BRUCE SCHNEIER

SZÓSTY ZMYSŁ HAKERA

O HAKOWANIU SYSTEMÓW SPOŁECZNYCH I PRZYWRACANIU SPRAWIEDLIWYCH ZASAD GRY

Helion

Tytuł oryginału: A Hacker's Mind: How the Powerful Bend Society's Rules, and How to Bend them Back

Tłumaczenie: Katarzyna Ellerik

ISBN: 978-83-289-1101-7

Drogi Czytelniku!
Jeżeli chcesz ocenić tę książkę, zajrzyj pod adres
https://helion.pl/user/opinie/umyhak
Możesz tam wpisać swoje uwagi, spostrzeżenia, recenzję.

Helion S.A.
ul. Kościuszki 1c, 44-100 Gliwice
tel. 32 230 98 63
e-mail: *helion@helion.pl*
WWW: *https://helion.pl* (księgarnia internetowa, katalog książek)

Printed in Poland.

Spis treści

VII. HAKOWANIE SYSTEMÓW SZTUCZNEJ INTELIGENCJI

Rozdział 1.

Czym jest hakowanie?

Słowa „hack", „hakowanie" i „haker" mają silne konotacje i szczególną reputację[1]. Moja definicja nie jest ani dokładna, ani ogólnie przyjęta. Nie przeszkadza mi to. Mam zamiar pokazać, że myślenie kategoriami hakowania ułatwia zrozumienie szerokiego wachlarza systemów, tego, jak zawodzą, i tego, jak uczynić je bardziej odpornymi.

> **Hack** (definicja)[2]
>
> 1. Sprytne, niezamierzone zastosowanie systemu, które (a) podważa reguły lub normy systemu (b) kosztem kogoś, kto podlega działaniu tego systemu.
> 2. Coś, na co system pozwala, ale co pozostaje niezamierzone i nieprzewidziane przez twórców systemu.

Hakowanie to nie to samo co oszukiwanie. Hack może być też oszustwem, ale raczej nim nie jest. Gdy ktoś oszukuje, robi coś wbrew regułom — coś, czego system jawnie zabrania. Wpisywanie na stronie czyjejś nazwy użytkownika i hasła bez zgody tej osoby albo nieujawnienie wszystkich swoich dochodów w zeznaniu podatkowym czy ściągniecie czyichś odpowiedzi na sprawdzianie — to wszystko oszustwa. Żadne z nich nie jest hakowaniem.

Hakowanie to nie to samo co poprawki, rozszerzenia czy innowacje. Ćwiczenie swojego serwisu i powrót na kort jako lepszy tenisista to poprawienie się. Gdy Apple doda nową funkcjonalność do swojego iPhone'a, to jest to rozszerzenie. Znalezienie zmyślnego sposobu na wykorzystanie arkusza kalkulacyjnego może być innowacją. Czasami hack jest też innowacją lub wzbogaceniem — gdy na przykład włamiesz się na własne urządzenie od Apple, by zainstalować na nim funkcjonalności, których firma nie akceptuje — ale nie zawsze tak jest.

Hakowanie bierze na cel system i zwraca go przeciw niemu samemu, nie uszkadzając go w tym procesie. Jeśli rozbiję szybę w oknie Twojego samochodu i zrobię zwarcie

w zapłonie, to nie będzie to hack. Jeśli znajdę sposób, żeby system dostępu do auta bez klucza otworzył mi drzwi i uruchomił silnik — to to będzie hack.

Zwróć uwagę na różnicę. Hacker jest nie tylko sprytniejszy od swojej ofiary. On znajduje uchybienie w regułach systemu. Robi coś, czego nie powinien móc zrobić, ale jednak może. Jest sprytniejszy od systemu. Wynika z tego także, że jest sprytniejszy od projektantów systemu.

Hakowanie podważa zamiar systemu przez podważenie jego reguł czy norm. To prowadzenie gry z systemem. Plasuje się gdzieś między oszustwem a innowacją.

„Hack" to pojęcie subiektywne. W tej dziedzinie wiele kwestii należy do kategorii „jak to zobaczę, to będę wiedział, czy to hack". Niektóre działania to oczywiste hakowanie. Inne w tak samo oczywisty sposób nie są hakowaniem. A inna pula aktywności plasuje się w szarej strefie pomiędzy nimi. Szybkie czytanie — to nie hack. Ukrycie mikrokropki w znaku interpunkcyjnym na stronie drukowanego tekstu — to zdecydowanie hack. Cliff Notes — może, nie jestem pewien.

Hacki są błyskotliwe. Skutkują zawistnym podziwem (być może obok słusznej złości) i zawierają element „och, szkoda, że na to sam nie wpadłem", nawet jeśli ich sedno jest czymś, czego nigdy byś nie zrobił. Jest to prawda, nawet gdy autorami hacka są źli ludzie o morderczych zamiarach. Swoją książkę *Beyond Fear* z 2003 r. rozpocząłem długim wyjaśnieniem, dlaczego ataki na World Trade Center były „zdumiewające"[3]. Terroryści złamali niepisane reguły porywania samolotów. Przed 11 września 2001 r. porwanie zakładało zmuszenie statku powietrznego do lotu *dokądś*, pewną liczbę żądań politycznych, negocjacje z rządem i policją i zasadniczo pokojowe wyjście z sytuacji. To, co zrobili terroryści w ataku na WTC, było okropne i przerażające, ale dostrzegam też pomysłowość ich hacka. Posłużyli się tylko bronią, która została przepuszczona przez lotniskową kontrolę bezpieczeństwa, zamienili cywilne samoloty w sterowane pociski odrzutowe i zmienili reguły związane z terroryzmem powietrznym.

Hakerzy i ich praca każą nam inaczej myśleć o systemach w naszym świecie. Ujawniają nasze założenia i to, co bierzemy za pewnik, często zawstydzając przy tym ludzi władzy, niejednokrotnie odbywa się to straszliwym kosztem.

Zostawmy temat terroryzmu, ludzie kochają hacki, bo są sprytne. MacGyver był hakerem. Kryminały i filmy o ucieczkach z więzienia są pełne błyskotliwych hacków: *Rififi*, *Wielka ucieczka*, *Papillon*, *Mission Impossible*, *Ocean's 11*, *12*, *13* i *8*.

