

Building My Own Linux-based Wireless Access Point

Contents

Context.....	3
First, My Router – A Linux-based PC	3
I Do This Because.....	3
“Small Plastic Routers”: For The Sane	3
Aftermarket Firmware.....	3
Limited Lifespan	3
I Decided To Not Be Sane	3
Benefits versus Drawbacks Of Linux-PC-As-A-Router	4
What Does This Have To Do With A Wireless Access Point?.....	4
Linux and Wi-Fi.....	4
Gathering The Hardware.....	6
The PC	6
Firmware Settings	7
Verdict	8
The Wi-Fi Cards	8
Wi-Fi Cards	8
Turning The Cards Into Something That Can Be Connected To A PC	8
More Pigtails	9
Antennas	10
Assembling The Cards	11
SSDs for the operating system	12
Assembly	14
Installing the SSDs	14
Installing the Wi-Fi cards.....	16
Bolt Cutters And Zip Ties.....	16
Installing Linux.....	19
Appliance Philosophy – Not Desktop.....	19
Why I Chose Debian	19

The Debian netinst image..... 19

 Getting The Bootable netinst image..... **Error! Bookmark not defined.**

 Creating Bootable Media **Error! Bookmark not defined.**

Context

First, My Router – A Linux-based PC

Before we get into my project, we'll talk about a journey I traveled before: building my own Linux-based router.

I Do This Because...

I like tinkering with networking. I find the ability and mechanics of running a small network from a residence, and having it securely interconnect with the world through the Internet, to be extremely interesting.

“Small Plastic Routers”: For The Sane

“Small plastic routers” are common household devices now. (This is in contrast to “huge metal rackmount enterprise routers”.)

Normal people tend to use equipment provided by their ISP that are “combo” boxes – those combine the functions of modem, router, wired Ethernet switch and wireless Ethernet switch into one box. Or they may buy their own equipment – modem, router, switch, or any combination of those.

Normal people just want a convenient, out-of-the-box solution to getting devices in their home online, and ISP-provided or store-bought devices are fine. These devices are obviously geared toward typical consumer Internet use and typical consumer technical knowledge, and that is quite lacking for the technically interested, curious network tinkerer.

However, there is a bigger problem with these small plastic routers ... keep reading.

Aftermarket Firmware

Is there middle ground? Sure – in the late 2000's and 2010's most small plastic routers (most already running Linux) could be flashed with aftermarket firmware – DD-WRT likely being the most famous, but others such as Tomato and OpenWRT also out there. You're leaving “normal person” territory by doing this.

Limited Lifespan

Most small plastic routers have a limited lifespan, because source code for the networking hardware drivers are often not released-only binary modules. This is particularly a problem for devices with Broadcom hardware. This means the Linux kernel on these devices is not upgradeable because binary modules only work for a specific kernel version. With the source code you can rebuild for any kernel version that's compatible.

- You always want to be able to upgrade the kernel to take care of security vulnerabilities.
- Open source drivers often get updates for longer than manufacturers who don't release the code and abandon drivers for nonprofitable or old product lines.

I Decided To Not Be Sane

Hence I permanently left the small plastic router world about two years ago, and replaced my very-long-in-the-tooth Linksys E2100 with a Debian Linux-based PC. It's not the first time I used a PC as a router-- I've been making a go at doing that in some fashion or another since I got my first broadband connection all the way back around 2005.

Benefits versus Drawbacks Of Linux-PC-As-A-Router

The benefits of using a Linux PC as a router are:

- No need to worry about out-of-date firmware.
 - o Debian—the operating system I chose--always gets updates.
- What is running is completely under my control
 - o No cloud or remote-management services running that I don't want or can't turn off.
- Able to put gobs of RAM and storage in the system to run advanced stuff right on the router.
 - o If I WANT cloud services running on the router, I can add the RAM and storage they need to thrive.
- Able to put as many NICs and other hardware in the system as I want.
 - o No need for VLANs when physically separate LANs can be used!

There are drawbacks:

- You need to know a lot about Linux, or be willing to learn.
 - o It took me a long time to get this knowledge, and is something I enjoy. Not for everyone.
- You are 100% responsible for security.
 - o This is a big deal.
- You are 100% responsible for everything, really. No more calling your ISP because your combo box stopped working.
 - o This means you need to know your stuff with networking – I think I do a little bit.
- Power consumption is higher than a small plastic router.
- A PC takes up more space and is heavier than a small plastic router.

For my home network's router, I judged the benefits to outweigh the drawbacks for me – especially since learning and using Linux and its networking features are a source of fun to me.

What Does This Have To Do With A Wireless Access Point?

Any modern home network needs Wi-Fi. We're fully in the post-PC age now and wired devices are going to be in the minority. My lovely self-built router above is wired only.

Linux and Wi-Fi

Your basic non-wireless router needs 2 network interfaces. So if you want to take a PC that has 1 onboard NIC, and add a second NIC to make a wired-only router, it's easy – just buy the cheapest wired PCIe NIC you can find on Amazon and throw it in.

Can you do the same with Wi-Fi? Yep. Just like there are PCIe cards that add wired network jacks to a PC, there are PCIe cards that add Wi-Fi radios. But things are more complicated with Wi-Fi.

Wi-Fi interfaces have roles (modes)

Wi-Fi, apart from working using radio waves, has other important differences from wired networking. One of those is the fact that there are roles for Wi-Fi interfaces and the interface can only be in one of them: the important ones here are STA ("station" - system wanting Wi-Fi) or AP ("access point" – system providing Wi-Fi).

