

INTRODUCTION

We mean to start a revolution with this book. But not a big revolution—at least, not at first. Our revolution does not rely on sweeping reforms, on a comprehensive Year Zero reinvention of society, or on the seamless and perfectly uniform adoption of a new technology. It is built on preexisting components—what a philosopher would call tools ready-to-hand, what an engineer would call commodity hardware—that are available in everyday life, in movies, in software, in murder mysteries, and even in the animal kingdom. Although its lexicon of methods can be, and has been, taken up by tyrants, authoritarians, and secret police, our revolution is especially suited for use by the small players, the humble, the stuck, those not in a position to decline or opt out or exert control over our data emanations. The focus of our limited revolution is on mitigating and defeating present-day digital surveillance. We will add concepts and techniques to the existing and expanding toolkit for evasion, noncompliance, outright refusal, deliberate sabotage, and use according to *our* terms of service. Depending on the adversary, the goals, and the resources, we provide methods for disappearance, for time-wasting and analysis-frustrating, for prankish disobedience, for collective protest, for acts of individual redress both great and small. We draw an outline around a whole domain of both established and emerging instances that share a common approach we can generalize and build into policies, software, and action. This outline is the banner under which our big little revolution rides, and the space it defines is called *obfuscation*.

In a sentence: *Obfuscation is the deliberate addition of ambiguous, confusing, or misleading information to interfere with surveillance and data collection.* It's a simple thing with many different, complex applications and uses. If you are a software developer or designer, obfuscation you build into your software can keep user data safe—even from yourself, or from whoever acquires your startup—while you provide social networking, geolocation, or other services requiring collection and use of personal information. Obfuscation also offers ways for government agencies to accomplish many of the goals of data collection while minimizing the potential misuses. And if you are a person or a group wanting to live in the modern world without being a subject of pervasive digital surveillance (and an object of subsequent analysis),

obfuscation is a lexicon of ways to put some sand in the gears, to buy time, and to hide in the crowd of signals. This book provides a starting point.

Our project has tracked interesting similarities across very different domains in which those who are obliged to be visible, readable, or audible have responded by burying salient signals in clouds and layers of misleading signals. Fascinated by the diverse contexts in which actors reach for a strategy of obfuscation, we have presented, in chapters 1 and 2, dozens of detailed instances that share this general, common thread. Those two chapters, which make up part I of the book, provide a guide to the diverse forms and formats that obfuscation has taken and demonstrate how these instances are crafted and implemented to suit their respective goals and adversaries. Whether on a social network, at a poker table, or in the skies during the Second World War, and whether confronting an adversary in the form of a facial-recognition system, the Apartheid government of 1980s South Africa, or an opponent across the table, properly deployed obfuscation can aid in the protection of privacy and in the defeat of data collection, observation, and analysis. The sheer range of situations and uses discussed in chapters 1 and 2 is an inspiration and a spur: What kind of work can obfuscation do for you?

The cases presented in chapter 1 are organized into a narrative that introduces fundamental questions about obfuscation and describes important approaches to it that are then explored and debated in part II of the book. In chapter 2, shorter cases illustrate the range and the variety of obfuscation applications while also reinforcing underlying concepts.

Chapters 3–5 enrich the reader’s understanding of obfuscation by considering why obfuscation has a role to play in various forms of privacy work; the ethical, social, and political problems raised by using obfuscatory tactics; and ways of assessing whether obfuscation works, or can work, in particular scenarios. Assessing whether an obfuscation approach works entails understanding what makes obfuscation distinct from other tools and understanding its particular weaknesses and strengths. The titles of chapters 3–5 are framed as questions.

The first question, asked in chapter 3, is “Why is obfuscation necessary?” In answering that question, we explain how the challenges of present-day digital privacy can be met by obfuscation’s utility. We point out how obfuscation may serve to counteract *information asymmetry*, which occurs when data

about us are collected in circumstances we may not understand, for purposes we may not understand, and are used in ways we may not understand. Our data will be shared, bought, sold, managed, analyzed, and applied, all of which will have consequences for our lives. Will you get a loan, or an apartment, for which you applied? How much of an insurance risk or a credit risk are you? What guides the advertising you receive? How do so many companies and services know that you're pregnant, or struggling with an addiction, or planning to change jobs? Why do different cohorts, different populations, and different neighborhoods receive different allocations of resources? Are you going to be, as the sinister phrase of our current moment of data-driven anti-terrorism has it, "on a list"? Even innocuous or seemingly benign work in this domain has consequences worth considering. Obfuscation has a role to play, not as a *replacement* for governance, business conduct, or technological interventions, or as a one-size-fits-all solution (again, it's a deliberately small, distributed revolution), but as a tool that fits into the larger network of privacy practices. In particular, it's a tool particularly well suited to the category of people without access to other modes of recourse, whether at a particular moment or in general—people who, as it happens, may be unable to deploy optimally configured privacy-protection tools because they are on the weak side of a particular information-power relationship.

Similarly, context shapes the ethical and political questions around obfuscation. Obfuscation's use in multiple domains, from social policy to social networks to personal activity, raises serious concerns. In chapter 4, we ask "Is obfuscation justified?" Aren't we encouraging people to lie, to be willfully inaccurate, or to "pollute" with potentially dangerous noise databases that have commercial and civic applications? Aren't obfuscators who use commercial services free riding on the good will of honest users who are paying for targeted advertising (and the services) by making data about themselves available? And if these practices become widespread, aren't we going to be collectively wasting processing power and bandwidth? In chapter 4 we address these challenges and describe the moral and political calculus according to which particular instances of obfuscation may be evaluated and found to be acceptable or unacceptable.

What obfuscation can and can't accomplish is the focus of chapter 5. In comparison with cryptography, obfuscation may be seen contingent, even shaky. With cryptography, precise degrees of security against brute-force

attacks can be calculated with reference to such factors as key length, processing power, and time. With obfuscation such precision is rarely possible, because its strength as a practical tool depends on what users *want* to accomplish and on what specific barriers they may face in respective circumstances of use. Yet complexity does not mean chaos, and success still rests on careful attention to systematic interdependencies. In chapter 5 we identify six common goals for an obfuscation project and relate them to design dimensions. The goals include buying some time, providing cover, deniability, evading observation, interfering with profiling, and expressing protest. The aspects of design we identify include whether an obfuscation project is individual or collective, whether it is known or unknown, whether it is selective or general, and whether it is short-term or long-term. For some goals, for instance, obfuscation may not succeed if the adversary knows that it is being employed; for other goals—such as collective protest or interference with probable cause and production of plausible deniability—it is better if the adversary knows that the data have been poisoned. All of this, of course, depends on what resources are available to the adversary—that is, how much time, energy, attention, and money the adversary is willing to spend on identifying and weeding out obfuscating information. The logic of these relationships holds promise because it suggests that we can learn from reasoning about specific cases how to improve obfuscation in relation to its purpose. Will obfuscation work? Yes—but only in context.

Let's begin.



An Obfuscation Vocabulary

There are many obfuscation strategies. They are shaped by the user's purposes (which may range from buying a few minutes of time to permanently interfering with a profiling system), by whether the users work alone or in concert, by its target and its beneficiaries, by the nature of the information to be obfuscated, and by other parameters we will discuss in part II. (Parts I and II can be read independently—you are encouraged to skip ahead if you have questions about obfuscation's purposes, about ethical and political quandaries, or about the circumstances that, we argue, make obfuscation a useful addition to the privacy toolkit.) Before we get to that, though, we want you to understand how of the many specific circumstances of obfuscation can be generalized into a *pattern*. We can link together a family of seemingly disparate events under a single heading, revealing their underlying continuities and suggesting how similar methods can be applied to other contexts and other problems. Obfuscation is contingent, shaped by the problems we seek to address and the adversaries we hope to foil or delay, but it is characterized by a simple underlying circumstance: unable to refuse or deny observation, we create many plausible, ambiguous, and misleading signals within which the information we want to conceal can be lost.

To illustrate obfuscation in the ways that are most salient to its use and development now, and to provide a reference for the rest of the book, we have selected a set of core cases that exemplify how obfuscation works and what it can do. These cases are organized thematically. Though they aren't suited to a simple typology, we have structured them so that the various choices particular to obfuscation should become clear as you read. In addition to these cases, we present a set of brief examples that illustrate some of obfuscation's other applications and some of its more unusual contexts. With these cases and explanations, you will have an index of obfuscation across all the domains in which we have encountered it. Obfuscation—positive and negative, effective and ineffective, targeted and indiscriminate, natural and artificial, analog and digital—appears in many fields and in many forms.

1 CORE CASES

1.1 Chaff: defeating military radar

During the Second World War, a radar operator tracks an airplane over Hamburg, guiding searchlights and anti-aircraft guns in relation to a phosphor dot whose position is updated with each sweep of the antenna. Abruptly, dots that seem to represent airplanes begin to multiply, quickly swamping the display. The actual plane is in there somewhere, impossible to locate owing to the presence of “false echoes.”¹

The plane has released chaff—strips of black paper backed with aluminum foil and cut to half the target radar’s wavelength. Thrown out by the pound and then floating down through the air, they fill the radar screen with signals. The chaff has exactly met the conditions of data the radar is configured to look for, and has given it more “planes,” scattered all across the sky, than it can handle.

