Chapter 8

Economics

The great fortunes of the information age lie in the hands of companies that have established proprietary architectures that are used by a large installed base of locked-in customers.
– CARL SHAPIRO AND HAL VARIAN

There are two things I am sure of after all these years: there is a growing societal need for high assurance software, and market forces are never going to provide it.
– EARL BOEBERT

The law locks up the man or woman
Who steals the goose from off the common
But leaves the greater villain loose
Who steals the common from the goose.
– TRADITIONAL, 17th CENTURY

8.1 Introduction

Round about 2000, we started to realise that many security failures weren't due to technical errors so much as to wrong incentives: if the people who guard a system are not the people who suffer when it fails, then you can expect trouble. In fact, security mechanisms are often designed deliberately to shift liability, which can lead to even worse trouble.

Economics has always been important to engineering, at the raw level of cost accounting; a good engineer was one who could build a bridge safely with a thousand tons of concrete when everyone else used two thousand tons. But the perverse incentives that arise in complex systems with multiple owners make economic questions both more important and more subtle for the security engineer. Truly global-scale systems like the Internet arise from the actions of millions of independent principals with divergent interests; we hope that reasonable global outcomes will result from selfish local actions. The outcome we get is typically a

market equilibrium, and often a surprisingly stable one. Attempts to make large complex systems more secure, or safer, will usually fail if this isn't understood. At the macro level, cybercrime patterns have been remarkably stable through the 2010s even though technology changed completely, with phones replacing laptops, with society moving to social networks and servers moving to the cloud. Network insecurity is somewhat like air pollution or congestion, in that people who connect insecure machines to the Internet do not bear the full consequences of their actions while people who try to do things right suffer the side-effects of others' carelessness.

In general, people won't change their behaviour unless they have an incentive to. If their actions take place in some kind of market, then the equilibrium will be where the forces pushing and pulling in different directions balance each other out. But markets can fail; the computer industry has been dogged by monopolies since its earliest days. The reasons for this are now understood, and their interaction with security is starting to be.

Security economics has developed rapidly as a discipline since the early 2000s. It provides valuable insights not just into 'security' topics such as privacy, bugs, spam, and phishing, but into more general areas of system dependability. For example, what's the optimal balance of effort by programmers and testers? (For the answer, see section 8.6.3 below.) It also enables us to analyse many important policy problems – such as the costs of cybercrime and the most effective responses to it. And when protection mechanisms are used to limit what someone can do with their possessions or their data, questions of competition policy and consumer rights follow – which we need economics to analyse. There are also questions of the balance between public and private action: how much of the protection effort should be left to individuals, and how much should be borne by vendors, regulators or the police? Everybody tries to pass the buck.

In this chapter I first describe how we analyse monopolies in the classical economic model, how information goods and services markets are different, and how network effects and technical lock-in make monopoly more likely. I then look at asymmetric information, another source of market power. Next is game theory, which enables us to analyse whether people will cooperate or compete; and auction theory, which lets us understand the working of the ad markets that drive much of the Internet – and how they fail. These basics then let us analyse key components of the information security ecosystem, such as the software patching cycle. We also get to understand why systems are less reliable than they should be: why there are too many vulnerabilities and why too few cyber-crooks get caught.

8.2 Classical economics

Modern economics is an enormous field covering many different aspects of human behaviour. The parts of it that have found application in security so far are largely drawn from microeconomics, game theory and behavioral economics. In this section, I'll start with a helicopter tour of the most relevant ideas from microeconomics. My objective is not to provide a tutorial on economics, but to get across the basic language and ideas, so we can move on to discuss security

economics.

The modern subject started in the 18th century when growing trade changed the world, leading to the industrial revolution, and people wanted to understand what was going on. In 1776, Adam Smith's classic *'The Wealth of Nations'* [1446] provided a first draft: he explained how rational self-interest in a free market leads to progress. Specialisation leads to productivity gains, as people try to produce something others value to survive in a competitive market. In his famous phrase, "It is not from the benevolence of the butcher, the brewer, or the baker, that we can expect our dinner, but from their regard to their own interest." The same mechanisms scale up from a farmers' market or small factory to international trade.

These ideas were refined by nineteenth-century economists; David Ricardo clarified and strengthened Smith's arguments in favour of free trade, while Stanley Jevons, Léon Walras and Carl Menger built detailed models of supply and demand. One of the insights from Jevons and Menger is that the price of a good, at equilibrium in a competitive market, is the marginal cost of production. When coal cost nine shillings a ton in 1870, that didn't mean that every mine dug coal at this price, merely that the marginal producers – those who were only just managing to stay in business – could sell at that price. If the price went down, these mines would close; if it went up, even more marginal mines would open. That's how supply responded to changes in demand. (It also gives us an insight into why so many online services nowadays are free; as the marginal cost of duplicating information is about zero, lots of online businesses can't sell it and have to make their money in other ways, such as from advertising. But we're getting ahead of ourselves.)

By the end of the century Alfred Marshall had combined models of supply and demand in markets for goods, labour and capital into an overarching 'classical' model in which, at equilibrium, all the excess profits would be competed away and the economy would be functioning efficiently. By 1948, Kenneth Arrow and Gérard Debreu had put this on a rigorous mathematical foundation by proving that markets give efficient outcomes, subject to certain conditions, including that the buyers and sellers have full property rights, that they have complete information, that they are rational and that the costs of doing transactions can be neglected.

Much of the interest in economics comes from the circumstances in which one or more of these conditions aren't met. For example, suppose that transactions have side-effects that are not captured by the available property rights. Economists call these *externalities*, and they can be either positive or negative. An example of a positive externality is scientific research, from which everyone can benefit once it's published. As a result, the researcher doesn't capture the full benefit of their work, and we get less research than would be ideal (economists reckon we do only a quarter of the ideal amount of research). An example of a negative externality is environmental pollution; if I burn a coal fire, I get the positive effect of heating my house but my neighbour gets the negative effect of smell and ash, while everyone shares the negative effect of increased CO_2 emissions.

Externalities, and other causes of market failure, are of real importance to the

computer industry, and to security folks in particular, as they shape many of the problems we wrestle with, from industry monopolies to insecure software. Where one player has enough power to charge more than the market clearing price, or nobody has the power to fix a common problem, then markets alone may not be able to sort things out. Strategy is about acquiring power, or preventing other people having power over you; so the most basic business strategy is to acquire market power in order to extract extra profits, while distributing the costs of your activity on others to the greatest extent possible. Let's explore that now in more detail.

8.2.1 Monopoly

As an introduction, let's consider a textbook case of monopoly. Suppose we have a market for apartments in a university town, and the students have different incomes. We might have one rich student able to pay $4000 a month, maybe 300 people willing to pay at least $2000 a month, and (to give us round numbers) at least 1000 prepared to pay at least $1000 a month. That gives us the *demand curve* shown in Figure 8.1 below.

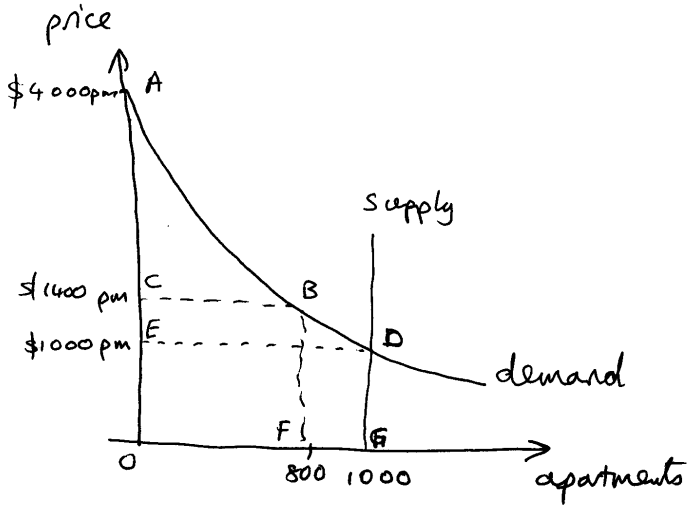

Figure 8.1: the market for apartments

So if there are 1000 apartments being let by many competing landlords, the market-clearing price will be at the intersection of the demand curve with the vertical supply curve, namely $1000. But suppose the market is rigged – say the landlords have set up a cartel, or the university makes its students rent through a tied agency. A monopolist landlord examines the demand curve, and notices that if he rents out only 800 apartments, he can get $1400 per month for each of them. Now 800 times $1400 is $1,120,000 per month, which is more than the million dollars a month he'll make from the market price at $1000. (Economists would say that his 'revenue box' is the box CBFO rather than EDGO in figure

8.1.) So he sets an artificially high price, and 200 apartments remain empty.

This is clearly inefficient, and the Italian economist Vilfredo Pareto invented a neat way to formalise this. A *Pareto improvement* is any change that would make some people better off without making anyone else worse off, and an allocation is *Pareto efficient* if there isn't any Pareto improvement available. Here, the allocation is not efficient, as the monopolist could rent out one empty apartment to anyone at a lower price, making both him and them better off. Now Pareto efficiency is a rather weak criterion; both perfect communism (everyone gets the same income) and perfect dictatorship (the king gets the lot) are Pareto-efficient. In neither case can you make anyone better off without making someone else worse off! Yet the simple monopoly described here is not efficient even in this very weak sense.