AP mode is needed

Not all Wi-Fi cards support AP mode – or more specifically, their *drivers* don't support AP mode or the manufacturer doesn't make them available. I can't blame them because most people buy Wi-Fi cards for laptops or other devices that connect to an AP. I'm the weird one here trying to build my own AP.

Furthermore, that's not Linux's fault - everything on the Linux end supports the AP mode fine and has for quite some time – because... most small plastic routers actually run Linux.

Hardware that supports AP mode is hard to find

It would be lovely if you could just open up a small plastic router, take out its mini-PCIe card, and throw it in a PC. However, the Wi-Fi hardware in these plastic boxes is often specific to the box and part of the motherboard, and not something you can just buy in a PC Wi-Fi card. (You actually could pull the mini-PCI (not PCIe!) card out of the first hardware revision of the famous Linksys WRT54G).

- Because of this, I currently actually *do* have a small plastic router connected to my PC-as-a-router – but it's in AP mode and doesn't do any routing. More on that later.

It's actually a challenge to find PCIe Wi-Fi cards that both support modern Wi-Fi speeds (802.11ac) and AP mode, but they are out there – and I bought a couple.

Gathering The Hardware

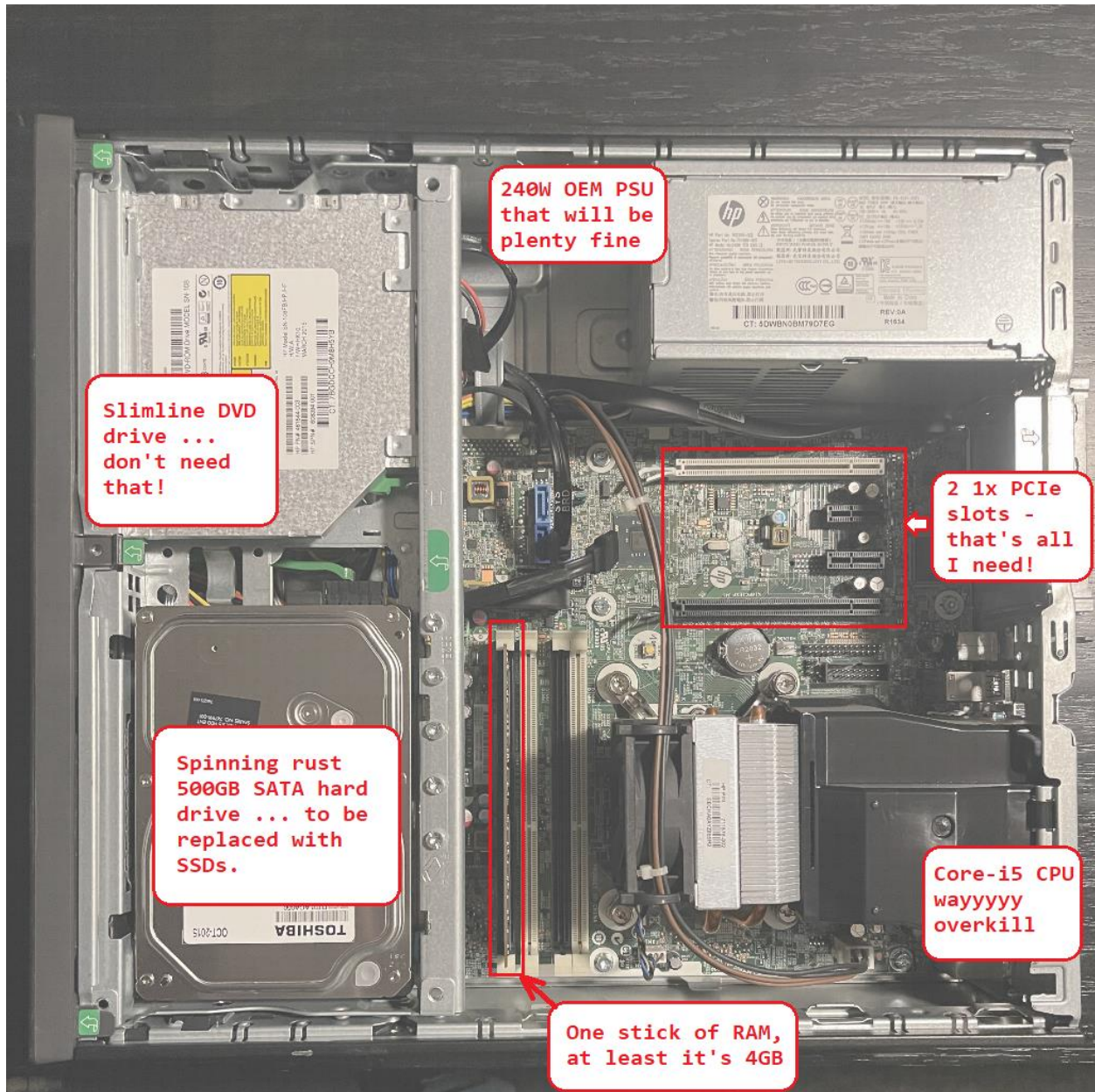
The PC

Well, if we're gonna build a PC-based wireless access point, we need a ... PC!

My workplace graciously let me have a decommissioned HP EliteDesk 800 G1 SFF. Decent desktop office computer from circa 2015.



It's big, but not too big. Let's have a look at the innards.



So we have what's probably a dual-core Intel Core i5, single stick of 4GBytes of RAM, enough PCIe slots for our Wi-Fi cards, and enough SATA ports to install 2 SSDs in a RAID-1 configuration.

Importantly it features an Ethernet port on the back, so we can bridge the Wi-Fi into a wired network or router. I haven't checked yet but I'm certain it's gigabit.