This may well be the purest, simplest example of the obfuscation approach. Because discovery of an actual airplane was inevitable (there wasn’t, at the time, a way to make a plane invisible to radar), chaff taxed the time and bandwidth constraints of the discovery system by creating too many potential targets. That the chaff worked only briefly as it fluttered to the ground and was not a permanent solution wasn’t relevant under the circumstances. It only had to work well enough and long enough for the plane to get past the range of the radar.

As we will discuss in part II, many forms of obfuscation work best as time-buying “throw-away” moves. They can get you only a few minutes, but sometimes a few minutes is all the time you need.

The example of chaff also helps us to distinguish, at the most basic level, between approaches to obfuscation. Chaff relies on producing echoes—imitations of the real thing—that exploit the limited scope of the observer. (Fred Cohen terms this the “decoy strategy.”²) As we will see, some forms of obfuscation *generate genuine but misleading signals*—much as you would protect the contents of one vehicle by sending it out accompanied by several other identical vehicles, or defend a particular plane by filling the sky with other planes—whereas other forms *shuffle genuine signals*, mixing data in an effort to make the extraction of patterns more difficult. Because those who scatter chaff have exact knowledge of their adversary, chaff doesn’t have to do either of these things.

If the designers of an obfuscation system have specific and detailed knowledge of the limits of the observer, the system they develop has to work for only one wavelength and for only 45 minutes. If the system their adversary uses for observation is more patient, or if it has a more comprehensive set of capacities for observation, they have to make use of their understanding of the adversary's internal agenda—that is, of what useful information the adversary hopes to extract from data obtained through surveillance—and undermine that agenda by manipulating genuine signals.

Before we turn to the manipulation of genuine signals, let's look at a very different example of flooding a channel with echoes.

1.2 Twitter bots: filling a channel with noise

The two examples we are about to discuss are a study in contrasts. Although producing imitations is their mode of obfuscation, they take us from the Second World War to present-day circumstances, and from radar to social networks. They also introduce an important theme.

In chapter 3, we argue that obfuscation is a tool particularly suited to the “weak”—the situationally disadvantaged, those at the wrong end of asymmetrical power relationships. It is a method, after all, that you have reason to adopt if you can't be invisible—if you can't refuse to be tracked or surveilled, if you can't simply opt out or operate within professionally secured networks. This doesn't mean that it isn't also taken up by the powerful. Oppressive or coercive forces usually have better means than obfuscation at their disposal. Sometimes, though, obfuscation becomes useful to powerful actors—as it did in two elections, one in Russia and one in Mexico. Understanding the choices faced by the groups in contention will clarify how obfuscation of this kind can be employed.

During protests over problems that had arisen in the 2011 Russian parliamentary elections, much of the conversation about ballot-box stuffing and other irregularities initially took place on LiveJournal, a blogging platform that had originated in the United States but attained its greatest popularity in Russia—more than half of its user base is Russian.³ Though LiveJournal is quite popular, its user base is very small relative to those of Facebook's and Google's various social systems; it has fewer than 2 million active accounts.⁴ Thus, LiveJournal is comparatively easy for attackers to shut down by means of distributed denial of service (DDoS) attack—that is, by using computers

scattered around the world to issue requests for the site in such volume that the servers making the site available are overwhelmed and legitimate users can't access it. Such an attack on LiveJournal, in conjunction with the arrests of activist bloggers at a protest in Moscow, was a straightforward approach to censorship.⁵ When and why, then, did obfuscation become necessary?

The conversation about the Russian protest migrated to Twitter, and the powers interested in disrupting it then faced a new challenge. Twitter has an enormous user base, with infrastructure and security expertise to match. It could not be taken down as easily as LiveJournal. Based in the United States, Twitter was in a much better position to resist political manipulation than LiveJournal's parent company. (Although LiveJournal service is provided by a company set up in the U.S. for that purpose, the company that owns it, SUP Media, is based in Moscow.⁶) To block Twitter outright would require direct government intervention. The LiveJournal attack was done independently, by nationalist hackers who may or may not have the approval and assistance of the Putin/Medvedev administration.⁷ Parties interested in halting the political conversation on Twitter therefore faced a challenge that will become familiar as we explore obfuscation's uses: time was tight, and traditional mechanisms for action weren't available. A direct technical approach—either blocking Twitter within a country or launching a worldwide denial-of-service attack—wasn't possible, and political and legal angles of attack couldn't be used. Rather than stop a Twitter conversation, then, attackers can overload it with noise.

During the Russian protests, the obfuscation took the form of thousands of Twitter accounts suddenly piping up and users posting tweets using the same hashtags used by the protesters.⁸ Hashtags are a mechanism for grouping tweets together; for example, if I add #obfuscation to a tweet, the symbol # turns the word into an active link—clicking it will bring up all other tweets tagged with #obfuscation. Hashtags are useful for organizing the flood of tweets into coherent conversations on specific topics, and #триумфальная (referring to Triumfalnaya, the location of a protest) became one of several tags people could use to vent their anger, express their opinions, and organize further actions. (Hashtags also play a role in how Twitter determines “trending” and significant topics on the site, which can then draw further attention to what is being discussed under that tag—the site's Trending Topics list often draws news coverage.⁹)

If you were following #триумфальная, you would have seen tweet after tweet from Russian activists spreading links to news and making plans. But those tweets began to be interspersed with tweets about Russian greatness, or tweets that seemed to consist of noise, gibberish, or random words and phrases. Eventually those tweets dominated the stream for #триумфальная, and those for other topics related to the protests, to such a degree that tweets relevant to the topic were, essentially, lost in the noise, unable to get any attention or to start a coherent exchange with other users. That flood of new tweets came from accounts that had been inactive for much of their existence. Although they had posted very little from the time of their creation until the time of the protests, now each of them was posting dozens of times an hour. Some of the accounts' purported users had mellifluous names, such as imelixvyq, wyqufahij, and hihexiq; others had more conventional-seeming names, all built on a firstname_lastname model—for example, latifah_xander.¹⁰

Obviously, these Twitter accounts were “Twitter bots”—programs purporting to be people and generating automatic, targeted messages. Many of the accounts had been created around the same time. In numbers and in frequency, such messages can easily dominate a discussion, effectively ruining the platform for a specific audience through overuse—that is, obfuscating through the production of false, meaningless signals.

The use of Twitter bots is becoming a reliable technique for stifling Twitter discussion. The highly contentious 2012 Mexican elections provide another example of this strategy in practice, and further refined.¹¹ Protesters opposed to the front-runner, Enrique Peña Nieto, and to the Partido Revolucionario Institucional (PRI), used #marchaAntiEPN as an organizing hashtag for the purposes of aggregating conversation, structuring calls for action, and arranging protest events. Groups wishing to interfere with the protesters' organizing efforts faced challenges similar to those in the Russian case. Rather than thousands of bots, however, hundreds would do—indeed, when this case was investigated by the American Spanish-language TV network Univision, only about thirty such bots were active. Their approach was both to interfere with the work being done to advance #marchaAntiEPN and to overuse that hashtag. Many of the tweets consisted entirely of variants of “#marchaAntiEPN #marchaAntiEPN #marchaAntiEPN #marchaAntiEPN #marchaAntiEPN #marchaAntiEPN.” Such repetition, particularly by accounts already showing

suspiciously bot-like behavior, triggers systems within Twitter that identify attempts to manipulate the hashtagging system and then remove the hashtags in question from the Trending Topics list. In other words, because the items in Trending Topics become newsworthy and attract attention, spammers and advertisers will try to push hashtags up into that space through repetition, so Twitter has developed mechanisms for spotting and blocking such activity.¹²

The Mexican-election Twitter bots were deliberately engaging in bad behavior in order to trigger an automatic delisting, thereby keeping the impact of #marchaAntiEPN “off the radar” of the larger media. They were making the hashtag unusable and removing its potential media significance. This was obfuscation as a destructive act. Though such efforts use the same basic tactic as radar chaff (that is, producing many imitations configured to hide the real thing), they have very different goals: rather than just buying time (for example, in the run-up to an election and during the period of unrest afterward), they render certain terms unusable—even, from the perspective of a sorting algorithm, toxic—by manipulating the properties of the data through the use of false signals.

1.3 CacheCloak: location services without location tracking

CacheCloak takes an approach to obfuscation that is suited to location-based services (LBSs).¹³ It illustrates two twists in the use of false echoes and imitations in obfuscation. The first of these is making sure that relevant data can still be extracted by the user; the second is trying to find an approach that can work indefinitely rather than as a temporary time-buying strategy.

Location-based services take advantage of the locative capabilities of mobile devices to create various services, some of them social (e.g., Four-Square, which turns going places into a competitive game), some lucrative (e.g., location-aware advertising), and some thoroughly useful (e.g., maps and nearest-object searches). The classic rhetoric of balancing privacy against utility, in which utility is often presented as detrimental to privacy, is evident here. If you want the value of an LBS—for example, if you want to be on the network that your friends are on so you can meet with one of them if you and that person are near one another—you will have to sacrifice some privacy, and you will have to get accustomed to having the service provider know where you are. CacheCloak suggests a way to reconfigure the tradeoff.

“Where other methods try to obscure the user’s path by hiding parts of it,” the creators of CacheCloak write, “we obscure the user’s location by surrounding it with other users’ paths”¹⁴—that is, through the propagation of ambiguous data. In the standard model, your phone sends your location to the service and gets the information you requested in return. In the CacheCloak model, your phone predicts your possible paths and then fetches the results for several likely routes. As you move, you receive the benefits of locative awareness—access to what you are looking for, in the form of data cached in advance of potential requests—and an adversary is left with many possible paths, unable to distinguish the beginning from the end of a route and unable to determine where you came from, where you mean to go, or even where you are. From an observer’s perspective, the salient data—the data we wish to keep to ourselves—are buried inside a space of other, equally likely data.