So what can the monopolist do? There is one possibility – if he can charge everyone a different price, then he can set each student's rent at exactly what they are prepared to pay. We call such a landlord a *price-discriminating monopolist*; he charges the rich student exactly $4000, and so on down to the 1000th student whom he charges exactly $1000. The same students get apartments as before, yet almost all of them are worse off. The rich student loses $3000, money that he was prepared to pay but previously didn't have to; economists refer to this money he saved as *surplus*. The discriminating monopolist manages to extract all the consumer surplus.

Merchants have tried to price-discriminate since antiquity. The carpet seller in Istanbul who expects you to haggle down his price is playing this game, as is an airline selling first, business and cattle class seats. The extent to which firms can charge people different prices depends on a number of factors, principally their *market power* and their *information asymmetry*. Market power is a measure of how close a merchant is to being a monopolist; under monopoly the merchant is a *price setter*, while under perfect competition he is a *price taker* who has to accept whatever price the market establishes. Merchants naturally try to avoid this. Information asymmetry can help them in several ways. A carpet seller has much more information about local carpet prices than a tourist who's passing through, and who won't have the time to haggle in ten different shops. So the merchant may prefer to haggle rather than display fixed prices. An airline is slightly different. Thanks to price-comparison sites, its passengers have good information on base prices, but if it does discount to fill seats, it may be able to target its offers using information from the advertising ecosystem. It can also create its own loyalty ecosystem by offering occasional upgrades. Technology tends to make firms more like airlines and less like small carpet shops; the information asymmetry isn't so much whether you know about average prices, as what the system knows about you and how it locks you in.

Monopoly can be complex. The classic monopolist, like the landlord or cartel in our example, may simply push up prices for everyone, resulting in a clear loss of consumer surplus. Competition law in the USA looks for welfare loss of this kind, which often happens where a cartel operates price discrimination. During the late 19th century, railroad operators charged different freight rates to different customers, depending on how profitable they were, how perishable their goods were and other factors – basically, shaking them all down according to their ability to pay. This led to massive resentment and to railway regulation.

In the same way, telcos used to price-discriminate like crazy; SMSes used to cost a lot more than voice, and voice a lot more than data, especially over distance. This led to services like Skype and WhatsApp which use data services to provide cheaper calls and messaging, and also to net neutrality regulation in a number of countries. This is still a tussle space, with President Trump's appointee at the FCC reversing many previous net neutrality rulings.

However, many firms with real market power like Google and Facebook give their products away free to most of their users, while others, like Amazon (and Walmart), cut prices for their customers. This challenges the traditional basis that economists and lawyers used to think about monopoly, in the USA at least. Yet there's no doubt about monopoly power in tech. We may have gone from one dominant player in the 1970s (IBM) to two in the 1990s (Microsoft and Intel) and a handful now (Google, Facebook, Amazon, Microsoft, maybe Netflix) but each dominates its field; although ARM managed to compete with Intel, there has been no new search startup since Bing in 2009 (whose market share is slipping), and no new social network since Instagram in 2011 (now owned by Facebook). So there's been a negative effect on innovation.

To understand what's going on, we need to dive more deeply into how information monopolies work.

8.3 Information economics

The information and communications industries are different from traditional manufacturing in a number of ways, and among the most striking is that these markets have been very concentrated for generations. Even before computers came along, newspapers tended to be monopolies, except in the biggest cities. Much the same happened with railways, and before that with canals. When electrical tabulating equipment came along in the late 19th century, it was dominated by NCR, until a spin-off from NCR's Manhattan sales office called IBM took over. IBM dominated the computer industry in the 1960s and 70s, then Microsoft came along and took pole position in the 90s. Since then, Google and Facebook have come to dominate advertising, Apple and Google sell phone operating systems, ARM and Intel do CPUs, while many other firms dominate their own particular speciality. Why should this be so?

8.3.1 Why information markets are different

Recall that in a competitive equilibrium, the price of a good should be its marginal cost of production. But for information that's almost zero! That's why there is so much free stuff online; zero is its fair price. If two or more suppliers compete to offer an operating system, or a map, or an encyclopedia, that they can duplicate for no cost, then they will keep on cutting their prices without limit. Take for example encyclopedias; the Britannica used to cost $1,600 for 32 volumes; then Microsoft brought out Encarta for $49.95, forcing Britannica to produce a cheap CD edition; and now we have Wikipedia for free [1393]. One firm after another has had to move to a business model in which the goods are given away free, and the money comes from advertising or

in some parallel market. And it can be hard to compete with services that are free, or are so cheap it's hard to recoup the capital investment you need to get started. So other industries with high fixed costs and low marginal costs tend to be concentrated – such as newspapers, airlines and hotels.

Second, there are often *network externalities*, whereby the value of a network grows more than linearly in the number of users. Networks such as the telephone and email took some time to get going because at the start there were only a few other enthusiasts to talk to, but once they passed a certain threshold in each social group, everyone needed to join and and network became mainstream. The same thing happened again with social media from the mid-2000s; initially there were 40–50 startups doing social networks, but once Facebook started to pull ahead, suddenly all young people had to be there, as that was where all your friends were, and if you weren't there then you missed out on the party invitations. This *positive feedback* is one of the mechanisms by which network effects can get established. It can also operate in a *two-sided market* which brings together two types of user. For example, when local newspapers got going in the nineteenth century, businesses wanted to advertise in the papers with lots of readers, and readers wanted papers with lots of small ads so they could find stuff. So once a paper got going, it often grew to be a local monopoly; it was hard for a competitor to break in. The same thing happened when the railways allowed the industrialisation of agriculture; powerful firms like Cargill and Armour owned the grain elevators and meat-packers, dealing with small farmers on one side and the retail industry on the other. We saw the same pattern in the 1960s when IBM mainframes dominated computing: firms used to develop software for IBM as they'd have access to more users, while many users bought IBM because there was more software for it. When PCs came along, Microsoft beat Apple for the same reason; and now that phones are replacing laptops, we see a similar pattern with Android and iPhone. Another winner was eBay in the late 1990s: most people wanting to auction stuff will want to use the largest auction, as it will attract more bidders. Network effects can also be negative; once a website such as Myspace starts losing custom, negative feedback can turn the loss into a rout.

Third, there's often lock-in stemming from *interoperability*, or a lack thereof. Once a software firm commits to using a platform such as Windows or Oracle for its product, it can be expensive to change. This has both technical and human components, and the latter are often dominant; it's cheaper to replace tools than to retrain programmers. The same holds for customers, too: it can be hard to close a sale if they not only have to buy new software and convert files, but retrain their staff too. These *switching costs* deter migration. Earlier platforms where interoperability mattered included the telephone system, the telegraph, mains electricity and even the railways.

These three features separately – low marginal costs, network externalities and technical lock-in – can lead to industries with dominant firms; together, they are even more likely to. If users want to be compatible with other users (and vendors of complementary products such as software) then they will logically buy from the vendor they expect to win the biggest market share.

8.3.2 The value of lock-in

There is an interesting result, due to Carl Shapiro and Hal Varian: that the value of a software company is the total lock-in (due to both technical and network effects) of all its customers [1393]. To see how this might work, consider a firm with 100 staff each using Office, for which it has paid $150 per copy. It could save this $15,000 by moving to a free program such as LibreOffice, so if the costs of installing this product, retraining its staff, converting files and so on – in other words the total switching costs – were less than $15,000, it would switch. But if the costs of switching were more than $15,000, then Microsoft would put up its prices.

As an example of the link between lock-in, pricing and value, consider how prices changed over a decade. In the second edition of this book, this example had the cost of Office as $500; since then, cloud-based services that worked just like Office, such as Google Docs, cut the costs of switching – so Microsoft had to slash its prices. As I write in 2019, I see standalone Office for sale at prices ranging between $59.99 and £124. Microsoft's response since 2013 has been trying to move its customers to an online subscription service (Office365) which costs universities a few tens of pounds per seat depending on what options they choose and how good they are at negotiating, while Google is also trying to move organisations away from their free services to paid G Suite versions that cost about the same. Charging $30 a year for an online service is better business than charging $60 for a program that the customer might use for five years or even seven. There's a new form of lock-in, namely that the cloud provider now looks after all your data.

Lock-in explains why so much effort gets expended in standards wars and antitrust suits. It also helps explain the move to the cloud (though cost cutting is a bigger driver). It's also why so many security mechanisms aim at controlling compatibility. In such cases, the likely attackers are not malicious outsiders, but the owners of the equipment, or new firms trying to challenge the incumbent by making compatible products. This doesn't just damage competition, but innovation too. Locking things down too hard can also be bad for business, as innovation is often incremental, and products succeed when new firms find killer applications for them [732]. The PC, for example, was designed by IBM as a machine to run spreadsheets; if they had locked it down to this application alone, then a massive opportunity would have been lost. Indeed, the fact that the IBM PC was more open than the Apple Mac was a factor in its becoming the dominant desktop platform. (That Microsoft and Intel later stole IBM's lunch is a separate issue.)