Firmware Settings

When entering the firmware screens, I was expecting HP's newer graphical UEFI menu, and was pleasantly disappointed. Random opinion no one asked for: I like the no-nonsense Compaq menu quite a lot.

Anyway, looking around reveals that this box supports virtualization (both VMX and VT-d), Secure Boot (disabled), and Intel AMT (also disabled).

We don't need any of that, so we'll disable all of those functions.

Verdict

This PC will be far more than adequate for this purpose, from a base hardware standpoint.

The Wi-Fi Cards

Wi-Fi Cards

So here is my Amazon order for my selected Wi-Fi card – the WLE900VX, supporting 802.11ac.



This was selected after some research about the QCA9880, and finding that the QCA9880 had good Linux support, AP mode support, and everything needed for an access point.

- You'll notice the date – November 4, 2020.
- Yep, a few years ago.
- Confession: I actually tried to get wireless working right on my main PC-based router, but it did not work out, and I'll detail that later.
- So this isn't really my first rodeo with this, but maybe it'll work this time!

You'll notice I ordered two. That's because this card's radio works on *either* the 2.4GHz or 5Ghz bands, but not both. So two radios are needed.

- You're seeing a snapshot of my Amazon order because--it's not available anymore! Guess I'm lucky I got it when I did.

Turning The Cards Into Something That Can Be Connected To A PC

So the motherboard of the PC I'm using does not have onboard mini-PCIe slots. Not surprising – I think having one for Wi-Fi on desktop motherboards is becoming a thing but it definitely wasn't a thing on business computers in 2015.

No big deal, we'll just get two of these:



Roll over image to zoom in



PCI Express to Mini PCI Express Card Adapter Wireless Network Ethernet LAN Mini PCI-E Riser PC to Laptop Network Card Adapter Converter Antenna WiFi

Visit the Zerone Store
 ★★★★★ 39 ratings | 5 answered questions

\$13¹⁹

✓prime
 FREE Returns

Coupon: ☐ Apply 5% coupon Terms

Save up to 10% with business pricing. Sign up for free Amazon Business account

Extra Savings Promotion A... 1 Applicable Promotion

Brand	Zerone
Hardware Interface	Ethernet, PCI
Color	default
Compatible Devices	Laptop, Desktop
Item Dimensions LxWxH	4.8 x 2.76 x 0.79 inches
Data Link Protocol	Ethernet
Item Weight	0.03 Kilograms


About this item

- [Function]—This is just an network adapter to connect a mini pcie wireless Card to a PC, it can be a PC method and mini PCIE


The mini-PCIe card will connect right to that, and it has convenient holes for the antenna pigtails.

More Pigtails

The WLE900VX has 3xMIMO meaning ... it uses 3 antennas. Glad the adapter I chose had 3 holes for the pigtails, but it only came with 1 pigtail. No biggie, just had to get 2 sets of thesez:



Roll over image to zoom in



Pack of 4 RF U.FL(IPEX/IPX) Mini PCI to SMA Female Pigtail Antenna Wi-Fi Coaxial RG-178 Low Loss Cable (12 inches (30 cm))

Brand: GP Electric
 ★★★★★ 172 ratings

\$18⁸⁹ (\$4.72 / Foot)

✓prime
 FREE Returns

Get a \$150 Gift Card: Pay \$0.00 ~~\$18.89~~ upon approval for the Amazon Prime Rewards Visa Card. No annual fee.

Size: 12 inches (30 cm)

2.5 inches (6.5 cm) \$11.84 (\$28.88 / Foot) ✓prime	4 inches (10 cm) \$11.99 (\$9.15 / Foot) ✓prime	5.5 inches (14 cm) \$13.98 ✓prime	7 inches (17.8 cm) \$13.09 (\$5.62 / Foot) ✓prime
---	---	---	---

12 inches (30 cm)
\$18.89 (\$4.72 / Foot)
 ✓prime

Incidentally, I learned the name of the connector that the antennas screw into is called SMA.

Antennas

I had the feeling I would need antennas, so I got two of those:



SMA compatible—and also the screw connectors look like they'll fit.