1.4 TrackMeNot: blending genuine and artificial search queries

TrackMeNot, developed in 2006 by Daniel Howe, Helen Nissenbaum, and Vincent Toubiana, exemplifies a software strategy for concealing activity with imitative signals.¹⁵ The purpose of TrackMeNot is to foil the profiling of users through their searches. It was designed in response to the U.S. Department of Justice’s request for Google’s search logs and in response to the surprising discovery by a *New York Times* reporter that some identities and profiles could be inferred even from anonymized search logs published by AOL Inc.¹⁶

Our search queries end up acting as lists of locations, names, interests, and problems. Whether or not our full IP addresses are included, our identities can be inferred from these lists, and patterns in our interests can be discerned. Responding to calls for accountability, search companies have offered ways to address people’s concerns about the collection and storage of search queries, though they continue to collect and analyze logs of such queries.¹⁷ Preventing any stream of queries from being inappropriately revealing of a particular person’s interests and activities remains a challenge.¹⁸

The solution TrackMeNot offers is not to hide users’ queries from search engines (an impractical method, in view of the need for query satisfaction), but to obfuscate by automatically generating queries from a “seed list” of terms. Initially culled from RSS feeds, these terms evolve so that different users develop different seed lists. The precision of the imitation is continually refined by repopulating the seed list with new terms generated from returns to search

queries. TrackMeNot submits queries in a manner that tries to mimic real users' search behaviors. For example, a user who has searched for "good wi-fi cafe chelsea" may also have searched for "savannah kennels," "freshly pressed juice miami," "asian property firm," "exercise delays dementia," and "telescoping halogen light." The activities of individuals are masked by those of many ghosts, making the pattern harder to discern so that it becomes much more difficult to say of any query that it was a product of human intention rather than an automatic output of TrackMeNot. In this way, TrackMeNot extends the role of obfuscation, in some situations, to include plausible deniability.

1.5 Uploads to leak sites: burying significant files

WikiLeaks used a variety of systems for securing the identities of both visitors and contributors. However, there was a telltale sign that could undercut the safety of the site: uploads of files. If snoops could monitor the traffic on WikiLeaks, they could identify acts of submitting material to WikiLeaks' secure server. Especially if they could make informed guesses as to the compressed sizes of various collections of subsequently released data, they could retroactively draw inferences as to what was transmitted, when it was transmitted, and (in view of failures in other areas of technical and operations security) by whom it was transmitted. Faced with this very particular kind of challenge, WikiLeaks developed a script to produce false signals. It launched in the browsers of visitors, generating activity that looked like uploads to the secure server.¹⁹ A snoop would therefore see an enormous mob of apparent leakers (the vast majority of whom were, in actuality, merely reading or looking through documents already made available), a few of whom might really be leakers. It didn't seek to provide *particular* data to interfere with data mining or with advertising; it simply sought to imitate and conceal the movements of some of its users.

Even encrypted and compressed data contain pertinent metadata, however, and the proposal for OpenLeaks—an ultimately unsuccessful variant on WikiLeaks, developed by some of the disaffected participants in the original WikiLeaks system—includes a further refinement.²⁰ After a statistical analysis of the WikiLeaks submissions, OpenLeaks developed a model of fake uploads that would keep to the same ratios of *sizes* of files typically appearing in the upload traffic of a leak site. Most of the files ranged in size from 1.5 to 2

megabytes, though a few outliers exceeded 700 megabytes. If an adversary can monitor upload traffic, form can be as telling as content, and as useful in sorting real signals from fake ones. As this example suggests, obfuscation mechanisms can gain a great deal from figuring out all the parameters that can be manipulated—and from figuring out what the adversary is looking for, so as to give the adversary a manufactured version of it.

1.6 False tells: making patterns to trick a trained observer

Consider how the same basic pattern of obfuscation can be called to service in a context lighter than concealing the work of whistleblowers: poker.

Much of the pleasure and much of the challenge of poker lies in learning to infer from expressions, gestures, and body language whether someone is bluffing (that is, pretending to hold a hand weaker than the one he or she actually holds) in hopes of drawing a call. Central to the work of studying one's opponents is the "tell"—some unconscious habit or tic that an opponent displays in response to a strong or a weak hand, such as sweating, glancing worriedly, or leaning forward. Tells are so important in the informational economy of poker that players sometimes use *false tells*—that is, they create mannerisms that may appear to be parts of a larger pattern.²¹ In common poker strategy, the use of a false tell is best reserved for a crucial moment in a tournament, lest the other players figure out that it is inaccurate and use it against you in turn. A patient analysis of multiple games could separate the true tells from the false ones, but in the time-bound context of a high-stakes game the moment of falsehood can be highly effective. Similar techniques are used in many sports that involve visible communication. One example is signaling in baseball—as a coach explained to a newspaper reporter, "Sometimes you're giving a sign, but it doesn't even mean anything."²²

1.7 Group identity: many people under one name

One of the simplest and most memorable examples of obfuscation, and one that introduces the work of the *group* in obfuscation, is the scene in the film *Spartacus* in which the rebel slaves are asked by Roman soldiers to identify their leader, whom the soldiers intend to crucify.²³ As Spartacus (played by Kirk Douglas) is about to speak, one by one the others around him say "I am Spartacus!" until the entire crowd is claiming that identity.

Many people assuming the same identity for group protection (for example, Captain Swing in the English agricultural uprisings of 1830, the ubiquitous “Jacques” adopted by the radicals in Dickens’s *A Tale of Two Cities*, or the Guy Fawkes mask in the graphic novel *V for Vendetta*, now associated with the hacktivist group known as Anonymous) is, at this point, almost a cliché.²⁴ Marco Deseriis has studied the use of “improper names” and collective identities in the effacement of individual responsibility and the proliferation of action.²⁵ Some forms of obfuscation can be conducted solo; others rely on groups, teams, communities, and confederates.

1.8 Identical confederates and objects: many people in one outfit

There are many examples of obfuscation by members of a group working in concert to produce genuine but misleading signals within which the genuine, salient signal is concealed. One memorable example from popular culture is the scene in the 1999 remake of the film *The Thomas Crown Affair* in which the protagonist, wearing a distinctive Magritte-inspired outfit, is suddenly in a carefully orchestrated mass of other men, dressed in the same outfit, circulating through the museum and exchanging their identical briefcases.²⁶ The bank-robbery scheme in the 2006 film *Inside Man* hinges on the robbers’ all wearing painters’ overalls, gloves, and masks and dressing their hostages the same way.²⁷ Finally, consider the quick thinking of Roger Thornhill, the protagonist of Alfred Hitchcock’s 1959 film *North By Northwest*, who, in order to evade the police when his train arrives in Chicago, bribes a redcap (a baggage handler) to lend him his distinctive uniform, knowing that the crowd of redcaps at the station will give the police too much of something specific to look for.²⁸

Identical objects as modes of obfuscation are common enough and sufficiently understood to recur in imagination and in fact. The *ancilia* of ancient Rome exemplify this. A shield (*ancile*) fell from the sky—so the legend goes—during the reign of Numa Pompilius, Rome’s second king (753–673 BCE), and was interpreted as a sign of divine favor, a sacred relic whose ownership would guarantee Rome’s continued imperium.²⁹ It was hung in the Temple of Mars along with eleven exact duplicates, so would-be thieves wouldn’t know which one to take. The six plaster busts of Napoleon from which the Sherlock Holmes story gets its title offers another example. The villain sticks a black pearl into the wet plaster of an object that not only has five duplicates but also

is one of a larger class of objects (cheap white busts of Napoleon) that are ubiquitous enough to be invisible.³⁰

A real-world instance is provided by the so-called Craigslist robber. At 11 a.m. on Tuesday, September 30, 2008, a man dressed as an exterminator (in a blue shirt, goggles, and a dust mask), and carrying a spray pump, approached an armored car parked outside a bank in Monroe, Washington, incapacitated the guard with pepper spray, and made off with the money.³¹ When the police arrived, they found thirteen men in the area wearing blue shirts, goggles, and dust masks—a uniform they were wearing on the instructions of a Craigslist ad that promised a good wage for maintenance work, which was to start at 11:15 a.m. at the bank's address. It would have taken only a few minutes to determine that none of the day laborers was the robber, but a few minutes was all the time the robber needed.

