CHAPTER 5

Free Software and Software-defined Radio: An Overview of New FCC Rules

Matt Norwood

A. ABSTRACT

The US Federal Communication Commission (FCC) recently promulgated rules governing the use of free and open source software (FOSS) in software-defined radio devices. While the rules encourage the use of FOSS for some applications, they also express reservations about its employment in certain security-critical systems, taking the position that secrecy results in better security system design. This paper examines the immediate implications of this mixed ruling for hardware manufacturers and independent software developers, as well as its likely meaning for the long-term relationship between the FCC and developers of new technologies. The paper also makes a normative argument that regulators should re-examine their reliance on secrecy as a method for ensuring the design of secure systems.

B. INTRODUCTION

The free software movement has brought about a paradigm shift in software production. The retail software model — where initial production, incremental development, and ongoing support of software was centralized with a single vendor — has given way to decentralization of all these activities, carried out by entities motivated by market and non-market forces. This shift presents new challenges for regulators, who are forced to rethink old incentive models to account for this decentralization. Regulators must also examine the decoupling of activities once considered unitary because production, distribution, modification, end-user support, and deployment

of software may each be performed separately by hardware manufacturers, software vendors, independent software developers, and users.

One domain of government regulation implicated in this shift from proprietary software to FOSS is the regulation of radio-spectrum use. As the wireless industry steps up production of radio devices whose operating parameters are controlled by computer software, regulators find themselves re-examining their assumptions about the most efficient points of accountability in the wireless device industry, and about how the industry distributes the labor of software production and support. In adjusting or rewriting rules to account for the new industrial landscape, there is a real danger of leaving certain assumptions unexamined even after they have lost their usefulness.

In the US, the FCC has recently made a positive step toward coming to terms with the new realities of the wireless industry, but it has failed to rethink some of its old assumptions that no longer hold, potentially inhibiting the rate of innovation among developers of new devices. This paper examines the positive changes effected by the FCC in lifting some of the regulatory barriers to wireless innovation, but also examines the ways in which the new regulations fall short of their stated purpose by hindering development of new technologies without any reciprocal gain in public welfare.

C. BACKGROUND

1) FCC Software-defined Radio Rules

The FCC regulates radio and wire communications in the US. Among its regulatory duties is the protection of radio broadcasts from harmful interference caused by consumer products. To this end, it requires manufacturers of electronic devices to certify their products at FCC-approved testing labs to ensure that they comply with rules limiting the power and patterns of radio transmissions at various frequencies. This certification process has traditionally examined only the behaviour of the hardware itself, as the broadcast characteristics of most radio devices on the market were limited only by the device's antenna and other hardware components. In the late 1980s and 1990s, however, a new class of devices emerged whose radio characteristics were limited in part by computer software. These devices showed promise in a number of respects: they were much simpler and cheaper to manufacture than devices with complex hard-wired logic, and they could be reconfigured for different behaviours through a simple alteration of the operating software. This also meant that one device could be marketed in

several regions under different regulatory regimes, and software could be used to bring the device into compliance with local regulations; this strategy presented significant efficiencies over the manufacture of different chipsets for sale in different countries.

The rise of these software-configurable radio devices presented the FCC with a regulatory challenge. Reconfiguration of software was one of the devices' main advantages, but under the existing certification rules any change to a device — including changes to its software — would require lengthy, expensive recertification with FCC labs. The FCC made efforts to accommodate the new devices under their existing certification rules, but it also put out a notice of proposed rule-making to streamline and standardize the treatment of software-configurable devices during certification.[1]

On 11 March 2005, the FCC released a set of rules outlining an alternative method for certification of devices whose radio frequency and power characteristics can be modified by software, designating such devices software defined radio (SDR) devices.[2] These rules allow manufacturers who have certified under the new process to update the software on the devices without recertifying the devices with the FCC.

The rules require any manufacturer certifying a device under the new process to take steps to prevent "unauthorized" changes to the software on the device that might alter its radio frequency and power parameters in a way that takes it out of compliance with the regulations known as FCC Part 15 regulations.[3] The specific technology implemented to accomplish this task is left to the manufacturers seeking certification, although the FCC suggests several possible mechanisms that can serve as such "security measures."[4]

In response to a petition from Cisco Systems, Inc., the FCC issued a *Memorandum Opinion and Order* on 25 April 2007, making two clarifications to the rules.[5] First, the FCC clarified the scope of the rules to require

---

1 US, Federal Communications Commission, *Notice of Proposed Rule Making and Order* (Doc. No. 03-322) (2003), online: http://hraunfoss.fcc.gov/edocs_public/attachmatch/FCC-03-322A1.pdf.