So the law in many countries gives companies a right to reverse-engineer their competitors' products for compatibility [1335]. Incumbents try to build ecosystems in which their offerings work better together than with their competitors'. They lock down their products using digital components such as cloud services and cryptography so that even if competitors have the legal right to try to reverse engineer these products, they are not always going to succeed in practice. Incumbents also use their ecosystems to learn a lot about their customers, the better to lock them in; while a variety of digital mechanisms are to control aftermarkets and enforce planned obsolescence. I will discuss these more

complex ecosystem strategies in more detail below in section 8.6.4.

8.3.3 Asymmetric information

Another way markets can fail, beyond monopoly and public goods, is when some principals know more than others, or know it slightly earlier, or can find it out more cheaply. We discussed how an old-fashioned carpet trader has an information advantage over tourists buying in his store; but the formal study of *asymmetric information* was kicked off by a famous paper in 1970 on the 'market for lemons' [25], for which George Akerlof won a Nobel prize. It presents the following simple yet profound insight: suppose that there are 100 used cars for sale in a town: 50 well-maintained cars worth $2000 each, and 50 'lemons' worth $1000. The sellers know which is which, but the buyers don't. What is the market price of a used car?

You might think $1500; but at that price, no good cars will be offered for sale. So the market price will be close to $1000. This is why, if you buy a new car, maybe 20% falls off the price the second you drive it out of the dealer's lot. Asymmetric information is also why poor security products dominate some markets. When users can't tell good from bad, they might as well buy the cheapest. When the market for antivirus software took off in the 1990s, people would buy the $10 product rather than the $20 one. (Nowadays there's much less reason to buy AV, as the malware writers test their code against all available products before releasing it – you should focus on patching systems instead. That people still buy lots of AV is another example of asymmetric information.)

A further distinction can be drawn between hidden information and hidden action. For example, Volvo has a reputation for building safe cars that help their occupants survive accidents, yet Volvo drivers have more accidents. Is this because people who know they're bad drivers buy Volvos so they're less likely to get killed, or because people in Volvos believe they're safer and drive faster? The first is the hidden-information case, also known as *adverse selection*, and the second is the hidden-action case, also known as *moral hazard*. Both effects are important in security, and both may combine in specific cases. (In the case of drivers, people adjust their driving behaviour to keep their risk exposure at the level with which they're comfortable. This also explains why mandatory seat-belt laws tend not to save lives overall, merely to move fatalities from vehicle occupants to pedestrians and cyclists [15].)

Asymmetric information explains many market failures in the real world, from low prices in used-car markets to the high price of cyber-risks insurance (firms who know they cut corners may buy more of it, making it expensive for the careful). In the world of information security, it's made worse by the fact that most stakeholders are not motivated to tell the truth; police and intelligence agencies, as well as security vendors, try to talk up the threats while software vendors, e-commerce sites and banks downplay them [91].

8.3.4 Public goods

An interesting case of positive externalities is when everyone gets the same quantity of some good, whether they want it or not. Classic examples are air quality, national defense and scientific research. Economists call these *public goods*, and the formal definition is that such goods are non-rivalrous (my using them doesn't mean there's less for you) and non-excludable (there's no practical way to stop people consuming them). Uncoordinated markets are generally unable to provide public goods in socially optimal quantities.

Public goods may be supplied by governments directly, as with national defense, or by using indirect mechanisms such as laws on patents and copyrights to encourage people to produce inventions, books and music by giving them a temporary monopoly. Very often, public goods are provided by some mix of public and private action; scientific research is done in universities that get some public subsidy, earn some income from student fees, and get some research contracts from industry (which may get patents on the useful inventions).

Many aspects of security are public goods. I do not have an anti-aircraft gun on the roof of my house; air-defense threats come from a small number of actors, and are most efficiently dealt with by government action. So what about Internet security? Certainly there are strong externalities; people who connect insecure machines to the Internet end up dumping costs on others, as they enable bad actors to build botnets. Self-protection has some aspects of a public good, while insurance is more of a private good. So what should we do about it?

The answer may depend on whether the bad actors we're concerned with are concentrated or dispersed. In our quick survey of cybercrime in section 2.3 we noted that many threats have consolidated as malware writers, spammers and others have became commercial. By 2007, the number of serious spammers had dropped to a handful, and by 2019, the same had become true of denial-of-service (DoS) attacks: there seems to be one dominant DoS-for-hire provider. This suggests a more centralised defence strategy, namely, finding the bad guys and throwing them in jail.

Some have imagined a gentler government response, with rewards paid to researchers who discover vulnerabilities, paid for by fines imposed on the firms whose software contained them. To some extent this happens already via bug bounty programs and vulnerability markets, without government intervention. But a cynic will point out that in real life what happens is that vulnerabilities are sold to cyber-arms manufacturers who sell them to governments who then stockpile them – and industry pays for the collateral damage, as with NotPetya. So is air pollution the right analogy – or air defense? This brings us to game theory.

8.4 Game theory

Game theory has some of the most fundamental insights of modern economics. It's about when we cooperate, and when we fight.

There are really just two ways to get something you want if you can't find or make it yourself. You either make something useful and trade it; or you take what you need, by force, by the ballot box or whatever. Choices between cooperation and conflict are made every day at all sorts of levels, by both humans and animals.

The main tool we can use to study and analyse them is *game theory* – the study of problems of cooperation and conflict among independent decision makers. Game theory provides a common language used by economists, biologists and political scientists as well as computer scientists, and is a useful tool for building collaboration across disciplines. We're interested in games of strategy, and we try to get to the core of a decision by abstracting away much of the detail. For example, consider the school playground game of 'matching pennies': Alice and Bob toss coins and reveal them simultaneously, upon which Alice gets Bob's penny if they're different and Bob gets Alice's penny if they're the same. I'll write this as in Figure 7.2:

		Bob	
		H	T
Alice	H	-1,1	1,-1
	T	1,-1	-1,1

Figure 7.2 – matching pennies

Each entry in the table shows first Alice's outcome and then Bob's. Thus if the coins fall (H,H) Alice loses a penny and Bob gains a penny. This is an example of a *zero-sum game*: Alice's gain is Bob's loss.

Often we can solve a game quickly by writing out a *payoff matrix* like this. Here's an example (Figure 7.3):

		Bob	
		Left	Right
Alice	Top	1,2	0,1
	Bottom	2,1	1,0

Figure 7.3 – dominant strategy equilibrium

In game theory, a *strategy* is just an algorithm that takes a game state and outputs a move[1]. In this game, no matter what Bob plays, Alice is better off playing 'Bottom'; and no matter what Alice plays, Bob is better off playing 'Left'. Each player has a *dominant strategy* – an optimal choice regardless of what the other does. So Alice's strategy should be a constant 'Bottom' and Bob's a constant 'Left'. We call this a *dominant strategy equilibrium*.

Another example is shown in Figure 7.4:

[1] In business and politics, a strategy a means of acquiring power, such as monopoly power or military advantage, by a sequence of moves; the game-theoretic meaning is a somewhat simplified version, to make problems more tractable.

		Bob: Left	Bob: Right
Alice	Top	2,1	0,0
	Bottom	0,0	1,2

Figure 7.4 – Nash equilibrium

Here each player's optimal strategy depends on what they think the other player will do. We say that two strategies are in Nash equilibrium when Alice's choice is optimal given Bob's, and vice versa. Here there are two symmetric Nash equilibria, at top left and bottom right. You can think of them as being like local optima while a dominant strategy equilibrium is a global optimum.

8.4.1 The prisoners' dilemma

We're now ready to look at a famous problem that applies to many situations from international trade negotiations through cooperation between hunting animals to whether the autonomous systems that make up the Internet cooperate effectively to protect its infrastructure. It was first studied by scientists at the Rand corporation in 1950 in the context of US and USSR defense spending; Rand was paid to think about possible strategies in nuclear war. But they presented it using the following simple example.

Two prisoners are arrested on suspicion of planning a bank robbery. The police interview them separately and tell each of them: "If neither of you confesses you'll each get a year for carrying a concealed firearm without a permit. If only one of you confesses, he'll go free and the other will get 6 years for conspiracy to rob. If both of you confess, you will each get three years."

What should the prisoners do? Here's their payoff matrix:

		Benjy: Confess	Benjy: Deny
Alfie	Confess	-3,-3	0,-6
	Deny	-6,0	-1,-1

Figure 7.5 – the prisoners' dilemma

When Alfie looks at this table, he will reason as follows: "If Benjy's going to confess then I should too as then I get 3 years rather than 6; and if he's going to deny then I should still confess as I'll walk rather than doing a year". Benjy will reason similarly. The two of them confess, and get three years each. This is not just a Nash equilibrium; it's a dominant strategy equilibrium. Each prisoner should confess regardless of what the other does.