Then there is the powerful story, often retold though factually inaccurate, of the king of Denmark and a great number of Danish gentiles wearing the Yellow Star so that the occupying Germans couldn't distinguish and deport Danish Jews. Although the Danes courageously protected their Jewish population in other ways, the Yellow Star wasn't used by the Nazis in occupied Denmark, for fear of arousing more anti-German feeling. However, "there were documented cases of non-Jews wearing yellow stars to protest Nazi anti-Semitism in Belgium, France, the Netherlands, Poland, and even Germany itself."³² This legend offers a perfect example of cooperative obfuscation: gentiles wearing the Yellow Star as an act of protest, providing a population into which individual Jews could blend.³³

1.9 Excessive documentation: making analysis inefficient

Continuing our look at obfuscation that operates by adding in genuine but misleading signals, let us now consider the overproduction of documents as a form of obfuscation, as in the case of over-disclosure of material in a lawsuit. This was the strategy of Augustin Lejeune, chief of the General Police Bureau in the Committee of Public Safety, a major instrument in the Terror phase of the French Revolution. Lejeune and his clerks produced the reports that laid the groundwork for arrests, internments, and executions. Later, in an effort to excuse his role in the Terror, Lejeune argued that the exacting, overwhelmingly detailed quality of the reports from his office had been deliberate: he had instructed his clerks to overproduce material, and to report "the most minor

details,” in order to slow the production of intelligence for the Committee without the appearance of rebellion. It is doubtful that Lejeune’s claims are entirely accurate (the numbers he cites for the production of reports aren’t reliable), but, as Ben Kafka points out, he had come up with a bureaucratic strategy for creating slowdowns through oversupply: “He seems to have recognized, if only belatedly, that the proliferation of documents and details presented opportunities for resistance, as well as for compliance.”³⁴ In situations where one can’t say No, there are opportunities for a chorus of unhelpful Yeses—for example, don’t send a folder in response to a request; send a pallet of boxes of folders containing potentially relevant papers.

1.10 Shuffling SIM cards: rendering mobile targeting uncertain

As recent reporting and some of Edward Snowden’s disclosures have revealed, analysts working for the National Security Agency use a combination of signals-intelligence sources—particularly cell-phone metadata and data from geolocation systems—to identify and track targets for elimination.³⁵ The metadata (showing what numbers were called and when they were called) produce a model of a social network that makes it possible to identify particular phone numbers as belonging to persons of interest; the geolocative properties of mobile phones mean that these numbers can be situated, with varying degrees of accuracy, in particular places, which can then be targeted by drones. In other words, this system can proceed from identification to location to assassination without ever having a face-to-face visual identification of a person. The closest a drone operator may come to setting eyes on someone may be the exterior of a building, or a silhouette getting into a car. In view of the spotty records of the NSA’s cell-phone-metadata program and the drone strikes, there are, of course, grave concerns about accuracy. Whether one is concerned about threats to national security remaining safe and active, about the lives of innocent people taken unjustly, or about both, it is easy to see the potential flaws in this approach.

Let us flip the situation, however, and consider it more abstractly from the perspective of the targets. Most of the NSA’s targets are obligated to always have, either with or near them, a tracking device (only the very highest-level figures in terrorist organizations are able to be free of signals-generating technology), as are virtually all the people with whom they are in contact. The calls and conversations that sustain their organizations also provide the

means of their identification; the structure that makes their work possible also traps them. Rather than trying to coordinate anti-aircraft guns to find a target somewhere in the sky, the adversary has complete air superiority, able to deliver a missile to a car, a street corner, or a house. However, the adversary also has a closely related set of systemic limitations. This system, remarkable as it is in scope and capabilities, ultimately relies on SIM (subscriber identity module) cards and on physical possession of mobile phones—a kind of narrow bandwidth that can be exploited. A former drone operator for the Joint Special Operations Command has reported that targets therefore take measures to mix and confuse genuine signals. Some individuals have many SIM cards associated with their identity in circulation, and the cards are randomly redistributed. One approach is to hold meetings at which all the attendees put their SIM cards into a bag, then pull cards from the bag at random, so that who is actually connected to each device will not be clear. (This is a time-bound approach: if metadata analysis is sufficiently sophisticated, an analyst should eventually be able to sort the individuals again on the basis of past calling patterns, but irregular re-shuffling renders that more difficult.) Re-shuffling may also happen unintentionally as targets who aren't aware that they are being tracked sell their phones or lend them to friends or relatives. The end result is a system with enormous technical precision and a very uncertain rate of actual success, whether measured in terms of dangerous individuals eliminated or in terms of innocent noncombatants killed by mistake. Even when fairly exact location tracking and social-graph analysis can't be avoided, using obfuscation to mingle and mix genuine signals, rather than generating false signals, can offer a measure of defense and control.

1.11 Tor relays: requests on behalf of others that conceal personal traffic

Tor is a system designed to facilitate anonymous use of the Internet through a combination of encryption and passing the message through many different independent “nodes.” In a hybrid strategy of obfuscation, Tor can be used in combination with other, more powerful mechanisms for concealing data. Such a strategy achieves obfuscation partially through the mixing and interleaving of genuine (encrypted) activity. Imagine a message passed surreptitiously through a huge crowd to you. The message is a question without any identifying information; as far as you know, it was written by the last person to hold it,

the person who handed it to you. The reply you write and pass back vanishes into the crowd, following an unpredictable path. Somewhere in that crowd, the writer receives his answer. Neither you nor anyone else knows exactly who the writer was.

If you request a Web page while working through Tor, your request will not come from your IP address; it will come from an “exit node” (analogous to the last person who hands the message to its addressee) on the Tor system, along with the requests of many other Tor users. Data enter the Tor system and pass into a labyrinth of relays—that is, computers on the Tor network (analogous to people in the crowd) that offer some of their bandwidth for the purpose of handling Tor traffic from others, agreeing to pass messages sight unseen. The more relays there are, the faster the system is as a whole. If you are already using Tor to protect your Internet traffic, you can turn your computer into a relay for the collective greater good. Both the Tor network and the obfuscation of individuals on the network improve as more people make use of the network.

Obfuscation, Tor’s designers point out, augments its considerable protective power. In return for running a Tor relay, “you do get better anonymity against some attacks. The simplest example is an attacker who owns a small number of Tor relays. He will see a connection from you, but he won’t be able to know whether the connection originated at your computer or was relayed from somebody else.”³⁶ If someone has agents in the crowd—that is, if someone is running Tor relays for surveillance purposes—the agents can’t read a message they pass, but they can notice who passed it to them. If you are on Tor and not running a relay, they know that you wrote the message you gave to them. But if you are letting your computer operate as a relay, the message may be yours or may be just one among many that you are passing on for other people. Did that message start with you, or not? The information is now ambiguous, and messages you have written are safe in a flock of other messages you pass along. This is, in short, a significantly more sophisticated and efficient way to render particular data transactions ambiguous and to thwart traffic analysis by making use of the volume of the traffic. It doesn’t merely mix genuine signals (as shaking up SIM cards in a bag does, with all the consequent problems of coordination); it gets each message to its destination. However, each message can serve to make the sources of other messages uncertain.

1.12 Babble tapes: hiding speech in speech

An old cliché about mobsters under threat from the FBI involved a lot of talking in bathrooms: the splash and hiss of water and the hum of the ventilation fan, so the story went, made conversations hard to hear if the house was bugged or if someone in the room was wearing a wire. There are now refined (and much more effective) techniques for defeating audio surveillance that draw more directly on obfuscation. One of these is the use of so-called babble tapes.³⁷ Paradoxically, babble tapes have been used less by mobsters than by attorneys concerned that eavesdropping may violate attorney-client privilege.

A babble tape is a digital file meant to be played in the background during conversations. The file is complex. Forty voice tracks run simultaneously (thirty-two in English, eight in other languages), and each track is compressed in frequency and time to produce additional “voices” that fill the entire frequency spectrum. There are also various non-human mechanical noises, and a periodic supersonic burst (inaudible to adult listeners) engineered specifically to interfere with the automatic gain-control system of an eavesdropping device configures itself to best pick up an audio signal. Most pertinent for present purposes, the voices on a babble tape used by an attorney include those of the client and the attorney themselves. The dense mélange of voices increases the difficulty of discerning any single voice.

1.13 Operation Vula: obfuscation in the struggle against Apartheid

We close this chapter with a detailed narrative example of obfuscation employed in a complex context by a group seeking to get Nelson Mandela released from prison in South Africa during the struggle against Apartheid. Called Operation Vula (short for Vul’indlela, meaning Opening the Road), it was devised by leaders of the African National Congress within South Africa who were in contact with Mandela and were coordinating their efforts with those of ANC agents, sympathizers, and generals around the world.

The last project of this scale that the ANC had conducted had resulted in the catastrophe of the early 1960s in which Mandela and virtually all of the ANC’s top leaders had been arrested and the Liliesleaf Farm documents had been captured and had been used against them in court. This meant that Operation Vula had to be run with absolutely airtight security and privacy practices. Indeed, when the full scope of the operation was revealed in the 1990s, it came

as a surprise not just to the South African government and to international intelligence services but also to many prominent leadership figures within the ANC. People purportedly receiving kidney transplants or recovering from motorcycle accidents had actually gone deep underground with new identities and then had returned to South Africa, “opening the road” for Mandela’s release. Given the surveillance inside and outside South Africa, the possible compromise of pre-existing ANC communications channels, and the interest of spies and law-enforcement groups around the world, Operation Vula had to have secure ways of sharing and coordinating information.

The extraordinary tale of Operation Vula has been told by one of its chief architects, Tim Jenkin, in the pages of the ANC’s journal *Mayibuye*.³⁸ It represents a superb example of operations security, tradecraft, and managing a secure network.

Understanding when and how obfuscation came to be employed in Operation Vula requires understanding some of the challenges its architects faced. Using fixed phone lines within South Africa, each linked to an address and a name, wasn’t an option. The slightest compromise might lead to wiretaps and to what we would now call metadata analysis, and thus a picture of the activist network could be put together from domestic and overseas phone logs. The Vula agents had various coding systems, each of them hampered by the difficulty and tedium of doing the coding by hand. There was always the temptation to fall back on “speaking in whispers over phones again,” especially when crises happened and things began moving fast. The operation had to be seamlessly coordinated between South Africa (primarily Durban and Johannesburg) and Lusaka, London, Amsterdam, and other locations around the world as agents circulated. Postal service was slow and vulnerable, encrypting was enormously time consuming and often prone to sloppiness, use of home phones was forbidden, and coordinating between multiple time zones around the world seemed impossible.