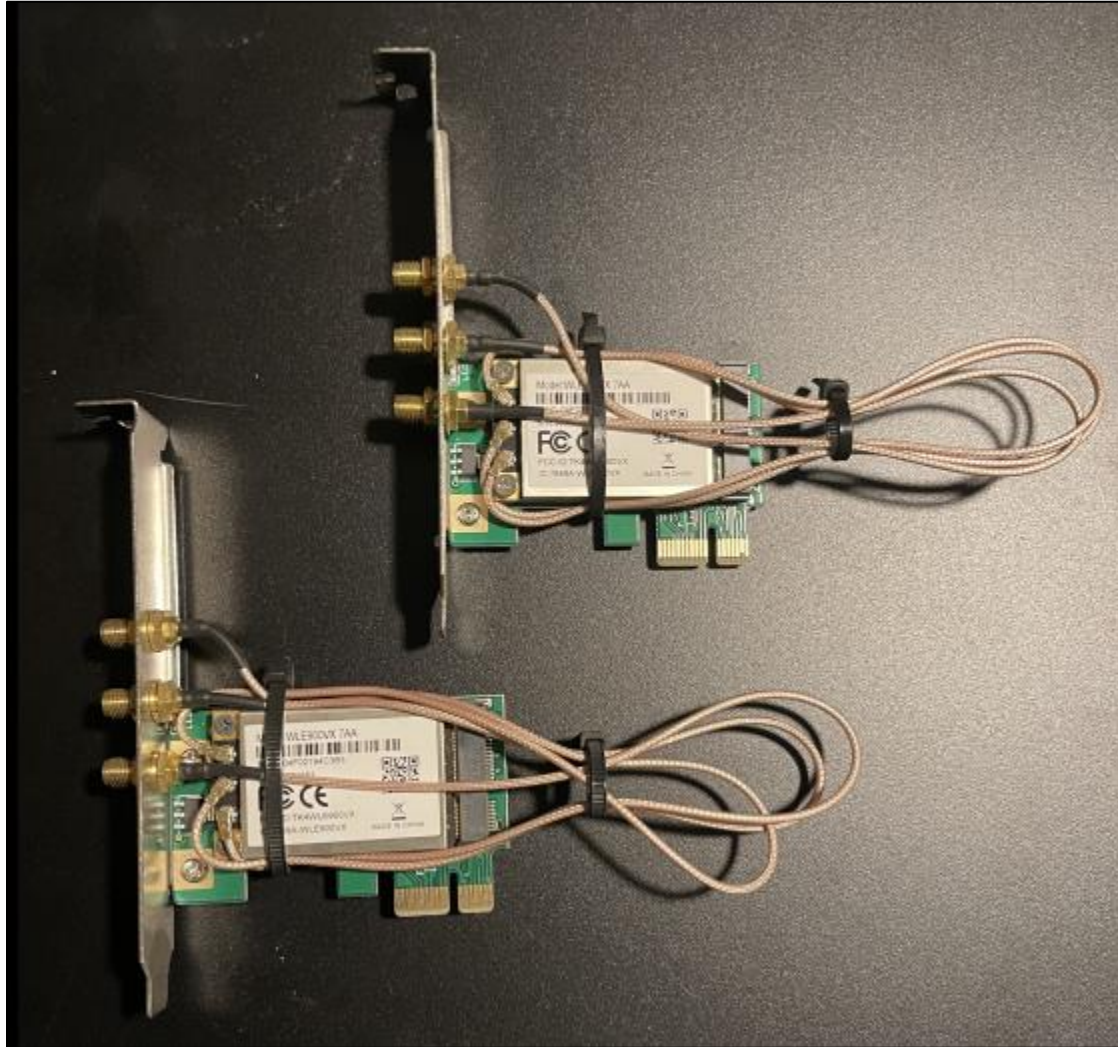
So those will plug right into the back of the cards from the PC.

Another cool thing about these not mentioned in the product description is that the base is magnetic. It conveniently attaches to metal.

Assembling The Cards

Some assembly required: namely, putting the mini-PCIe cards in the adapters, and connecting the pigtails from the cards to the back of the adapter bracket.

To keep the pigtails out of the way, I used 2 zip ties. Here's the result:



Ugly, sure—and that will be a recurring theme here.

I'm not sure if this is good for a radio interference perspective. I'm wondering if shorter ones would be better, but it's functional. Maybe if I put aluminum foil in between the wires to protect the signal...

Just kidding, I didn't do that.

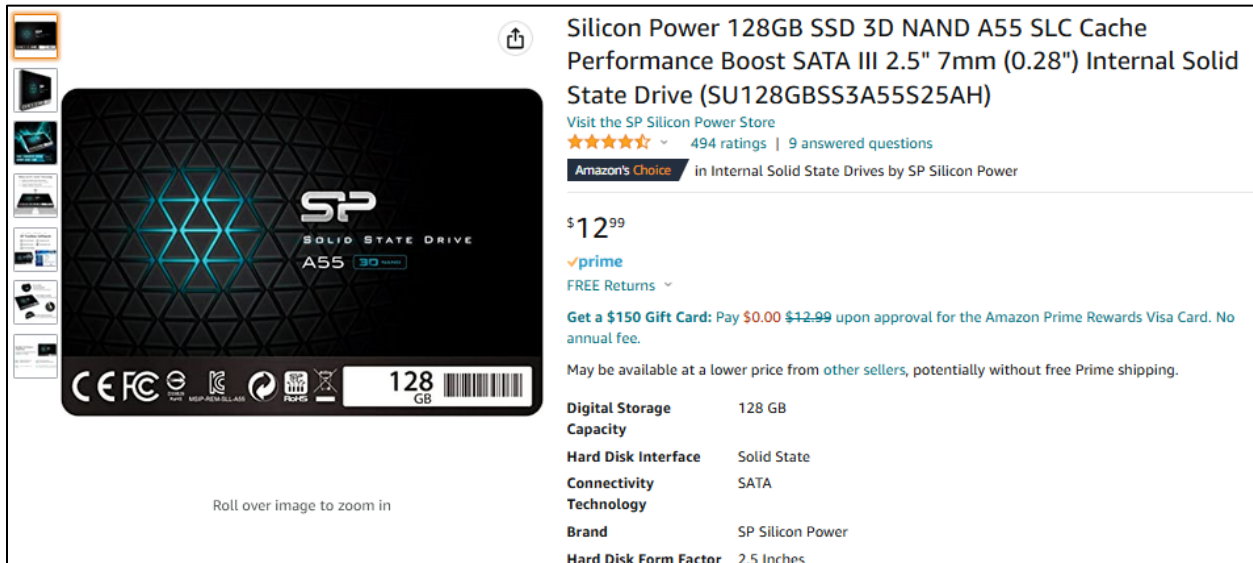
SSDs for the operating system

So we need to provide a home for Linux on this box.

If you were paying attention to the photos of the PC above, this system came with a 500GB mechanical hard drive. There is no reason in 2023 to be using a mechanical hard drive for anything other than bulk storage, especially in an “infrastructure” device that’s going to sit out of sight, potentially for years, and not even have a monitor connected.

