

```
/* * INET          An implementation of the TCP/IP
protocol suite for the LINUX
*                operating system.  INET is implemented
using the        BSD Socket
```

Encode + Decode / 297

The (copylefted) Source Code for the Ethical Production of Information Freedom

BIELLA COLEMAN

Free software hackers, and hackers in general, have an almost maniacal penchant and love, I mean true LOVE, for 'information freedom'. It is at once an ethical stance, an aesthetic sensibility, and increasingly an overt political issue. In this essay, I will address the emergence, constitution, and solidification of the hacker's ethical stance towards information freedom and sharing. In particular, how this is expressed by the free software developers of the Debian project: a non-commercial distribution of the Linux OS with over 800 developers worldwide. I will also address the macro and micro social spheres in which ethics are socially produced and expressed by free software hackers. The macro context is the larger socio-political milieu in which the practice of hacking generally unfolds, while the micro context relates to actual participation in the sphere of free software, especially those of development projects. Given the messiness that is social life, these two spheres are clearly not entirely distinct, but will nevertheless be analytically separated to clarify how ethics become a solid and social value, from what are initially more fuzzy, ambiguous and individual stances.

By looking at these two structural shapers of ethics, I will additionally argue that although hacking is often perceived as apolitical, hacking always tends to evoke political elements due to the nature of knowledge in our society. The quest for knowledge, which is an unmistakable core component of hacking, is a politics of transgression because the 'knowledge' that is sought is often inaccessible (or potentially so) at either a technological or legal level. It is this political condition that helps to explain the initial entry into the 'ethical realm' for hackers. But it is through more sustained participation in the world of free software, especially projects, that a deeper ethical practice is cultivated. Free software projects are spaces in which hackers can flesh out the skeleton of their ethics, especially through its 'exercise' or application.

Hacker politics don't look, smell or feel like your traditional forms of party politics, coalitions and protests. In fact, there is a seemingly paradoxical situation in which, on the one hand, many hackers state something to the effect of 'I stay away from politics', or are accused by activists of being totally apolitical, while on the other hand the question of politics seems to loom at every corner of the practice and philosophy of hacking. There are, of course, a number of historical and current examples of explicitly political movements or organisations, like that of the EFF (Electronic Frontier Foundation) and FSF (Free Software Foundation). However, it does seem like the large majority of different types of hackers, whether it is your phone phreaker, cyberpunk, gamer, or open source developer, are NOT

```
*                Tegge          : Arp bug fixes.
*                Florian        : Removed many unnec-
essary functions, code cleanup
```

political. How then does one understand the political traces scattered all over the hacker cultural sphere? What does one make of dramatic hacker manifestos, DeCSS poetry, underground BBS and zines, and technical discussions about copyright and patent law that would leave any 3L Berkeley Law Student breathless (or possibly ashamed, because nackers know just as much about computer law as law students without stepping in a classroom)?

Clearly, while it is difficult to put a finger on hacker politics, there is indeed something very political going on within and through hacking. I believe this has to do with the formal (as opposed to substantive) nature of hacker politics. The political is not something that hackers do – as an activist might when organising a protest or coalition – instead it is done by and through the very act of hacking. The politics, often of transgression, is embedded within the fibers of the practice of hacking. The political dimension remains obscure since it comes from the rationalised practice of programming and technological manipulation.

Before I explain what I mean more fully, I need to answer the basic, yet incredibly complex, question of what hacking is. Hacking is not easy to define; there are a range of ways one could explain the phenomena of hacking across time and space. However, I am going to be bold and present what I think is one of the fundamental traits of hacking. Here I will actually defer to a hacker and academic, Patrice Reimens, who offers one of the clearest definitions of hacking. Fittingly enough, in a piece that examines the difference between hacking and political activism, he writes: "To put it simply... the hacker ethic runs strikingly parallel to the formula – *art pour l'art* (art for art's sake). What matters here, is the realisation that unlike activists, hackers are focussed on the pursuit of knowledge and the exercise of curiosity for its own sake. Therefore, the obligations that derive from the hacker ethic are perceived by genuine hackers as sovereign and not instrumental, and always prevail above other aims or interests, whatever these may be, and if there are any at all..."

