Before the

FEDERAL COMMUNICATIONS COMMISSION

Washington, DC 20554

In the Matter of	)	
	)	
Amendment of Part 0, 1, 2, 15 and 18 of the	)	ET Docket No. 15-170
Commission's Rules regarding Authorization	)	
Of Radio frequency Equipment	)	
	)	
Request for the Allowance of Optional	)	RM-11673
Electronic Labeling for Wireless Devices	)	

Summary

The rules laid out in ET Docket No. 15-170 should not go into effect as written. They would cause more harm than good and risk a significant overreach of the Commission's authority. Specifically, the rules would limit the ability to upgrade or replace firmware in commercial, off-the-shelf home or small-business routers. This would damage the compliance, security, reliability and functionality of home and business networks. It would also restrict innovation and research into new networking technologies.

We present an alternate proposal that better meets the goals of the FCC, not only ensuring the desired operation of the RF portion of a Wi-Fi router within the mandated parameters, but also assisting in the FCC's broader goals of increasing consumer choice, fostering competition, protecting infrastructure, and increasing resiliency to communication disruptions.

If the Commission does ***not*** intend to prohibit the upgrade or replacement of firmware in Wi-Fi devices, the undersigned would welcome a clear statement of that intent.

Introduction

We recommend the FCC pursue an alternative path to ensuring Radio Frequency (RF) compliance from Wi-Fi equipment. We understand there are significant concerns regarding existing users of the Wi-Fi spectrum, and a desire to avoid uncontrolled change. However, we most strenuously advise against prohibiting changes to firmware of devices containing radio components, and furthermore advise against allowing non-updatable devices into the field. Doing so would block efforts to address serious ongoing problems with the Internet infrastructure. Instead, we propose the FCC mandate best practices for software development, and that the FCC mandate that the Wi-Fi vendors:

- Provide public, full, and maintained source code for review and improvement
- Assure that secure firmware updates are available and under owner control
- Address known security vulnerabilities in source and binary within specific time frames
- Be made aware that noncompliance could result in decertification.

These points are expanded upon below, in the section labeled, “An Alternate Approach”.

Background

Wi-Fi routers are an area of rapid innovation. These devices play an essential role in home and small business networks, and are responsible for the integrity of these networks: a vulnerable router leads to a vulnerable network. Despite the importance of these devices, most home router vendors lack the financial incentive to improve home router functionality, security and performance. This is a result of the current economic climate, where there is no market feedback for vendors with higher-quality products to increase prices or offer ongoing maintenance.

As a result, the academic community, the Free and Open Source Software Community (FOSS), and the Internet Engineering Task Force (IETF) have all joined forces to pick up the slack. For over four years

we have been resolving these problems by releasing standards-conformant router software to the public. Work produced by academics and open source developers in these areas is often adopted by Wi-Fi vendors in subsequent products. This proof-of-concept work simply would not be possible without the ability to update the firmware on Wi-Fi routers.

We believe the ability to replace the firmware of Wi-Fi devices will be in jeopardy following the implementation of the rules proposed in ET Docket No. 15-170. While the proposed rulemaking does address concerns for modular transmitters operating in licensed use, it will also create an array of problems. The rulemaking may protect portions of the radio spectrum against non-compliant consumer routers, but will result in the loss of vital work in the public interest. The rulemaking also risks permanently locking in place buggy and insecure software. Recent examples, most notably the Volkswagen emissions scandal, very clearly shows that having closed systems whose operation cannot be inspected is exactly the opposite of what is needed: Only by having software systems be open, inspectable, and verifiable ***by the owner of the equipment*** can compliance be ensured.

Some FCC guidance to the proposed rulemaking asks router vendors to "describe in detail how the device is protected from 'flashing' and the installation of third-party firmware such as DD-WRT." This is exactly backwards; the options provided by projects like OpenWrt are the only realistic hope of addressing the severe problems with the present network and installed base. Many improvements have already been made and delivered in Internet security, IPv6 capability, code correctness, and network performance via the "WRT" family of open-source router operating systems: OpenWrt, DD-WRT, CeroWrt and derivatives.

