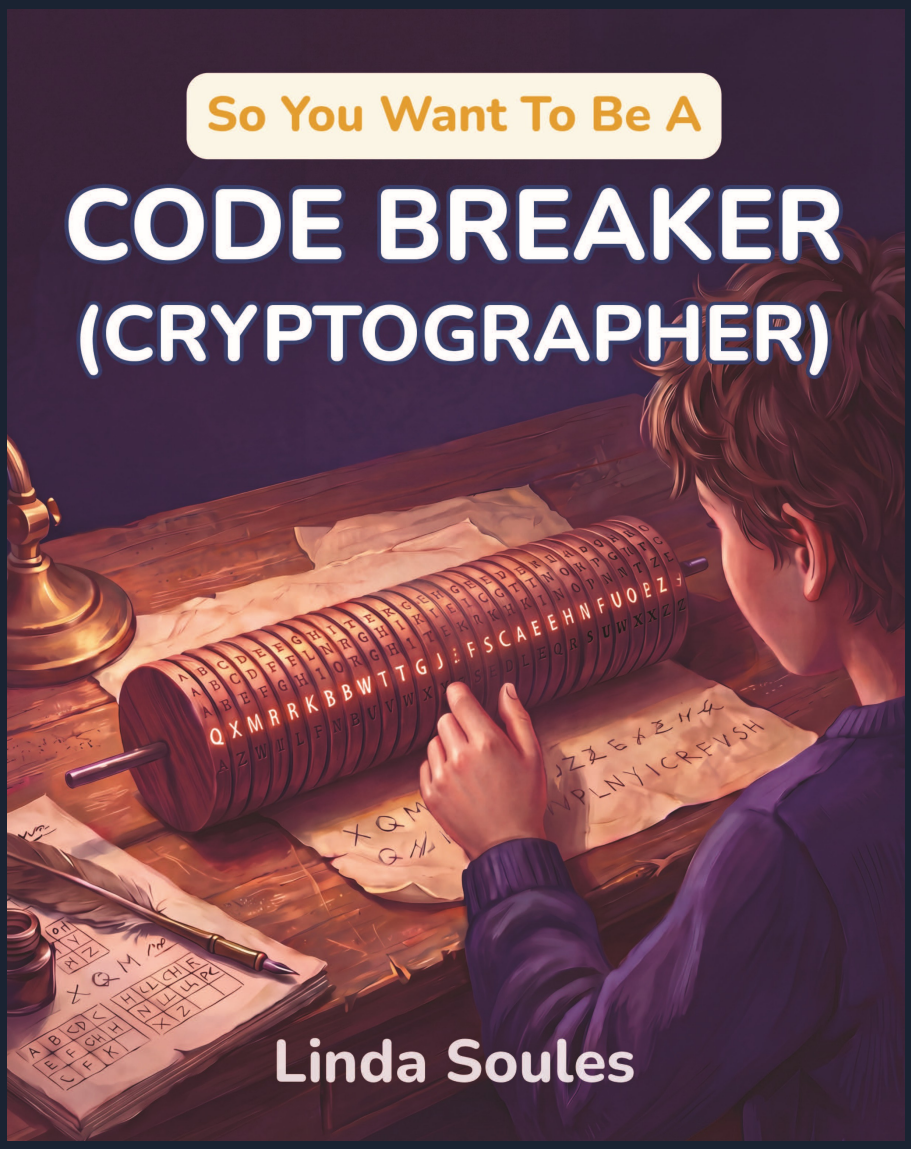
15 min Summary
BookGist.ai



So You Want To Be A Code Breaker (Cryptographer)

Linda Soules

BookGist Summary

So You Want To Be A Code Breaker (Cryptographer)

by Linda Soules

Genre: Children's Nonfiction

Key Takeaways

• Cryptography is both the art of making information unreadable to outsiders and the science of analyzing or breaking systems that hide it. A readable message is plaintext; a scrambled message is ciphertext; a cipher supplies the scrambling rule, and a key sets how it works.

• Code breakers rely on patterns and context as much as computation. Frequency analysis, likely words called cribs, repeated letter groups, and the sender’s habits can all reveal a message or expose a key.

