Free Software Foundation recommendations for free operating system distributions considering Secure Boot

John Sullivan
Executive Director

June 30, 2012

1 Introduction

We have been working hard the last several months to stop Restricted Boot, a major threat to user freedom, free software ideals, and free software adoption. Under the guise of security, a computer afflicted with Restricted Boot refuses to boot any operating systems other than the ones the computer distributor has approved in advance. Restricted Boot takes control of the computer away from the user and puts it in the hands of someone else.

To respect user freedom and truly protect user security, computer makers must either provide users a way of disabling such boot restrictions, or provide a sure-fire way that allows the computer user to install a free software operating system of her choice.

Distributors of restricted systems usually appeal to security concerns. They claim that if unapproved software can be used on the machines they sell, malware will run amok. By only allowing software they approve to run, they can protect us.

This claim ignores the fact that *we* need protection from *them*. We don't want a machine that only runs software approved by *them* – our computers should always run only software approved by *us*. We may choose to trust someone else to help us make those approval decisions, but we should never be locked into that relationship by force of technological restriction or law. Software that enforces such restrictions *is* malware. Companies like Microsoft that push these restrictions also have a terrible track record when it comes to security, which makes their platitudes about restricting us for our own good both hollow and deceitful.

The GNU General Public License (GPLv3) shields our freedom against such restrictions. When you buy or rent a computer containing GPLv3-covered software, the license protects your freedom to use modified versions of that software in your computer. GPLv2 always required that users be able to do this, but one of the improvements in GPLv3 ensures that the freedoms all GPL versions are meant to provide can't be taken away by hardware that refuses to run modified software.

When it comes to security measures governing a computer's boot process, GPLv3's terms lead to one simple requirement: Provide clear instructions and functionality for users to disable or fully modify any boot restrictions, so that they will be able to install and run a modified version of any GPLv3-covered software on the system.

Secure Boot is one such measure, defined by a UEFI standard, but discus-

sion about it has primarily revolved around the rules established by Microsoft in its Windows 8 Logo program. These rules say what computer distributors have to do in order to have their systems be Microsoft-approved. Part of this includes implementing Secure Boot in specific ways.

In order to comply with Microsoft's rules as currently published, distributors of x86 computers will have to provide users both the option to customize Secure Boot by using their own security keys, and the option to disable it completely.

Secure Boot, done right, embodies the free software view of security, because it puts users – whether individuals, government agencies, or organizations – in control of their machines. Our thought experiment to demonstrate this is simple: Microsoft may be worried about malware written to take over Windows machines, but we view Windows itself as malware and want to keep *it* away from our machines. Does Secure Boot enable us to keep Windows from booting on a machine? It does: We can remove Microsoft's key from the boot firmware, and add our own key or other keys belonging to free software developers whose software we wish to trust.

2 So what's the problem?

In theory, there should be no problem. In practice, the situation is more complicated. As currently proposed, Secure Boot impedes free software adoption. It is already bad enough that nearly all computers sold come with Microsoft Windows pre-installed. In order to convince users to try free software, we must convince them to remove the operating system that came on their computers (or to divide their hard drives and make room for a new system, perceptually risking their data in the process).

With Secure Boot, new free software users must take an additional step to install free software operating systems. Because these operating systems do not have keys stored in every computer's firmware by default like Microsoft does, users will have to disable Secure Boot before booting the new system's installer. Proprietary software companies may present this requirement under the guise of "disable security on your computer," which will mislead new users into thinking free software is insecure.

Without a doubt, this is an obstacle we don't need right now, and it is highly questionable that the security gains realized from Secure Boot outweigh the difficulties it will cause in practice for users trying to *actually*

provide for their own security by escaping Microsoft Windows.

It's also a problem because the Windows 8 Logo program currently *mandates* Restricted Boot on all ARM systems, which includes popular computer types like tablets and phones. It says that users must not be able to disable the boot restrictions or use their own signing keys. In addition to being unacceptable in its own right, this requirement was a reversal from Microsoft's initial public position, which claimed that the Windows 8 program would not block other operating systems from being installed. With this deception, Microsoft has demonstrated that they can't be trusted. While we are interpreting their current guidelines, we must keep in mind that they could change their mind again in the future and expand the ARM restrictions to more kinds of systems.

The best way out of all of this (other than having all computers come pre-installed with free software) would be for free software operating systems to *also* be installable by default on any computer, without needing to disable Secure Boot. In the last few weeks, we've seen two major GNU/Linux distributions, Fedora and Ubuntu, sketch out two different paths in an attempt to achieve this goal.