2 US, Federal Communications Commission, *Report and Order* (Doc. No. 05-57, 20 FCC Rcd 5486) (2005), online: http://hraunfoss.fcc.gov/edocs_public/attachmatch/FCC-05-57A1.pdf [*Report and Order*].

3 *Ibid*. at 5488.

4 *Ibid*. at 5509.

5 US, Federal Communications Commission, *Memorandum Opinion and Order* (Doc. No. 07-66) (2007), online: http://fjallfoss.fcc.gov/edocs_public/attachmatch/FCC-07-66A1.pdf [*Memorandum Opinion and Order*].

certification under the new process of any device that uses software to comply with the Part 15 regulations, if such software is "designed or expected to be modified by a party other than the manufacturer."[6] Second, the FCC stated a position regarding the use of FOSS on SDR devices. The FCC acknowledged the use of FOSS by device manufacturers and noted some of the advantages of FOSS for the industry. However, citing concerns regarding publishing information relating to security measures, the FCC stated that an SDR device that uses FOSS to build the "security measures" protecting the software against modification would face a "high burden" during the certification process "to demonstrate that it is sufficiently secure."[7]

2) Linux-based Wireless Devices

The FCC notes in its *Memorandum Opinion and Order* that the industry now commonly uses both the kernel named Linux and complete FOSS operating systems in wireless radio devices under its regulatory control. For example, many 802.11 wireless network routers use FOSS to provide a fully functional system, including network address translation (NAT), network firewalling, intranet web servers, and other network management features. Such functionality, which if licensed as individual proprietary components can be prohibitively expensive, is readily available with any FOSS operating system. Many manufacturers have therefore chosen to configure their devices with these FOSS systems rather than proprietary alternatives. In such a configuration, almost none of the FOSS on the device interacts directly with the FCC-regulated radio hardware. Typically, only small pieces of code, either running as a Linux kernel module and/or as a wholly independent firmware on the chip itself, actually constitute an SDR component whose modification would need to be limited by the "security measures" described in the FCC's rules. This paper focuses primarily on the impact of the rules on these small components, but the reader should not lose sight of how much software on the standard FOSS-based, FCC-regulated consumer devices remains far from the regulatory control of the FCC.

3) Industrial Adoption of FOSS

Free and open source software has been widely adopted by industries where its advantages of stability, standardization, and community support and de-

---

6 *Ibid.* at 3; *Report and Order*, above note 2 at 5504; 47 C.F.R. § 2.1.

7 *Memorandum Opinion and Order*, above note 5 at 4.

velopment outweigh the advantages gained by exclusive control and ownership of a proprietary software product. Although many FOSS projects are primarily developed by volunteer programmers, the support and development of FOSS is increasingly carried out by private industry, with companies and consultants sharing the development and support work among themselves and with the volunteer community. This decentralization of contribution to the success of FOSS projects distinguishes it from proprietary software, which is usually developed and supported by a single entity. FOSS is also distinguished by the common availability of its source code, where proprietary software vendors tend to closely guard the secrecy of their code. This last distinction is the salient characteristic of FOSS seized on by the FCC's *Memorandum Opinion and Order* as presenting a problem for the design of secure systems: in its rules, the FCC discourages manufacturers from making their security software public if doing so would increase the likelihood of it being circumvented.

D. LEGAL CONSEQUENCES FOR DEVELOPERS AND MANUFACTURERS

After an extended period of uncertainty among software developers and wireless hardware manufacturers alike as to the FCC's stance on FOSS in wireless devices, the new rules provide substantial clarity and leeway to both hardware and software developers using FOSS. The rules eliminate the uncertainty in the industry as to FCC's stance on FOSS by explicitly addressing the subject. They unequivocally permit the use of FOSS for most applications used in wireless devices. And the only prohibition they create for FOSS is qualified in two respects: it applies only when FOSS can be shown to make a system less secure, and even then it is not an absolute bar on FOSS but only a "high burden" to prove the device's security.