Our Concerns with Today's Wi-Fi Routers

The United States has hundreds of millions of commercial off-the-shelf routers in homes, small businesses, and enterprises. These routers, installed over the last 15 years, have all passed FCC certification, but still have many severe software flaws, and as we note below, compliance failures.

Insecure Router Implementations

Most home routers ship with outdated software that is usually four or more years out of date. Too often, the firmware is known to be insecure at FCC certification time. This issue was highlighted by Independent Security Evaluators (ISE), which demonstrated multiple vulnerabilities in common routers, which can be exploited without active user participation. Other security researchers have presented similar results. Recently, in August 2015, a brand-new router was found to have a critical vulnerability that could compromise user data or allow an attacker complete control of the device.

The economics of the present day home router market suggests the security situation is unlikely to change, which makes the ability to install corrected, open firmware essential for the future safety and integrity of the Internet.

Lack of Functionality and missing ongoing support

Most Wi-Fi routers, even the newest ones, do not or only barely support IPv6, with poor implementations of IPv6 leading to problems with interoperability. Few vendors are tracking the developments of IETF working groups, and nearly none are working on retrofitting their previously shipped products. Furthermore, we are not aware of any COTS Wi-Fi routers that correctly implement the Domain Name System Security (DNSSEC) specification, which protects against DNS attacks.

Inability to Verify Proper Operation

The recent Volkswagen case shows that using closed and uninspectable source code can lead to devices (vehicles) operating out of specification. Similarly, the inability to inspect the RF-controlling source code of routers makes it impossible to determine whether they are operating within specifications under all circumstances.

Inability to Correct/Improve Operation

Under the proposed rulemaking, owners of non-compliant devices would be unable to make repairs. This directly conflicts with the principle that it is the owner's responsibility to ensure that a device

meets operation regulations.

Work by the IETF, FOSS members, and the CeroWrt Project

The undersigned are experts in modern software engineering and networking technologies. We are deeply familiar with best practices for software development, network deployment, and Internet security.

The IETF is the group of network professionals who developed the Internet, and whose continuing goal is “to make the Internet work better.” They are deeply knowledgeable about networking best practices, and the design of future capabilities.

The FOSS community is a group of software experts who collaborate to produce software that is generally available at no cost, and without restrictions on its use. Volunteers and paid professionals work on FOSS projects to build useful software.

These groups would not be able to collaborate without access to free and open tools and inexpensive equipment (e.g. commercial, off-the-shelf Wi-Fi routers). With the use of these open source tools, research teams are able to gain new insights into network phenomena and improve network operation for the public. Projects such as OpenWrt, CeroWrt, DD-WRT, Tomato, Gargoyle, and others operate in much the same collaborative fashion.

The CeroWrt Project

Members of the CeroWrt router research project have worked over the past four years to improve network performance under load. Many network connections perform for a single user, but break down when more than one user shares the network connection. This project—made possible only by entirely open and modifiable firmware—had a breakthrough in 2012, speeding up the edge of the Internet often by an order of magnitude or more under load.

The problem is called “Bufferbloat”, and the best known current solution, developed through the work of the CeroWrt project, is an algorithm called “fq_codel”. Bufferbloat causes poor network performance for voice, video-conferencing, gaming, DNS and web traffic. It is easily tested for, and due to legacy home router equipment and slow adoption by router vendors and ISPs, bufferbloat is still at epidemic proportions across the edge of the Internet.

Fq_codel has been shown to work extremely well on wired packet transports (DSL, cable, fiber, ethernet), and can provide substantial improvements to point-to-point wireless links. Standardization efforts for fq_codel are nearly complete within the IETF “aqm” working group, and work is in progress for a successor algorithm, “cake”, which addresses a few edge cases that fq_codel did not.

The work on CeroWrt and fq_codel was instrumental in Apple’s recent decision to enable “Explicit Congestion Notification” (ECN) across their IOS and OSX operating systems, making possible loss-free, low latency, and congestion-controlled network video transfers.