3 Fedora's approach

Fedora's default approach to official distribution has the project joining a Microsoft and Verisign developer program which will give them a key that can then be used to sign a "shim" bootloader. This shim loads GRUB 2, the GPLv3-covered GNU program whose job it is to boot the operating system kernel, in this case Linux. Because Fedora's key will be "vouched" by Microsoft, it will be recognized by the firmware on the vast majority of desktops and laptops available.

Fedora also suggests that others use this option for unofficial distribution of modified Fedora software, or any other free software operating system. Anyone can pay $99 and get their own Microsoft-backed key which they can then use to sign the software they wish to run and/or share.

Fedora is not *requiring* users to join the Microsoft program. Users can also create and use their own keys, with a little more work. The Microsoft program is the route they chose for official Fedora distributions, but they will also provide utilities and support for users wishing to work with their own self-generated keys.

There is much to like about Fedora's thinking, as explained by Matthew Garrett. Their process of deliberation evinced concern for user freedom; it's clear that the Fedora team sought a solution that would work not just for their own GNU/Linux distribution, but for as many free software users and distributions as possible. Their discussion was also mindful of the desirability of empowering users to sign and run their own modified software without being treated as second-class citizens. Unsurprisingly, with those concerns guiding their thinking, they have ended on a proposal which as described is compliant with GPLv3.

Unfortunately, while it is compliant with the license of GRUB 2 and any other GPLv3-covered software, we see two serious problems with the Microsoft program approach.

1. Users wishing to run in a Secure Boot environment will have to trust Microsoft in order to boot official Fedora. The Secure Boot signing format currently allows only one signature on a binary – so Fedora's shim bootloader can be signed only by the Microsoft-vouched key. If a user removes Microsoft's key, official Fedora will no longer boot, as long as Secure Boot is on.

2. We reject the recommendation that others join the Microsoft developer program. In addition to the $99 expense being a barrier for many people around the world, the process for joining this program is objectionable. A nonexhaustive list of the problems includes: restrictive terms in multiple of the half-dozen contracts that must be signed, a forced commitment "to receive targeted advertisements and periodic member email messages from Microsoft," and a requirement to provide notarized proof of government-issued identification and a credit card.

These are not acceptable conditions for modifying or using your operating system. For the time being, we should instead rely on the approach Fedora will support for unofficial distribution – providing tools and materials for users who want to install and use their own keys.

Software signed with self-generated keys has the downside of not working on the majority of computers right off the shelf, without the user taking some extra steps. We acknowledge that this is an issue, but in addition to insisting on (and contributing to) documentation to make the necessary process easy to follow, we will strive to solve this problem through political action against

manufacturers and proprietary software companies who impede free software adoption. Encouraging free software distributors and users to trust Microsoft or any other proprietary software company as a precondition to exercising their freedoms is simply not an acceptable solution.

4 Ubuntu's approach

Ubuntu has also announced a plan which is further described in a message to the Ubuntu developers' mailing list. Their plan addresses software distributed through three different channels:

1. Machines sold as "Ubuntu Certified," preinstalled with Ubuntu, will have an Ubuntu-specific key, generated by Canonical, in their firmware. Additionally, they will be required by the certification guidelines to have the Microsoft key installed.

2. Ubuntu CDs, distributed separately from hardware, will also depend on the presence of Microsoft's key in the machine's firmware to boot, when Secure Boot is active.

3. Ubuntu bootloader images distributed online from the official Ubuntu archive will be signed by Ubuntu's own key.

In the first two situations, because of the requirement to have the Microsoft key, their approach has the same issue as Fedora's official method. Users have to trust Microsoft in order to boot official Ubuntu CDs. Their certification program amplifies this problem, because it means no one can sell certified Ubuntu machines without trusting Microsoft.

As with Fedora, on a system with Secure Boot properly implemented, Ubuntu users will be able to add their own keys, or Ubuntu's key.

Our main concern with the Ubuntu plan is that because they are afraid of falling out of compliance with GPLv3, they plan to drop GRUB 2 on Secure Boot systems, in favor of another bootloader with a different license that lacks GPLv3's protections for user freedom. Their stated concern is that someone might ship an Ubuntu Certified machine with Restricted Boot (where the user cannot disable it). In order to comply with GPLv3, Ubuntu thinks it would then have to divulge its private key so that users could sign and install modified software on the restricted system.

This fear is unfounded and based on a misunderstanding of GPLv3. We have not been able to come up with any scenario where Ubuntu would be forced to divulge a private signing key because a third-party computer manufacturer or distributor shipped Ubuntu on a Restricted Boot machine. In such situations, the computer distributor – not Canonical or Ubuntu – would be the one responsible for providing the information necessary for users to run modified versions of the software.