Jenkin was aware of the possibilities of using personal computers to make encryption faster and more efficient. Based in London after his escape from Pretoria Central Prison, he spent the mid 1980s working on the communications system needed for Operation Vula, which ultimately evolved into a remarkable network. Encryption happened on a personal computer, and the ciphered message was then expressed as a rapid series of tones recorded onto a portable cassette player. An agent would go to a public pay phone and

dial a London number, which would be picked up by an answering machine that Jenkin had modified to record for up to five minutes. The agent would play the cassette into the mouthpiece of the phone. The tones, recorded on the cassette's other side, could be played through an acoustic modem into the computer and then decrypted. (There was also an "outgoing" answering machine. Remote agents could call from a pay phone, record the tones for their messages, and decrypt them anywhere they had access to a computer that could run the ciphering systems Jenkin had devised.)

This was already an enormously impressive network—not least because large parts of its digital side (including a way of implementing error-handling codes to deal with the noise of playing back messages over international phone lines from noisy booths) had to be invented from scratch. However, as Operation Vula continued to grow and the network of operatives to expand, the sheer quantity of traffic threatened to overwhelm the network. Operatives were preparing South Africa for action, and that work didn't leave a lot of time for finding pay phones that accepted credit cards (the sound of coins dropping could interfere with the signal) and standing around with tape players. Jenkin and his collaborators would stay up late, changing tapes in the machines as the messages poured in. The time had come to switch to encrypted email, but the whole system had been developed to avoid the use of known, owned telephone lines within South Africa.

Operation Vula needed to be able to send encrypted messages to and from computers in South Africa, in Lukasa, and in London without arousing suspicion. During the 1980s, while the network we have described was taking shape, the larger milieu of international business was producing exactly the kind of background against which this subterfuge could hide itself. The question was, as Jenkin put it, "Did the enemy have the capacity to determine which of the thousands of messages leaving the country every day was a 'suspicious' one?" The activists needed a typical user of encrypted email—one without clear political affiliation—to find out if their encrypted messages could escape notice in the overall tide of mail. They needed, Jenkin later recalled, to "find someone who would normally use a computer for communicating abroad and get that person to handle the communications."

They had an agent who could try this system out before they switched their communications over to the new approach: a native South African who was about to return to his homeland after working abroad for many years as a

programmer for British telecommunications companies. Their agent would behave just as a typical citizen sending a lot of email messages every day would, using a commercial email provider rather than a custom server and relying on the fact that many businesses used encryption in their communications. “This was a most normal thing for a person in his position to do,” Jenkin recalled. The system worked: the agent’s messages blended in with the ordinary traffic, providing a platform for openly secret communications that could be expanded rapidly.

Posing as computer consultants, Tim Jenkin and Ronnie Press (another important member of the ANC Technical Committee) were able to keep abreast of new devices and storage technologies, and to arrange for their purchase and delivery where they were needed. Using a combination of commercial email providers and bulletin-board services run off personal and pocket computers, they were able to circulate messages within South Africa and around the world, and also to prepare fully formatted ANC literature for distribution. (The system even carried messages from Mandela, smuggled out by his lawyer in secret compartments in books and typed into the system.) The ordinary activity of ordinary users with bland business addresses became a high-value informational channel, moving huge volumes of encrypted data from London to Lukasa and then into South Africa and between Vula cells in that country. The success of this system was due in part to historical circumstance—personal computers and email (including encrypted email) had become common enough to avoid provoking suspicion, but not so common as to inspire the construction of new, more comprehensive digital surveillance systems such as governments have today.

The Vula network, in its ultimate stage, wasn’t naive about the security of digital messages; it kept everything protected by a sophisticated encryption system full of inventive details, and it encouraged its users to change their encryption keys and to practice good operations security. Within that context, however, it offers an excellent example of the role obfuscation can play in building a secure and secret communications system. It illustrates the benefits of finding the right existing situation and blending into it, lost in the hubbub of ordinary commerce, hidden by the crowd.

2 OTHER EXAMPLES

2.1 Orb-weaving spiders: obfuscating animals

Some animals (and some plants too) have ways to conceal themselves or engage in visual trickery. Insects mimic the appearance of leaves or twigs, rabbits have countershading (white bellies) to eliminate the cues of shape that enables a hawk to easily see and strike, and spots on butterflies' wings mimic the eyes of predatory animals.

A quintessential obfuscator in the animal world is *Cyclosa mulmeinensis*, an orb-weaving spider.¹ This spider faces a particular problem for which obfuscation is a sound solution: its web must be somewhat exposed in order to catch prey, but that makes the spider much more vulnerable to attack by wasps. The spider's solution is to make stand-ins for itself out of remains of its prey, leaf litter, and spider silk, with (from the perspective of a wasp) the same size, color, and reflectivity of the spider itself, and to position these decoys around the web. This decreases the odds of a wasp strike hitting home and gives *Cyclosa mulmeinensis* time to scuttle out of harm's way.

2.2 False orders: using obfuscation to attack rival businesses

The obfuscation goal of making a channel noisier can be employed not only to conceal significant traffic, but also to raise the costs of organization through that channel—and so raise the cost of doing business. The taxi-replacement company Uber provides an example of this approach in practice.

The market for businesses that provide something akin to taxis and car services is growing fast, and competition for both customers and drivers is fierce. Uber has offered bonuses to recruit drivers from competing services, and rewards merely for visiting the company's headquarters. In New York, Uber pursued a particularly aggressive strategy against its competitor Gett, using obfuscation to recruit Gett's drivers.² Over the course of a few days, several Uber employees would order rides from Gett, then would cancel those orders shortly before the Gett drivers arrived. This flood of fruitless orders kept the Gett drivers in motion, not earning fees, and unable to fulfill many legitimate requests. Shortly after receiving a fruitless order, or several of them, a Gett driver would receive a text message from Uber offering him money to switch jobs. Real requests for rides were effectively obfuscated by Uber's fake requests, which reduced the value of a job with Gett. (Lyft, a ride-

sharing company, has alleged that Uber has made similar obfuscation attacks on its drivers.)

2.3 French decoy radar emplacements: defeating radar detectors

Obfuscation plays a part in the French government's strategy against radar detectors.³ These fairly common appliances warn drivers when police are using speed-detecting radar nearby. Some radar detectors can indicate the position of a radar gun relative to a user's vehicle, and thus are even more effective in helping drivers to avoid speeding tickets.

In theory, tickets are a disincentive to excessively fast and dangerous driving; in practice, they serve as a revenue source for local police departments and governments. For both reasons, police are highly motivated to defeat radar detectors.

The option of regulating or even banning radar detectors is unrealistic in view of the fact that 6 million French drivers are estimated to own them. Turning that many ordinary citizens into criminals seems impolitic. Without the power to stop surveillance of radar guns, the French government has taken to obfuscation to render such surveillance less useful in high-traffic zones by deploying arrays of devices that trigger radar detectors' warning signals without actually measuring speed. These devices mirror the chaff strategy in that the warning chirps multiply and multiply again. One of them may, indeed, indicate actual speed-detecting radar, but which one? The meaningful signal is drowned in a mass of other plausible signals. Either drivers risk getting speeding tickets or they slow down in response to the deluge of radar pings. And the civic goal is accomplished. No matter how one feels about traffic cops or speeding drivers, the case holds interest as a way obfuscation serves to promote an end not by destroying one's adversaries' devices outright but by rendering them functionally irrelevant.

2.4 AdNauseam: clicking all the ads

In a strategy resembling that of the French radar-gun decoys, AdNauseam, a browser plug-in, resists online surveillance for purposes of behavioral advertising by clicking all the banner ads on all the Web pages visited by its users. In conjunction with Ad Block Plus, AdNauseam functions in the background, quietly clicking all blocked ads while recording, for the user's interest, details about ads that have been served and blocked.

The idea for AdNauseam emerged out of a sense of helplessness: it isn't possible to stop ubiquitous tracking by ad networks, or to comprehend the intricate institutional and technical complexities constituting its socio-technical backend. These include Web cookies and beacons, browser fingerprinting (which uses combinations and configurations of the visitor's technology to identify their activities), ad networks, and analytics companies. Efforts to find some middle ground through a Do Not Track technical standard have been frustrated by powerful actors in the political economy of targeted advertising. In this climate of no compromise, AdNauseam was born. Its design was inspired by a slender insight into the prevailing business model, which charges prospective advertisers a premium for delivering viewers with proven interest in their products. What more telling evidence is there of interest than clicks on particular ads? Clicks also sometimes constitute the basis of payment to an ad network and to the ad-hosting website. Clicks on ads, in combination with other data streams, build up the profiles of tracked users. Like the French radar decoy systems, AdNauseam isn't aiming to destroy the ability to track clicks; instead it functions by diminishing the value of those clicks by obfuscating the real clicks with clicks that it generates automatically.