Hacki to nowości. „A to tak można?" i „Nie wiedziałem, że można tak zrobić!" to częste reakcje na hakowanie. To, co jest hackiem lub nim nie jest, zmienia się z czasem. Reguły i normy się zmieniają. „Powszechna wiedza" się zmienia. Ponieważ ostatecznie hacki przeważnie zostają albo dozwolone, albo zabronione, coś, co kiedyś było hackiem, może już nim nie być. Kiedyś trzeba było zhakować swój telefon, żeby zrobić z niego bezprzewodowy hotspot. Obecnie hotspot WiFi to standardowa funkcjonalność smartfonów Android i iOS. Ukrycie metalowego pilnika w cieście wysłanym uwięzionemu

konfederatowi początkowo było hackiem, ale stało się tropem filmowym, na który strażnicy więzienni się nie nabiorą.

W 2019 r. ktoś posłużył się dronem, żeby dostarczyć telefon komórkowy i marihuanę do więzienia w Ohio[4]. W tamtym momencie nazwałbym to hackiem. W tej chwili używanie dronów w pobliżu więzień jest oficjalnie niedozwolone w niektórych stanach i nie sądzę, żeby nadal był to hack. Czytałem ostatnio o użyciu wędki do przerzucenia kontrabandy przez więzienny mur[5]. A także o kocie, którego złapano, gdy przenosił karty SIM do telefonów i narkotyki do pewnego więzienia na Sri Lance (kot później uciekł)[6]. To zdecydowanie hack.

Hacki często są legalne. Ponieważ przestrzegają litery prawa, ale ignorują jego ducha, są nielegalne tylko wtedy, gdy panuje jakaś nadrzędna reguła zabraniająca ich. Gdy księgowa znajdzie jakąś lukę w prawie podatkowym, jej wykorzystanie jest prawdopodobnie legalne, o ile żadne bardziej ogólne prawo tego nie zabrania.

W języku włoskim funkcjonuje nawet określenie *furbizia* — pomysłowość, z jaką Włosi podchodzą do biurokracji i omijania niewygodnych praw. W języku hindi można znaleźć podobne pojęcie — *jugaad* — które podkreśla spryt i obrotność w radzeniu sobie. W brazylijskiej odmianie języka portugalskiego odpowiednikiem jest słowo *gambiarra*.

Hakowanie bywa etyczne. Niektórzy przyjmują, że kiedy jakieś działanie lub zachowanie jest legalne, automatycznie jest też etyczne, ale świat oczywiście nie jest tak prosty. Tak jak istnieją nieetyczne prawa, tak są też etyczne zbrodnie. Większość hacków, o których będę pisał w tej książce, w teorii należy do legalnych, ale niezgodnych z duchem prawa (a system prawny to tylko jeden z rodzajów systemów, które można zhakować).

Początki słowa „hack" datuje się na 1955 r. i wiąże z grupą Tech Model Railroad Club na uniwersytecie MIT[7]. Pojęcie to szybko spopularyzowało się w rodzącej się dziedzinie komputeryzacji. Pierwotnie oznaczało sposób rozwiązania problemu, zakładający pomysłowość, innowację czy zaradność, bez jakiegokolwiek kontekstu kryminalnego czy choćby wrogiego. Ale już w latach 80. XX w. „hakowanie" najczęściej oznaczało włamywanie się do komputerowych systemów bezpieczeństwa lub ich łamanie. Nie chodziło tylko o to, żeby komputer zrobił coś nowego, ale też o zmuszenie go, żeby zrobił coś, czego nie powinien być w stanie zrobić.

Z mojej perspektywy hakowanie komputerów dzieli tylko jeden niewielki krok od hakowania gospodarki, polityki czy społeczeństwa. Wszystkie te systemy to tylko zbiory reguł, a czasem norm. Są dokładnie tak samo podatne na hakowanie, jak systemy komputerowe.

To nic nowego. Hakujemy systemy społeczne od zarania dziejów.

Rozdział 2.

Hakowanie systemów

Hacki można stosować wobec dowolnego systemu, ale porównania między różnymi rodzajami systemów mogą przydać się do naświetlenia, jak takie hacki działają. Weźmy za przykład porównanie kodeksu podatkowego (ang. *tax code*) i kodu komputerowego (ang. *computer code*).

Kodeks podatkowy to nie oprogramowanie. Nie działa na komputerze. Ale nadal możesz rozpatrywać go jako „kod" w komputerowym znaczeniu tego słowa. To seria algorytmów, które przyjmują dane wejściowe (informacje finansowe z danego roku) i produkują dane wyjściowe (należną kwotę podatku).

Kodeks podatkowy jest niesamowicie złożony. Zawarto w nim całe mnóstwo detali, wyjątków i szczególnych przypadków, choć może nie dla większości z nas jako jednostek, ale dla ludzi bogatych oraz wszelkiej maści firm. Składają się na niego ustawy, rozporządzenia administracyjne, wyroki sądowe i interpretacje. Obejmuje on także prawo oraz regulacje dotyczące korporacji i różnego rodzaju spółek. Trudno trafić na wiarygodne szacunki co do jego wielkości, nawet specjaliści nie potrafili udzielić odpowiedzi na moje pytania na ten temat. W Stanach Zjednoczonych samo prawo podatkowe zajmuje ok. 2600 stron[1]. Regulacje izby skarbowej IRS oraz podatkowe orzeczenia sądowe poszerzają ten zbiór do 70 tysięcy stron. Prawo dotyczące struktur korporacji i spółek jest równie złożone, więc machnę na to ręką i przyjmę 100 tysięcy stron — czyli 3 miliony wierszy — jako objętości kodeksu podatkowego Stanów Zjednoczonych. Microsoft Windows 10 mieści się w 50 milionach linii kodu[2]. Trudno porównywać wiersze tekstu i linie kodu komputerowego, ale nadal stanowi to użyteczne ćwiczenie. W obu przypadkach duża część złożoności związana jest z tym, jak różne składowe całego korpusu wchodzą ze sobą w interakcje.