In summary, the CeroWrt team made an important research contribution to solve the Bufferbloat problem, a contribution that would not have been possible without the ability to flash and update custom firmware against many brands of routers.

Making Wi-Fi Fast

The CeroWrt team is now working on the Make-Wifi-Fast project, which will vastly improve Wi-Fi functionality, particularly when multiple stations are in use, using open, unpatented, and standards-compliant algorithms and protocols. This work can be incorporated into new Wi-Fi products, and can be retrofitted into hundreds of millions of non-locked existing products. This is all within the existing capabilities of Wi-Fi, and does not affect regulatory compliance.

Research is also in progress to improve spatial reuse of the RF spectrum, and to dramaticallyreduce power use. All these changes will improve sharing of Wi-Fi spectrum in higher density situations: for example, in apartment buildings and shared industrial settings in which multiple companies’ network

infrastructure radiation spheres overlap. This is directly in line with the FCC's core mandate to prevent destructive interference.

Dave Täht has outlined some of these solutions in talks to the IEEE 802.11 working group and at the BattleMesh conference.

Security

The CeroWrt project's last generation of firmware was considerably more secure than the average home router, and all its innovations were pushed into OpenWrt and other projects. It incorporated multiple advanced hardening features and the latest security standards such as DNSSEC. Our work fully supported IPv6 protocols, and included many measurement and diagnostic tools, the latest fq_codel and "cake" Bufferbloat-fighting algorithms, and the latest IETF "homenet" working group outputs – hnetd and babel.

Best Practices

The latest OpenWrt production version includes all of CeroWrt's innovations and more, independently developed by hundreds of researchers and developers across the globe.The central Linux and OpenWrt software repositories combine those projects together to make a whole. The result is a software platform that supports the next generation of IPv6 Internet as well as the latest, most secure versions of the protocols that run on the Internet, and delivers performance adequate to the speeds produced by the most innovative service providers, such as Comcast, Google Fiber and Verizon FiOS.

Barriers caused by Rulemaking

One of the biggest barriers to a full implementation of the Make Wi-Fi Fast project is the FCC's latest regulatory actions, particularly as makers of new hardware and firmware have understood them.

The most common interpretation of these rules is that they require vendors to deny access to inspect or

modify their code to the FOSS, IETF, and academic research communities, as well as other potential investigators who do not have the purchasing power to request special treatment.

In some cases, vendors have used FCC rules as an excuse to keep their source code private, unmodifiable, and unfixable. As a result, bugs and security vulnerabilities are locked in place, with the potential to pose a significant danger to the national security of the United States. Router firmware is a target for both criminal exploitation and cyber warfare. To mitigate these risks, in-field upgrading and open-source development are both essential features.

Our understanding is that portions of existing and proposed FCC rules would prevent in-field testing and fixes for router firmware, fixes to buggy code throughout the software stack, and improvements to IPv6 and Wi-Fi performance. The rules will also hinder in-field compliance with future FCC regulations.

The proposed rule could prevent anyone other than the original vendor from making modifications. Members of the community have observed several new cases of vendor firmware being locked down where it had not been previously. At least one router vendor has stated that FCC compliance requires firmware activation codes, and at least one chipset vendor has stated that locking down firmware would be “the easiest way to comply”. Together, this evidence strongly suggests that the tendency of vendors to lock down firmware will increase if these new rules are implemented.

The software portion on common Wi-Fi routers that affects the correct functioning of the radio is a tiny fraction of the operating system: a “device driver” and the “co-processor firmware”. This is a small, separable and specialized component; the majority of the router firmware consists of the operating system, user interface, routing and switching code, and other functions unrelated to radio. To lock down the entire firmware causes collateral damage by preventing improvements to these other components.

Prohibiting the owner of a router to replace *any* part of its operating system will have a chilling effect on our ability to implement new algorithms. We are currently limited in our attempts to fix the ath9k

and mt76 devices, and stopped cold on addressing similar problems in the universally closed firmware in 802.11ac devices. Proposed upcoming rules may prevent further work on the project. The proposed rules would also hinder the FCC's own measurement studies attempting to analyze Wi-Fi behavior!