But hang on, you say, if they had agreed to keep quiet then they'll get a year each, which is a better outcome for them! In fact the strategy (deny,deny) is Pareto efficient, while the dominant strategy equilibrium is not. (That's one reason it's useful to have concepts like 'Pareto efficient' and 'dominant strategy equilibrium' rather than just arguing over 'best'.)

So what's the solution? Well, so long as the game is going to be played once only, and this is the only game in town, there isn't a solution. Both prisoners will confess and get three years.

You may think this is fair enough, as it serves them right. However, the Prisoners' Dilemma can be used to model all sorts of interactions where we decide whether or not to cooperate: international trade, nuclear arms control, fisheries protection, the reduction of CO_2 emissions, and the civility of political discourse. Even matters of self-control such as obesity and addiction can be seen as failures of cooperation with our future selves. In these applications, we really want cooperation so we can get good outcomes, but the way a single-shot game is structured can make them really hard to achieve. We can only change this if somehow we can change the game itself.

There are many possibilities: there can be laws of various kinds from international treaties on trade to the gangster's *omertà*. In practice, a prisoner's dilemma game is changed by altering the rules or the context so as to turn it into another game where the equilibrium is more efficient.

8.4.2 Repeated and evolutionary games

Suppose the game is played repeatedly – say Alfie and Benjy are career criminals who expect to be dealing with each other again and again. Then of course there can be an incentive for them to cooperate. There are at least two ways of modelling this.

In the 1970s, Bob Axelrod started thinking about how people might play many rounds of prisoners' dilemma. He set up a series of competitions to which people could submit programs, and these programs played each other repeatedly in tournaments. He found that one of the best strategies overall was *tit-for-tat*, which is simply that you cooperate in round one, and at each subsequent round you do to your opponent what he or she did in the previous round [127]. It began to be realised that strategy evolution could explain a lot. For example, in the presence of noise, players tend to get locked into (defect, defect) whenever one player's cooperative behaviour is misread by the other as defection. So in this case it helps to 'forgive' the other player from time to time.

A parallel approach was opened up by John Maynard Smith and George Price [1015]. They considered what would happen if you had a mixed population of aggressive and docile individuals, 'hawks' and 'doves', with the behaviour that doves cooperate; hawks take food from doves; and hawks fight, with a risk of death. Suppose the value of the food at each interaction is v and the risk of death in a hawk fight is c per encounter. Then the payoff matrix looks like Figure 7.6:

	Hawk	Dove
Hawk	$\frac{v-c}{2}, \frac{v-c}{2}$	v,0
Dove	0, v	$\frac{v}{2}, \frac{v}{2}$

Figure 7.6 – the hawk-dove game

Here, if $v > c$, the whole population will become hawk, as that's the dominant strategy, but if $c > v$ (fighting is too expensive) then there is an equilibrium where the probability p that a bird is a hawk sets the hawk payoff and the dove payoff equal, that is

$$p\frac{v-c}{2} + (1-p)v = (1-p)\frac{v}{2}$$

which is solved by $p = v/c$. In other words, you can have aggressive and docile individuals coexisting in a population, and the proportion of aggressive individuals will be a function of the costs of aggression; the more dangerous a fight is, the fewer combative individuals there will be. Of course, the costs can change over time, and diversity can a good thing in evolutionary terms as a society with some hard men may be at an advantage when war breaks out. But it takes generations for a society to move to equilibrium. Perhaps our current high incidence of aggression reflects conditions in pre-state societies. Indeed, anthropologists believe that tribal warfare used to be endemic in such societies; the archaeological record shows that until states came along, about a quarter to a third of men and boys died of homicide [926]. We just haven't lived long enough in civilised societies for evolution to catch up.

Such insights, along with Bob Axelrod's simulation methodology, got many people from moral philosophers to students of animal behaviour interested in evolutionary game theory. They offer further insights into how cooperation evolved. It turns out that many primates have an inbuilt sense of fairness and punish individuals who are seen to be cheating – the instinct for vengeance is one mechanism to enforce sociality. Fairness can operate in a number of different ways at different levels. For example, doves can get a better result against hawks if they can recognise each other and interact preferentially, giving a model for how some social movements and maybe even some religions establish themselves [1442]. Online reputation systems, as pioneered by eBay and now used by firms like Uber and AirBnB, perform a similar function: they help doves avoid hawks by making interactions into iterated games.

Of course, the basic idea behind tit-for-tat goes back a long way. The Old Testament has 'An eye for an eye' and the New Testament 'Do unto others as you'd have them do unto you' – the latter formulation being the more fault-tolerant – and versions of it can be found in Aristotle, in Confucius and elsewhere. More recently, Thomas Hobbes used similar arguments in the seventeenth century to argue that a state did not need the Divine Right of Kings to exist, paving the way for revolutions, republics and constitutions in the eighteenth.

Since 9/11, people have used hawk-dove games to model the ability of fundamentalists to take over discourse in religions at a time of stress. Colleagues and I have used evolutionary games to model how insurgents organise themselves into cells [1106]. Evolutionary games also explain why cartel-like behaviour can appear in industries even where there are no secret deals.

For example, Internet service in the UK involves a regulated monopoly that provides the local loop, and competing retail companies that sell Internet service to households. If the local loop costs the ISPs £6 a month, how come the ISPs

all charge about £35? Well, if one were to undercut the others, they'd all retaliate by cutting their own prices, punishing the defector. It's exactly the same behavior you see if there are three airlines operating a profitable route, and one lowers its prices to compete for volume; the others will often respond by cutting prices even more sharply to punish it and make the route unprofitable. And just as airlines offer all sorts of deals, air miles and so on to confuse the customer, so also the telecomms providers offer their own confusion pricing. Similar structures lead to similar behaviour. Tacit collusion can happen in both industries without the company executives actually sitting down and agreeing to fix prices (which would be illegal). As pricing becomes more algorithmic, both lawyers and economists may need to understand more computer science; and computer scientists need to understand economic analysis tools such as game theory and auction theory.

8.5 Auction Theory

Auction theory is vital for understanding how Internet services work, and what can go wrong. Much online activity is funded by the ad auctions run by firms like Google and Facebook, and many e-commerce sites run as auctions.

Auctions have been around for millennia, and are the standard way of selling livestock, fine art, mineral rights, bonds and much else; many other transactions from corporate takeovers to house sales are also really auctions. They are the fundamental way of discovering prices for unique goods. There are many issues of game play, asymmetric information, cheating – and some solid theory to guide us.

Consider the following five traditional types of auction.

1. In the English, or ascending-bid, auction, the auctioneer starts at a reserve price and then raises the price until only one bidder is left. This is used to sell art and antiques.

2. In the Dutch, or descending-bid, auction, the auctioneer starts out at a high price and cuts it gradually until someone bids. This is used to sell flowers.

3. In the first-price sealed-bid auction, each bidder is allowed to make one bid. After bidding closes, all the bids are opened and the highest bid wins. This has been used to auction TV rights; it's also used for government contracts, where it's the lowest bid that wins.

4. In the second-price sealed-bid auction, or Vickrey auction, we also get sealed bids and the highest bid wins, but that bidder pays the price in the second-highest bid. This is familiar from eBay, and is also how online ad auctions work; it evolved to sell rare postage stamps, though the earliest known use was by the poet Goethe to sell a manuscript to a publisher in the 18th century.

5. In the all-pay auction, every bidder pays at every round, until all but one drop out. This is a model of war, litigation, or a winner-take-all market

race between several tech startups. It's also used for charity fundraising.

The first key concept is *strategic equivalence*. The Dutch auction and the first-price sealed-bid auction give the same result, in that the highest bidder gets the goods at his *reservation price* – the maximum he's prepared to bid. Similarly, the English auction and the Vickrey auction give the same result (modulo the bid increment). However the two pairs are not strategically equivalent. In a Dutch auction, you should bid low if you believe your valuation is a lot higher than anybody else's, while in a second-price auction it's best to bid truthfully.

The second key concept is *revenue equivalence*. This is a weaker concept; it's not about who will win, but how much money the auction is expected to raise. The interesting result here is the *revenue equivalence theorem*, which says that you get the same revenue from any well-behaved auction under ideal conditions. These conditions include risk-neutral bidders, no collusion, Pareto efficiency (the highest bidder gets the goods) and independent valuations (no externalities between bidders). In such circumstances, the bidders adjust their strategies and the English, Dutch and all-pay auctions all yield the same. So when you design an auction, you have to focus on the ways in which the conditions aren't ideal. For details and examples, see Paul Klemperer's book [860].

And there are many things that can go wrong. There may be bidding rings, where all the buyers collude to lowball the auction; here, a first-price auction is best as it takes only one defector to break ranks, rather than two. Second, there's entry detection: in one UK auction of TV rights, bidders had to submit extensive programming schedules, which involved talking to production companies, so everyone in the industry knew who was bidding and the franchises with only one bidder went for peanuts. Third, there's entry deterrence: bidders in corporate takeovers often declare that they will top any other bid. Fourth, there's risk aversion: if you prefer a certain profit of \$1 to a 50% chance of \$2, you'll bid higher at a first-price auction. Fifth, there are signaling games; in US spectrum auctions, some bidders broke anonymity by putting zip codes in the least significant digits of their bids, to signal what combinations of areas they were prepared to fight for, and to deter competitors from starting a bidding war there. And then there are budget constraints: if bidders are cash-limited, all-pay auctions are more profitable.