Każdy kod komputerowy zawiera defekty. To błędy: błędy w specyfikacji, błędy w programowaniu, błędy zachodzące gdzieś w procesie tworzenia oprogramowania, błędy tak prozaiczne, jak literówki czy błędy ortograficzne. Współczesne aplikacje mają setki, jeśli nie tysiące, defektów. Znajdują się one w każdym elemencie oprogramowania, z którego korzystasz: na komputerze, smartfonie, dowolnym urządzeniu z obszaru internetu rzeczy (IoT), jakim dysponujesz w domu lub w pracy. Fakt, że całe to oprogramowanie działa

bez zarzutu przez większość czasu, świadczy tylko o tym, jak niezrozumiałe i nielogiczne są te defekty. Jest mało prawdopodobne, że kiedykolwiek się na nie natkniesz, ale istnieją (podobnie jak wiele części kodeksu podatkowego, z którymi nigdy nie będziesz mieć styczności).

Niektóre z tych defektów oznaczają dziury w bezpieczeństwie. Mam tu na myśli coś bardzo konkretnego: atakujący może celowo wykorzystać defekt, by osiągnąć pewien efekt niechciany przez projektantów oprogramowania i programistów. W żargonie bezpieczeństwa komputerowego nazywamy je „podatnościami".

Kodeks podatkowy również ma defekty. Mogą to być błędy w sposobie sformułowania praw podatkowych: pomyłki słowne, nad którymi głosował Kongres i które prezydent podpisał, zatwierdzając ustawę. Mogą to być błędy w interpretacji tych praw. Mogą to być przeoczenia w sposobie skonstruowania pewnych części tego kodeksu lub pominięcia tego lub innego rodzaju. Mogą powstać w wyniku miriad interakcji między różnymi elementami ustawodawstwa w zakresie podatków.

Niedawny przykład to ustawa Tax Cuts and Jobs Act z 2017 r. Ustawę tę spisano naprędce i w tajemnicy, a przegłosowano, nie dawszy legislatorom czasu na analizę czy choćby powtórne przeczytanie pierwszego szkicu. Część projektu została spisana odręcznie i w zasadzie jest niemożliwe, żeby ktokolwiek, kto głosował za lub przeciw, widział dokładnie, co ten projekt zawierał. Tekst zawierał natomiast błąd, który przypadkowo klasyfikował rentę po zmarłym wojskowym jako dochód podlegający opodatkowaniu. W konsekwencji tej pomyłki rodzinom poległych niespodziewanie wystawiono rachunki na 10 tysięcy dolarów lub więcej[3]. To defekt.

Nie jest to jednak podatność, bo nikt nie może tego wykorzystać, by obniżyć swoje zobowiązanie podatkowe. Niemniej pewne defekty w kodeksie podatkowym są podatnościami. Istniała na przykład podatkowa sztuczka nazywana „Double Irish with a Dutch Sandwich" (dosłownie „podwójny Irlandczyk z holenderską kanapką"). To podatność, która wynikała z interakcji prawa podatkowego w różnych krajach, usunięta wreszcie przez Irlandczyków.

Oto na czym polegała[4]. Amerykańska firma przelewa środki na konto swojej irlandzkiej filii. Ta filia obciąża amerykańską spółkę olbrzymimi tantiemami za sprzedaż amerykańskim klientom. Drastycznie obniża to podatki firmy w Stanach Zjednoczonych, a irlandzkie podatki od tantiem są ustawowo niskie. Następnie, posługując się kruczkiem w irlandzkim prawie podatkowym, firma może przenieść zyski do jednego z rajów podatkowych takich jak Bermudy, Belize, Mauritius czy Kajmany — aby mieć pewność, że dochód pozostanie nieopodatkowany. A teraz dodaj do tego jeszcze jedną irlandzką firmę, tym razem zajmującą się sprzedażą klientom europejskim, również nisko opodatkowaną. Wreszcie trzeba posłużyć się kolejną podatnością, tym razem zakładającą istnienie holenderskiej spółki pośredniczącej, by przelać zyski z powrotem do pierwszej irlandzkiej firmy i dalej do rajów podatkowych. Firmy technologiczne po części są

dobrze przystosowane do wykorzystania tych podatności: mogą przypisywać prawa własności intelektualnej zagranicznym podmiotom zależnym, które następnie przelewają środki pieniężne do rajów podatkowych.

W ten sposób firmy takie jak Google czy Apple unikały płacenia należnych podatków w Stanach Zjednoczonych, mimo że są to firmy amerykańskie. Jest to zdecydowanie nieprzewidziane wykorzystanie praw podatkowych w trzech krajach, choć Irlandia celowo stawiała na swobodne regulacje podatkowe, by przyciągnąć amerykańskie spółki. Może to być bardzo zyskowne dla hakerów. Szacuje się, że amerykańskie spółki uniknęły w ten sposób obciążeń podatkowych w wysokości prawie 200 miliardów dolarów tylko w roku 2017[5]. Do czego doszło kosztem wszystkich innych.

W świecie podatków defekty i podatności nazywa się kruczkami. Atakujący korzystają z nich i nosi to nazwę uchylania się od płacenia podatków. Istnieją też tysiące specjalistów — w świecie bezpieczeństwa komputerowego nazwalibyśmy ich „badaczami black-hat" — którzy biorą na warsztat każdy wiersz kodeksu podatkowego w poszukiwaniu podatności do wykorzystania. To doradcy podatkowi i księgowi.

Umiemy naprawiać podatności w kodzie komputerowym. Po pierwsze, możemy użyć wielu narzędzi do ich wykrywania, zanim kod zostanie ukończony. Po drugie, gdy kod działa już w środowisku produkcyjnym, istnieją liczne sposoby znajdowania podatności i — co najważniejsze — szybkiego ich eliminowania.

Tych samych metod moglibyśmy użyć w stosunku do kodeksu podatkowego. Prawo podatkowe z 2017 r. wprowadziło limit na odliczenia z tytułu podatku dochodowego w przypadku podatków od nieruchomości[6]. Zapisy te weszły w życie dopiero w 2018 r., więc ktoś wpadł na sprytny pomysł, by zapłacić podatek od nieruchomości za rok 2018 awansem w 2017. Tuż przed końcem roku IRS orzekło, kiedy takie działanie jest legalne, a kiedy nie, i w ten sposób usunęło możliwość wykorzystania tej luki. W skrócie: w większości przypadków było to nielegalne.