This sentiment has been expressed by countless others, including Levy's now cult classic *Hackers* (1984), and was revisited recently by Castells in the *Internet Galaxy* (2001). My experiences with free software hackers support this fundamental tenet. The spirit of exploration that forms the basis of hacking might start by taking apart a household blender, much to a mother's horror; then lead to learning how to programme at the age of 5, much to the delight of the parental unit; then transform into locking oneself in the bedroom to read every computer manual, which parents duly confuse with pre-teen angst; then to learning every in and out of that simple operating system known as UNIX, discovering every last topographical and temporal feature of the Net much to the amazement of the anthropologist; and finally to volunteering their time to code on free software projects, often to the dismay, again, of their parents. Bruce Sterling expresses this deep proclivity for learning quite accurately when he writes that hackers are "possessed not merely by curiosity but by a positive lust to know". Though hacking as a practice and even a philosophy is much more complex than this, the pursuit of knowledge and learning through the material substratum of computers and the network is a basic undeniable element of computer hacking.

Most commentators parse out the political from this very core substance of hacking. For example, Castells notes that as opposed to your general hacking "there are, however, hacker subcultures built on political principles, as well as personal revolt", going on to note the über-example of Richard Stallman and the Free Software Foundation. Reimens who

```
#include <linux/socket.h>
#include <linux/in.h>
#include <linux/inet.h>
#include <linux/ip.h>
#include <linux/netdevice.h>
```

notes that political activists “do have objectives and aims that precede their action. Hackers on the other hand, are usually happy with the ‘mere’ but unrestricted pursuit of knowledge which reduces their political programme if that can so be called, to the freedom of learning and enquiry”.

But beyond this, I would like to suggest, that in order to better understand the social nature of hacking and the hacker proclivity toward ethical codes, we must see hacking as something fundamentally political. It is political because of the where and the how of hacking itself. This is the macro-context that I referred to in my introduction. Hackers’ insatiable quest for knowledge has historically existed, and currently exists, in a larger socio-political and economic context where ‘knowledge’ – or at least the knowledge they want – is legally, technologically, or institutionally inaccessible for learning and especially for using. Art for art’s sake does not occur in a vacuum, but in a context that gives the pursuit varying shades of a political, transgressive hue. This is a hue that lends itself towards adopting or thinking about ethical questions in the first place.

While transgression may not apply to taking apart the blender (yet) many activities of a hacker have been, are and have the potential to be illegal. One can even think about a good portion of the history of UNIX as a political battle over access to its source code. Inaccessible knowledge may come in the form of a patented algorithm, cryptography, the telephone system, or an API. Corporate policy, patents, trade secrets, copyrights, technological copy control measures, and currently more draconian legal schemes like the DMCA, are the main legal and institutional vehicles that over decades have created the categories of forbidden knowledge and illegal access, and thus have created the socio-political conditions that lay the ground for a politics of transgression. Even without explicitly political intentions, hacking still keeps alive the question of how the boundaries between public and private knowledge should be defined.

* I would like to ask how it is that hackers on the ground experience this condition that I am marking as the necessary groundwork for the cultivation of ethics? And how does one express political subjectivity anyway? Hackers may not operate within a sphere where the threat of the political hammer is knocking on their heads, but there is a ghostly knock that can be heard in the halls of hacking. A ghost has taken more material form through the very visible hacker crackdowns of the early 1990s: the arrest and persecution of Kevin Mitnick, the visible arrest of Dmitry Sklyarov at Defcon 9, the legal threat to Jo Johansson over DeCoss, the looming legal threat of the DMCA, the constant barrage of negative press and the media portrayal of hackers as criminal underground bandits and pirates.