The clarity and permissiveness of the rules allow wireless software and hardware developers to operate in a much more certain legal environment than has been available for several years. The boundaries of the activities permitted and barred by the rules can now be demarcated with a reasonable degree of confidence, as can the boundaries between the parties subject to the rules and those unaffected by them. Specifically, FCC's lack of jurisdiction over independent software developers and the rules' explicit applicability only to device manufacturers seeking certification mean that independent software developers with no ties to radio hardware manufacturers are not the subject of these regulations, and their activities are not affected by the SDR device certification rules. Hardware manufacturers,

meanwhile, are still afforded significant leeway in the use of FOSS in their devices, with most applications being expressly cleared for implementation through FOSS. Only a completely FOSS-driven device would be implicated by the FCC's cautionary statement, and even then the FCC has presented guidance on how their resistance to FOSS can be overcome by a manufacturer seeking certification.

1) FCC Jurisdictional Limits

It is unlikely that the FCC could promulgate rules regulating the activities of software developers unless those developers were engaged in the manufacture or distribution of devices capable of causing radio interference. The FCC is a statutory body with a grant of jurisdiction strictly defined by Congress.[8] First, it has the power to regulate "interstate and foreign communication by wire or radio."[9] This primary jurisdictional grant gives it wide latitude to make and enforce rules for broadcasters and telecommunication carriers. Second, it has ancillary jurisdiction to make rules and regulations necessary for carrying out its primary responsibilities.[10] This ancillary jurisdiction includes the power to make "reasonable regulations . . . governing the interference potential of devices which in their operation are capable of emitting radio frequency energy."[11] It is much more narrowly constrained than the FCC's primary jurisdiction over broadcasters and carriers because courts, fearful that the FCC's powers would become "unbounded" if it were allowed to make rules governing any activity or party that might have an effect on radio or wire transmissions, have precluded such a reading of the FCC's jurisdictional grant.[12] Thus, while the FCC's ancillary jurisdiction reasonably extends to regulating the marketing and sale of devices that create active radio interference, it does not extend to such activities as the construction of buildings that might interfere with radio signals.[13]

Similarly, the FCC's ancillary jurisdiction cannot reasonably extend to the development of software by parties uninvolved in the marketing or sale of radio devices. Congress did not contemplate the FCC as a gen-

8 *Michigan v. EPA*, 268 F.3d 1075 (D.C. Cir. 2001); *Louisiana Public Service Commission v. FCC*, 476 U.S. 355 (1986).

9 47 U.S.C. § 152(a).

10 47 U.S.C. § 154(i).

11 47 U.S.C. § 302(a).

12 *FCC v. Midwest Video Corp.*, 440 U.S. 689 (1979).

13 *Illinois Citizens Committee for Broadcasting v. FCC*, 467 F.2d 1397 (7th Cir. 1972) [*Illinois*].

eric technology-regulatory agency, and courts have repeatedly limited the FCC's reach when it attempted to make rules outside of the realm of the distribution or marketing of equipment capable of wire or radio signal transmission.[14] Attempts by the FCC to regulate the activities of software developers not engaged in the importation or marketing of radio devices and not employed by telecommunication carriers are likely to be met with similar judicial restriction.

2) Scope of the Certification Rules

Even if the FCC did have the power to regulate independent software development, it has promulgated no rules governing such activity. Its new rules related to SDR are addressed to "manufacturers" of radio "equipment," modifying the rules for certification of such "hardware-based device[s]" prior to their "marketing or importation."[15] No other parties or activities are affected by the regulations. Thus, unless a given entity engages in the marketing or importation of hardware-based radio equipment, it is unaffected by these regulations.

3) "Equipment" vs. Software

The FCC has promulgated the SDR rules as a modification to its existing regulations governing the certification for marketing and sale of devices that may interfere with radio transmissions.[16] These rules limit the ability to "manufacture, import, sell, offer for sale, or ship devices or home electronic equipment and systems, or use devices, which fail to comply with regulations promulgated pursuant to this section."[17]

Since software is a representation of a mathematical algorithm, it is not a "device," "home electronic equipment," or a "home electronic . . . system."[18]

---

14 *American Library Association v. FCC*, 406 F.3d 689 (D.C. Cir. 2005) (the FCC has no power to regulate television components unrelated to signal reception); *Illinois*, above note 13 (the FCC lacks jurisdiction over objects that interfere with television transmissions).

