Mesh networking: setting up Google OnHub-based Wi-Fi

Brian Dipert - November 06, 2017

In a recent post, I mentioned that I planned to try out mesh Wi-Fi networking as a potential successor to the conventional router-plus-multiple-access point arrangement I'd long been using across multiple residences spanning ... oh ... 15 years or so. Shortly after submitting that earlier writeup, I decided to actualize my aspirations. I'd previously mentioned that I planned to test two different Google-branded configurations; the AC1200-based Google Wifi three-pack system and a set of AC1900-based, TPLink-developed OnHub routers (ASUS also partnered with Google on OnHub; these particular devices had an altered antenna array configuration and somewhat different cosmetics):
I began my testing with the AC1900 OnHubs and, after a positive experience with them, decided to just return the AC1200 Google Wifi units unopened for full refund from Newegg. Note that from everything I've read, the two options are identical from setup and functional standpoints. And as you can probably already tell from the above photos, the Google Wifi units are significantly smaller than the OnHub alternatives. So if aesthetics are important to you, I feel comfortable recommending the Google Wifi alternative, at the tradeoff of slightly less performance especially in large-client-count setups (as I type this, for example, I have 41 active devices on my residence network, most wireless-connected, with other network clients asleep or turned off right now).

In my previous post, I mentioned that Google doesn't seem very enthused about folks using OnHubs as mesh access points, although steps for doing so are documented on their support site and plenty of users have reported success in doing so. I believe I've figured out the root cause for the lack of enthusiasm, as well as how to alleviate Google's concerns. The core issue, I suspect, is that each unit needs to be upgraded to at least last December's firmware version before it can operate in a mesh configuration, either as a mesh node or the "root" node. And there's no way to "sideload" or manually "force" an online firmware update.

For the "root" node, i.e. the router, this version requirement isn't an issue ... the router is by default connected to the WAN (and from there to Google's update servers) via the broadband modem intermediary, so it'll receive the necessary update automatically. But for the mesh nodes connected to the router, they won't be "seen" by it unless they've got the requisite firmware release already pre-installed. So what I did as the initial step in the setup process was to sequentially hook up each of my three OnHubs to the cable modem, power-cycling the modem each time to ensure it saw the new unit connected to it. A firmware upgrade was then immediately initiated and took only a few minutes to complete; I could tell the update status via the unit's LED color and cadence, or could alternatively monitor progress via the setup app:
Speaking of the setup app, as the above screenshot suggests, the Google Wifi mesh system (whether based on OnHubs, Wifi hardware, or a combination of the two) is exclusively managed by an Android- or iOS-based app; there's no web browser-based configuration option as with most conventional routers. This limitation isn't unique to Google Wifi; the only mesh system I'm aware of that currently offer a web GUI option is Netgear's Orbi. It doesn't preclude network management via the WAN; since it's tied to your Google account login, you can monitor and configure the network from anywhere (well, OK, maybe not China ... ). Pragmatically, the Android or iOS requirement isn't much of a limitation (with apologies to the lingering Windows Mobile users out there), given the near-100% dominance of the combination of the two O/Ss in smartphones. And I'm not worried about ongoing support for the Android app, since the mesh network and O/S come from the same source. But had I chosen an alternative mesh network system, especially from a not-ye-established supplier (eero, EnGenius, Plume, etc.), the lack of a web browser-based management option might give me pause.

The app initially communicates with the hardware over Bluetooth:
Once you've enabled Bluetooth on the smartphone or tablet, the "root" node (i.e. router, which I set up first) is found in short order:
In my particular case, the initial setup "handshake" took place (I suspect) over NFC (near-field communication). For NFC-less Android devices as well as for use in the iOS version of the Google Wifi app (which I also briefly tested using my iPad), the setup code is alternatively printed on the OnHub underside:
At that point, you're off to the setup races (including firmware update, per the previously shown screenshot, if necessary):
Next step: setup of the mesh (access point) nodes. The steps shown in the screenshots that follow were identical for each node’s setup sequence. And the OnHub Bluetooth broadcast range seems to be pretty impressive; when I captured the screenshots, I was next to the OnHub on one end of the house, with the other OnHub at the other end of the house (the app saw them both):
### Where is your Wifi point?

This will help you identify it later.

<table>
<thead>
<tr>
<th>Location</th>
</tr>
</thead>
<tbody>
<tr>
<td>Family Room</td>
</tr>
<tr>
<td>Front yard</td>
</tr>
<tr>
<td>Garage</td>
</tr>
<tr>
<td>Hallway</td>
</tr>
<tr>
<td>Kitchen</td>
</tr>
<tr>
<td>Living Room</td>
</tr>
<tr>
<td>Master Bedroom</td>
</tr>
<tr>
<td>Office</td>
</tr>
</tbody>
</table>

NEXT →
And now that the mesh network is up and running, I'm able at any time to test both its robustness and that of the nodes' shared WAN connection to the Internet:
As you can see, setup was relatively straightforward, and successful. But how well did the resultant mesh network function, and how well did Google balance the seemingly contradictory goals of robust functionality and simple management? Stay tuned to my next post in this series for those and other details. Until then, I welcome your thoughts in the comments.

**Read the next part in this series:** [The pros and cons of mesh networking](#)

**Related articles:**

- Is it time to upgrade to mesh networking?
- Laundry detergent, cat litter, and Wi-Fi
- Plug and play mesh IoT sensor system unveiled
- Nest Shares Thread Code
- Simplify embedded Wi-Fi connectivity with Near-Field Communications
- Seven keys to understanding Thread protocol
- Maximize your Wi-Fi signal range with using these solutions