128GB SATA SSDs are dirt cheap at the moment. So I got 2 of them – a non-major brand that I’ve used before.

We will install Linux on a RAID-1 for reliability. For \$12.99, why not.



Silicon Power 128GB SSD 3D NAND A55 SLC Cache Performance Boost SATA III 2.5" 7mm (0.28") Internal Solid State Drive (SU128GBSS3A55S25AH)

Visit the SP Silicon Power Store
★★★★★ 494 ratings | 9 answered questions
Amazon's Choice in Internal Solid State Drives by SP Silicon Power

\$12.99
✓prime
FREE Returns

Get a \$150 Gift Card: Pay \$0.00 ~~\$12.99~~ upon approval for the Amazon Prime Rewards Visa Card. No annual fee.

May be available at a lower price from other sellers, potentially without free Prime shipping.

Digital Storage Capacity	128 GB
Hard Disk Interface	Solid State
Connectivity Technology	SATA
Brand	SP Silicon Power
Hard Disk Form Factor	2.5 Inches

Pros and cons of other options here

Of course... other options were possible:

- Get 2 NVMe to PCI adapters and use NVMe,
 - o BUT that’s more expensive.
 - o Also I don’t know if this motherboard is weird about devices other than graphics cards in the x16 slot. This is business OEM hardware here that sometimes is weird like that.
- Could have installed and ran Linux off of USB storage device,
 - o BUT they’re not too reliable, slow, and might be accidentally removed.
 - o RAID over multiple USB storage devices is possible but probably a big pain.
 - o Regarding slowness, while I don’t need super high storage performance I would like it to boot quickly

Run it from the optical drive that’s in the system,

- o BUT
 - Probably would take ages to boot.
 - The optical drive might not even work.
 - I don’t even know if I have any blank CDs or DVDs anywhere.
 - It’s 2023.

- Setup PXE boot off of my main router or home server,
 - o BUT I want this to be a standalone device and not dependent on my router.
 - However with this I could have avoided putting any storage in the box at all.
 - Maybe another time.
- Get 2 SD-card-to-SATA adapters and use SD cards.
 - o BUT that's actually more expensive than 2 128GB SSDs.
 - o It'll also probably be slow unless I get expensive SD cards.
 - o SD cards are sometimes unreliable.

Is 128GB enough?

If you think it isn't, you are looking at this from a Windows or desktop Linux perspective. We'll be installing the minimum we need to get this working.

- The base Debian Linux operating system is very lean and can comfortably run in 8GB if you don't need GUI tools.
- If we were to do things like run servers, web interfaces, etc. we would need more space.

So 128GB is more than enough. I would have selected a lower capacity SSD drive if it was cheaper than the 128GB; it seems 128GB is the sweet spot right now, at least on Amazon.