15 *Report and Order*, above note 2 at 5505.

16 *Ibid*. at 5487. Codified at 47 C.F.R. § 0.457, § 2.1, § 2.932, § 2.944, § 2.1033, § 2.1043, and §15.202.

17 47 U.S.C. § 302.

18 The only place where the FCC's rules or its enabling statute include software within the definition of any of these terms is in the context of regulating telecommunications carriers: the definition of "Telecommunications Equipment" in 47 U.S.C.

Further, there is no precedent for applying the device certification rules to software except as installed as a component of a specific hardware device. Indeed, the FCC has explicitly limited the certification requirements to "hardware-based device[s]."[19] Both of these facts make it clear that the FCC rules do not apply to software by itself, but only to hardware-based devices.

4) FCC Recognition of FOSS

The FCC's *Memorandum Opinion and Order*, clarifying its SDR certification rules, acknowledges the use of "open source software" by SDR device manufacturers and notes the advantages FOSS provides to the industry. It declines to forbid or restrict the use of FOSS on SDR devices. However, it discourages the use of FOSS in the "hardware and software security elements" of SDR devices by stating that systems "*wholly* dependent on open source elements" would have a "high burden" to demonstrate their security during the certification process.[20]

Nowhere do the rules restrict any party other than a manufacturer seeking certification for a device. Even in that case, they do not restrict the manufacturer's activities directly, but simply warn that the certification will be less likely to be granted if the manufacturer relies "*wholly*" on FOSS in building the device. The rules acknowledge the activities of the FOSS community in developing radio device software and, in keeping with the FCC's jurisdictional limitations, decline to formulate rules governing any participant in this activity except manufacturers seeking certification for their devices.

This reluctance on the part of the FCC to create regulations for technological development — beyond regulating the actual marketing of de-

---

§ 153(45) states that the term "includes software integral to such equipment." This definition explicitly excludes "customer premises equipment." Thus, it applies only to equipment used by carriers, on the premises of the carriers, to provide telecommunications services. The careful limitation of this definition to carrier equipment is consistent with the FCC's broad jurisdiction over carriers' business practices, allowing the FCC to regulate software in this specific domain where Congress is unwilling to grant jurisdiction over software developed or used by other parties not under the FCC's direct jurisdiction. In addition, the explicit inclusion of "software integral to such equipment" in this definition implicitly excludes software from other uses of the term "equipment" in the statute. If Congress had intended the general term "equipment" to include software, it would have defined it accordingly, as it did in the limited case of "Telecommunications Equipment."

19 *Report and Order*, above note 2 at 5505.

20 *Memorandum Opinion and Order*, above note 5 at 4 [emphasis added].

vices — is consistent with the FCC's position on experimental or specialized equipment. Such equipment is exempt from the certification requirements if it is not marketed to the public and is only used under controlled conditions.[21] It is allowed to be developed and distributed as long as it is used for such limited purposes. Thus, even entities that install and run software (FOSS or otherwise) on radio hardware devices for the purposes of testing, research, or development are exempt from the new SDR certification rules as long as they abide by the other applicable FCC rules.

E. POLICY ANALYSIS

The FCC's rules on SDR device certification present two positive developments for FOSS deployment in the wireless technology space. First, the FCC permits the use of FOSS in all but a very narrowly constrained subsystem on a specific type of device. Second, the FCC has provided much greater regulatory clarity than has existed on this issue since the commencement of the SDR rule-making process in 2003. This freedom to deploy FOSS and the clarity of the FCC's position allows hardware manufacturers and FOSS developers to openly collaborate on most parts of SDR device design, and it frees software developers from concerns that they might have to deal with legal issues related to the FCC.

Unfortunately, the FCC has not gone as far as it could in embracing and encouraging the use of FOSS by SDR device manufacturers. Its rules express reservations about the security of FOSS-based systems based on the notion that "making information on security measures publicly available could assist parties in determining ways to defeat them."[22] However, there is broad consensus among software security experts that public disclosure of security design actually results in more secure systems, especially when the system is subject to repeated, low-cost attacks by attackers who can share information freely among themselves.[23] Disclosure of security designs for such devices tends to benefit designers more than attackers, as the designers can learn from each other and improve their designs accordingly, offset-

21 47 C.F.R. § 2.803.