Advertisement auctions are big business, with Google, Facebook and Amazon making about \$50bn, \$30bn and \$10bn respectively in 2019, while the rest of the industry gets about \$40bn. The ad auction mechanism pioneered by Google is a second-price auction tweaked to optimise revenue. Bidders offer to pay prices b_i, the platform estimates their ad quality as e_i, based on the ad's relevance and clickthrough rate. It then calculates 'ad rank' as $a_i = b_i e_i$. The idea is that if my ad is five times as likely to be clicked on as yours, then my bid of 10c is just as good as your bid of 50c. This is therefore a second-price auction, but based on ranking a_i rather than b_i. Thus if I have five times your ad quality, I bid 10c and you bid 40c, then I get the ad and pay 8c. It can be shown that under reasonable assumptions, this maximises platform revenue.

There's one catch, though. Once media become social, then ad quality can easily segue into virality. If your ads are good clickbait and people click on them, you pay less. One outcome was that in the 2016 US Presidential Election, Hilary

Clinton paid a lot more per ad than Donald Trump did [1001]. Some people feel this should be prohibited by electoral law; certainly it's one more example of mechanisms with structural tension between efficiency and fairness. Auction theory also shows how the drive to optimise platform revenue may lead to ever more provocative and extreme content, so it's not just about elections, or even just about fairness. In fact, in the UK, election ads aren't permitted on TV, along with some other categories such as tobacco; maybe the cleanest solution in such jurisdictions is to ban them online too, just like tobacco. And ad pricing is not the only way social media promote extreme content; as former Googler Tristan Harris has explained, the platforms' recommender algorithms are also optimised to maximise the time people spend on site, which means not just scrolling feeds and followers, but a bias towards anxiety and outrage. The real action is at the boundary between economics and psychology, but economic tools such as auctions can often be used to map it.

8.6 The economics of security and dependability

Economists used to see a simple interaction between economics and security; richer nations could afford bigger armies. But after 1945, nuclear weapons were thought to decouple national survival from economic power, and the fields of economics and strategic studies drifted apart [1003]. It has been left to the information security world to re-establish the connection.

Round about 2000, a number of us noticed persistent security failures that appeared at first sight to be irrational, but which we started to understand once we looked more carefully at the incentives facing the various actors. I observed odd patterns of investment by banks in information security measures [41, 42]. Hal Varian looked into why people were not spending as much money on anti-virus software as the vendors hoped [1571]. When the two of us got to discussing these cases in 2001, we suddenly realised that there was an interesting and important research topic here, so we contacted other people with similar interests and organised a workshop for the following year. I was writing the first edition of this book at the time, and found that describing many of the problems as incentive problems made the explanations much more compelling; so I distilled what I learned from the book's final edit into a paper 'Why Information Security is Hard – An Economic Perspective". This paper, plus the first edition of this book, got people talking [60]. By the time they came out, the 9/11 attacks had taken place and people were searching for new perspectives on security.

We rapidly found many other examples of security failure associated with institutional incentives, such as hospital systems bought by medical directors and administrators that support their interests but don't protect patient privacy. (Later, we found that patient safety failures often had similar roots.) Jean Camp had been writing about markets for vulnerabilities, and two startups had set up early vulnerability markets. Networking researchers were starting to use auction theory to design strategy-proof routing protocols. The Department of Defense had been mulling over its failure to get vendors to sell them secure systems, as you can see in the second quote at the head of this chapter. Microsoft was thinking about the economics of standards. All these ideas came together at the

Workshop on the Economics of Information Security at Berkeley in June 2002, which launched security economics as a new field of study. The picture that started to emerge was of system security failing because the people guarding a system were not the people who suffered the costs of failure. Sometimes, security mechanisms are used to dump risks on others, and if you are one of those others you'd be better off with an insecure system. Put differently, security is often a power relationship; the principals who control what it means in a given system often use it to advance their own interests.

This was the initial insight, and the story of the birth of security economics is told in [66]. But once we started studying the subject seriously, we found that there's a lot more to it than that.

8.6.1 Why is Windows so insecure?

The hot topic in 2002, when security economics got going, was this. Why is Windows so insecure, despite Microsoft's dominant market position? It's possible to write much better software, and there are fields such as defense and healthcare where a serious effort is made to produce dependable systems. Why do we not see a comparable effort made with commodity platforms, especially since Microsoft has no real competitors?

By then, we understood the basics of information economics: the combination of high fixed and low marginal costs, network effects and technical lock-in makes platform markets particularly likely to be dominated by single vendors, who stand to gain vast fortunes if they can win the race to dominate the market. In such a race, the Microsoft philosophy of the 1990s – 'ship it Tuesday and get it right by version 3' – is perfectly rational behaviour. In such a race, the platform vendor must appeal not just to users but also to complementers – to the software companies who decide whether to write applications for its platform or for someone else's. Security gets in the way of applications, and it tends to be a lemons market anyway. So the rational vendor engaged in a race for platform dominance will enable all applications to run as root on his platform[2], until his position is secure. Then he may add more security – but will be tempted to engineer it in such a way as to maximise customer lock-in, or to appeal to complementers in new markets such as digital media.

The same pattern was also seen in other platform products, from the old IBM mainframe operating systems through telephone exchange switches to the early Symbian operating system for mobile phones. Products are insecure at first, and although they improve over time, many of the new security features are for the vendor's benefit as much as the user's. And this is exactly what we saw with Microsoft's product lines. DOS had no protection at all and kick-started the malware market; Windows 3 and Windows 95 were dreadful; Windows 98 was only slightly better; and security problems eventually so annoyed Microsoft's customers that finally in 2003 Bill Gates decided to halt development until all its engineers had been on a secure coding course. This was followed by investment in better testing, static analysis tools, and regular patching. The number and

[2]To make coding easier, and enable app developers to steal the user's other data for sale in secondary markets.

lifetime of exploitable vulnerabilities continued to fall through later releases of Windows. But the attackers got better too, and the protection in Windows isn't all for the user's benefit. As Peter Gutmann points out, much more effort went into protecting premium video content than into protecting users' credit card numbers [682].

From the viewpoint of the consumer, markets with lock-in are often 'bargains then rip-offs'. You buy a nice new printer for $39.95, then find to your disgust after just a few months that you need two new printer cartridges for $19.95 each. You wonder whether you'd not be better off just buying a new printer. From the viewpoint of the application developer, markets with standards races based on lock-in look a bit like this. At first it's really easy to write code for them; later on, once you're committed, there are many more hoops to jump through. From the viewpoint of the poor consumer, they could be described as 'poor security, then security for someone else'.

The same pattern can be seen with externalities from security management costs to infrastructure decisions that the industry takes collectively. When racing to establish a dominant position, vendors are tempted to engineer products so that most of the cost of managing security is dumped on the user. A classic example is SSL/TLS encryption. This was adopted in the mid-1990s as Microsoft and Netscape battled for dominance of the browser market. As we discussed in Chapter 5, SSL leaves it up to the user to assess the certificate offered by a web site and decide whether to trust it; and this led to all kinds of phishing and other attacks. Yet dumping the compliance costs on the user made perfect sense at the time; competing protocols such as SET would have saddled banks with the cost of issuing certificates to every customer who wanted to buy stuff online, and that would just have cost too much [433]. The world ended up with an insecure system of credit card payments on the Internet, and with most of the stakeholders trying to dump liability on others in ways that block progress towards a better system.

There are also network effects for bads, and well as for goods. Most malware writers targeted Windows rather than Mac or Linux through the 2000s and 2010s as there are simply more Windows machines to infect – leading to an odd equilibrium in which people who were prepared to pay more for their laptop could have a more secure one, albeit one that didn't run as much software. This model replicated itself when smartphones took over the world in the 2010s; since Android took over from Windows as the world's most popular operating system, we're starting to see a lot of bad apps for Android, while people who pay more for an iPhone get better security but less choice. (There, the more stringent policies of Apple's app store are more important now than market share.)

8.6.2 Managing the patching cycle

The second big debate in security economics was about how to manage the patching cycle. If you discover a vulnerability, should you just publish it, which may force the vendor to patch it but may leave people exposed for months until they do so? Or should you report it privately to the vendor – and risk getting a lawyer's letter threatening an expensive lawsuit if you tell anyone else, after which the vendor just doesn't bother to patch it?

This debate goes back a long way; as we noted in the preface, the Victorians agonised over whether it was socially responsible to publish books about lockpicking, and eventually concluded that it was [1533]. People have worried more recently about whether the online availability of the US Army Improvised Munitions Handbook [1552] helps terrorists; in some countries it's a crime to possess a copy.