Często jednak sprawy nie są takie proste. Pewne hacki są zapisane w prawie i nie można ich po prostu oddalić. Uchwalenie jakiejkolwiek ustawy związanej z podatkami to wielki wysiłek, zwłaszcza w Stanach Zjednoczonych, gdzie jest to kwestia ścisłego podziału politycznego, który budzi wielkie kontrowersje. Usuwanie defektu związanego z opodatkowaniem z tytułu dochodu zarobkowego zasiłku dla rodzin poległych żołnierzy zaczęto dopiero w 2021 r. Ściśle rzecz biorąc, Kongres nie naprawił defektu z 2017 r., ale starsze uchybienie, które wchodziło w interakcję z problemem z 2017 r. Pełna poprawka pojawiła się dopiero w 2023 r. (a to prosty przypadek, wszyscy byli zgodni, że to błąd)[7]. Nie mamy możliwości łatania kodeksu podatkowego z taką zwinnością, z jaką łata się oprogramowanie.

Jest i inna droga: podatności się nie usuwa, a z czasem jej używanie staje się normą. Wiele kruczków prawnych przechodzi tę ścieżkę. Czasami IRS je akceptuje. Czasem sądy potwierdzają ich legalność. Być może nie oddają intencji twórców prawa podatkowego,

ale litera prawa na nie zezwala. Niekiedy są nawet legalizowane przez Kongres z mocą wsteczną, gdy znajdą większość zwolenników. Dzięki temu procesowi systemy ewoluują.

Hack podważa intencje systemu. Jakąkolwiek jurysdykcję ma system rządów, blokuje hacki lub zezwala na nie. Czasem zezwala na nie jawnie, innym razem nie robi nic, przez co zezwala na hacki w sposób dorozumiany.

Rozdział 3.

Czym jest system?

Hack przestrzega reguł systemu, ale zadaje kłam ich duchowi i zamiarom.

Aby powstał hack, musi istnieć system reguł do zhakowania. Muszę więc zrobić krok w tył i doprecyzować, co oznacza słowo „system", przynajmniej w takim zakresie, w jakim ja się nim posługuję.

> **System** (definicja)
>
> Złożony proces ograniczony przez zbiór reguł lub norm, stworzony z zamiarem generowania jednego pożądanego rezultatu lub ich większej liczby.

Edytor tekstu, przy którego użyciu napisałem ten akapit, to system: zbiór sygnałów elektronicznych ograniczony zestawem reguł oprogramowania charakterystycznych dla wytwarzania pisma, tak by słowa pojawiały się na ekranie — mój pożądany efekt. Stworzenie tej książki to produkt — rezultat — innego systemu, w którym proces obejmuje projektowanie stron, drukowanie ich, zszywanie we właściwej kolejności, obkładanie całości folią zabezpieczającą i pakowanie do wysyłki. Każdy z tych procesów jest realizowany na podstawie pewnego zbioru reguł. A te dwa systemy, plus kilka innych, dają papierową książkę, którą trzymasz w dłoniach, elektroniczny plik, który otwierasz na czytniku, lub inny plik elektroniczny, który jest odtwarzany w dowolnym systemie audiobooków, z którego korzystasz. Ten schemat jest prawdziwy zarówno, gdy elementy systemu znajdują się pod jednym dachem, jak i wtedy, gdy są rozproszone po całym świecie. Jest prawdziwy także wtedy, gdy rezultaty są realne lub wirtualne, darmowe albo przepłacone, marnie wykonane lub trudno dostępne. Co najmniej jeden system zawsze działa.

Systemy mają zasady. Zwykle zasady prawa, ale czasem to zasady gry, nieformalne reguły grupy lub procesu, nienazwane reguły społeczne. Systemy poznawcze też przestrzegają prawa — prawa natury.

Pamiętaj, że hack jest czymś, na co system pozwala. Pisząc „pozwala", mam na myśli coś bardzo konkretnego. Nie chodzi o to, że pewna czynność jest legalna, dozwolona, społecznie akceptowana czy choćby moralna — choć może mieć jedną z tych cech lub wszystkie jednocześnie. Chodzi o to, że system — przy tym, jak go skonstruowano — nie zapobiega zaistnieniu hacka w ramach ograniczeń systemu. System nie pozwala na hacki celowo, a jedynie incydentalnie i przez przypadek ze względu na sposób, w jaki go zaprojektowano. W systemach technicznych oznacza to przeważnie, że oprogramowanie pozwala na zaistnienie hacka. W systemach społecznych zwykle oznacza to, że zasady — często prawo — regulujące system nie zabraniają jawnie hacków. Dlatego właśnie czasami te hacki nazywa się lukami.

Oznacza to, że hakowanie ma miejsce w systemie, w którym uczestnicy uprzednio zgodzili się — wprost lub w sposób dorozumiany — przestrzegać wspólnego zbioru reguł. Czasami reguły systemu nie są takie same jak prawo, któremu system podlega. Wiem, że to zagmatwane, więc wyjaśnię to na przykładzie. Komputer jest kontrolowany przez zestaw reguł zawierających oprogramowanie uruchomione na tej maszynie. Zhakowanie komputera oznacza podważenie oprogramowania. Ale istnieje także prawo, które wyznacza zakres tego, co można legalnie zrobić. Dla przykładu, w Stanach Zjednoczonych ustawa Computer Fraud and Abuse Act klasyfikuje większość form hakowania jako przestępstwo (zwróć uwagę, co się tu dzieje — to system komputerowy jest hakowany, ale bardziej ogólny system prawny go chroni). Istnieje wiele wątpliwości co do tego, jak ogólne jest to prawo, ale właśnie ze względu na tę ogólność stało się powszechną wyrocznią, według której każde hakowanie jest nielegalne.

Zawodowy sport jest nieustannie hakowany, bo kieruje nim jawny zbiór zasad. Prawo jest hakowane non stop, bo składa się wyłącznie z reguł.

W niektórych systemach oczywiście prawo stanowi reguły albo przynajmniej dyktuje wiele z nich. Jak zobaczysz przy omawianiu hakowania finansów i samego systemu prawnego, literówki czy mylący język ustawy, rozporządzenia czy opinii prawnej mogą otworzyć drzwi nieustannie wykorzystywanym lukom, niezamierzonym przez ustawodawców czy sędziów.