• Modern cryptographers design security systems, prove and test their strength, and try to break them before attackers do. Strong designs are openly examined so weaknesses can be found and fixed.

• Cryptography is a team profession with opportunities in government agencies, technology companies, banking, universities, standards organizations, historical research, and puzzle design.

• The work can be thrilling when a jumble of symbols suddenly resolves into a message, but it also demands patience: investigations may stall for years, and a correct solution may come only after many failed attempts.

• Cryptographic security protects ordinary activities such as banking, messaging, shopping, password storage, and software updates. Its deeper purpose is to protect

privacy, trust, and people's ability to communicate safely.

• The best cryptographers combine pattern recognition, persistence, logical reasoning, creativity, integrity, language skills, clear writing, and the ability to imagine how a system could be exploited.

• Security must anticipate change. Quantum computers could eventually threaten some widely used public-key systems, so researchers are developing and deploying quantum-resistant methods before that risk becomes practical.

• Young readers can begin exploring cryptography now by solving substitution puzzles, practicing frequency analysis, learning basic programming and mathematics, studying languages, and inventing ciphers with friends.

Who Should Read This Book

This book is ideal for curious readers, especially children and teens, who enjoy puzzles, secret messages, mathematics, computers, history, or imagining a future career. It explains technical ideas in accessible language while giving readers practical ways to experiment: decode Caesar ciphers, analyze letter frequencies, build simple codes, and explore beginner programming. Parents, teachers, and librarians can also use it to spark conversations about privacy, trust, teamwork, and how technology affects everyday life. Readers looking for a hands-on introduction will appreciate that the book connects historical episodes—from Enigma and Bletchley Park to the Navajo Code Talkers—with modern encryption and the coming challenge of quantum computing. Unlike a book devoted only to cipher puzzles or a technical textbook focused on algorithms, it combines career guidance, human stories, practical activities, and an honest account of the work's frustrations and responsibilities. It is a welcoming starting point for further reading, study, or simply solving a first secret message with a friend.

Chapter Summaries

What If You Were a Cryptographer?

The book introduces cryptography as an age-old contest between people who hide messages and people who try to uncover them. It explains the basic vocabulary: plaintext is the original message, ciphertext is the scrambled version, a cipher is the rule used to scramble it, and a key is the secret setting that controls the rule. Encryption locks information; decryption unlocks it. The distinction between a cipher, which changes letters, and a code, which substitutes whole words or phrases using a codebook, gives readers a foundation for the

examples that follow. A Caesar cipher demonstrates the principles in action: shifting each letter three places turns A into D, and readers can reverse the shift to decode a sample message. The text also notes that Caesar's successor Augustus used a one-letter shift but handled the end of the alphabet differently. These playful historical details show that even simple systems depend on clear rules—and that knowing the key can turn apparent nonsense back into meaning.

What a Cryptographer Actually Does

A person who breaks a cipher without being given its key practices cryptanalysis. The book presents frequency analysis, described by the scholar al-Kindi around 1,200 years ago, as a classic technique: because some letters occur more often than others in a language, counting symbols can suggest which letters they represent. Code breakers also use cribs—words or phrases they suspect are present—and search for repetitions, predictable message formats, and habits that reveal how a system is being used. Contemporary cryptographers do more than crack old codes. They create encryption methods for phones, websites, and other digital services, build them into software, and use mathematical proofs and attacks to assess their strength. A design is not considered dependable simply because its creator believes in it; it must withstand scrutiny. The work often involves extended concentration at desks, whiteboards, and computers, and the book emphasizes curiosity, persistence, and a willingness to revise an idea when testing exposes a flaw.

Where Cryptographers Work

Cryptographers work across public and private sectors. Government agencies such as the U.S. National Security Agency and Britain's GCHQ protect sensitive information and study foreign communications. Technology companies hire specialists to secure messaging apps, phones, and cloud services; banks and payment companies depend on cryptography to protect transactions; and universities support research and teaching. NIST helps shape the field by organizing competitions that select encryption standards for widespread use. The book also shows that code breaking can lead to less conventional careers. Historians and volunteers examine archives, as did the international team that decoded more than fifty letters written by Mary, Queen of Scots. Puzzle makers create ciphers for games, escape rooms, and newspapers, following in the tradition of Edgar Allan Poe, who challenged readers to send him messages. For students, useful preparation includes mathematics, computer science, and languages; many specialists study these subjects in college and some continue through graduate school. The book stresses that curiosity and determination matter alongside formal qualifications.