Although the child programmers, barely out of the crib and hacking away with BASIC, may not be aware of the political nature of what they do (thankfully), it is not too long before many programmers become at some level aware that what they do is in some dimension illegal. There is a basic but ambiguous cognisance of what I have referred to as a politics of transgression. In fact, now younger hackers are becoming even more aware that hacking can have very legal and political repercussions. This is due to exposure early on to what I will call the hacker public sphere: a sphere composed of the online and off-line sites and domains in which hackers congregate to talk about all things technical, and increasingly, all things political (such as Slashdot, irc channels, local 2600 and LUG meetings, mailing

```
* Set the protocol type. For a packet of
type ETH_P_802_3 we put the length
```

lists). Many of my life experiences give testament to the minute, but powerful ways in which programmers came to realise that what they do is illegal, or at least potentially so. One of the funniest stories, which I have heard a number of times from Linux 'old timers', is of being accused of pirating software by computer lab staff while downloading (Free) Linux, when in fact they were JUST downloading the early versions of Linux which filled about forty floppies.

Given the inclination for knowledge and curiosity, the very act of circumventing access controls – whether human, legal or technological – has become an end in and of itself. Along with a politics of transgression, a poetics of transgression has come to occupy a special place in certain hacker spheres, which admittedly tends not to be in the free software sphere. Forbidden fruit is sweet. Some of the most 'fun forms of hacking' are those in which you try to access forbidden knowledge, the harder to get, the more ripe for hacking.

Thus, while hacking may not necessarily look like activist politics at a formal level, it is often fundamentally political. Indeed, the very act of hacking makes visible the constraints of knowledge in our society. It is in this amorphous, but still very real, contextualised space where the first seeds of ethical sentiments to information freedom are born.

Now I would like to switch to the micro-context of the free software project and look at how participation on a project like Debian contributes to the solidification of the ethical principles first developed out of the experience of transgression. It is interesting to note that within the space of free software development transgression does not occur because the copyleft, the main legal licence for free software projects, has materially and symbolically reterritorialised certain forms of knowledge. Copyleft has made source code permanently and legally accessible. Instead of engaging in the more subtle forms of transgressive politics that arise on the restrictive space of copyright and patents, Richard Stallman came up with a legal licence and political organisation that could confront the issue of knowledge head on.

Developers tend to come to the Debian project with some sort of ethical stance about information freedom, although it is usually not all that well thought out. Initially, some already hold the idea that 'information wants to be free' for moral and universalist reasons, while most Debian developers feel that 'information should be free' for practical reasons such as building better software. However, through continued participation in Debian many developers come to subscribe to a certain degree to both a moralistic and functional understanding of this hacker ethic of information freedom and sharing. As one developer noted, "We are hard-core about being free. Red Hat will bundle non-free. What Debian throws into the mix is that we are free and we are serious about being free. Certainly, you don't have to have such a devotion to it, but the fact is that there is a group of people that are so dedicated to freedom and openness".

Why the devotion? One might think that this sort of 'hard-core' position is simply an ethical belief that developers have and bring to projects, but it is more accurate to say that participation in projects is the site where the skeleton of ethics is given its flesh. Many developers talk about the fact that, although they were always committed to freedom, their knowledge about the legal issues surrounding software and content was bare until participating on Debian. The act of entry into the project itself nicely illustrates the ethical sociali-

Ethernet MAC header. This is called after an ARP
* (or in future other address resolution) has
completed on this

```

*   sk_buff. We now let ARP fill in the other fields.
*
*   This routine CANNOT use cached dst->neigh
*   Really, it is used only when dst->neigh is wrong.
*/

```

Encode + Decode / 301

sation that developers first experience when entering the realm of the free software project. As part of the process to become a 'Debian Maintainer' the prospective developer has to pass a short test about the Debian Free Software Guidelines and their Social Contract. Although this is a very informal process, it marks a rite of passage into a project where ethics are made manifest through initiation exercises. Ethics are also realised in the detailed and complex discussions that occur on mailing lists, IRC and conferences about software licences, the practicalities of what licence to apply to your code, and even reflective discussion on the political nature of Debian itself. This was exemplified in a recent thread on the main Debian developer mailing list where there was a discussion on whether Debian is an example of political anarchy in action.