Assembly

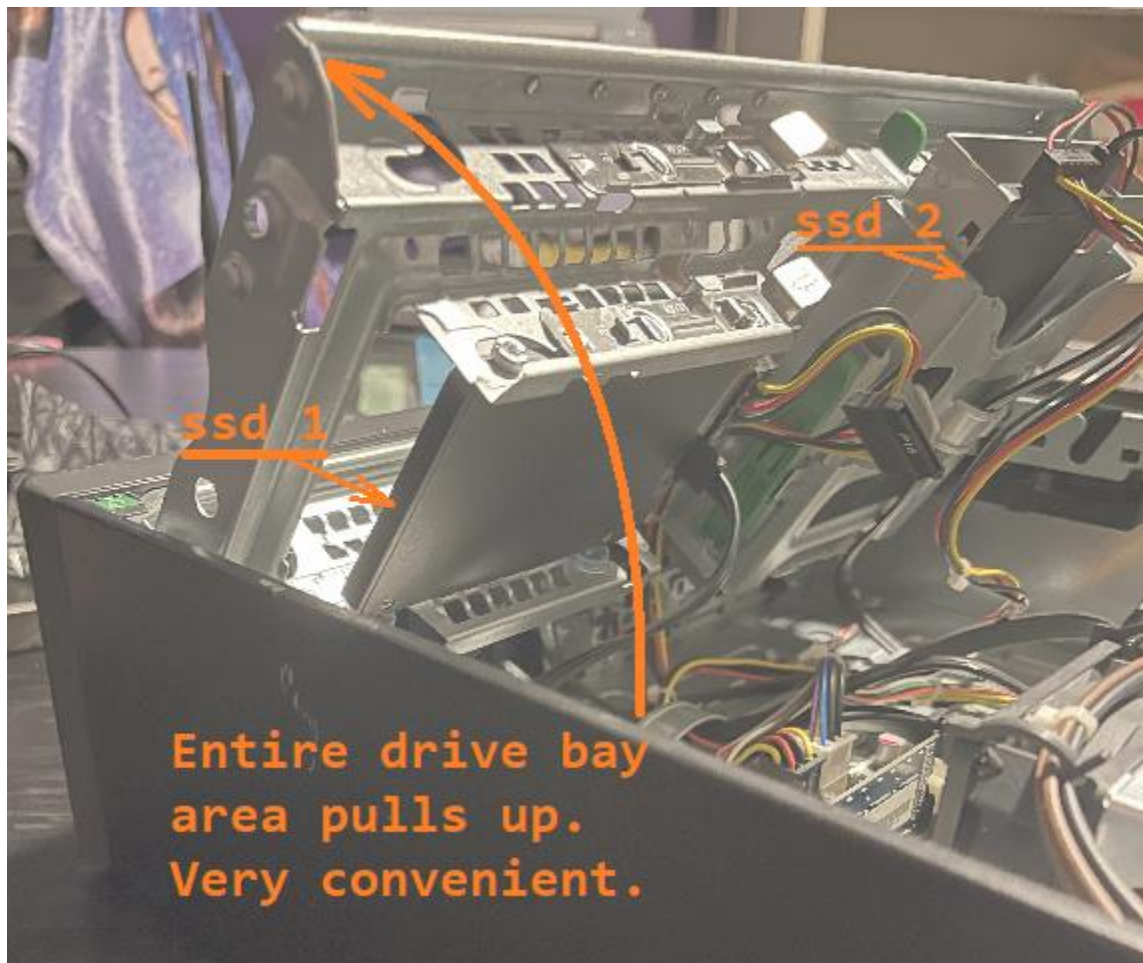
Installing the SSDs

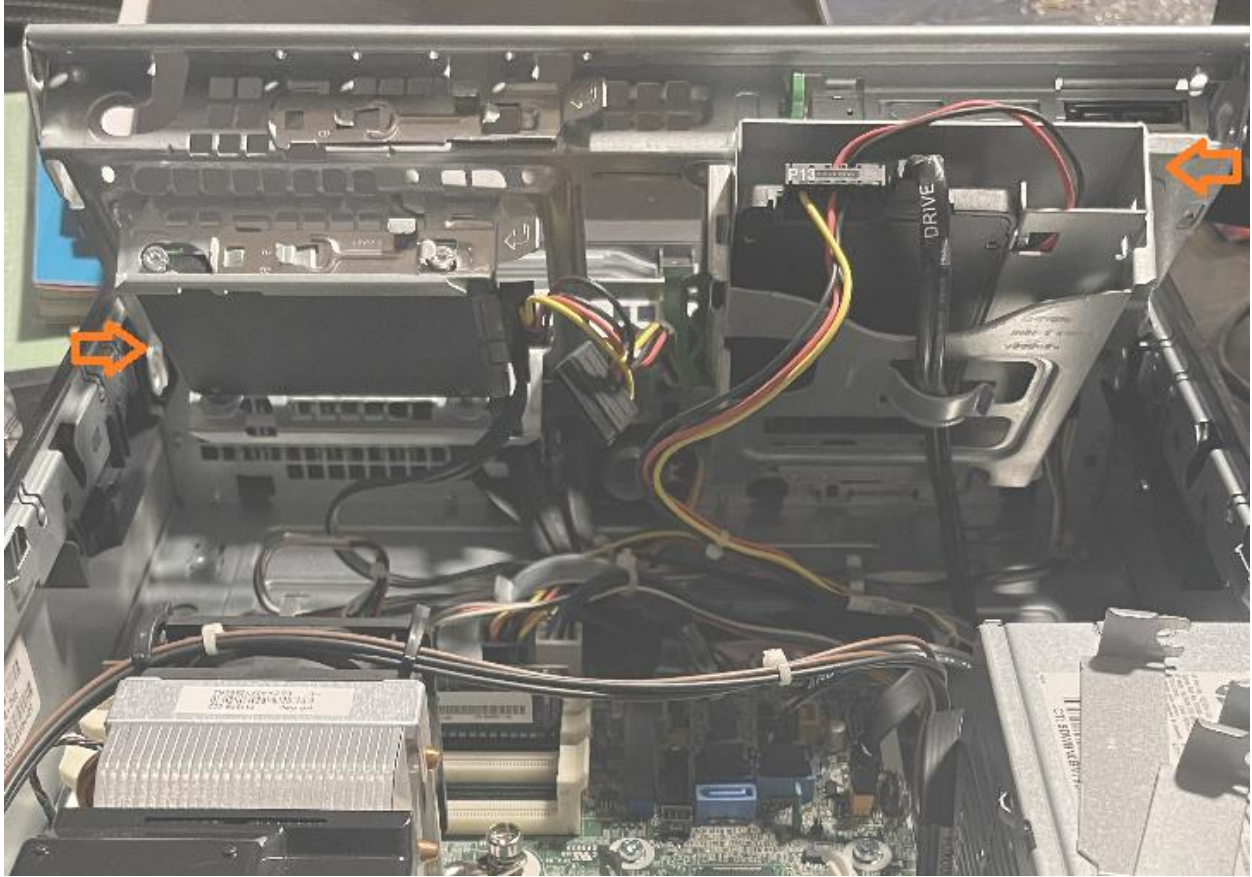
This wasn't too bad.

Looks like this PC has room enough for 2 3-inch hard drives and 1 2.5-inch hard drive.

One SSD will go in the 2.5-inch space and I have an adapter to put the other in the 3-inch space. Below you see the SSDs installed and the old mechanical hard drive removed.

I know these pictures are terrible, but it's OK. This whole thing is terrible. I should not be doing this.





We will plug our SSDs into SATA0 and SATA1 on the motherboard.