Not even the small and specialized portion of the firmware that *is* related to radio operation should be locked down. It must be able to change and evolve in conjunction with the other code in use in order to address bugs, new requirements, or regulatory compliance issues that have been discovered after shipping.

The FCC ***should not allow*** the development or deployment of equipment that cannot be inspected, as that has led to software that covertly avoids regulatory compliance, as has recently been found to be the case with Volkswagen cars with diesel engines. Source code transparency in this case would have assisted necessary regulation, and can do so for network hardware as well.

In our work we have found and fixed many bugs, significantly mitigating misuses of the Wi-Fi spectrum. For one example among hundreds, we found and fixed a quite egregious infinite retry bug. This bug caused the device driver to become locked sending the same data over and over again for tens of seconds, improperly dominating the channel, and needlessly interfering with other users. Preventing upgrades to this software would force the router to remain out of compliance, the opposite of the FCC's intentions.

As an additional example, the Make-Wi-Fi-Fast project currently has upgrades pending that we cannot further develop or deploy in today's regulatory environment.

While the goal of protecting some radar installations from non-compliant equipment is important, the restrictions this rulemaking would create would have a much broader effect, potentially reaching the majority of users of the Internet, not only within the United States but also abroad. We believe that the FCC should instead focus on ways to improve how Wi-Fi and software defined radio (SDR) are developed and used.

Software Engineering Best Practices

Software and hardware development are iterative processes. No hardware, software or firmware is perfect on first release. In addition, all software and hardware must be able to evolve. No one-time certification is able to meet regulatory compliance, while still addressing new bugs and meeting the ever-changing needs and threats on the Internet.

Bugs and functional limitations in more conventional server and client operating systems (OSes) are routinely corrected using a process of transparent code access, frequent updates and peer review.

Like the other mainstream OS projects mentioned, the source code to the OpenWrt project is under full change control, with the author of each patch publicly identified.

We recommend that this same degree of professional source code management and change control be applied to all products submitted to the FCC, and to other U.S. regulatory bodies such as the FTC and EPA. We encourage its application not only to the general purpose networking and operating system software with which we are primarily concerned, but also to the radio device drivers and on-board firmware with which you have concerns.

For example, OpenWrt contains a built-in configuration data to ensure that radios are used in compliance with the local laws and regulations throughout the world. The Linux community has long made available a secured, signed and regularly updated worldwide database of regulatory power and channel constraints for all devices that use Wi-Fi. This database, and code to access it, is published openly online using a distributed change-control system called Git, and is fully available to anyone with an Internet connection. This is presently the best mechanism for ensuring that devices—once configured properly for their locality—are compliant.

In combination with unambiguous locality configuration, this database can be used to ensure that a

device is compliant; noncompliance would only be possible by willful negligence of the owner. Mandating the use of such an open database is well within the scope of what the FCC can demand of vendors and manufacturers of devices operating in this spectrum. Indeed, regular FCC review of the database and its contents would be a better way to ensure that all devices are in compliance than piecemeal certification of each individual device.

Don’t Prohibit Third-Party Software

Most Wi-Fi router software today is shipped with ancient code, rife with security holes and bugs. Yet the recent guidance provided to vendors by the FCC suggests that vendors of new wireless hardware describe “how the device is protected from ‘flashing’ and the installation of third-party firmware such as DD-WRT”. We believe this constitutes an ill-advised attempt to regulate the means instead of the end results, which is not in the public interest. Third-party software is demonstrably better in many regards to most stock vendor firmware, as demonstrated by the fact that several vendors have incorporated it into their products. Examples of this include:

1. DD-WRT (an OpenWrt relative) is shipped as a factory default by at least one vendor (Buffalo). OpenWrt is also shipped as the default firmware in many devices, and is used by Qualcomm Atheros as the default operating system supplied to their original design vendors (ODMs). Other big-name vendors in the Wi-Fi market known to use derivatives of OpenWrt are Meraki and Ubiquiti; no doubt many others exist. Many other firmware builds are based on open source software distributions such as Debian (Ubiquiti), or Buildroot (Google Fiber).