2.5 Quote stuffing: confusing algorithmic trading strategies

The term "quote stuffing" has been applied to bursts of anomalous activity on stock exchanges that appear to be misleading trading data generated to gain advantage over competitors on the exchange. In the rarefied field of high-frequency trading (HFT), algorithms perform large volumes of trades far faster than humans could, taking advantage of minute spans of time and differences in price that wouldn't draw the notice of attention of human traders. Timing has always been critical to trading, but in HFT thousandths of a second separate profit and loss, and complex strategies have emerged to accelerate your trades and retard those of your competitors. Analysts of market behavior began to notice unusual patterns of HFT activity during the summer of 2010: bursts of quote requests for a particular stock, sometimes thousands of them in a second. Such activity seemed to have no economic rationale, but one of the most interesting and plausible theories is that these bursts are an obfuscation tactic. One observer explains the phenomenon this way: "If you could generate a large number of quotes that your competitors have to process, but you can ignore since you generated them, you gain valuable processing time."⁴

Unimportant information, in the form of quotes, is used to crowd the field of salient activity so that the generators of the unimportant information can accurately assess what is happening while making it more difficult and time consuming for their competitors to do so. They create a cloud that only they can see through. None of the patterns in that information would fool or even distract an analyst over a longer period of time—it would be obvious that they were artificial and insignificant. But in the sub-split-second world of HFT, the time it takes merely to observe and process activity makes all the difference.

If the use of “quote stuffing” were to spread, it might threaten the very integrity of the stock market as a working system by overwhelming the physical infrastructure on which the stock exchanges rely with hundreds of thousands of useless quotes consuming bandwidth. “This is an extremely disturbing development,” the observer quoted above adds, “because as more HFT systems start doing this, it is only a matter of time before quote-stuffing shuts down the entire market from congestion.”⁵

2.6 Swapping loyalty cards to interfere with analysis of shopping patterns

Grocery stores have long been in the technological vanguard when it comes to working with data. Relatively innocuous early loyalty-card programs were used to draw repeat customers, extracting extra profit margins from people who didn’t use the card and aiding primitive data projects such as organizing direct mailings by ZIP code. The vast majority of grocers and chains outsourced the business of analyzing data to ACNielsen, Catalina Marketing, and a few other companies.⁶ Although these practices were initially perceived as isolated and inoffensive, a few incidents altered the perception of purpose from innocuous and helpful to somewhat sinister.

In 1999, a slip-and-fall accident in a Los Angeles supermarket led to a lawsuit, and attorneys for the supermarket chain threatened to disclose the victim’s history of alcohol purchases to the court.⁷ A string of similar cases over the years fed a growing suspicion in the popular imagination that so-called loyalty cards were serving ends beyond the allotment of discounts. Soon after their widespread introduction, card-swapping networks developed. People shared cards in order to obfuscate data about their purchasing patterns—initially in *ad hoc* physical meetings, then, with the help of mailing lists and online social networks, increasingly in large populations and over wide

geographical regions. Rob's Giant Bonus Card Swap Meet, for instance, started from the idea that a system for sharing bar codes could enable customers of the DC-area supermarket chain Giant to print out the bar codes of other customers and then paste them onto their cards.⁸ Similarly, the Ultimate Shopper project fabricated and distributed stickers imprinted with the bar code from a Safeway loyalty card, thereby creating "an army of clones" whose shopping data would be accrued.⁹ Cardexchange.org, devoted to exchanging loyalty cards by mail, presents itself as a direct analogue to physical meet-ups held for the same purpose. The swapping of loyalty cards constitutes obfuscation as a group activity: the greater the number of people who are willing to share their cards, and the farther the cards travel, the less reliable the data become.

Card-swapping websites also host discussions and post news articles and essays about differing approaches to loyalty-card obfuscation and some of the ethical issues they raise. Negative effects on grocery stores are of concern, as card swapping degrades the data available to them and perhaps to other recipients. It is worth noting that such effects are contingent both on the card programs and on the approaches to card swapping. For example, sharing of a loyalty card within a household or among friends, though it may deprive a store of individual-level data, may still provide some useful information about shopping episodes or about product preferences within geographic areas. The value of data at the scale of a postal code, a neighborhood, or a district is far from insignificant. And there may be larger patterns to be inferred from the genuine information present in mixed and mingled data.

2.7 BitTorrent Hydra: using fake requests to deter collection of addresses

BitTorrent Hydra, a now-defunct but interesting and illustrative project, fought the surveillance efforts of anti-file-sharing interests by mixing genuine requests for bits of a file with dummy requests.¹⁰ The BitTorrent protocol broke a file into many small pieces and allowed users to share files with one another by simultaneously sending and receiving the pieces.¹¹ Rather than download an entire file from another user, one assembled it from pieces obtained from anyone else who had them, and anyone who needed a piece that you had could get it from you. This many-pieces-from-many-people approach expedited the sharing of files of all kinds and quickly became the method of choice for moving large files, such as those containing movies and music.¹² To help users

of BitTorrent assemble the files they needed, “torrent trackers” logged IP addresses that were sending and receiving files. For example, if you were looking for certain pieces of a file, torrent trackers would point you to the addresses of users who had the pieces you needed. Representatives of the content industry, looking for violations of their intellectual property, began to run their own trackers to gather the addresses of major unauthorized uploaders and downloaders in order to stop them or even prosecute them. Hydra counteracted this tracking by adding random IP addresses drawn from those previously used for BitTorrent to the collection of addresses found by the torrent tracker. If you had requested pieces of a file, you would be periodically directed to a user who didn’t have what you were looking for. Although a small inefficiency for the BitTorrent system as a whole, it significantly undercut the utility of the addresses that copyright enforcers gathered, which may have belonged to actual participants but which may have been dummy addresses inserted by Hydra. Doubt and uncertainty had been reintroduced to the system, lessening the likelihood that one could sue with assurance. Rather than attempt to destroy the adversary’s logs or to conceal BitTorrent traffic, Hydra provided an “I am Spartacus” defense. Hydra didn’t avert data collection; however, by degrading the reliability of data collection, it called any specific findings into question.

2.8 Deliberately vague language: obfuscating agency

According to Jacquelyn Burkell and Alexandre Fortier, the privacy policies of health information sites use particularly obtuse linguistic constructions when describing their use of tracking, monitoring, and data collection.¹³ Conditional verbs (e.g., “may”), passive voice, nominalization, temporal adverbs (e.g., “periodically” and “occasionally”), and the use of qualitative adjectives (as in “small piece of data”) are among the linguistic constructions that Burkell and Fortier identify. As subtle as this form of obfuscation may seem, it is recognizably similar in operation to other forms we have already described: in place of a specific, specious denial (e.g., “we do not collect user information”) or an exact admission, vague language produces many confusing gestures of possible activity and attribution. For example, the sentence “Certain information may be passively collected to connect use of this site with information about the use of other sites provided by third parties” puts the particulars of what a site does with certain information inside a cloud of possible interpretations.

These written practices veer away from obfuscation *per se* into the more general domain of abstruse language and “weasel words.”¹⁴ However, for purposes of illustrating the range of obfuscating approaches, the style of obfuscated language is useful: a document must be there, a straightforward denial isn’t possible, and so the strategy becomes one of rendering who is doing what puzzling and unclear.

2.9 Obfuscation of anonymous text: stopping stylometric analysis

How much in text identifies it as the creation of one author rather than another? Stylometry uses only elements of linguistic style to attribute authorship to anonymous texts. It doesn’t have to account for the possibility that only a certain person would have knowledge of some matter, for posts to an online forum, for other external clues (such as IP addresses), or for timing. It considers length of sentences, choice of words, and syntax, idiosyncrasies in formatting and usage, regionalisms, and recurrent typographical errors. It was a stylometric analysis that helped to settle the debate over the pseudonymous authors of the Federalist Papers (for example, the use of “while” versus “whilst” served to differentiate the styles of Alexander Hamilton and James Madison), and stylometry’s usefulness in legal contexts is now well established.¹⁵

Given a small amount of text, stylometry can identify an author. And we mean small—according to Josyula Rao and Pankaj Ratangi, a sample consisting of about 6,500 words is sufficient (when used with a corpus of identified text, such as email messages, posts to a social network, or blog posts) to make possible an 80 percent rate of successful identification.¹⁶ In the course of their everyday use of computers, many people produce 6,500 words in a few days.

Even if the goal is not to identify a specific author from a pool of known individuals, stylometry can produce information that is useful for purposes of surveillance. The technology activist Daniel Domscheit-Berg recalls the moment when he realized that if WikiLeaks’ press releases, summaries of leaks, and other public texts were to undergo stylometric analysis it would show that only two people (Domscheit-Berg and Julian Assange) had been responsible for all those texts rather than a large and diverse group of volunteers, as Assange and Domscheit-Berg were trying to suggest.¹⁷ Stylometric analysis offers an adversary a more accurate picture of an “anonymous” or

secretive movement, and of its vulnerabilities, than can be gained by other means. Having narrowed authorship down to a small handful, the adversary is in a better position to target a known set of likely suspects.

Obfuscation makes it practicable to muddle the signal of a public body of text and to interfere with the process of connecting that body of text with a named author. Stylometric obfuscation is distinctive, too, in that its success is more readily tested than with many other forms of obfuscation, whose precise effects may be highly uncertain and/or may be known only to an uncooperative adversary.

Three approaches to beating stylometry offer useful insights into obfuscation. The first two, which are intuitive and straightforward, involve assuming a writing style that differs from one's usual style; their weaknesses highlight the value of using obfuscation.

Translation attacks take advantage of the weaknesses of machine translation by translating a text into multiple languages and then translating it back into its original language—a game of Telephone that might corrupt an author's style enough to prevent attribution.¹⁸ Of course, this also renders the text less coherent and meaningful, and as translation tools improve it may not do a good enough job of depersonalization.