Furthermore, addressing the threat of Restricted Boot by weakening the license of the bootloader is backwards. With a weaker license, companies will now have a form of *advance permission* to obstruct the user's ability to run modified software. Rather than work to make sure this situation does not happen – for example by enforcing the proper Secure Boot implementation they say they "strongly support in [their] own firmware guidelines" – Ubuntu has chosen a path which explicitly *allows* Restricted Boot.

No representative from Canonical contacted the FSF about these issues prior to announcing the policy. This is unfortunate because the FSF, in addition to being the primary interpreter of the license in question, is the copyright holder of GRUB 2, the main piece of GPLv3-covered software at issue.

It is not too late to change. We urge Ubuntu and Canonical to reverse this decision, and we offer our help in working through any licensing concerns. We also hope that Ubuntu, like Fedora, will actively support users generating and using their own signing keys to run and share any versions of the software, and not require users to install a key from Canonical to get the full benefit of their operating system.

5 What's the FSF doing to help solve these problems?

Secure Boot raises many issues for protecting user freedom, promoting free software ideals, and encouraging free software adoption. Addressing it requires a multifaceted approach. Assessing the solutions popular GNU/Linux distributions have proposed is one aspect, but we will also take proactive measures of our own.

- We will continue to build public support around our statement against Restricted Boot. Over 31,000 people and 25 organizations have signed

this statement, pledging not to buy any computer that they cannot install a free operating system on, and to advise others to do the same. We were pleased last week to add Debian GNU/Linux as an official organizational supporter of the statement. Subsequently, Trisquel and gNewSense have also added their signatures. When further actions need to be taken to stand up for this freedom as Secure Boot and Restricted Boot are rolled out, we will call upon this base of support. If you haven't yet signed, please do.

- We will fight Microsoft's attempt at enforcing Restricted Boot on ARM devices like smartphones and tablets. Like any other computer, users must be able to install free software operating systems on these devices. We will monitor Microsoft's behavior to make sure they do not deceive the public again by expanding these restrictions to other kinds of systems.

- We will work with (and when necessary, pressure) manufacturers and distributors to make the user instructions for working with Secure Boot on all systems extremely clear, so that users will be able to disable it and modify the approved keys with little difficulty and no bias. We will also work to make sure that users can change *all* of the software running on their machine, including the boot firmware itself.

- We will offer our licensing and compliance resources to any free software developers to help them make sure they are complying with the GPL and other licenses as they implement Secure Boot. We will monitor distributions of signed GPLv3 software to ensure that they respect the necessary user freedoms, including providing installation instructions and materials.

- We have already started exploring ways in which the FSF can work with manufacturers on behalf of the entire free software community to make free software operating systems installable with default Secure Boot hardware settings.

- We will continue to work with companies like Lemote, Freedom Included, ZaReason, ThinkPenguin, Los Alamos Computers, Garlach44, and InaTux to make computers available that are *preinstalled* with fully free GNU/Linux distributions.

- We will help provide information about which computers and components are most compatible with free software, including making people aware of which machines have Restricted Boot. Much of this information will be found at `http://h-node.org`.

6 Conclusion and recommendations

What we've offered here is our position based on the details published by all parties involved so far – we will continue to assess the situation as these plans are actually put into practice, or changes are announced.

Our focus is to evaluate proposed solutions to the issues posed by Secure Boot on the basis of how well they protect user freedom, to recommend the solutions that do the best job of that, and to stop attempts to turn Secure Boot into Restricted Boot.

The best solution currently available for operating system distributions includes:

1. fully supporting user-generated keys, including providing tools and full documentation for booting and installing both modified and official versions of the distribution using this method;

2. using a GPLv3-covered bootloader to help protect users against the dangers of Restricted Boot;

3. avoiding requiring or encouraging users to trust Microsoft or any company which makes proprietary software; and

4. joining the FSF and the broader free software movement in pressuring computer distributors to facilitate easy and independent installation of free software operating systems on any computer.

We will do what we can to help all free software operating system distributions follow this path, and we will work on a political level to reduce the practical difficulties that adhering to these principles might pose for expedient installation of free software. The FSF does want everyone to be able to easily install a free operating system – our ultimate goal is for everyone to do so, and the experience of trying out free software is a powerful way to communicate the importance of free software ideals to new people. But we

cannot in the name of expediency or simplicity accept systems that direct users to put their trust in entities whose goal it is to extinguish free software. If that's the tradeoff, we better just turn Secure Boot off.

Please support the FSF's work in this area by joining as a member or making a one-time donation.