Over time a deep dedication to the organisation develops because of what developers learn and gain by participation. As much as Debian developers give their own time and know-how many feel they personally gain a tremendous amount from participation. They gain things like free technical tools to use at home or for work, the satisfaction of building a quality product that fellow peers use and admire, collaborative skills, new forms of knowledge and a sense of belonging to a community. One long-time participant has expressed it in the following way:

"I've learned about the intricacies and history and every detail of the Debian distribution, how its disparate components fit together, how its packaging system works. I've learned all sorts of little oddities of technical lore, and I've picked up a few programming languages and a lot of general programming knowledge. I've learned how to collaborate with folks spread out over the world and across time zones. I've learned how to argue effectively online, and I've learned that even though I tend to shy away from arguments, there are things that are worth arguing for. I've learned how to think about the large effects a work can have on a project, and how to take responsibility for and plan out those effects beforehand."

The experience of learning (non-technical and technical skills) and sharing helps to explain the strong allegiance that developers develop towards Debian, and the stronger ethical stance for information freedom and sharing that develops over time. The hacker drive for knowledge becomes more overtly ethical and socialised through participation on a free software project. Many free software projects provide a social space for the practice of sharing in which the classic geek tendency for elitism and bravado also gives way to a desire to help others, when developers come to recognise how much they have gained from others – whether it is in the form of software, help, or more intense learning. On Debian, the fundamental hacker pursuit for knowledge becomes an endeavour that is recognised as a social process that requires human interfacing and coders tend to honour the ethic of sharing. So while hackers might still love doing art for art's sake, they additionally come to do art for the sake of others.

I have explained how the experience of programming for a large project helps to substantially flesh out and solidify nascent ethical commitments that are part of the motivation for joining a project in the first place. Although we can conceptually separate the moral drive from the more practical or self-motivated reasons that form the underlying basis to join a project, the two eventually work to bolster each other. The technical success of free soft-

```

(struct ethhdr *eth;
 unsigned char *rawp;

```

ware projects, and the personal gains from them, reflects the influence of the hacker belief in openness and sharing of information.

In conclusion, I hope that I was able to at least offer a rudimentary blue print of the conditions in which the hacker ethic gets constituted. Instead of presenting the hacker ethic for information freedom as a static and ideologically misguided commitment, I wanted to offer a processual account of the distinct social spaces where these ethics emerge, and the site or praxis where these ethics are solidified. In the process of doing this, I hope to have set the stage for inquiry into the peculiar and particular politics that emerge out of the techocultural practice of hacking, as well as the broader relationship between ethics and politics. Moreover, there will still have to be an inquiry that asks how the politics of freedom of information in the hacker sphere relates to a larger tradition in American politics of placing value in information freedom to overcome the tyranny of bureaucracy, to develop self-understanding of the law, and to empower the 'public' against false rhetoric. However, this can only be undertaken with a concrete understanding of the form and content of hacker ethics.

```

*      so don't forget to remove it.
*
*      Seems, you forgot to remove it. All silly
devices
*      seems to set IFF_PROMISC.
*/

else if(1 /*dev->flags&IFF_PROMISC*/)
{
    if(memcmp(eth->h_dest,dev->dev_addr,
ETH_ALEN))
        skb->pkt_type=PACKET_OTHERHOST;
}

if (ntohs(eth->h_proto) == 1536)
    return eth->h_proto;

rawp = skb->data;

/*
 *      This is a magic hack to spot IPX pack-
ets. Older Novell breaks
 *      the protocol design and runs IPX over
802.3 without an 802.2 LLC

```