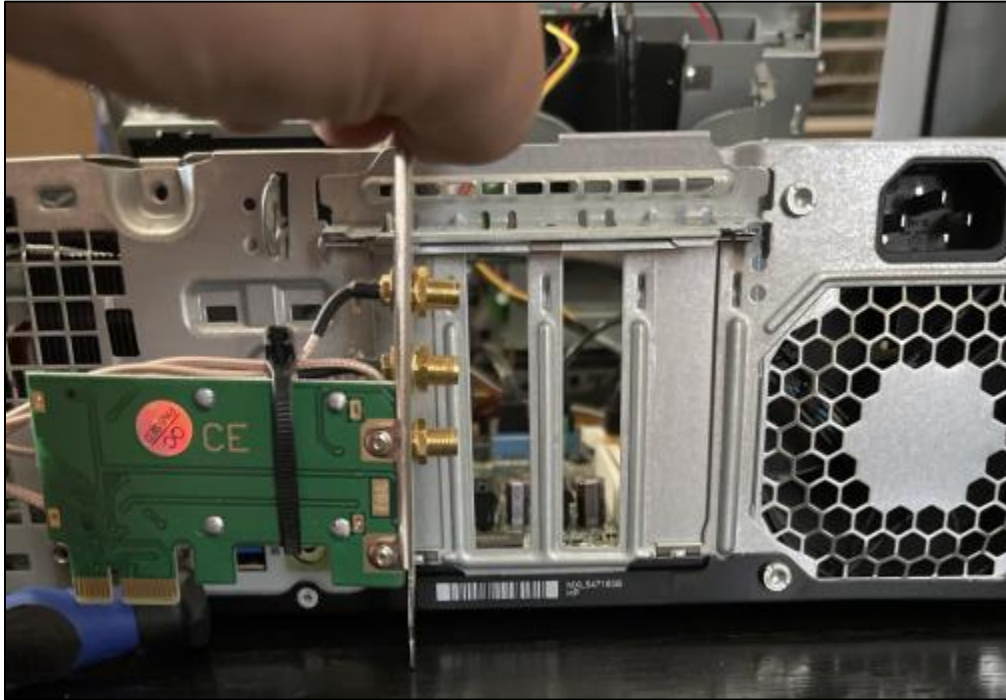
I'm leaving the optical drive disconnected but in the chassis – don't really have a use for that and don't have anything to replace it at the moment.

(If you're looking at the motherboard picture, you'll see SATA3 and SATA4 and I think a SATA5—there's solder pads but no port there.)

Installing the Wi-Fi cards

Well, looks like I made a mistake.

If you were particularly astute you may have noticed it and were chuckling up to this point.



I didn't order LOW-PROFILE adapters – but to be fair, the *original* computer I put these into (my precious PC-based Linux router I talked about earlier) was not low-profile.

I could, of course, just stick the cards in there without a bracket and let the pigtails drop outside of the chassis. I'm afraid I will accidentally pull on something and damage it, though.

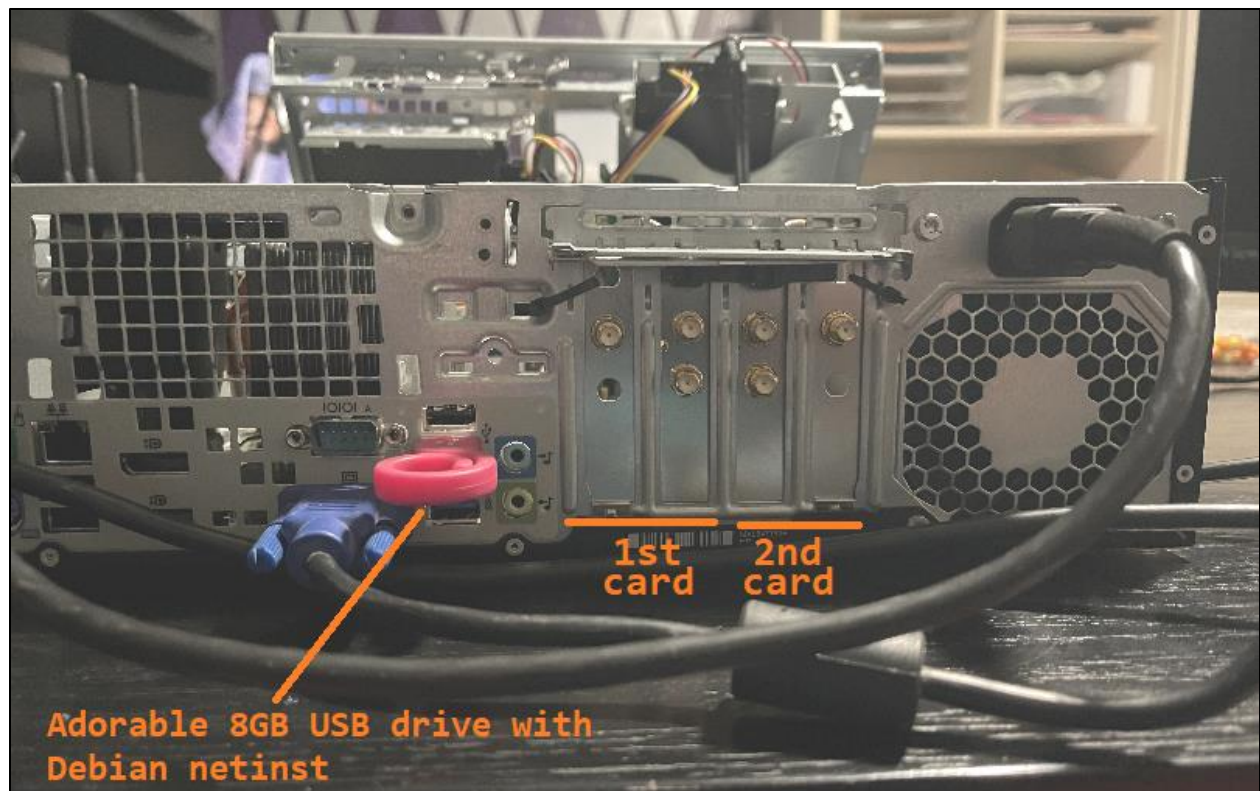
Bolt Cutters And Zip Ties

What I can do is just cut the existing brackets down to size.

That presents a second problem, the top antenna hole is covered by the clasp that keeps the cards secured. I could remove that clasp (somehow), or ... for some reason I have some additional normal-profile brackets with 4 antenna holes—I really don't know why I have these.