Zwróć uwagę na ważny element: reguły nie muszą być wyrażone wprost. W naszym świecie funkcjonuje wiele systemów, zwłaszcza systemów społecznych, ograniczonych przez normy. Normy są mniej formalne niż reguły, często niepisane, niemniej jednak posiadające moc kierowania zachowaniem. Normy społeczne ograniczają nas cały czas — różne w różnych sytuacjach. Nawet polityką kierują normy w tym samym stopniu co prawo — o tym często przekonujemy się w Stanach Zjednoczonych w ostatnich latach, gdy kolejne normy są łamane.

Moja definicja systemu zawiera słowa „stworzony z zamiarem". Zakłada to istnienie jakiegoś projektanta: kogoś, kto wskazuje pożądany rezultat pracy systemu. To ważna część definicji, ale tylko czasami jest poprawna.

W przypadku komputerów hakowane systemy są umyślnie tworzone przez osobę lub organizację, co oznacza, że haker jest bystrzejszy od projektantów systemu. Tak samo jest w przypadku systemów reguł stworzonych przez pewne ciała rządzące: procedury korporacyjne, zasady sportowe czy traktaty ONZ.

Wiele z systemów, o których będę wspominał w tej książce, nie ma jednego projektanta. Nikt indywidualnie nie zaprojektował wolnego rynku kapitalizmu — w ciągu lat do jego ewolucji przyczyniło się wiele osób. Tak samo było w przypadku procesów demokratycznych: w Stanach Zjednoczonych to połączenie konstytucji, legislacji, orzeczeń sądowych i norm społecznych. A gdy ktoś hakuje system społeczny, polityczny czy gospodarczy, przewyższa sprytem pewną liczbę projektantów systemu, społeczny proces, dzięki któremu system ewoluował, oraz normy, które nim rządzą.

Nasze systemy poznawcze z czasem również ewoluowały, mimo braku projektanta. Ta ewolucja jest normalną częścią opartego na biologii systemu: pojawiają się nowe zastosowania istniejących systemów, stare systemy zyskują nowe cele, systemy niepotrzebne zanikają. Mówi się jednak o „celu" systemu biologicznego — o celu funkcjonowania trzustki czy ciała migdałowatego. Ewolucja to sposób systemu na „zaprojektowanie" samego siebie bez projektanta. Przy takich systemach punktem wyjścia jest ich funkcja w ciele czy ekosystemie — nawet jeśli nikt jej nie zaprojektował.

Hakowanie to naturalne rozszerzenie myślenia systemowego. Systemy rozprzestrzeniają się w wielu sferach naszego życia. Leżą one u podstaw większości złożonych społeczności i same stają się coraz bardziej złożone, podczas gdy rośnie złożoność społeczna. A wykorzystywanie tych systemów — hakowanie — staje się coraz ważniejsze. Ogólnie, jeśli rozumiesz system dobrze i dogłębnie, nie musisz posługiwać się tymi samymi zasadami co wszyscy inni. Możesz szukać wad i pominięć w regułach. Zauważasz, gdzie nie sprawdzają się ograniczenia, które system na Ciebie nakłada. Naturalnie hakujesz system. A jeśli jesteś bogaty i masz władzę, prawdopodobnie nie poniesiesz żadnych konsekwencji.

Rozdział 4.

Cykl życia hakowania

W żargonie nauk komputerowych na hack składają się dwa elementy: podatność i exploit.

Podatność to funkcjonalność w systemie, która pozwala na zaistnienie hacka. W systemach komputerowych to wada. To błąd lub niedopatrzenie: w projekcie, specyfikacji czy samym kodzie. Może to być drobnostka taka jak niedomknięty nawias lub coś tak poważnego, jak właściwość architektury oprogramowania. To czynnik, który sprawia, że hack działa. Exploit to mechanizm wykorzystania podatności.

Jeśli logujesz się na stronie, która pozwala, aby Twoja nazwa użytkownika i hasło były przesyłane w sieci bez enkrypcji — to podatność. Exploit to program, który nasłuchuje połączeń internetowych, rejestruje Twoją nazwę użytkownika oraz hasło i wykorzystuje je później, by uzyskać dostęp do Twojego konta. Jeśli jakieś oprogramowanie umożliwia Ci przeglądanie prywatnych plików innego użytkownika, to podatność. Exploit to program, który pozwoli mi je zobaczyć. Jeśli zamek w drzwiach da się otworzyć bez klucza — to także podatność. Exploit to łom czy inne narzędzie, którego trzeba będzie użyć, by sforsować zamek.

Posłużę się przykładem komputerowym — EternalBlue. To kryptonim, jaki NSA nadała exploitowi na systemy operacyjne Windows, z którego agencja korzystała przez co najmniej pięć lat przed rokiem 2017, gdy Rosjanie wykradli go NSA. EternalBlue wykorzystuje podatność w użytej przez Microsoft implementacji protokołu Server Message Block (SMB), który kontroluje komunikację serwera i klienta. Ze względu na sposób napisania kodu SMB wysłanie uważnie przygotowanego pakietu danych przez internet do komputera z systemem Windows pozwalało atakującemu wykonać dowolny kod na komputerze ofiary i w ten sposób przejąć nad nim kontrolę. Generalnie NSA była w stanie wykorzystywać EternalBlue, by zdalnie zarządzać każdym komputerem z systemem Windows podłączonym do internetu.

Kilka różnych rodzajów ludzi — każdy z innym zestawem kompetencji — może być zaangażowanych w powstanie hacków i wszystkich ich nazywa się hakerami, co jest niezwykle mylące. Po pierwsze, istnieją kreatywni hakerzy, którzy wykorzystują swoją

ciekawość i wiedzę do odkrycia hacka i stworzenia exploita. W przypadku EternalBlue to specjalista nauk komputerowych z NSA odkrył podatność. W przypadku sztuczki „Double Irish" był to jakiś ekspert międzynarodowego prawa podatkowego, który do bólu zgłębiał różne zapisy i ich wzajemne zależności.