22 *Memorandum Opinion and Order*, above note 5 at 4.

23 Peter P. Swire, "A Model for When Disclosure Helps Security: What is Different about Computer and Network Security?" (2004) 2 J. on Telecomm. & High Tech. L. 163. Swire makes a technical, economic, and regulatory argument for different disclosure models maximizing security based on the nature of the system being designed. He finds that disclosure tends to result in higher security for systems like SDR consumer products that are exposed to multiple attackers who can share information with each other.

ting the advantages the attackers already have in information-gathering and sharing.[24] SDR devices used by consumers fit this description: the software on a cell phone or 802.11 card is exposed to virtually unlimited attack by anyone who purchases the device, and anything learned from such an attack can be communicated at low cost to fellow attackers. Design details disclosed by manufacturers, on the other hand, have tremendous benefits for other manufacturers' ability to spot security flaws, but they provide little benefit to attackers. These arguments are articulated in greater depth in a Petition for Reconsideration submitted to the FCC in June 2007 by the SDR Forum, a wireless industry consortium with a vested interest in the SDR rule-making process.[25] The SDR Forum's petition argues for the right of manufacturers to collaborate openly on the design of security systems. It notes the advantages for security design gained by open standards and public collaboration, condemning "security through obscurity" as an unworkable and self-defeating strategy for designing robust, secure systems.[26] It also points out the ambiguity of the FCC's language regulating disclosure of security design details: while the rules prohibit disclosure "if doing so would increase the risk" of circumvention, they do not make clear who determines whether this risk is increased, decreased, or unaffected by disclosure.[27]

On a positive note, the FCC does signal in its *Memorandum Opinion and Order* that it is open to persuasion on the subject of FOSS-based security systems. It states:

> manufacturers should not intentionally make the distinctive elements that implement that manufacturer's particular security measures in a software defined radio public, if doing so would increase the risk that these security measures could be defeated or otherwise circumvented to allow operation of the radio in a manner that violates the FCC's rules.[28]

This qualification suggests that the FCC is not committed to this proposition, as does the doubly qualified message that only devices "wholly" (not

---

24 Bruce Schneier, *Beyond Fear: Thinking Sensibly About Security in an Uncertain World* (New York: Copernicus Books, 2003) at 119–28. Schneier deconstructs the myth that "security through obscurity" is an effective strategy, noting the brittleness of systems designed under that approach.

25 SDR Forum, *Petition for Reconsideration* (Doc. No. SDRF-07-A-0012-V0.0.0) (2007), online: www.sdrforum.org/pages/documentLibrary/documents/SDRF-07-A-0012-V0_0_0_Response_to_MOO.pdf .

26 *Ibid*. at 3.

27 *Ibid*., quoting *Memorandum Opinion and Order*, above note 5 at 4.

28 *Memorandum Opinion and Order*, above note 5 at 4 [emphasis added].

mostly, or partly) based on FOSS would face a "high" (not insurmountable) burden during certification. The FCC's refusal to categorically prohibit FOSS in any part of an SDR system reads as an invitation to the wireless device industry and the FOSS community to demonstrate the suitability of FOSS for the design of all software subsystems of SDR devices.

F. CONCLUSION

The SDR rules promulgated by the FCC represent a positive development for FOSS developers working in the wireless space. The rules allow FOSS developers not affiliated with device manufacturers to continue work on their software without restriction. They allow SDR manufacturers to employ FOSS for most of the functionality of their devices, and leave open the possibility that a device using a purely FOSS-based software platform could also pass FCC certification if it managed to demonstrate the soundness of its security strategy. The rules should spur FOSS developers and hardware manufacturers to collaborate on design strategies that maximize the efficiency, robustness, and freedom inherent in the FOSS development process, while ensuring that manufacturers satisfy the FCC's security mandate.

On the other hand, the SDR rules' overly conservative position on secrecy suggests that the FCC's regulatory decisions are being unduly influenced by the proprietary software production model that is currently losing market share to newer, more open and collaborative models. It also suggests that the FCC is willing to mandate specific technology choices for the industries it regulates, potentially hampering innovation as new technologies are not allowed to develop. Neither of these trends is positive, but each represents an opportunity for dialogue between the FCC, the wireless device industry, and the FOSS community, on how best to foster innovation while meeting the FCC's regulatory goals.