2. The onboard firmware for at least one 802.11ac chip (QCA ath10k) is based on the BSD driver stack. Despite it having the open-source BSD license, concerns about the FCC requiring lockdowns have been a huge factor in delaying the open public release of that firmware. This in turn prevents it from receiving needed fixes, and the firmware is still, after 2+ years of shipping, not ETSI CCA compliant. Concerns over the FCC's rules have made it impossible for those with source licenses to that firmware to collaborate – even including the original authors of the BSD-

based code.

An Alternate Approach

In place of these regulations, we suggest that the Commission adopt rules to foster innovation and improve security and usage of the Wi-Fi spectrum for everybody.

Specifically, we advocate that rather than denying users the ability to make any changes to the router whatsoever, router vendors be required to open access to their code (especially code that controls RF parameters) to describe and document the safe operating bounds for the software defined radios within the Wi-Fi router.

In this alternative approach, the FCC could mandate that:

1. Any vendor of SDR, wireless, or Wi-Fi radio must make public the full and maintained source code for the device driver and radio firmware in order to maintain FCC compliance. The source code should be in a buildable, change controlled source code repository on the Internet, available for review and improvement by all.
2. The vendor must assure that secure update of firmware be working at shipment, and that update streams be under ultimate control of the owner of the equipment. Problems with compliance can then be fixed going forward by the person legally responsible for the router being in compliance.
3. The vendor must supply a continuous stream of source and binary updates that must respond to regulatory transgressions and Common Vulnerability and Exposure reports (CVEs) within 45 days of disclosure, for the warranted lifetime of the product, the business lifetime of the vendor, or until five years after the last customer shipment, whichever is longer.
4. Failure to comply with these regulations should result in FCC decertification of the existing product and, in severe cases, bar new products from that vendor from being considered for

certification.

5. Additionally, we ask the FCC to review and rescind any rules for anything that conflict with open source best practices, produce unmaintainable hardware, or cause vendors to believe they must only ship undocumented “binary blobs” of compiled code or use lockdown mechanisms that forbid user patching. This is an ongoing problem for the Internet community committed to best practice change control and error correction on safety-critical systems.

This path has the following advantages:

- **Inspectability -** Skilled developers can verify the correctness of software drivers that are now hidden in binary “blobs”.
- **Opportunity for innovation -** Many experiments can be performed to make the network “work better” without affecting compliance.
- **Improved spectrum utilization -** A number of techniques to improve the use of Wi-Fi bands remain theoretical possibilities. Field trials with these proposed algorithms could prove (or disprove) their utility, and advance the science of networking.
- **Fulfillment of legal (GPL) obligations -** Allowing router vendors to publish their RF-controlling source code in compliance with the license under which they obtained it will free them from the legal risk of being forced to cease shipping code for which they no longer have a license.

Requiring all manufacturers of Wi-Fi devices to make their source code publicly available and regularly maintained, levels the playing field as no one can behave badly. The recent Volkswagen scandal with uninspected computer code that cheated emissions testing demonstrates that this is a real concern.

Conclusion

These current and pending FCC regulations for the design and use of the public's Wi-Fi devices act as a barrier to the process of making routers faster, safer, and better in general.

The FCC should step back, and prepare rules to enhance the security, reliability and functionality of the routers that operate home and business networks. These rules should increase visibility into the source code that operates these routers, and encourage best software practices to create a better future for billions of Wi-Fi devices already deployed, and the billions to come, as well as a freer, faster, and safer Internet.

As individuals involved in making Wi-Fi better, we want both to comply fully with the law and to move the state of the art forward. All our measurements show that the state of Wi-Fi today is dismal – and we know why. If the FCC permits us to soldier on, we promise to make Wi-Fi, and the Internet, a whole lot better.

Sincerely,

Dave Täht US Citizen Co-founder, bufferbloat.net	Vint Cerf US Citizen Co-Inventor of the Internet