Po drugie, są osoby, które posługują się powstałym exploitem w praktyce. W NSA to pracownik, który zrealizował atak exploitem na konkretny cel. W firmie księgowej był to ten księgowy, który posłużył się irlandzkim i holenderskim prawem i zastosował strategię unikania opodatkowania danej firmy. Haker, który wykonuje tego typu hack, wykorzystuje cudzą kreatywność. W świecie technologicznym szyderczo przezywamy takie osoby *script kiddies* (z ang. „skryptowymi dzieciaczkami"). Nie są wystarczająco bystrzy czy kreatywni, żeby znaleźć nowe hacki, ale potrafią uruchamiać programy komputerowe — skrypty — które automatycznie spuszczają ze smyczy rezultaty czyjejś kreatywności.

Wreszcie jest też organizacja czy osoba, na rzecz której to wszystko się dzieje. Możemy więc mówić, że NSA hakuje obcą sieć albo Rosja hakuje USA czy Google hakuje prawo podatkowe.

To wszystko jest ważne, ponieważ będę wracał do omawiania tego, jak bogaci i silni hakują system. Nie mówię, że pieniądze i władza czynią z kogoś technicznie lepszego hakera, dają mu po prostu lepszy dostęp do hakowania. Jak w przypadku Rosji, Stanów Zjednoczonych czy Google ktoś bogaty i możny będzie w stanie zatrudnić odpowiednich specjalistów, by z powodzeniem zhakować system.

Hacki są zarówno wynajdowane, jak i odkrywane. Ściśle rzecz ujmując, podatności się odkrywa, a exploity to wynalazki. Stosuje się oba pojęcia, ale ja wolę koncepcję „odkrycia", bo podkreśla ona istnienie uśpionej możliwości w systemie, nawet zanim ktokolwiek zda sobie z niej sprawę.

To, co się stanie, gdy hack zostanie odkryty, zależy od tego, kto tego dokona. Ogólnie osoba lub organizacja, która wymyśli hack, użyje go dla swojej korzyści. W kontekście systemów komputerowych może to być haker-przestępca lub państwowa służba wywiadowcza taka jak NSA — lub cokolwiek pomiędzy. W zależności od tego, kto zaczyna wykorzystywać hack i w jaki sposób, inni się o tym dowiedzą lub nie, a inne osoby mogą go odkryć niezależnie. Ten proces może trwać tygodnie, miesiące, a nawet lata.

W przypadku innych systemów użyteczność hacka zależy od tego, jak często i na ile jawnie jest on wykorzystywany. Trudna do zrozumienia podatność w systemie bankowym może być od czasu do czasu wykorzystywana przez przestępców i mogą minąć całe lata, zanim bank się zorientuje, co się dzieje. Skuteczny hack systemu podatkowego będzie w powszechnym użyciu, bo ktokolwiek jest jego właścicielem, prawdopodobnie sprzedaje swoją wiedzę[1]. Sprytna manipulacja psychologiczna może wyjść na jaw, gdy wystarczająco dużo osób o niej mówi — lub może funkcjonować w ukryciu przez pokolenia.

Ostatecznie system zareaguje. Hack może zostać zneutralizowany, jeśli leżąca u jego podstaw podatność zostanie załatana. Rozumiem przez to aktualizację systemu w celu usunięcia podatności lub uniemożliwienia jej wykorzystania w inny sposób. Nie ma podatności, nie ma hacka. Proste.

Zakłada się przy tym, że istnieje ktoś, kto kontroluje docelowy system i nadzoruje proces aktualizacji. To oczywiste, gdy mówimy o systemie operacyjnym Microsoft Windows lub każdym innym dużym pakiecie oprogramowania i stojącymi za tym programistami. Firmy takie jak Microsoft i Apple wytrenowały się w naprawianiu swoich systemów w zakresie bezpieczeństwa.

Ten proces działa także w przypadku oprogramowania wolnoźródłowego i znajdującego się w domenie publicznej — zwykle związana jest z nim jakaś organizacja lub osoba, a kod może przejrzeć każdy. Gorzej sprawa ma się z tanimi urządzeniami IoT, z których wiele jest projektowanych za granicą, przy minimalnym marginesie dochodu, przez zespoły, które rozwiązuje się natychmiast po zakończeniu prac programistycznych. Co gorsza, wielu urządzeń internetu rzeczy w ogóle nie da się załatać. Nie jest to kwestia braku kompetencji. Po prostu wiele tego typu sprzętów wbudowuje swój kod w fizyczne urządzenie, a nie w oprogramowanie, co sprawia, że zupełnie nie da się go zmienić — nawet jeśli chodzi o naprawę. Problem pogarsza się, gdy firmy likwidują linie produkcyjne lub upadają, pozostawiając za sobą miliony osieroconych urządzeń podłączonych do internetu.

W przypadku systemów technicznych hacki łata się często tuż po ich odkryciu. Inaczej sprawa ma się w przypadku systemów społecznych, które omawiam w tej książce. Zaktualizowanie kodeksu podatkowego — dla przykładu — wymaga przeprowadzenia wieloletniego procesu legislacyjnego. Osoby czerpiące korzyści z hacka mogą lobbować przeciwko wszelkim zmianom w prawie. Może rozgorzeć słuszna debata na temat tego, czy hack rzeczywiście jest korzystny dla społeczeństwa czy nie. A jak przekonasz się na kolejnych stronach książki, bogaci i silni mają nieproporcjonalnie duży wpływ na efekty realizacji nominalnie demokratycznego procesu.

Jeśli system nie zostanie załatany, to hack wnika w reguły systemu. Staje się nową normą. Tak więc coś, co początkowo jest hackiem, szybko może stać się standardową działalnością operacyjną. Taką trajektorię obiera wiele nieetycznych hacków, które będę opisywał w tej książce.

Rozdział 5.

Powszechność hakowania

Niezależnie od tego, jak zamknięty jest system, podatności zawsze powstaną, a hacki zawsze będą możliwe. W 1930 r. pochodzący z Austro-Węgier matematyk Kurt Gödel udowodnił, że wszystkie systemy matematyczne są albo niezupełne, albo niespójne. Mam taką teorię, że to twierdzenie jest prawdziwe także w szerszym kontekście. Każdy system będzie miał niejasności, niespójności, niedopatrzenia i zawsze będzie można je wykorzystać. Systemy zasad w szczególności muszą uważać na delikatną równowagę między kompletnością i zrozumiałością wśród wielu ograniczeń ludzkiego języka i zrozumienia. To w połączeniu z naturalnym dążeniem ludzi do naciskania na granice i testowania limitów oraz nieuniknionymi podatnościami składa się na rzeczywistość, w której wszystko jest ciągle hakowane.

