details about blessing devices on the Electric Imp documentation center.*

<http://electricimp.com/docs/manufacturing/>

### Reference documents for blessing

- Sample factory firmware code
- Blessing FAQs for troubleshooting factory firmware, blessing and webhooks

### Reference for Blessing APIs

- `server.bless`

### Reference for Factory Webhooks

- Factory Webhooks