The Tools of the Trade

Despite the power of modern computers, a code-breaking investigation may begin with pencil and paper. Analysts arrange a message neatly, count symbols, create frequency charts, and

test possible interpretations. Doing this by hand can help reveal patterns that a computer will not recognize without the right instructions. Cribs and codebooks provide other kinds of leverage: an expected phrase can expose a setting, while a captured list of code words may unlock a system directly. The story of a bored German officer repeatedly reporting that he had nothing to report illustrates how a sender's routine can supply code breakers with a useful daily crib. Computers handle repetitive work at scale, testing keys and analyzing large collections of messages. The book connects their early development to World War II code breaking, including Britain's secret Colossus machines. Whiteboards remain essential for modern cryptographers, who use mathematics involving probability and prime numbers to explain why a design should be secure. Teams debate, erase, and refine their reasoning; a section of the board may be marked "DO NOT ERASE" to preserve an unfinished idea.

The Best Parts of the Job

For many code breakers, the most satisfying moment comes when a long stretch of uncertainty gives way to a solution. The Polish mathematician Marian Rejewski described numbers flowing from his pencil "as by magic" when he worked out the secret wiring of Germany's Enigma machine. The puzzle itself is also part of the appeal: cryptographers can spend their days thinking through difficult problems with colleagues who share their curiosity and humor. The book illustrates how this work can matter at both global and everyday scales. The cipher now known as AES, developed by Joan Daemen and Vincent Rijmen, protects information across billions of devices. In World War II, U.S. Navy code breakers used a deceptive plain-language message about Midway's water supply to confirm that the Japanese code name "AF" referred to the island. The resulting intelligence helped the Navy prepare for the Battle of Midway. The episode shows how painstaking analysis, a clever experiment, and collaboration can turn a small clue into a consequential discovery.

The Hardest Parts of the Job

Cryptanalysis is often slow and frustrating. A promising pattern may lead nowhere, and difficult problems can occupy experts for years without yielding an answer. The stakes can also be high: a wartime message decoded too late may cost lives, while a software flaw can expose large numbers of people. The Heartbleed bug, discovered in 2014, arose from a small programming mistake in widely used website-security software. It had been present for two years and could allow attackers to extract fragments of private information, including passwords. Security work does not end when a system launches. Cryptographers keep looking for weaknesses, communicate risks, and help develop fixes. Government roles may add strict citizenship, background-check, and secrecy requirements, meaning employees cannot freely discuss their work with family. The field also requires years of study and continuous learning because methods once considered secure may later be broken. The book's practical lesson is to become comfortable with uncertainty and to recognize incremental progress—a single understood letter, for example—as a meaningful step.

The Most Surprising Thing About the Job

A strong cryptographic system should remain secure even when its design is public; the key, not the method, is what must remain secret. This principle, associated with Auguste Kerckhoffs, is compared to a combination lock: people may know how the lock works, but they cannot open it without the combination. Accordingly, modern cryptographers publish their designs and invite others to attack them. Public contests put proposed systems through years of scrutiny, and failures are useful because they reveal vulnerabilities before a system becomes widely relied upon. The chapter also looks ahead to quantum computing. A sufficiently powerful quantum computer could threaten some public-key cryptography, which the book compares to a mailbox that anyone can use to send a letter but only its owner can open. Researchers are preparing for this possibility partly because attackers might collect encrypted communications now and decrypt them later. NIST has therefore selected quantum-resistant standards, and organizations are beginning the lengthy transition to them. The central insight is that security is ongoing: it requires anticipating future threats, testing assumptions, and replacing outdated protections in time.