Security economics provides both a theoretical and a quantitative framework for discussing some issues of this kind. We started in 2002 with simple models in which bugs were independent, identically distributed and discovered at random; these have nice statistical properties, as attackers and defenders are on an equal footing, and the dependability of a system is a function only of the initial code quality and the total amount of time spent testing it [62]. But is the real world actually like that? Or is it skewed by correlated bugs, or by the vendor's inside knowledge? This led to a big policy debate. Eric Rescorla argued that software is close enough to the ideal that removing one bug makes little difference to the likelihood of an attacker finding another one later, so frequent disclosure and patching were an unnecessary expense unless the same vulnerabilities were likely to be rediscovered [1292]. Ashish Arora and others responded with data showing that public disclosure made vendors fix bugs more quickly; attacks increased to begin with, but reported vulnerabilities declined over time [113]. In 2006, Andy Ozment and Stuart Schechter found that the rate at which unique vulnerabilities were disclosed for the core OpenBSD operating system decreased over a six-year period [1200]. In short, software is more like wine than like milk – it improves with age.

Several further institutional factors helped settle the debate in favour of *responsible disclosure*, also known as *coordinated disclosure*, whereby people report bugs to vendors or to third parties that keep them confidential for a period until patches are available, then let the reporters get credit for their discoveries. One was the political settlement at the end of Crypto War I whereby bugs would be reported to CERT which would share them with the NSA during the bug-fixing process, as I will discuss later in section 26.2.7.3. This got governments on board. The second was the emergence of commercial vulnerability markets such as those set up by iDefense and TippingPoint, where security researchers could sell bugs; these firms would then disclose each bug responsibly to the vendor, and also work out indicators of compromise that could be sold to firms operating firewall or intrusion-detection services. Third, smart software firms started their own bug-bounty programs, so that security researchers could sell their bugs directly, cutting out middlemen such as CERT and iDefense.

This marketplace sharpened considerably after Stuxnet drove governments to stockpile vulnerabilities. We've seen the emergence of firms like Zerodium that buy bugs and sell them to state actors, and to cyberweapons suppliers that also sell to states; zero-day exploits for platforms such as the iPhone can now sell for a million dollars or more. This had knock-on effects on the supply chain. For example, in 2012 we came across the first case of a volunteer deliberately contributing vulnerable code to an open-source project[3], no doubt in the hope of a six-figure payoff if it had found its way into widely-used platforms. Already in 2010, Sam Ransbotham had shown that although open-source and proprietary

[3]Webkit, which is used in mobile phone browsers

software are equally secure in an ideal model, bugs get turned into exploits faster in the open source world, so attackers target it more [1277]. In 2014, Abdullah Algarni and Yashwant Malaiya surveyed vulnerability markets and interviewed some of the more prolific researchers; a combination of curiosity and economic incentives draw in many able young men, many from less developed countries, some disclose responsibly, some use vulnerability markets to get both money and recognition, while others sell for more money to the black hats; some will offer bugs to the vendor, but if not treated properly will offer them to the bad guys instead. Vendors have responded with comparable offers: at Black Hat 2019, Apple announced a bug bounty schedule that goes up to $1m for exploits that allow zero-click remote command execution on iOS. Oh, and many of the bug hunters retire after a few years [29]. Like it or not, volunteers running open-source projects now find themselves some capable motivated opponents if their projects get anywhere, and even if they can't match Apple's pocket, it's a good idea to keep as many of the researchers onside as possible.

The lifecycle of a vulnerability now involves not just its discovery, but perhaps some covert use by an intelligence agency or other black-hat actor; then its rediscovery, perhaps by other black hats but eventually by a white hat; the shipment of a patch; and then further exploitation against users who didn't apply the patch. There are tensions between vendors and their customers over the frequency and timing of patch release, as well as with complementers and secondary users over trust. A vulnerability in Linux doesn't just affect the server in your lab and your kid's Raspberry Pi. Linux is embedded everywhere: in your air-conditioner, your smart TV and even your car. This is why responsible disclosure is being rebranded as coordinated disclosure. There may be simply too many firms using a platform for the core developers to trust them all about a forthcoming patch release. There are also thousands of vulnerabilities, of which dozens appear each year in the exploit kits used by criminals (and some no doubt used only once against high-value targets, so they never become known to defense systems). We have to study multiple overlapping ecosystems – of the vulnerabilities indexed by their CVE numbers; of the Indicators of Compromise (IoCs) that get fed to intrusion detection systems; of disclosure to vendors directly, via markets, via CERTs and via ISACs; of the various botnets, crime gangs and state actors; and of the various recorded crime patterns. We have partial correlations between these ecosystems, but the data are generally noisy. I'll come back to all this in Part III.

8.6.3 Structural models of attack and defence

The late Jack Hirshleifer, the founder of conflict theory, told the story of Anarchia, an island whose flood defences were constructed by individual families each of whom maintained a section of the flood wall. The island's flood defence thus depended on the weakest link, that is, the laziest family. He compared this with a city whose defences against missile attack depend on the single best defensive shot [735]. Another example of best-shot is medieval warfare, where there could be a single combat between the two armies' champions. This can lead to different political systems. Medieval Venice, the best example of weakest-link defence because of the risk of flooding, had strong central government, with the

merchant families electing a Doge with near-dictatorial powers over flood defence. In much of the rest of late medieval Europe, kings or chieftains led their own armies to kill enemies and seize land; the strongest king built the biggest empire, and this led to a feudal system that optimised the number of men at arms.

Hal Varian extended this model to the dependability of information systems – where performance can depend on the minimum effort, the best effort, or the sum-of-efforts [1573]. This last case, the sum-of-efforts, is the modern model for warfare: we pay our taxes and the government hires soldiers. It's more efficient than best-shot (where most people will free-ride behind the heroes), which in turn is more efficient than weakest-link (where everyone will be vulnerable via the laziest). Information security is an interesting mix of all three modes. Program correctness can depend on minimum effort (the most careless programmer introducing a vulnerability) while software vulnerability testing may depend on the sum of everyone's efforts. Security may also depend on the best effort – the actions taken by an individual champion such as a security architect. As more agents are added, systems become more reliable in the total-effort case but less reliable in the weakest-link case. So as software companies get bigger, they end up hiring more testers and fewer (but more competent) programmers; Microsoft found by the early 2000s that they had more test engineers than software engineers.

Other models of attack and defence include epidemic models of malware spread, which were important back when computer viruses spread from machine to machine via floppy disks, but are of less interest now that we see relatively few wormable exploits; and models of security games that hinge on timing, notably the game of FlipIt by Ron Rivest and colleagues [463]; indeed, there's a whole conference (Gamesec) devoted to game theory and information security. There are also models of social networks. For example, most social networks owe their connectivity to a relatively small number of nodes that have a relatively high number of links to other nodes [1610]. Knocking out these nodes can rapidly disconnect things; William the Conqueror consolidated England after 1066 by killing the Anglo-Saxon nobility and replacing them with Normans, while Stalin killed the richer peasants. US and British forces similarly targeted highly-connected people in counterinsurgency operations during the Iraq war (and the resulting social breakdown in Sunni areas helped the emergence of ISIS). Such models also suggest that for insurgents to form into cells is the natural and most effective response to repeated decapitation attacks [1106].

George Danezis and I also showed that where solidarity is needed for defence, smaller and more homogeneous groups will be more effective [422]. Rainer Böhme and Tyler Moore studied what happens where it isn't – if people use defense mechanisms that bring only private benefit, then the weakest-link model becomes one of low-hanging fruit. Examples include spammers who simply guess enough weak passwords to replenish their stock of compromised email accounts, and card-not-present fraud against e-commerce websites [234].

In short, the technology of conflict in any age can have deep and subtle effects on politics, as it conditions the kind of institutions that can survive and thrive. These institutions in turn shape the security landscape. Tyler Moore, Allan Friedman and Ariel Procaccia studied whether a national agency such as the

NSA with both defensive and offensive missions would disclose vulnerabilities so they could be fixed, or stockpile them; they concluded that if it could ignore the social costs that fall on others, it would stockpile [1084]. However the biggest institutions in the security ecosystem are probably not government agencies but the dominant firms.

8.6.4 The economics of lock-in, tying and DRM

Technical lock-in is one of the factors that lead to dominant-firm markets, and software firms have spent billions over more than thirty years on mechanisms that make it hard for their customers to leave but easy for their competitors to defect. The 1980s saw file format wars where companies tried to stop anyone else accessing the word-processing files or spreadsheets their software generated. By the 1990s, the fight had shifted to network compatibility as Microsoft tried to exclude other operating systems from LANs, until SAMBA created interoperability with Apple; in the wake of a 1993 anti-trust suit, Microsoft held back from using the Windows contract to block it. Adversarial interoperability emerged as a kind of judo to fight network effects [471]. Similar mechanisms are used to control markets in neighbouring or complementary goods and services, examples being tying ink cartridges to printers, and digital rights management (DRM) systems that lock music and videos to a specific machine or family of machines, by preventing users from simply copying them as files. In an early security-economics paper, Hal Varian pointed out in 2002 that their unfettered use could damage competition [1572].

