Why Cyberattacks Are Easier Than Cyberdefense

Estevan H. Ramos and Vladik Kreinovich
Department of Computer Science, University of Texas at El Paso
ehramos@miners.utep.edu, vladik@utep.edu

Formulation of the problem. In general, in military confrontations, defense is easier than an attack. However, in cybersecurity, the inverse is true: cyberattacks are easier than cyberdefense; see, e.g., [1]: many times college kids who have not yet finished their education managed to penetrate sophisticated cybersecurity arrangements of Pentagon and other heavily protected targets. How can we explain this?

Our explanation. For each system s and attack a, let $S(a, s)$ indicate that the attack a was successful against the system s. For each pair a and s, it is feasible to check whether $S(a, s)$ is true: to check this, it is sufficient to launch the attack and see if it succeeds. In other words, the predicate $S(a, s)$ is feasible: its truth value can be computed by a feasible (= time-polynomial) algorithm.

In these terms, finding a successful attack means finding a for which $S(a, s)$ is true. Once someone proposes a possible attack, it take polynomial time to check whether this attack was successful. In other words, if we consider "algorithms" including guessing steps – such "algorithms" are known as *non-deterministic algorithms* – then such a non-deterministic algorithm can solve the problem of finding a successful attack in polynomial time. The class of all the problems that can be solved by such *n*on-deterministic *p*olynomial time is usually denoted by NP. So, the problem of finding a successful attack belongs to the class NP; see, e.g., [2] for a general description of this and other complexity classes.

In NP-problems, the existence of a successful attack can be described as $\exists a\, S(a, s)$, i.e., as a formula with one existential quantifier. An existential quantifier is, in effect, an "or" (over all possible attacks), and in digital design, "or" is usually describe by a sum Σ. Thus, the class NP is also described as $\Sigma_1\mathbf{P}$.

On the other hand, finding a successful defense means finding s for which for every a, we have $\neg S(a, s)$. The formula describing the existence of such s is $\exists s\, \forall a\, \neg S(a, s)$. This formula also starts with $\exists$, but now it has two quantifiers, so the class of such formulas is denoted by $\Sigma_2\mathbf{P}$; it is one of the classes next to $\Sigma_1\mathbf{P}$ in the so-called *polynomial hierarchy*. At present, it is not known whether problems from the class $\Sigma_2\mathbf{P}$ are, in general, more complex to solve that problems from $\Sigma_1\mathbf{P}$. However, most computer scientists believe that, in general, problems $\Sigma_2\mathbf{P}$ are more complex.

This explains why cyberattacks are easier than cyberdefense.

Comment. Why does not the same logic apply to the military attacks and defense? Because in cybersecurity success or failure of an attack depends on its ingenuity, brute force is a minor factor. In contrast, in military conflicts, the situation is different: there, brute force is an important – often dominant – factor.

References

[1] N. Kshetri, "Economics of Artifical Intelligence in cybersecurity", *IT Professional*, September/October 2021, pp. 73–77.

[2] C. Papadimitriou, *Computational Complexity*, Addison-Wesley, Reading, Massachusetts, 1994.