A Day in the Life

A fictional day at a messaging-app company makes cryptography work concrete. The cryptographer reads a research paper claiming a weakness in an existing system, brings it to the team, and helps decide how to test it. In an attack session, the narrator examines a colleague’s new key-handling design and finds a flaw; the colleague is glad it was caught before the design reached millions of users. Later testing shows the paper’s reported weakness is real but not immediately dangerous, so the team documents it and plans a stronger setting. The day also includes the less tidy realities of the work. A quantum-resistant update fails on older phones, and the team cannot fix it before going home. The narrator later realizes the likely cause while brushing their teeth and writes down the idea for the next day. At home, a child sends a Caesar-shifted message, turning the family into part of the puzzle. The episode shows that cryptography involves reading research, collaboration, testing, documentation, debugging, and sometimes letting a problem rest until a new insight appears.

The People You Work With

Code breaking has always depended on varied talents. At Bletchley Park, nearly 10,000 people—including mathematicians, language specialists, crossword solvers, chess players, and clerks—worked together, and roughly three-quarters were women. Polish mathematicians had made crucial early discoveries about Enigma; Alan Turing and Gordon Welchman developed bombes to search for daily settings; and Wrens operated the machines around the clock. The history demonstrates that major breakthroughs often depend on people contributing different kinds of expertise. Modern teams likewise include cryptographers, software engineers, security testers, lawyers, and policy experts. Collaboration requires being

willing to challenge one another's ideas. The development of RSA is a striking example: Ron Rivest and Adi Shamir proposed designs while Leonard Adleman tried to break them. Rivest eventually developed a system based on two large prime numbers: multiplying them is easy, but reversing the process to recover the primes is extremely difficult. Named for its three inventors, RSA became an important protection for internet communications.

How Cryptographers Help People

Cryptography quietly protects many ordinary parts of digital life. Encryption secures information sent to websites; chip cards make transactions harder to reuse fraudulently; end-to-end encryption limits message access to the sender and recipient; password hashes make stolen password databases less useful; and digital signatures help devices distinguish genuine software updates from malicious imitations. These systems allow people to communicate, shop, bank, and share medical information with greater confidence. The book also highlights code making as a matter of safety, dignity, and trust. Navajo Code Talkers used their language to create a military code that the enemy never broke, even though many had been punished in childhood for speaking Navajo. In everyday life, secure communication can protect a journalist's sources, a patient's medical results, or a child's private message to a friend. Cryptography's value is not only that it hides information, but that it gives people confidence that their words will reach the intended person.

Skills, Stories, and Getting Started

The book describes the habits and abilities that help people thrive in cryptography: spotting patterns, persisting through failed attempts, reasoning carefully, thinking creatively, and handling secrets responsibly. Language knowledge can reveal grammatical patterns that pure mathematics misses; strong writing helps explain designs and vulnerabilities clearly; and "thinking like a sneak" means considering how an attacker might exploit an overlooked opening. Stories of figures such as al-Kindi, Elizabeth Smith Friedman, Marian Rejewski, Joan Clarke, William D. Coffee, and Shafi Goldwasser show the field's range, from historical code breaking and wartime intelligence to mathematical proofs and privacy-preserving zero-knowledge systems. Readers are invited to explore famous successes and mysteries, including the once-mysterious Copiale manuscript, the still-unsolved symbols sent to Edward Elgar, the Voynich manuscript, and hidden messages in sculpture and gravestones. Hands-on activities include making a Spartan scytale or invisible ink, constructing a cipher wheel, solving cryptograms, counting letter frequencies, learning programming, studying prime numbers and languages, and practicing strong passwords with an adult. The book's closing encouragement is that a future cryptographer does not need special credentials to begin: a pencil, a puzzle-loving friend, patience, and a willingness to enjoy getting stuck are enough to start.

Notable Quotes

> *"“Every secret message starts with the words you want to send, called the plaintext.”"*

> *"“When a design breaks, its makers learn from the failure and try again.”"*

> *"“A good system, he said, should stay safe even if the enemy knows exactly how it works. Only the key needs to be secret.”"*

> *"“A big part of a cryptographer's job is cheering when someone finds a flaw, because now it can be fixed.”"*

> *"“Keeping someone’s words safe is a way of respecting them.”"*

> *"“The good ones learn to enjoy the struggle and celebrate small wins, like a single letter that finally makes sense.”"*

> *"“Every coded message was written by someone with something important to say, and one particular reader they trusted to understand it.”"*

> *"“The best ones learn to enjoy being stuck a little, because they know how good it feels when the first letter finally falls into place.”"*

Full Summary

A cryptographer may spend the day staring at scrambled letters, hoping to find the message concealed inside them—or designing a lock strong enough to resist anyone trying to get in. For thousands of years, generals, queens, and merchants have hidden messages, while other people have tried to uncover them. Today that contest reaches into ordinary life: when people send texts, shop online, or log into games, cryptographers’ mathematics helps keep their information from strangers.

