Regulating the Internet

David T. A. Wesley
Northeastern University, USA

INTRODUCTION

The early history of the Internet was a time of naive optimism that the medium would create a new era of global free speech. In 1996, John Barlow, an Internet Pioneer and founder of the digital rights organization known as the Electronic Frontier Foundation (EFF), wrote "A Declaration of the Independence of Cyberspace" in which he declared that governments "have no sovereignty" over the Internet. Cyberspace was "naturally independent of the tyrannies" of nation states, he wrote, and any attempts to control it would ultimately fail (Barlow, 1996).

More than 20 years later, Google is creating a search engine for China that censors any content deemed disagreeable to the central government, Facebook is being used by hate groups to organize lynch mobs and commit genocide, and Internet-connected cameras are being deployed for mass surveillance. The Internet has also become a primary tool for state propaganda and "fake news" aimed at undermining democracy and free speech.

Despite these challenges, court decisions that protect computer code under the First Amendment of the United States constitution have helped create a free flow of knowledge and tools that can be used to counter such threats. Given the ubiquity of American technology companies, U.S. legal precedents and decisions can have global implications.

FOCUS OF THE ARTICLE

Established U.S. law has long held that computer code is a language, like any other language, and is therefore subject to same free speech protections afforded other forms of speech under the First Amendment of the United States Constitution. Computer code also protects free speech through cryptography that enables protected communication between two or more parties.

This article will consider the legal history of computer code as free speech and how it can be used to promote other forms of free speech through cryptography and secure communications. It will further argue that the deep web and dark web are direct results of these precedents and while they can be abused by cybercriminals and malicious state actors, they are also indispensable in promoting free speech and human rights.

BACKGROUND

The telegraph was one of the first uses of technology to communicate information. In the early 20th century, the language of automatic telegraph operators in many ways resembled computer code. For example, when a telegraph operator would send a message, the message was encoded with special

DOI: 10.4018/978-1-5225-9715-5.ch026

terms – dux for duplex, mux for multiplex, gm for good morning, etc. With the advent of the automatic telegraph, words such as printers, machines, and bugs took on new meaning. A bug, for example, was an electrical disturbance that could cause the signal to drop out momentarily or even cause the system to hang (Brackbill, 1929).

Similarly, the first computer engineers created unique computer codes that later developed into their own languages. In the 1950s, as computers became more complex, engineers began to recognize the problem of making this increasingly complex machine language intelligible. Brownson (1953) wrote, "In communicating with each other, we seek to communicate concepts; in defining terms for manipulation by computers, man will have to find out exactly what he believes and make coherent and integrated sense of it."

Bar-Hillel (1953), a pioneer in machine translation, saw this as "a real challenge for structural linguists," but he also believed that it would only be a matter of time before machines acquired a "semantic organ." "One of the decisive steps in certain methods of machine translation is the determination of the syntactic structure of any given sentence in the source-language (i.e., the language from which we translate) to a required degree of explicitness." As a result, computers that "were originally designed to solve certain mathematical problems… might well be recombined to yield similar results in noncomputational fields" (Bar-Hillel & Bar-Hillel, 1951).

Already, Brownson and Bar-Hillel were expressing two views that would later become central in the debate over computer code as free speech. Brownson saw computer code as a form of communicating what one believes, whereas Bar-Hillel saw it as a means to solve problems, including those that were strictly mathematical and those that would later develop in noncomputational fields. If computer code expresses ideas and beliefs, it could be a protected form of speech under the First Amendment to the United States Constitution, but if, as Bar-Hillel says, it is merely a method to solve problems, then it should not be protected.

Two of the earliest cases to consider the question of the First Amendment status of computer code involved export restrictions on cryptography programs. In 1996, President Clinton issued an executive order restricting the exportation of encryption products in the interests of national security. Henceforth, encryption technology would be "designated as defense articles in Category XIII of the United States Munitions List and regulated by the United States Department of State pursuant to the Arms Export Control Act" (Executive Order 13026, 61 Fed. Reg. 58767).

In *Karn v. United States Department of State* (1996), the court made "no ruling as to whether source codes… fall within the protection of the First Amendment," but clearly suggested that it was not protected when it stated that "[s]ource codes are merely a means of commanding a computer to perform a function." In this case, the court applied the O'Brien test (*United States v. O'Brien*, 1968), holding that the "government regulation at issue here is clearly content-neutral" and that the government had a compelling national security interest at stake.

This contrasts the position in *Bernstein v. United States Department of State* (1996), in which the Honorable Marilyn Patel held that "the particular language one chooses [does not] change the nature of language for First Amendment purposes. This court can find no meaningful difference between computer language, particularly high-level languages as defined above, and German or French. All participate in a complex system of understood meanings within specific communities. Even object code, which directly instructs the computer, operates as a 'language.' When the source code is converted into the object code 'language,' the object program still contains the text of the source program. The expression of ideas, commands, objectives and other contents of the source program are merely translated into machine-