THE PATENT LINE

Dear Linux Magazine Reader,

Joe Casad, Editor in Chief

Many have begun to wonder whether the dark predictions regarding Linux patent infringement reflect a real danger, or whether the patent doomsday talk is intended primarily as a means for raising public awareness. The answer is that no one knows for sure – and the reason no one knows is perhaps the most unsettling part of the story.

It was announced recently that Linux may infringe on as many as 283 patents. If you followed the story, you'll recall that few details emerged as to which parts of Linux infringe on which patents. The reason for this lack of information is that, strangely, the more you know about patents, the more they can hurt you. US patent law imposes harsher penalties on those who violate patents knowingly than on those who violate without specific knowledge of the infringement. In other words, if you use or distribute Linux, you'd better avoid any specific knowledge of specific patent violations.

The present trepidation over patents is due, in part, to the fact that the lawyers who really understand which patents Linux may violate don't know enough engineering to evaluate whether the problems caused by those violations could be crippling to Linux. On the other hand, the Linux kernel developers, who are in the position to know what the affects of a violation might be, have a strong disincentive preventing them from looking closely at patent issues.

If this sounds like an unfortunate tangle of ill-conceived laws, that is because it is. The good news is that the great majority of these 283 patents are probably not enforcible. Only a small fraction of patents are airtight enough to bother with suing anyone over, and even then, many of the patents are overturned. The threshold for patentability shifts about in the gray space between good engineering and true innovation, and such matters are hard enough to judge when you're looking at a farm implement. How much more difficult would it be when you're gazing at mountains of source code?

The sad part of the patent situation is that, since no one is in a position to know the full scope of the problems, there is no way of knowing whether these problems could ultimately be addressed programmatically. Does Linux need a new brain or just a new hair-

cut? For the sake of a good story, or perhaps as a mental exercise to explore this most intractable quandary, allow me to propose that, before this is all over, the moment may come when a team of trusted kernel developers embarks on a secret mission – across the dangerous line of patent knowledge, there to dwell with the lawyers. The trusted developers and the lawyers will work together to evaluate exactly which patents are a problem (it will be way fewer than 283 – probably closer to 30) and to assemble conclusive proof for why the other patents aren't a threat.

Once they have crossed the line of patent knowledge, the trusted developers won't be able to go back. They won't be able to share what they know of the problem patents with other developers, and they won't even be able to work on Linux anymore – at least not on versions of Linux that are in active distribution. Their task will be to remain in this secret place, never to emerge until they have produced a version of Linux that is free of the offending patents. The other kernel developers could have a solemn feast for them when they embark and could drink a toast to them with Klingon blood wine.

But is it even possible – and is it even necessary – for a team of kernel developers to voluntarily exile themselves from the community in order to produce a version of Linux that is sanitized of patent vulnerabilities? I do not know, because I have never crossed the line of patent knowledge, but I can imagine scenarios where it might be a better option than sitting around waiting for the lawsuits to start.

Joe

LINUX MAGAZINE