A secret message begins as plaintext: the readable words someone wants to send. A cipher is the rule that scrambles them, producing ciphertext. Encrypting locks the message; decrypting turns it back into readable form. The key is the secret setting that determines how the cipher works. Julius Caesar reportedly used a substitution cipher, shifting each letter three places along the alphabet, so A became D and B became E. His key was the shift of three. Anyone who knew it could read his messages quickly; without it, the letters looked like nonsense. Caesar’s great-nephew Augustus used a shift of one, but when he reached the alphabet’s last letter, he did not loop back to A. He simply wrote AA.

A code differs from a cipher: rather than swapping individual letters, it replaces whole words or phrases, often using a codebook. A codebook might say that “eagle” means “attack at dawn” and “pancakes” means “run away.” Both codes and ciphers are part of a code

breaker’s territory.

Breaking a system without being given its key is called cryptanalysis, and it often begins with noticing patterns. Around twelve hundred years ago, the Baghdad scholar al-Kindi described a method based on the fact that some letters occur more often than others in any language. In English, E is most common, followed by T and A. If one symbol appears unusually often in a ciphertext, it may stand for E. Counting how often symbols appear is called frequency analysis.

Code breakers also look for cribs: words they can guess might be hidden in a message. A weather report may contain “weather”; a letter may begin with a greeting and end with a name. One plausible word can give a breaker a way to work backward toward the key. Repeated letter groups offer more clues, as do the habits of the people sending the messages. A sender who uses the same sign-off, or reuses a key, can leave a trail. Some messages yield quickly; others resist for months, and some have baffled experts for centuries.

Modern cryptographers do not only break codes; they design them, test them, and try to break them before anyone else can. The systems protecting phone calls, bank cards, and websites use mathematics so complex that guessing the key could take today’s computers longer than the age of the universe. Cryptographers develop these systems, write mathematical proofs of their strength, and build them into software. Then they attack other people’s designs—and invite others to attack their own. If a system fails, its makers learn from the flaw and try again. Many cryptographers also teach and research at universities, spending much of their working time with mathematics, whiteboards, and keyboards.

Government agencies employ many cryptographers. In the United States, the National Security Agency protects government secrets and studies foreign messages; it says it hires more mathematicians than any other employer in the country. Its secrecy once inspired the joke that NSA stood for “No Such Agency.” Britain’s GCHQ grew out of the code-breaking work at Bletchley Park during World War II. Technology companies hire cryptographers to protect messaging apps, cloud storage, and phones, while banks and credit-card companies need them to help keep money safe. Universities support teaching and research, and the National Institute of Standards and Technology, or NIST, runs worldwide contests to select new encryption methods.

Preparation often begins with mathematics, computer science, and foreign languages. Some agencies and technology companies offer summer programs for high school students. A future cryptographer does not have to be the fastest student in math class: curiosity and stubbornness matter, too. Many professionals study mathematics or computer science in college, and some continue to a master’s degree or PhD. In 2025, the median salary for mathematicians in the United States was about $127,000 a year, meaning half earned more and half less.

The work also turns up in less expected places. Historians and amateur code breakers search libraries and archives for unread letters. In 2023, three volunteers—a French computer scientist, a German music professor, and a Japanese physicist—decoded more than fifty letters Mary, Queen of Scots, had written in code to supporters while imprisoned in England. The letters had remained for centuries in France's national library, misfiled as Italian, though they were written in French. Other code enthusiasts create puzzles for escape rooms, games, and newspapers. Edgar Allan Poe invited readers to send him secret messages in 1839 and 1840, claiming he solved nearly all of them. His later story "The Gold-Bug," about a coded message leading to buried treasure, helped spark a cipher craze.