In 2003, Microsoft, Intel and others launched a 'Trusted Computing' initiative that extended rights management to other types of file, and Windows Server 2003 offered 'Information Rights Management' (IRM) whereby I could email you a Word document that you could only read on screen, not print, and only till the end of the month. There was obvious potential for competitive abuse; by transferring control of user data from the owner of the machine on which it is stored to the creator of the file in which it is stored, the potential for lock-in is hugely increased [61]. Think of the example in section 8.3.2 above, in which a firm has 100 staff, each with a PC on which they install Office for $150. The $15,000 they pay Microsoft is roughly equal to the total costs of switching to (say) LibreOffice, including training, converting files and so on. However, if control of the files moves to its thousands of customers, and the firm now has to contact each customer and request a digital certificate in order to migrate the file, then clearly the switching costs have increased – so you could expect the cost of Office to increase too. Now IRM failed to take off at the time: corporate America quickly understood that it was a lock-in play, European governments objected to the fact that the Trusted Computing initiative excluded small firms, and Microsoft couldn't get the mechanisms to work properly with Vista. However, now that email has moved to the cloud, both Microsoft and Google are offering restricted email services of just the type that was proposed, and objected to, back in 2003.

Another aspect concerns DRM and music. In the late 1990s and early 2000s, Hollywood and the music industry lobbied hard for mandatory DRM in consumer electronics equipment, and we still pay the costs of that in various ways;

for example, when you switch your presentation from a VGA adapter to HDMI and lose the audio. Hollywood's claim that unlicensed peer-to-peer filesharing would destroy the creative industries was always shaky; a 2004 study showed that downloads didn't harm music industry revenues overall [1172] while a later one suggested that downloaders actually bought more CDs [37]. However the real issue was explained in 2005 by Google's chief economist [1574]: that a stronger link between the tech industry and music would help tech firms more than the music industry, because tech was more concentrated (with only three serious music platforms then – Microsoft, Sony and Apple). The content industry scoffed, but by the end of that year music publishers were protesting that Apple was getting too large a share of the cash from online music sales. Power in the supply chain moved from the music majors to the platforms, so the platforms (now Apple, Google, Amazon and Spotify) got most of the money and the residual power in the music industry shifted from the majors to the independents – just as airline deregulation favoured aircraft makers and low-cost airlines. This is a striking demonstration of the predictive power of economic analysis. By fighting a non-existent threat, the record industry helped the computer industry eat its lunch.

DRM had become much less of an issue by 2019; the move from removable media to streaming services means that few people copy music or movies any more; the question is whether you pay a subscription to avoid the ads. Similarly, the move to cloud-based services means that few people steal software. As a result, crimes involving copyright infringement have dropped sharply [74].

However, the move to the cloud is making lock-in a more complex matter, operating at the level of ecosystems as well as of individual products. We discussed above how competition from Google Docs cut the price of Office, and so Microsoft responded with a move to Office365; and how the total cost of ownership of either that service or G-suite is greater than a standalone productivity product. So where is the lock-in? Well, if you opt for the Google ecosystem, you'll probably be using not just gmail and google docs but a google calendar, maps and much else. Although you can always download all your data, reinstalling it on a different platform (such as Microsoft's or Apple's) will be a lot of bother, so you'll probably just grit your teeth and pay for more storage when the free quota runs out. Similarly, if you start using tools like Slack or Splunk in an IT company, you'll end up customising them in all sorts of ways that make it difficult to migrate. Again, this is nothing new; my own university's dreadful accounting system has been a heavily customised version of Oracle Financials for about 20 years. Now everyone's playing the lock-in game by inducing customers to buy or build complementary assets, or even to outsource whole functions. Salesforce has taken over many companies' sales admin, Palantir has locked in many US police forces, and the big academic publishers are usurping the functions of university libraries. Where there's no viable competition – as in the second of these cases – there's a real policy issue. The depth of Microsoft lockin on public-sector IT is illustrated by the brave attempts made by the city of Munich to break away and use Linux in public administration: this was eventually reverted after 15 years, several visits of Bill Gates, and a new mayor [616].

The control of whole ecosystems by cartels is nothing new; Joshua Specht tells the history of how the big food companies like Cargill and Armour grabbed

control of the two-sided markets opened up by the railroads, consolidated their power by buying infrastructure such as grain elevators, dumped climate risk on small farmers, ran union organisers out of town and even got the politicians to pass 'ag-gag' laws that define animal-rights activism as terrorism [1464]. There are interesting echoes of this in the way the big IT service firms have built out their market power, controlling everything from the ad ecosystem through operating systems to datacentres. In fact, the whole global economy has become more monopolistic over the past couple of decades, and IT isn't the only reason for this – other industries (such as defence contracting) have their own dynamic, while natural monopolies such as utilities are traditionally dealt with using regulation, where regulators tend to be captured over time by lobbying. There is a growing literature on *moats* – structural barriers to competition, of which network effects and technical lock-in are merely two examples; others range from patents and regulatory capture to customer insight derived from control of data [1152]. The dynamics of the information industries compound many of these existing problems and can make effective competition even harder.

There are other policy issues raised by the use of security mechanisms to tie products to each other. European competition law forbids firms from using a dominant position in one market to establish one in another, and we've seen a whole series of judgements against Google and other big tech firms. Tying spare parts is also regulated, with specific laws in some sectors requiring vendors to let other firms make compatible spare parts, and in others requiring that they make spares available for a certain period of time. This links in with laws on planned obsolesence, which is reinforced for goods with digital components when the vendors limit the time period for which software updates are made available.

The rules have recently been upgraded in the European Union by an amendment to sale-of-goods law[4] which requires firms selling goods with digital components – whether embedded software, cloud services or associated phone apps – to maintain this software for at least two years after the good are sold, and for longer if this is the reasonable expectation of the customer (for cars and white goods it's likely to mean ten years). Such regulations will become more of an issue now we have software in durable goods such as cars and medical devices; I'll discuss the problems in much more detail in Part III.

8.6.5 Perversely motivated guards

"There's none sae blind as them that will na see", goes an old Scots proverb, and security engineering throws up lots of examples.

- There's very little police action against cybercrime, as they have found it simpler to ignore the problem and deter people from reporting it. As we noted in section 2.3, this enabled them to claim that crime was falling for many years even though it was just moving online like everything else.
- Governments have imposed a duty on banks to spot money laundering, especially since 9/11. However no banker really wants to know that one of his customers is a Mafioso; you'd have to resign his business. So banks

[4]Directive 2019/771

lobby for risk reduction to be formalised as due diligence; they press for detailed regulations that specify the forms of ID they need for new account opening, and the processing to be done to identify suspicious transactions.

- When it comes to fraud, spotting a rare bank fraud pattern means a payment service provider should now carry the loss rather than just telling the customer she must be mistaken or lying. So they're tempted to wait and learn about new fraud types from other people in the industry or from academics, rather than doing serious research of their own.
- Click fraud is similar. Spotting a pattern of 'inorganic clicks' from a botnet means you can't charge the advertisers for those clicks any more. You have to do some work to mitigate the worst of it, but if you have a dominant market position then the harder you work at fighting click fraud, the less revenue you earn.
- Finding bugs in your own code is another example. Of course you have to tweak the obvious bugs that stop it working, but what about the more subtle bugs that can be exploited by attackers? The more time you spend looking for them, the more time you have to spend fixing them. You can always go and buy a static analysis tool like Coverity, but then you'll find thousands more bugs and your ship date will slip by months. So firms tend to do that only if their customers demand it, and it's only cheap if you do it from the start of a project (but in that case you could just as well write the code in Rust rather than in C).

There are more subtle examples, such as when it's not politically acceptable to tell the truth about threats. In the old days, it was hard to talk to a board of directors about the insider threat, as directors mostly preferred to believe the best about their company; so a typical security manager would make chilling presentations about 'evil hackers' in order to get the budget to build internal controls. Nowadays, the security-policy space in many companies has been captured by the big four accountancy firms, whose consensus on internal controls is tied to their thought leadership on governance, which a cynic might say is optimised for the welfare not of their ostensible client, the shareholders, but for their real client, the CEO and the other senior executives. Executive frauds are rarely spotted unless they bring the company down; the effort goes instead into the annoying and irrelevant, such as changing passwords every month and insisting on original paper receipts.

Or consider the 2009 parliamentary expenses scandal in the UK described in section 2.3.6. Perhaps the officers of the Houses of Parliament didn't defend the expenses system more vigorously because they have to think of MPs and peers as 'honourable members' in the context of a government that was pushing harsh surveillance legislation with a slogan of 'If you've nothing to hide you have nothing to fear'. The author of that slogan, then Home Secretary Jacqui Smith, may have had nothing to hide, but her husband did: he was watching porn and charging it to her parliamentary expenses. Jacqui lost her job, and her seat in Parliament too. Had officers known that the information on the expenses server could cost a cabinet minister her job, they probably ought to have classified it Top Secret and kept it in a vault. But how could the extra costs have been justified to the Treasury? On that cheerful note, let's go on to privacy.