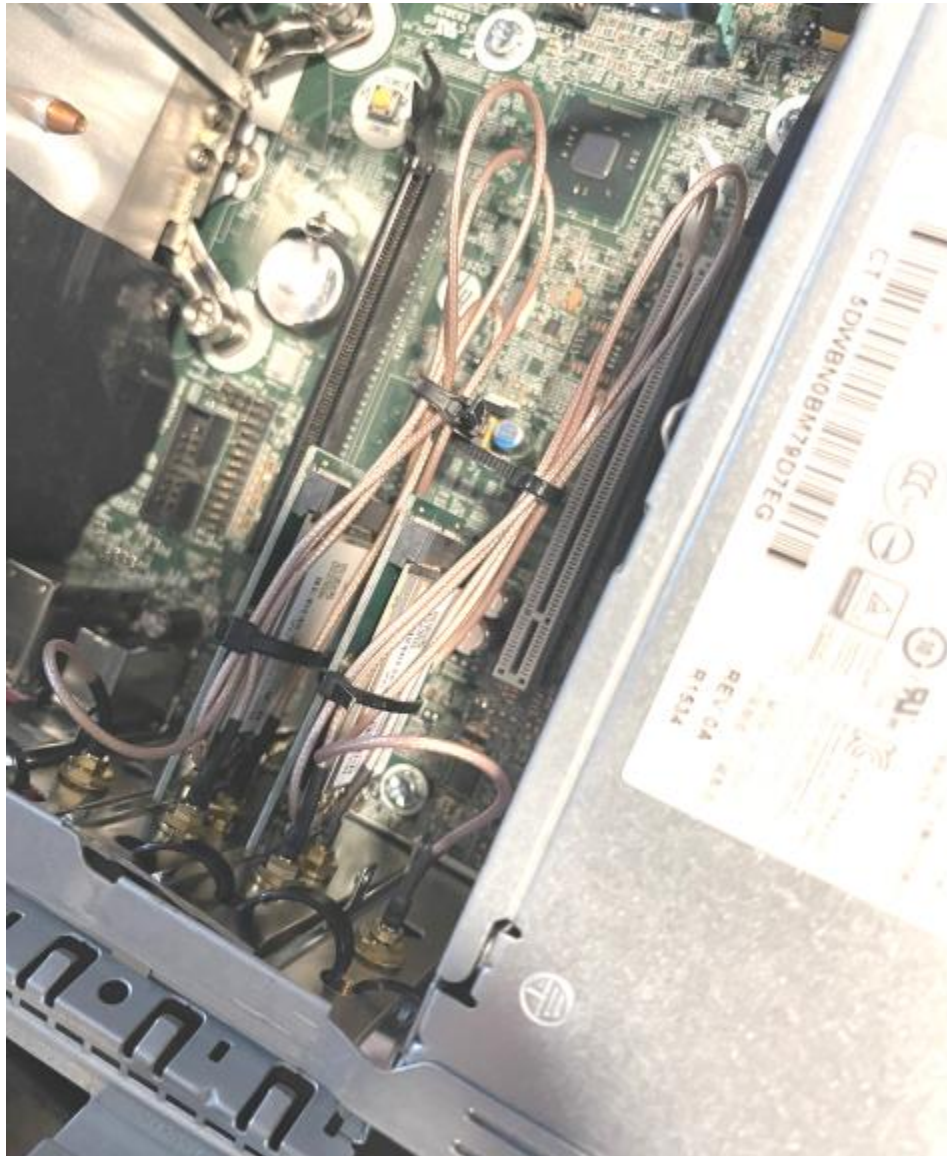
I can thread 2 antennas through the original brackets and 1 antenna through the other brackets.

And that's what I did. Here's the beautiful result.



Things are reasonably steady there.

Here's a top-down view.



Installing Linux

Appliance Philosophy – Not Desktop

My Linux-based access point is basically an appliance—one that's going to be directly providing connectivity. and I want to approach the operating system installation and configuration like an appliance. Meaning:

- The less installed, the better. Less software installed means less to update, less to break, and less to worry about. (Told you that 128GB SSD was overkill)
- We don't need desktop GUI stuff on a box that will basically be out of the way and invisible. We can SSH into this and do whatever we need.

I'm wanting to start with as barebones of a Linux installation as possible, but with the option to easily install anything that's needed.

Why I Chose Debian

Debian is famous for its dedication to FOSS ideals, dedication to stability over new features, dedication to keeping up with security updates, and has been around since 1993. It's not going anywhere. Plus I've been using it in one form or another since 2004.

Plus...

The Debian `netinst` image

Debian's `netinst` install fits our minimalist "appliance philosophy" perfectly.

Nothing will be installed at first except the bare minimum to get a useable system, but there are the literally tens of thousands of Debian packages to choose from and install later if needed (and of course external `.deb`'s can be installed as needed).

The intent of `netinst` is to allow installation over the Internet without having to download all of the CDs or DVDs that make the entire distribution. You are able to select collections of packages, then it downloads and installs straight from the Internet.

Of course you do not have to select any collections for a true, minimal, barebones system – you simply select nothing. Some of the collections are useful, such as "SSH Server" and "Basic system utilities."

Actual Installation Process

Legacy Mode

This hardware is UEFI based with support for "legacy mode." I attempted to use UEFI mode and ran into problems—documentation forthcoming. So a step was to put the firmware into legacy mode.

If interested in the nitty-gritty details..

For all the details about installing Debian, which could be applicable to more than just this guide, please see my separate document "[Installing Debian Linux netinst Image On A Two-128GB-SATA RAID-1](#)" for the steps of the installation process, if interested.