Despite supercomputers, a code breaker may begin with pencil and paper. They copy a message into neat rows, count each symbol, and make a frequency chart—a bar graph showing which symbols occur most often. Comparing that chart with the usual frequencies of English letters can suggest possible substitutions. Breakers test guesses, cross them out, and try again. Working by hand can help reveal patterns a computer would not know to seek.

They keep lists of possible cribs, too. At Bletchley Park, code breakers reportedly benefited from a German officer in the Egyptian desert who sent the same sort of report day after day: he had nothing to report. His predictable message supplied a fresh crib each morning. A captured codebook could provide even more direct help. In 1942, three men aboard a British warship retrieved two codebooks from a sinking German submarine. One of them, Tommy Brown, was a canteen assistant who was only sixteen.

Computers handle the heavy calculations. They can test millions of possible keys each second, search thousands of messages for repeated patterns, and run the mathematics used in modern encryption. Code breakers write programs to test new attacks. The first electronic computers were built partly to help break codes in World War II. Britain's Colossus machines filled entire rooms, and their existence remained so secret that most people did not know about them. After the war, most were dismantled, and the people who had worked on them stayed silent for about thirty years.

For modern cryptographers, the whiteboard is another essential tool. Before a system is built, its designers must use mathematics involving patterns, probability, and prime numbers—numbers divisible evenly only by one and themselves—to reason about its strength. A proof is a step-by-step argument in which every step is checked and none is skipped. Teams debate, erase, and redraw their ideas, sometimes reserving a corner marked "DO NOT ERASE" for a line of thought that is not finished yet.

For code breakers, the best part of the job is often one sudden moment after days or weeks of staring at a jumble. A guess fits; one letter emerges, then a word, and soon a whole sentence seems to pour out faster than it can be written down. The Polish mathematician Marian Rejewski knew that feeling when he worked out the secret wiring inside Enigma, the machine

Germany used to scramble military messages. Numbers, he wrote, began to flow from his pencil “as by magic.”

The puzzle itself is part of the pleasure, especially when it can be shared with clever, funny colleagues. Teams may trade jokes, hold puzzle contests, or rely on one person’s reliably terrible puns. There is satisfaction, too, in building a system that people can trust even if they never know who designed it. In the late 1990s, Belgian researchers Joan Daemen and Vincent Rijmen entered a worldwide contest with a cipher they called Rijndael, a combination of their last names. It won. Renamed AES, their cipher now helps protect Wi-Fi networks, phones, and secret government files. A design worked out by a small team is running inside billions of devices.

Sometimes the thrill is knowing that a solution mattered at a decisive moment in history. In 1942, U.S. Navy code breakers in Hawaii had spent months working on Japan’s naval code. They could tell that Japan planned to attack a place identified as “AF,” but they did not know where it was. They suspected Midway Island, so Midway sent a fake plain-English message saying its water supply had broken. Soon, the code breakers intercepted a Japanese message reporting that AF was short of water. Their guess was confirmed, and the Navy prepared. The Battle of Midway became a turning point in the Pacific war.

Some unexpected code breakers helped make that work possible. About twenty members of the Navy band survived when the battleship USS California sank at Pearl Harbor. Rather than sending them to another ship, the Navy assigned them to a code-breaking basement. No one is sure why, but musicians spend their lives noticing patterns on a page, and the band members proved good at recognizing them. They exchanged their instruments for codebooks and helped crack Japan’s naval code in time for Midway.

The work can also be deeply frustrating. A pattern may lead nowhere after weeks of effort, or a problem may remain unsolved for years. Some famous puzzles have resisted experts for centuries, and others may never yield their secrets. Meanwhile, the stakes can be serious: during wartime, a message decoded too late could cost lives; in everyday technology, one software mistake can expose private information on a huge scale.

That danger became clear in 2014, when researchers discovered a small programming error in software protecting a large portion of the world’s websites. Nicknamed Heartbleed, the flaw even acquired a dripping red-heart logo. It allowed attackers to trick websites into revealing scraps of private information, including passwords. The bug had gone unnoticed for two years before companies around the world rushed to repair it. Cryptographers cannot simply finish a system and forget it. A design may keep protecting millions of people for years, so they must continue watching for weaknesses, warning users, and helping create fixes.