8.6.6 Economics of privacy

The privacy paradox is that people say that they value privacy, yet act otherwise. If you stop people in the street and ask them their views, about a third say they are privacy fundamentalists and will never hand over their personal information to marketers or anyone else; about a third say they don't care; and about a third are in the middle, saying they'd take a pragmatic view of the risks and benefits of any disclosure. However, their shopping behavior – both online and offline – is quite different; the great majority of people pay little heed to privacy, and will give away the most sensitive information for little benefit. Privacy-enhancing technologies have been offered for sale by various firms, yet most have failed in the marketplace. Why should this be?

Privacy is one aspect of information security that interested economists before 2000. In 1978, Richard Posner defined privacy in terms of secrecy [1244], and the following year extended it to seclusion [1245]. In 1980, Jack Hirshleifer published a seminal paper in which he argued that rather than being about withdrawing from society, privacy was a means of organising society, arising from evolved territorial behavior; internalised respect for property supports autonomy. In 1996, Hal Varian analysed privacy in terms of information markets [1568]. Consumers want to not be annoyed by irrelevant marketing calls while marketers do not want to waste effort; yet both are frustrated, because of search costs, externalities and other factors. Varian suggested giving consumers rights in information about themselves, and letting contracts sort it out.

However, as we've seen, the information industries are prone to market failures leading to monopoly, and the proliferation of dominant, information-intensive business models demands a different approach. Andrew Odlyzko argued in 2003 that these monopolies simultaneously increase both the incentives and the opportunities for price discrimination [1176]. Companies mine online interactions for data revealing individuals' willingness to pay, and while the differential pricing we see in many markets from airline yield-management systems to telecommunications prices may be economically efficient, it is increasingly resented. Peter Swire argued that we should measure the externalities of privacy intrusion [1499]. If a telesales operator calls 100 prospects, sells three of them insurance, and annoys 80, then the conventional economic analysis considers only the benefit to the three and to the insurer. But persistent annoyance causes millions of people to go ex-directory, screen calls through an answering machine, or just not have a landline at all. The long-run societal costs of robocalls can be considerable. Empirical studies of people's privacy valuations have supported this.

The privacy paradox has generated a significant literature, and is compounded by at least three factors. First, there are many different types of privacy harm, from discrimination in employment, credit and insurance, through the kind of cybercrime that presents as payment fraud, to personal crimes such as stalking and non-consensual intimate imagery.

Second, the behavioral factors we discussed in section 3.2.5 play a large role. Laura Brandimarte and colleagues demonstrated the power of context with a neat experiment. She devised a 'privacy meter' in the form of a list of embarrassing questions; the score was how many questions a subject would

answer before they balked. She tried this on three groups of students: a control group in a neutral university setting, a privacy treatment group who were given strong assurances that their data would be encrypted, their IP addresses not stored, and so on; and a gamer treatment group that was taken to an external website (`howbadareyou.com` with a logo of a smiling devil). You might think that the privacy treatment group would disclose more, but in fact they disclosed much less – as privacy had been made salient to them. As for the gamer group, they happily disclosed everything [267].

Third, the industry understands this, and goes out of its way to make privacy risks less salient. Privacy policies are usually not on the front page, but are easily findable by concerned users; policies typically start with anodyne text and leave the unpleasant stuff to the end, so they don't alarm the casual viewer, but the vigilant minority can quickly find a reason not to use the site, so they also don't stop the other users clicking on the ads. The cookie warnings mandated in Europe are mostly anodyne, though some firms give users fine-grained control; as noted in section 3.2.5, the illusion of control is enough to reassure many.

So what's the overall effect? In the 2000s and early 2010s there was evidence that the public were gradually learning what we engineers already understood about the risks; we could see this for example in the steadily rising proportion of Facebook users who opt to use privacy controls to narrow that system's very open defaults.

In 2015, almost two years after the Snowden revelations, two surveys conducted by Pew Research disclosed a growing sense of learned helplessness among the US public. 93% of adults said that being in control of who can get information about them is important, and 90% that controlling what information is collected about them is important; 88% said it's important that no-one watch or listen to them without their permission. Yet just 6% of adults said they were 'very confident' that government agencies could keep their records private and secure, while another 25% said they were 'somewhat confident.' The figures for phone companies and credit card companies were similar while those for advertisers, social media and search engines were significantly worse. Yet few respondents had done anything significant, beyond occasionally clearing their browser history or refusing particularly inappropriate demands for personal information [980].

These tensions have been growing since the 1960s, and have led to complex privacy regulation that differs significantly between the US and Europe. I'll discuss this in much more detail in section 26.6.

8.6.7 Organisations and human behaviour

todo

Put in a page about organisational economics here, and organisational behaviour – linking forward to the short section on public choice at section 26.3.3.

8.6.8 Economics of cybercrime

todo: three pages on cyber-criminology

Costs of cybercrime 2012; Changing costs of cybercrime 2018. Emotional costs of cybercrime.

Measuring policy interventions, including Ben's stuff on booter takedown and Alice's on website takedown/

Security economics and the single market. Shady markets: 'Is the Internet for Porn?' by Gilbert Wondracek et al., WEIS 2010. Michel van Eeten on whether ISPs might clean up cybercrime: the top 10 account for 30% of spam and their effectiveness at botnet mitigation varies by two orders of magnitude. ROSI. empirical confirmation e.g. Moore and Clayton on websites that impersonate banks (see Moore and Anderson 2012).

Event studies of breach disclosures – Acquisti and others studying effects on the stock price of companies reporting a security or privacy breach [13, 322]. Breach disclosure laws made breaches into insurable events; if TJX loses 47m records and has to pay $5 to mail each customer, that's a claim.

Growth of cyber-insurance since 1980s; ideas that markets might drive companies to be more secure (WEIS 2010, 5% of Chubb's business at WEIS 2010 and growing briskly) dashed since about 2017 as the market has softened, and the emergence of Bitsight instead as the benchmark that everyone uses.

Tucker-Miller, WEIS 2010: encryption made breaches in hospitals more likely as people got careless. Later: breaches more likely at US hospitals in competitive markets, as opposed to local monopolies.

8.7 Summary

Many systems fail because the incentives are wrong, rather than because of some technical design mistake. As a result, the security engineer needs to understand basic economics as well as the basics of crypto, protocols, access controls and psychology. Security economics has grown rapidly to explain many of the things that we used to consider just 'bad weather'. It constantly throws up fascinating new insights into all sorts of questions from how to optimise the patching cycle through whether people really care about privacy.

Research problems

So far, three areas of economics have been explored for their relevance to security, namely microeconomics, game theory and behavioural economics. But economics is a vast subject. What other ideas might it give us?

In the history paper I wrote on the origins of security economics, I suggested a new research student might follow the following heuristics to select a research topic. First, think of security and X for other subfields X of economics. Second, think about the security economics of Y for different applications Y; there have

already been some papers on topics like payments, pornography, gaming, and censorship, but these aren't the only things computers are used for. Third, where you find gold, keep digging (e.g. behavioral privacy) [66]. Since then I would add the following.

Fourth, there is a lot of scope for data-driven research now that we're starting to make large datasets available to academics (via the Cambridge Cybercrime Centre) and many students are keen to develop skills in data science. Fifth, now we're starting to put software and online connectivity in durable safety-critical things like cars and medical devices, we need to know a lot more about the interaction between security and safety, and about how we can keep such systems patched and running for decades. This opens up all sorts of new topics in dependability and sustainability.

The current research in security economics is published mostly at the Workshop on the Economics of Information Security (WEIS), which has been held annually since 2002 [65]. There are liveblogs of all but one of the workshops, consisting of a summary of each paper and a link to it, which you can get on my blog or linked directly from my Economics and Security Resource Page at `http://www.cl.cam.ac.uk/~rja14/econsec.html`. If you're thinking of doing research in this field, then that's essential reading.

Further reading

The classic introduction to information economics is Shapiro and Varian's *'Information Rules'* which remains remarkably fresh for a book written twenty years ago [1393]. This is still on our student reading list. For those who want to go on to do research in the subject, I suggest a standard economics textbook such as Varian's *'Intermediate Microeconomics'* [1569] or the Core Economics website. Adam Smith's classic *'An inquiry into the nature and causes of the wealth of nations'* is still worth a look, while Dick Thaler's *'Misbehaving'* tells the story of behavioural economics.

The early story of security economics is told in [66]; there's an early (2007) survey of the field that I wrote with Tyler Moore at [90], and a more comprehensive 2011 survey, also with Tyler, at [91], which is perhaps a good starting point for the research literature. Then I'd suggest the WEIS liveblogs mentioned above.

A number of economists study related areas. I mentioned Jack Hirshleifer's conflict theory [736]; another important strand is the economics of crime, which was kick-started by Gary Becker [169], and has recently been popularised by Steve Levitt and Stephen Dubner's "Freakonomics" [934]. Diego Gambetta is probably the leading scholar of organised crime; his *' Codes of the Underworld: How Criminals Communicate'* is a classic [599]. There is now a growing research community and literature on cyber-criminology, for which the website of our Cambridge Cybercrime Centre might be a reasonable starting point.