In *imitation attacks*, the original author deliberately writes a document in the style of another author. One vulnerability of that approach has been elegantly exposed by research.¹⁹ Using the systems you would use to identify texts as belonging to the same author, you can determine the most powerful identifier of authorship between two texts, then eliminate that identifier from the analysis and look for the next-most-powerful identifier, then keep repeating the same process of elimination. If the texts really are by different people, accuracy in distinguishing between them will decline slowly, because beneath the big, obvious differences between one author and another there are many smaller and less reliable differences. If, however, both texts are by the same person, and one of them was written in imitation of another author, accuracy in distinguishing will decline rapidly, because beneath notable idiosyncrasies fundamental similarities are hard to shake.

Obfuscation attacks on stylometric analysis involve writing in such a way that there is no distinctive style. Researchers distinguish between “shallow” and “deep” obfuscation of texts. “Shallow” obfuscation changes only a small number of the most obvious features—for example, preference for “while” or

for “whilst.” “Deep” obfuscation runs the same system of classifiers used to defeat imitation, but does so for the author’s benefit. Such a method might provide real-time feedback to an author editing a document, identifying the highest-ranked features and suggesting changes that would diminish the accuracy of stylometric analysis—for example, sophisticated paraphrasing. It might turn the banalities of “general usage” into a resource, enabling an author to blend into a vast crowd of similar authors.

Anonymouth—a tool that is under development as of this writing—is a step toward implementing this approach by producing statistically bland prose that can be obfuscated within the corpus of similar writing.²⁰ Think of the car provided to the getaway driver in the 2011 movie *Drive*: a silver late-model Chevrolet Impala, the most popular car in California, about which the mechanic promises “No one will be looking at you.”²¹ Ingenious as this may be, we wonder about a future in which political manifestos and critical documents strive for great rhetorical and stylistic banality and we lose the next Thomas Paine’s equivalent to “These are the times that try men’s souls.”

2.10 Code obfuscation: baffling humans but not machines

In the field of computer programming, the term “obfuscated code” has two related but distinct meanings. The first is “obfuscation as a means of protection”—that is, making the code harder for human readers (or the various forms of “disassembly algorithms,” which help explicate code that has been compiled for use) to interpret for purposes of copying, modification, or compromise. (A classic example of such reverse engineering goes as follows: Microsoft sends out a patch to update Windows computers for security purposes; bad actors get the patch and look at the code to figure out what vulnerability the patch is meant to address; they then devise an attack exploiting the vulnerability they have noticed hitting.) The second meaning of “obfuscated code” refers to a form of art: writing code that is fiendishly complex for a human to untangle but which ultimately performs a mundane computational task that is easily processed by a computer.

Simply put, a program that has been obfuscated will have the same functionality it had before, but will be more difficult for a human to analyze. Such a program exhibits two characteristics of obfuscation as a category and a concept. First, it operates under constraint—you obfuscate because people *will* be able to see your code, and the goals of obfuscation-as-protection are

to decrease the efficiency of the analysis (“at least doubling the time needed,” as experimental research has found), to reduce the gap between novices and skilled analysts, and to give systems that (for whatever reason) are easier to attack threat profiles closer to those of systems that are more difficult to attack.²² Second, an obfuscated program’s code uses strategies that are familiar from other forms of obfuscation: adding significant-seeming gibberish; having extra variables that must be accounted for; using arbitrary or deliberately confusing names for things within the code; including within the code deliberately confusing directions (essentially, “go to line *x* and do *y*”) that lead to dead ends or wild goose chases; and various forms of scrambling. In its protective mode, code obfuscation is a time-buying approach to thwarting analysis—a speed bump. (Recently there have been advances that significantly increase the difficulty of de-obfuscation and the amount of time it requires; we will discuss them below.)

In its artistic, aesthetic form, code obfuscation is in the vanguard of counterintuitive, puzzling methods of accomplishing goals. Nick Montfort has described these practices in considerable detail.²³ For example, because of how the programming language C interprets names of variables, a programmer can muddle human analysis but not machine execution by writing code that includes the letters *o* and *O* in contexts that trick the eye by resembling zeroes. Some of these forms of obfuscation lie a little outside our working definition of “obfuscation,” but they are useful for illustrating an approach to the fundamental problem of obfuscation: how to transform something that is open to scrutiny into something ambiguous, full of false leads, mistaken identities, and unmet expectations.

Code obfuscation, like stylometry, can be analyzed, tested, and optimized with precision. Its functionality is expanding from the limited scope of buying time and making the task of unraveling code more difficult to something closer to achieving complete opacity. A recent publication by Sanjam Garg and colleagues has moved code obfuscation from a “speed bump” to an “iron wall.” A Multilinear Jigsaw Puzzle can break code apart so that it “fits together” like pieces of a puzzle. Although many arrangements are possible, only one arrangement is correct and represents the actual operation of the code.²⁴ A programmer can create a clean, clear, human-readable program and then run it through an obfuscator to produce something incomprehensible that can withstand scrutiny for a much longer time than before.

Code obfuscation—a lively, rich area for the exploration of obfuscation in general—seems to be progressing toward systems that are relatively easy to use and enormously difficult to defeat. This is even applicable to hardware: Jeyavijayan Rajendran and colleagues are utilizing components within circuits to create “logic obfuscation” in order to prevent reverse engineering of the functionality of a chip.²⁵

2.11 Personal disinformation: strategies for individual disappearance

Disappearance specialists have much to teach would-be obfuscators. Many of these specialists are private detectives or “skip tracers”—professionals in the business of finding fugitives and debtors—who reverse engineer their own process to help their clients stay lost. Obviously many of the techniques and methods they employ have nothing to do with obfuscation, but rather are merely evasive or concealing—for instance, creating a corporation that can lease your new apartment and pay your bills so that your name will not be connected with those common and publicly searchable activities. However, in response to the proliferation of social networking and online presence, disappearance specialists advocate a strategy of *disinformation*, a variety of obfuscation. “Bogus individuals,” to quote the disappearance consultant Frank Ahearn, can be produced in number and detail that will “bury” pre-existing personal information that might crop up in a list of Web search results.²⁶ This entails creating a few dozen fictitious people with the same name and the same basic characteristics, some of them with personal websites, some with accounts on social networks, and all of them intermittently active. For clients fleeing stalkers or abusive spouses, Ahearn recommends simultaneous producing numerous false leads that an investigator would be likely to follow—for example, a credit check for a lease on an apartment in one city (a lease that was never actually signed) and applications for utilities, employment addresses and phone numbers scattered across the country or the world, and a checking account, holding a fixed sum, with a debit card given to someone traveling to pay for expenses incurred in remote locations. Strategies suggested by disappearance specialists are based on known details about the adversary: the goal is not to make someone “vanish completely,” but to put one far enough out of sight for practical purposes and thus to use up the seeker’s budget and resources.

2.12 Apple's "cloning service" patent: polluting electronic profiling

In 2012, as part of a larger portfolio purchase from Novell, Apple acquired U.S. Patent 8,205,265, "Techniques to Pollute Electronic Profiling."²⁷ An approach to managing data surveillance without sacrificing services, it parallels several systems of technological obfuscation we have described already. This "cloning service" would automate and augment the process of producing misleading personal information, targeting online data collectors rather than private investigators.

A "cloning service" observes an individual's activities and assembles a plausible picture of his or her rhythms and interests. At the user's request, it will spin off a cloned identity that can use the identifiers provided to authenticate (to social networks, if not to more demanding observers) that represents a real person. These identifiers might include small amounts of actual confidential data (a few details of a life, such as hair color or marital status) mixed in with a considerable amount of deliberately inaccurate information. Starting from its initial data set, the cloned identity acquires an email address from which it will send and receive messages, a phone number (there are many online calling services that make phone numbers available for a small fee), and voicemail service. It may have an independent source of funds (perhaps a gift card or a debit card connected with a fixed account that gets refilled from time to time) that enables it to make small transactions. It may even have a mailing address or an Amazon locker—two more signals that suggest personhood. To these signals may be added some interests formally specified by the user and fleshed out with existing data made accessible by the scraping of social-network sites and by similar means. If a user setting up a clone were to select from drop-down menus that the clone is American and is interested in photography and camping, the system would figure out that the clone should be interested in the work of Ansel Adams. It can conduct searches (in the manner of TrackMeNot), follow links, browse pages, and even make purchases and establish accounts with services (e.g., subscribing to a mailing list devoted to deals on wilderness excursions, or following *National Geographic's* Twitter account). These interests may draw on the user's actual interests, as inferred from things such as the user's browsing history, but may begin to diverge from those interests in a gradual, incremental way. (One could also salt the profile of one's clone with demographically appropriate activities, automatically chosen, building on the basics of one's actual data by selecting

interests and behaviors so typical that they even out the telling idiosyncrasies of selfhood.)