The work may also require long training and a tolerance for uncertainty. Many jobs demand advanced mathematics and years of graduate study, and even experienced cryptographers keep learning because the field changes quickly. A technique considered safe a decade ago may no longer be. Government code breakers face another difficulty: they often cannot discuss their work. Getting hired by a U.S. agency such as the NSA involves citizenship requirements and extensive checks, including interviews with friends, neighbors, and former bosses, as well as a lie-detector test. Once on the job, they may have to answer their family's questions with a vague "Oh, you know. Stuff." The people who succeed learn to live with not knowing, take satisfaction in small discoveries, and keep trying after a promising idea fails.

One surprising principle of cryptography is that the strongest locks are not built around secret designs. In 1883, Dutch-born professor Auguste Kerckhoffs argued that a system should remain secure even if an enemy knows exactly how it works. Only the key should need to stay secret. It is like a combination lock: anyone can inspect the lock, but the bike remains protected as long as nobody learns the combination.

Modern cryptographers take this idea seriously. They publish their designs and invite people around the world to attack them. In 2016, the National Institute of Standards and Technology began a contest to choose new encryption standards, the official methods companies and governments build into apps, websites, and devices. Teams submitted eighty-two designs, then spent years testing and attacking one another's work. Many entries failed. SIKE, a design that had survived years of scrutiny, was eventually broken by two researchers using one ordinary computer in about an hour. Eight years after the contest began, the first winners became official standards in 2024.

Finding flaws is not treated as an insult; it is part of making a system safer. A bridge tested under heavy trucks and inspected for cracks is more trustworthy than one that has never been challenged. Cryptographers may even cheer when someone breaks a design, because the weakness can then be repaired before attackers exploit it.

They are also preparing for a powerful kind of computer that does not yet exist in its full form. Quantum computers use the strange rules of atoms and tiny particles to solve certain problems far faster than ordinary computers. Today's machines are still limited, error-prone, and sometimes need temperatures colder than outer space. But a sufficiently powerful quantum computer could break public-key cryptography, which protects much of the internet. That system works like a mailbox: anyone can drop a message through the slot, but only the owner has the key to open it.

The danger may arrive even before such a computer is built. In a "harvest now, decrypt later" attack, someone copies encrypted messages today and stores them, hoping to unlock them years from now. Some secrets remain valuable for decades. For that reason, the winners of the standards contest were chosen to resist quantum computers, and companies have

already begun switching to them. Replacing the locks across the internet is a vast undertaking, likely to take many years.

A cryptographer's day can move between urgent security work and ordinary puzzles. On the train to a job at a messaging-app company, you might read a research paper claiming that a system your company uses has a weakness, then bring it to the team meeting for discussion. Some colleagues may doubt the attack would work in practice; another may be concerned enough to test it. Later, you might try to break a teammate Priya's new method for handling keys. If you find a flaw, she may groan, laugh, and thank you: it is far better to repair a weakness now than after millions of people rely on the app.

Even lunch can become an impromptu challenge. A coded note on the refrigerator complains, "To whoever keeps eating my yogurt: I know who you are." In the afternoon, you test the research paper's attack and discover that the researchers were partly right. The weakness is real, but exploiting it would take years of computing time, so it is not an emergency; still, you write up the finding and plan to use a stronger setting. The day may end less neatly. A test of the new quantum-resistant system fails, and messages between two older phones will not decrypt. The team traces the code for hours without finding the cause. At dinner, your son hands you a Caesar-cipher message saying, "I fed my broccoli to the dog." You pretend to struggle because he is delighted to have stumped you. Then, while brushing your teeth, the phone problem suddenly makes sense. You jot down the answer on a toothpaste-smeared sticky note for tomorrow. Sometimes a solution appears only after you stop staring at the problem.

Code breaking has always depended on different kinds of minds working together. During World War II, nearly 10,000 people gathered at Britain's Bletchley Park, and about three-quarters were women. Mathematicians, language specialists, crossword fans, chess players, and clerks all contributed; some recruits had proved themselves by finishing a newspaper crossword in under twelve minutes. Because German Enigma settings changed each day, the work had to begin again every morning. Alan Turing and Gordon Welchman designed enormous clicking machines called bombes to search for the settings, building on discoveries by Polish mathematicians Marian Rejewski, Jerzy Ró■ycki, and Henryk Zygalski. Women known as Wrens, members of the Royal Navy's women's branch, operated the machines day and night.

