Sergey Bratus
6211 Sudikoff Laboratory
Dartmouth College
Hanover, NH 03755
Ph.: (603) 646-9224

March 14, 2015

Re: Draft report on "Human rights and technology: the impact of intrusion and surveillance systems on human rights in third countries", item #17, regulating "zero-day exploits".

Dear Ms. Schaake,

Free exchange of exploits is crucial to cybersecurity. There is no smart regulation of exploits that won't set back security. Regulation should take a different focus: on software for surveillance and stealing data from personal devices.

I am a security researcher. My job is to write about computer system vulnerabilities and about how to mitigate them and how to construct future systems without them. In 2014 I co-chaired a workshop by USENIX—an association of researchers and practitioners that frequently stood for rights and freedoms of computer users—on Offensive Computing, a study of computer attacks and vulnerabilities.

I am extremely concerned about the proposal to "regulate sales of zero-day exploits". I am convinced that it will have adverse effects not only on research in my area, but on the overall state of computer security. Instead of protecting users from cyber-attacks, it will weaken both personal computing technology and infrastructure, while attackers will develop their capabilities unburdened.

For this letter I will leave out most of the terminology and jargon I professionally use, and will focus on two terms only: "exploits" and "zero-day".

Let me first talk about exploits. My primary product is papers, but at the heart of each paper is an exploit or a class of exploits. Essentially, most of my papers are about exploits, some discovered by me, most discovered by other people, most of these working outside of academia, in private employment.

Exploits are proofs of vulnerabilities. Without a working program—an exploit—I and my colleagues cannot claim that the security vulnerabilities we write about actually exist, no more than a physicist can claim that a physical phenomenon exists without successful experiments. But, as computer security researchers, we deal with programs, and these are our proofs and our experiments.

An exploit is simply a program that makes other programs behave in ways not expected by their original designers. Exploits show conclusively and beyond doubt that software or hardware can do what was believed and trusted to be impossible.

In computer science theory, the question of what computations were impossible occupied such brilliant minds as Alan Turing and Alonzo Church. They discovered the fact that designers and programmers do not and cannot know all properties of programs they themselves write. In applied computing, these questions are settled empirically—with exploits.

Exploits are mostly software. Software takes time to write and test before it actually works (just as today's physics experiments require custom equipment that must be built and tested). People must be paid for working on it. Many key results that I employ in my work exist only because expert researchers were paid to develop exploits.

Now let me talk about "zero-day" for a minute. "Zero-day" means new, not known before. Any scientific result worth publishing is "zero-day"—previously unknown, just discovered. Science is pursuit of "zero-day" discoveries. Since we are computer scientists, our discoveries take the form of programs: "zero-day" programs.

Without "zero-day exploits", claims of new security phenomena—new vulnerabilities, new types of vulnerabilities, new risks—remain hypothetical. The industry cannot waste its effort on hypotheticals. Even if they wanted to, how would they know which hypotheticals are actually worth their effort, which is necessarily limited?

Why do I worry about "sales of zero-day exploits"? One can point out that I do not sell exploits, I publish about them openly. This is true, but I depend on public funding. My colleagues who practice the same skills, however, mostly work for private employers, and their results become public only eventually. Yet most of the results I build on were not obtained or published in academia. They were instead discovered privately, and they took weeks to months, sometimes years of privately paid labor to discover.

When does privately paid labor, work-for-hire on an exploit become a "sale"? Does a contractual deliverable of an independent researcher become a "sale"? What purpose would be served by eliminating researcher independence by "regulating sales"?

Public funds can only be spent deliberatively and judiciously; they cannot fund ventures with high risk of failure. Imagine what would become of the Silicon Valley if projects there were funded the way academic funds are distributed. Would it even exist? Most likely not.

Regulated enterprise moves slowly and avoids risk. At the same time, other countries where basic exploit research will not be so burdened will catch up and overtake our current level of defensive insight based on the kind of exploits we know to date. This may sound far-fetched, but it's real: famous security professionals like Microsoft's John Lambert point out that cyber-defense is informed by exploits, not the other way around. For a solid theory of security we must explore insecurity.

In summary, "regulation of zero-day exploit sales" suggested in the proposal goes against the proposal's own goals. By chilling development and exchange of exploits—i.e., code, a form of speech—it will chill the fundamental freedom of speech (#1). It will chill dissemination and deployment of software to protect privacy and digital freedom (#4)—what good is a privacy-protecting program if users cannot install it on their phones without jailbreaking/unlocking exploit tools enabling such "intrusion"? It will chill anti-censorship software (#5), for the same reason. It will chill access to knowledge and information about technology internals (#8), because exploits are key education tools that allow users to unlock their proprietary platforms for user's own programming. It will hinder the defense of the open internet (#20) by allowing vendors to leverage control over exploitless internet endpoint systems. It will chill the voice of independent security experts (#21), who need to be able to sustain themselves to remain independent. It will actually bolster restrictions on security research (#22), by making the essence of researcher professional communications, exploits, harder to exchange. It will undermine democratic oversight of police and other government cyber-activities (#23) by reducing the public-minded experts' ability—largely based on exploit techniques—to dissect the software involved.

As a concerned security professional, I urge you to refocus your proposal on the actual enabler of computing surveillance: *exfiltration* software that gathers and transmits data about users without their consent. We describe this approach in our public comment on the Wassenaar implementation, `http://www.cs.dartmouth.edu/~sergey/wassenaar/wassenaar-public-comment.pdf`

Best regards,
[signed]

Sergey Bratus, Ph.D.
Research Associate Professor
Institute for Security, Technology,
and Society,
Department of Computer Science
Dartmouth College