Club Penguin był grą online produkowaną przez Disneya, funkcjonującą w latach 2005 – 2017. Dzieci rozmawiające z nieznajomymi to zawsze wielka obawa, więc Disney stworzył tryb Ultimate Safe Chat (najbezpieczniejszy czat), w którym niemożliwe było pisanie własnych wypowiedzi, a komunikacja użytkowników była ograniczona do zbioru predefiniowanych zdań. Pomysł polegał na tym, by chronić dzieci przed przypadkowymi rozmowami z realnymi lub wyobrażonymi rozmówcami o złych zamiarach. A dzieci, jak to dzieci, chciały ze sobą rozmawiać. Zhakowały więc to ograniczenie przez użycie pozycji awatara do komunikowania liter czy cyfr.

Dzieci to naturalni hakerzy. Nie rozumieją intencji, więc w konsekwencji nie widzą ograniczeń systemu w ten sam sposób co dorośli. Patrzą na problemy holistycznie i mogą natknąć się na hack, nie zdając sobie sprawy z tego, co robią. Nie są tak ograniczone przez normy i z całą pewnością nie rozumieją prawa w ten sam sposób. Testowanie zasad to oznaka samodzielności.

Podobnie jak Club Penguin wiele innych gier online dla dzieci próbowało nałożyć ograniczenia na wypowiedzi, by zminimalizować przypadki zastraszania, nękania czy żerowania na naiwności dzieci. Dzieci zhakowały je wszystkie[1]. Sztuczki mające na celu wymknięcie się moderatorom czy filtrom przekleństw obejmują celowe błędy ortograficzne, takie jak *phuq*, rozdzielanie kluczowej informacji na kilka wiadomości, tak by

żadna nie łamała zasad, czy posługiwanie się akrostychami. Niektóre strony uniemożliwiały dzieciom używanie cyfr, więc dzieci odpowiadały, używając zamiast nich tak samo lub podobnie brzmiących słów, na przykład: *won* zamiast *one*, *too* zamiast *two*, *tree* zamiast *three* itd. Podobnie było z obelgami: *lose her* zamiast *loser* czy *stew putt* zamiast *stupid*.

Szkoły próbowały ograniczać uczniom możliwości korzystania z komputerów udostępnianych przez placówki, ale dzieci odpowiedziały złamaniem wszystkich tych restrykcji. Skuteczne hacki przekazują koleżankom i kolegom. Po tym, jak pewien okręg szkolny ograniczył adresy stron, jakie uczniowie mogli odwiedzać, uczniowie zorientowali się, że jeśli użyją sieci VPN, to ograniczeń nie da się wykryć i wymóc. Gdy z kolei inny okręg zablokował aplikacje do rozmów, uczniowie wpadli na pomysł, że mogą rozmawiać przez współdzielony dokument Google Docs.

Ten hack nie jest akurat nowy. Ma nawet nazwę: foldering[2]. W różnych przypadkach korzystali z niego: generał Petraeus, Paul Manafort i terroryści odpowiedzialni za ataki na WTC. Wszyscy oni zdawali sobie sprawę, że mogą uniknąć nadzoru komunikacji, jeśli wraz z pozostałymi konspiratorami będą dzielić konto e-mail i pisać do siebie wiadomości, zachowując je jako kopie robocze, nigdy ich nie wysyłając.

Z dzieciństwa pamiętam hacki związane z obchodzeniem reguł systemu telefonicznego. Jeśli jesteś za młody, by pamiętać, jak to działało — wyjaśnię. Osoba telefonująca łączyła się z operatorem, mówiła mu, kim jest, i zgłaszała potrzebę rozmowy na koszt odbiorcy. Operator przełączał połączenie i pytał tego, kto je odebrał, czy akceptuje płatną rozmowę przychodzącą od nadawcy. Takie rozmowy wiązały się z wysoką opłatą manipulacyjną. Niemniej jednak ponieważ operator inicjował połączenie, można było przekazać informację, zanim jakakolwiek opłata została pobrana. Tak więc wykonywaliśmy połączenie na koszt odbiorcy, odbiorca pytał drugą stronę — przeważnie naszych rodziców — czy przyjmują rozmowę na swój koszt. Nasi rodzice odmawiali i oddzwaniali do nas po zwykłych, niższych stawkach. Tego typu rozwiązanie można było jeszcze usprawnić. W niektórych rodzinach funkcjonowały listy imion podawanych operatorowi jako zakodowana wiadomość: „Bruce" znaczyło „dojechałem bezpiecznie", „Steve" oznaczało „oddzwoń" itd. (operator nie miał pojęcia, jak naprawdę nazywa się dzwoniący). Nawet dziś ludzie mają sposoby na reguły opłat telefonicznych. W Nigerii nazywa się to „flashing" — dzwonienie do kogoś i rozłączanie się, zanim ta osoba odbierze telefon[3]. To zjawisko cieszyło się olbrzymią popularnością w Indiach ok. 2010 r. ze względu na gigantyczną różnicę w opłatach za połączenia stacjonarne i komórkowe[4]. Zamysł wszystkich tych hacków polegał na podważeniu systemu telefonicznego, aby wymieniać informacje bez konieczności płacenia za ten przywilej.

Edukacja domowa w czasie pandemii Covid-19 wyzwoliła hakerskie inklinacje w wielu uczniach[5]. Jeden z uczniów zmienił swoje imię na „Reconnecting…" (z ang. łączenie) i wyłączył kamerę, tak by wyglądało to na problemy z łącznością internetową. W marcu

2020 r., w pierwszych miesiącach pandemii, wprowadzono lockdown w chińskim mieście Wuhan. Szkoły przeszły na zdalny tryb nauczania, a uczniowie zaczęli masowo wystawiać jednogwiazdkowe oceny aplikacji DingTalk do prac domowych w nadziei, że zostanie ona usunięta ze sklepów z aplikacjami (pomysł nie zadziałał)[6].