Modern teams bring together software engineers who build cryptographic ideas into apps, security experts who test them, and lawyers and policy specialists who consider how encryption should be used—including whether police should ever be able to unlock private messages. Collaboration means inviting others to find weaknesses, even when that means dismantling an idea you worked hard to create. In 1977, MIT researchers Ron Rivest, Adi Shamir, and Leonard Adleman set out to invent a new kind of lock. Rivest and Shamir

proposed designs while Adleman tried to break them; by his rough count, he defeated about forty-two. Then Rivest developed an idea based on two enormous prime numbers. Multiplying them is easy for a computer, but reversing the process to recover the original numbers would take even the fastest computers millions of years. Adleman could not break it. Named RSA after its inventors, the system protected a huge share of the internet for decades.

The work reaches into everyday life. A padlock beside a website address signals that encryption is scrambling what you send. Chip cards make each purchase different, so a stolen number cannot easily be reused. End-to-end encrypted messaging keeps messages readable only to sender and receiver, not even the company running the app. Websites store passwords as hashes, scrambled fingerprints that cannot simply be turned back into the original passwords. Phones and game consoles use cryptography to check that updates really came from their makers, rather than accepting harmful software disguised as an update.

Cryptography has also protected people in war. About 420 Navajo Code Talkers in the U.S. Marines used their language to create a code the enemy never broke. Many had been punished as children for speaking Navajo at government boarding schools; the language they had been told to abandon became a powerful secret weapon. Their service remained secret until 1968. More broadly, cryptography protects trust: it can help a grandparent pay a bill, a doctor send test results, a reporter protect a source, or a child share a worry with a friend without strangers listening. Securing someone’s words is a way of respecting the person who trusted you with them.

The qualities that help cryptographers succeed are not limited to advanced mathematics. Pattern spotting can make an overlooked detail decisive. At Bletchley Park, Mavis Lever noticed that one long Italian message contained no letter L. Since Enigma could not encrypt a letter as itself, she reasoned that an operator might have sent a practice message by repeatedly pressing L, producing a scrambled text without any L’s. She was right, and the mistake helped the team work out how the machine was wired. Persistence matters just as much: most guesses fail, and progress may mean ruling out one possibility after filling pages with attempts. Logical thinkers ask what else must be true if a guess is correct, while creative ones try a different angle when familiar methods stall.

Trustworthiness is essential because cryptographers handle information that could harm people if exposed. When they find a weakness, responsible researchers first warn the system’s makers privately and allow time for a fix. Companies may offer bug bounties to reward honest reports. Clear writing is part of the same responsibility: cryptographers explain research, designs, and flaws to other specialists, but also write reports for leaders who may not know the mathematics. Language skills help too, because every language has patterns and habits that can offer clues. A cryptographer also learns to think like a sneak, asking how someone might cheat or exploit a loophole before an attacker does.

You can begin without a security clearance. Try counting letters on a page and comparing their frequencies, solving substitution puzzles, listing prime numbers, or learning to code a program that encrypts a Caesar message and then tests all twenty-five possible shifts. Study another language, make a cipher wheel, or work with a friend or grandparent to invent messages and try to break each other’s. A long passphrase made from four random words is far safer than “123456,” and learning to protect your own passwords is practical cryptographic thinking.

The work can lead in many directions: mathematics, programming, ethical hacking, translation, history, spying, or puzzle-making. Sixteen-year-old Sarah Flannery, for example, won Ireland’s Young Scientist of the Year award in 1999 for a new encryption method. When a weakness was found, she explained it herself in a later contest entry and still won Europe’s top prize for young scientists. Making a system and learning how to break it belong together. Every encrypted message began with someone who had something important to say and hoped a particular person would hear it. The first step toward becoming a code breaker can be as simple as picking up a pencil, making a message, and patiently waiting for its first letter to fall into place.

Generated by BookGist.ai • November 2026

www.bookgist.ai