After performing some straightforward analysis, a clone can also take on a person's rhythms and habits. If you are someone who is generally offline on weekends, evenings, and holidays, your clone will do likewise. It won't run continuously, and you can call it off if you are about to catch a flight, so an adversary will not be able to infer easily which activities are not yours. The clones will resume when you do. (For an explanation of why we now are talking about *multiple* clones, see below.) Of course, you can also select classes of activities in which your clones will not engage, lest the actors feigning to be you pirate some media content, begin to search for instructions on how to manufacture bombs, or look at pornography, unless they must do so to maintain plausibility—making all one's clones clean-living, serious-minded network users interested only in history, charitable giving, and recipes might raise suspicions. (The reason we have switched from talking about a singular clone to speaking about multiple clones is that once one clone is up and running there will be many others. Indeed, imagine a Borgesian joke in which sufficiently sophisticated clones, having learned from your history, demography, and habits, create clones of their own—copies of copies.) It is in your interest to expand this population of possible selves, leading lives that could be yours, day after day. This fulfills the fundamental goal outlined by the patent: your clones don't dodge or refuse data gathering, but in complying they pollute the data collected and reduce the value of profiles created from those data.

2.13 Vortex: cookie obfuscation as game and marketplace

Vortex—a proof-of-concept game (of sorts) developed by Rachel Law, an artist, designer, and programmer²⁸—serves two functions simultaneously: to educate players about how online filtering systems affect their experience of the Internet and to confuse and misdirect targeted advertising based on browser cookies and other identifying systems. It functions as a game, serving to occupy and delight—an excellent venue for engaging users with a subject as seemingly dry and abstract as cookie-based targeted advertising. It is, in other words, a massively multi-player game of managing and exchanging personal data. The primary activities are “mining” cookies from websites and swapping them with other players. In one state of play, the game looks like a

few color-coded buttons in the bookmarks bar of your browser that allow you to accumulate and swap between cookies (effectively taking on different identities); in another state of play, it looks like a landscape that represents a site as a quasi-planet that can be mined for cookies. (The landscape representation is loosely inspired by the popular exploration and building game Minecraft.)

Vortex ingeniously provides an entertaining and friendly way to display, manage, and share cookies. As you generate cookies, collect cookies, and swap cookies with other players, you can switch from one cookie to another with a click, thereby effectively disguising yourself and experiencing a different Web, a different set of filters, a different online self. This makes targeted advertising into a kind of choice: you can toggle over to cookies that present you as having a different gender, a different ethnicity, a different profession, and a different set of interests, and you can turn the ads and “personalized” details into mere background noise rather than distracting and manipulative components that peg you as some marketer’s model of your identity. You can experience the Web as many different people, and you can make any record of yourself into a deniable portrait that doesn’t have much to do with you in particular. In a trusted circle of friends, you can share account cookies that will enable you to purchase things that are embargoed in your location—for example, video streams that are available only to viewers in a certain country.

Hopping from self to self, and thereby ruining the process of compiling demographic dossiers, Vortex players would turn online identity into a field of options akin to the inventory screens of an online role-playing game. Instead of hiding, or giving up on the benefits that cookies and personalization can provide, Vortex allows users to deploy a crowd of identities while one’s own identity is offered to a mob of others.

2.14 “Bayesian flooding” and “unselling” the value of online identity

In 2012, Kevin Ludlow, a developer and an entrepreneur, addressed a familiar obfuscation problem: What is the best way to hide data from Facebook?²⁹ The short answer is that there is no good way to *remove* data, and wholesale withdrawal from social networks isn’t a realistic possibility for many users. Ludlow’s answer is by now a familiar one.

“Rather than trying to hide information from Facebook,” Ludlow wrote, “it may be possible simply to overwhelm it with too much information.” Ludlow’s

experiment (which he called “Bayesian flooding,” after a form of statistical analysis) entailed entering hundreds of life events into his Facebook Timeline over the course of months—events that added up to a life worthy of a three-volume novel. He got married and divorced, fought cancer (twice), broke numerous bones, fathered children, lived all over the world, explored a dozen religions, and fought for a slew of foreign militaries. Ludlow didn’t expect anyone to fall for these stories; rather, he aimed to produce a less targeted personal experience of Facebook through the inaccurate guesses to which the advertising now responds, and as an act of protest against the manipulation and “coercive psychological tricks” embedded both in the advertising itself and in the site mechanisms that provoke or sway users to enter more information than they may intend to enter. In fact, the sheer implausibility of Ludlow’s Timeline life as a globe-trotting, caddish mystic-mercenary with incredibly bad luck acts as a kind of filter: no human reader, and certainly no friend or acquaintance of Ludlow’s, would assume that all of it was true, but the analysis that drives the advertising has no way of making such distinctions.

Ludlow hypothesizes that, if his approach were to be adopted more widely, it wouldn’t be difficult to identify wild geographic, professional, or demographic outliers—people whose Timelines were much too crowded with incidents—and then wash their results out of a larger analysis. The particular understanding of victory that Ludlow envisions, which we discuss in the typology of goals presented in second part of this book, is a limited one. His Bayesian flooding isn’t meant to counteract and corrupt the vast scope of data collection and analysis; rather, its purpose is to keep data about oneself both within the system and inaccessible. Max Cho describes a less extreme version: “The trick is to populate your Facebook with just enough lies as to destroy the value and compromise Facebook’s ability to sell you”³⁰—that is, to make your online activity harder to commoditize, as an act of conviction and protest.

2.15 FaceCloak: concealing the work of concealment

FaceCloak offers a different approach to limiting Facebook’s access to personal information. When you create a Facebook profile and fill in your personal information, including where you live, where you went to school, your likes and dislikes, and so on, FaceCloak allows you to choose whether to display this information openly or to keep it private.³¹ If you choose to display the information openly, it is passed to Facebook’s servers. If you choose to keep it

private, FaceCloak sends it to encrypted storage on a separate server, where it may be decrypted for and displayed only to friends you have authorized when they browse your Facebook page using the FaceCloak plug-in. Facebook never gains access to it.

What is salient about FaceCloak for present purposes is that it obfuscates its method by generating fake information for Facebook's required profile fields, concealing from Facebook and from unauthorized viewers the fact that the real data are stored elsewhere. As FaceCloak passes your real data to the private server, FaceCloak fabricates for Facebook a plausible non-person of a certain gender, with a name and an age, bearing no relation to the real facts about you. Under the cover of the plausible non-person, you can forge genuine connections with your friends while presenting obfuscated data for others.

2.16 Obfuscated likefarming: concealing indications of manipulation

Likefarming is now a well-understood strategy for generating the illusion of popularity on Facebook: employees, generally in the developing world, will "like" a particular brand or product for a fee (the going rate is a few U.S. dollars for a thousand likes).³² A number of benefits accrue to heavily liked items—among other things, Facebook's algorithms will circulate pages that show evidence of popularity, thereby giving them additional momentum.

Likefarming is easy to spot, particularly for systems as sophisticated as Facebook's. It is performed in narrowly focused bursts of activity devoted to liking one thing or one family of things, from accounts that do little else. To appear more natural, they employ an obfuscating strategy of liking a spread of pages—generally pages recently added to the feed of Page Suggestions, which Facebook promotes according to its model of the user's interests.³³ The paid work of systematically liking one page can be hidden within scattered likes, appearing to come from a person with oddly singular yet characterless interests. Likefarming reveals the diversity of motives for obfuscation—not, in this instance, resistance to political domination, but simply provision of a service for a fee.

2.17 URME surveillance: "identity prosthetics" expressing protest

The artist Leo Selvaggio wanted to engage with the video surveillance of public space and the implications of facial-recognition software.³⁴ After considering

the usual range of responses (wearing a mask, destroying cameras, ironic attention-drawing in the manner of the Surveillance Camera Players), Selvaggio hit on a particularly obfuscating response with a protester's edge: he produced and distributed masks of his face that were accurate enough so that other people wearing them would be tagged as him by Facebook's facial-recognition software.

Selvaggio's description of the project offers a capsule summary of obfuscation: "[R]ather than try to hide or obscure one's face from the camera, these devices allow you to present a different, alternative identity to the camera, my own."

2.18 Manufacturing conflicting evidence: confounding investigation

The Art of Political Murder: Who Killed the Bishop?—Francisco Goldman's account of the investigation into the death of Bishop Juan José Gerardi Conedera—reveals the use of obfuscation to muddy the waters of evidence collection.³⁵ Bishop Gerardi, who played an enormously important part in defending human rights during Guatemala's civil war of 1960–1996, was murdered in 1998.

As Goldman documented the long and dangerous process of bringing at least a few of those responsible within the Guatemalan military to justice for this murder, he observed that those threatened by the investigation didn't merely plant evidence to conceal their role. Framing someone else would be an obvious tactic, and the planted evidence would be assumed to be false. Rather, they produced *too much* conflicting evidence, too many witnesses and testimonials, too many possible stories. The goal was not to construct an airtight lie, but rather to multiply the possible hypotheses so prolifically that observers would despair of ever arriving at the truth. The circumstances of the bishop's murder produced what Goldman terms an "endlessly exploitable situation," full of leads that led nowhere and mountains of seized evidence, each factual element calling the others into question. "So much could be made and so much would be made to *seem to connect*," Goldman writes, his italics emphasizing the power of the ambiguity.³⁶

The thugs in the Guatemalan military and intelligence services had plenty of ways to manage the situation: access to internal political power, to money, and, of course, to violence and the threat of violence. In view of how opaque

the situation remains, we do not want to speculate about exact decisions, but the fundamental goal seems reasonably clear. The most immediately significant adversaries—investigators, judges, journalists—could be killed, menaced, bought, or otherwise influenced. The obfuscating evidence and other materials were addressed to the larger community of observers, a proliferation of false leads throwing enough time-wasting doubt over every aspect of the investigation that it could call the ongoing work, and any conclusions, into question.