Systemy zwykle są sztywne i ograniczone przez reguły. Wyznaczają to, co użytkownik może z nimi zrobić, ale niezmiennie niektórzy użytkownicy chcą robić coś innego. Więc hakujemy. Gdy zdobędziesz pewną wiedzę o tym, czym są systemy i jak funkcjonują, zaczniesz dostrzegać je wszędzie. A potem zaczniesz wszędzie dostrzegać hakowanie.

Nie oznacza to, że wszystkie systemy są popsute. Przypomnij sobie Gödla[7]. Wśród prawników powtarzane jest powiedzenie: „Wszystkie umowy są niekompletne". Umowa obowiązuje nie dlatego, że ściśle uniemożliwia stronom podważenie swoich zamiarów, ale dlatego, że większość luk wypełniają zaufanie i dobre intencje — a jeśli sprawy pójdą nie tak, do dyspozycji pozostaje cały repertuar arbitrażu i sądownictwa. Może to brzmieć naiwnie i idealistycznie, ale to właśnie systemy oparte na zaufaniu sprawiają, że społeczeństwo funkcjonuje[8]. Nie wymagamy w umowach bezwzględnej ochrony, ponieważ: (1) nie da się jej osiągnąć, (2) każda próba będzie zbyt długa i niezgrabna, (3) tak naprawdę wcale jej nie potrzebujemy.

Tak samo rzecz ma się z bardziej ogólnymi systemami. Tym, co sprawia, że system działa, nie jest przekonanie o braku podatności. To to samo połączenie zaufania i możliwości rozstrzygania sporów. Choć będę mówił o hackach i hakowaniu, to w dużej mierze stanowią one wyjątki. Większość ludzi nie hakuje systemów, a większość systemów przeważnie działa całkiem nieźle. Słusznie ufamy, że większość ludzi nie hakuje systemów. I mamy systemy do walki z hackami, gdy takie się pojawiają. To odporność. Dzięki temu społeczeństwo funkcjonuje. Tak jako ludzkość radziliśmy sobie z hakowaniem przez tysiąclecia.

Nie wszystkie systemy dają się hakować tak samo. W miarę jak będziesz się zagłębiać w tę książkę, poznasz cechy systemów, które sprawiają, że są one mniej lub bardziej podatne na hakowanie. Złożone systemy o wielu zasadach są szczególnie narażone po prostu dlatego, że istnieje więcej możliwości nieprzewidzianych i niezamierzonych konsekwencji. Z pewnością sprawdza się to w zakresie systemów komputerowych — pisałem kiedyś, że złożoność jest największym wrogiem bezpieczeństwa[9] — a także systemów takich jak kodeks podatkowy, regulacje finansowe i sztuczna inteligencja. Ludzkie systemy ujęte w ramy bardziej elastycznych norm społecznych i zasad są bardziej narażone na hakowanie, ponieważ pozostają bardziej otwarte na interpretacje, a co za tym idzie, mają więcej przestrzeni dla luk.

Z drugiej strony systemy niekrytyczne, działające na mniejszą skalę czy wręcz marginalnie — a więc potencjalnie bardziej eksperymentalne i słabiej doprecyzowane — wyrządzą mniej szkód, gdy ulegną awarii, więc prawdopodobnie korzystnie będzie pozwolić im ewoluować przez hakowanie, a nie martwić się o to, co może pójść źle.

Niewiele jest wartości, a za to dużo niebezpieczeństw w dopuszczeniu do hakowania procesu projektowania i budowy mostu. Pomyłka może mieć katastrofalne skutki. Więcej można powiedzieć o pozwoleniu na hakowanie, które skutkuje wspaniałymi, nieprzewidzianymi sposobami używania internetu.

Hakowanie to naturalna część kondycji ludzkiej. To powszechny, i jak się okaże, ewolucyjny proces: stały, niekończący się, będący w stanie stworzyć — jak ujałby to Darwin — „najpiękniejsze i najwspanialsze formy" lub formy najdziwniejsze i okropne.

Ten, kto ma władzę, nie musi nikogo za nic przepraszać.

Niccolò Machiavelli, *Książę*

Pomyśl o hakowaniu jako o metodzie wykorzystania luk w regułach systemu. Zauważ, jak bezlitośnie hakowane są praktycznie wszystkie systemy, na których opiera się funkcjonowanie społeczeństwa. Spójrz na prawo jak na skomplikowany system zawierający podatności, dzięki którym można unikać na przykład opodatkowania. W wyszukiwaniu luk specjalizują się hakerzy, w tym wypadku księgowi, doradcy podatkowi i prawnicy. Jeśli pójdziesz tym tropem, przekonasz się, że we wszystkich ważnych systemach pozostawia się luki, które służą wtajemniczonym do naginania reguł i czerpania korzyści kosztem innych.

Dzięki tej książce spojrzysz na hakowanie jak na zajęcie możnych, którzy podważają obowiązujące wszystkich innych reguły. Różnego rodzaju hacki pozwalają tym osobom na czerpanie korzyści i równocześnie utrwalają ich bogactwo i władzę. Przekonasz się, że hakowanie to główny powód, dla którego czujemy się bezradni wobec interesów korporacji i wpływowych ludzi. Dodatkowo stoimy u progu ery nowych hacków z użyciem systemów AI. Sztuczna inteligencja zacznie hakować nie tylko nasze komputery, ale także rządy, rynki, a nawet umysły. Czas najwyższy nauczyć się rozpoznawać hacki i poznać sposoby, jak się przed nimi bronić. Dzięki temu być może uda się stworzyć bardziej sprawiedliwy świat.

I właśnie o tym jest ta książka.

BRUCE SCHNEIER jest światowej klasy specjalistą w dziedzinie cyberbezpieczeństwa. Pracował w Departamencie Obrony USA i w AT&T. Jest autorem kilku bestsellerów, między innymi książki *Dane i Goliat*. Regularnie pisze dla „The Guardian" i „Wired". Aktywnie działa na rzecz ochrony prywatności, często występuje jako prelegent na konferencjach branżowych.

Helion
helion.pl
HELION S.A.
ul. Kościuszki 1c
44-100 Gliwice
tel.: 32 230 98 63
helion@helion.pl
KOD KORZYŚCI
Sięgnij po więcej!
ISBN 978-83-289-1101-7
9 788328 911017
Cena: 54,90 zł