


Encryption, Security, and Privacy

Steven M. Bellovin
https://www.cs.columbia.edu/~smb

Disclaimer

Everything I say is my opinion alone, and does not represent the opinion of any US government agency.

The “Going Dark” Debate

- For many years, the NSA and the FBI have worried about the spread of cryptography in the civilian world
- On the other hand, encryption is necessary to protect American computers and data
- Is there a problem? If so, is a compromise possible?

It's an Old Debate

- According to some reports, the need for civilian encryption was recognized in 1972 when the Soviets eavesdropped on US grain negotiators
- IBM proposed the "Lucifer" cipher, with 112-bit keys
- After refinement, the key size was 64 bits. NSA wanted 48 instead, to aid in their attacks; IBM and the NSA compromised on 56 bits
- *Is there a way to balance the need to protect American information with the need of law enforcement and intelligence agencies to (lawfully) intercept traffic. Is there even a problem?*

Cryptography is Hard

- Most non-government cryptographers oppose modifying encryption systems to permit government access
- Why? Because cryptography is hard *in the real world*
- Real-world cryptosystems are far more complex than high-level examples—and the complexity leads to trouble

Cryptographic Protocols

- When doing encryption, you need a *protocol*—a stylized set of messages and data formats
- Getting these wrong can result in security problems
- The very first academic paper on the subject (Needham and Schroeder,1978) ended with a warning: "Finally, protocols such as those developed here are prone to extremely subtle errors that are unlikely to be detected in normal operation. The need for techniques to verify the correctness of such protocols is great, and we encourage those interested in such problems to consider this area."
- They were right—a simple flaw in their design went unnoticed for *18 years*

Examples

- Incorrectly padding a short message to match the encryption algorithm's requirements has resulted in security flaws
- Not authenticating every encrypted message has resulted in flaws. (That was the essential flaw recently found in Apple's iMessage protocol.)
- Omitting sequence numbers from encrypted messages has resulted in flaws
- The *existence* of older, "exportable" algorithms in the key and algorithm negotiation protocol has resulted in flaws
- Trying to provide an "additional decryption key" for the government has resulted in flaws

Historical Example: The World War II Enigma Machine

Photo: public domain

Historical Example: The World War II Enigma Machine

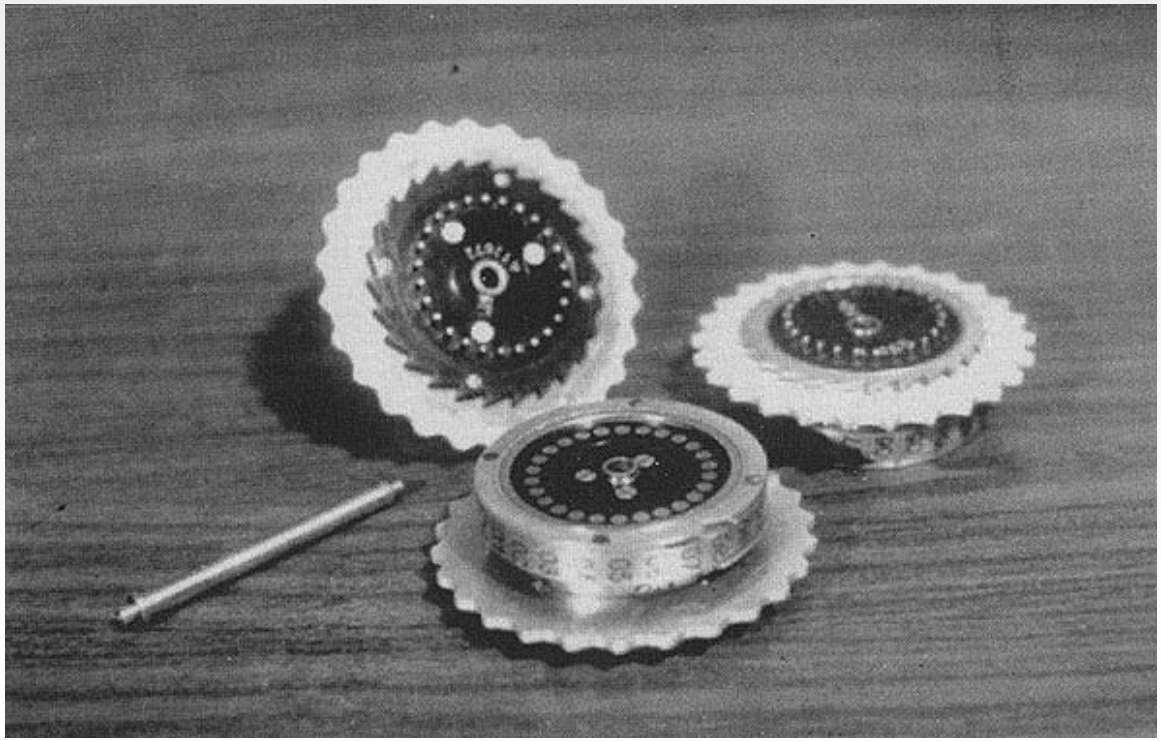

Photo: public domain

You select the proper rotors

Historical Example: The World War II Enigma Machine

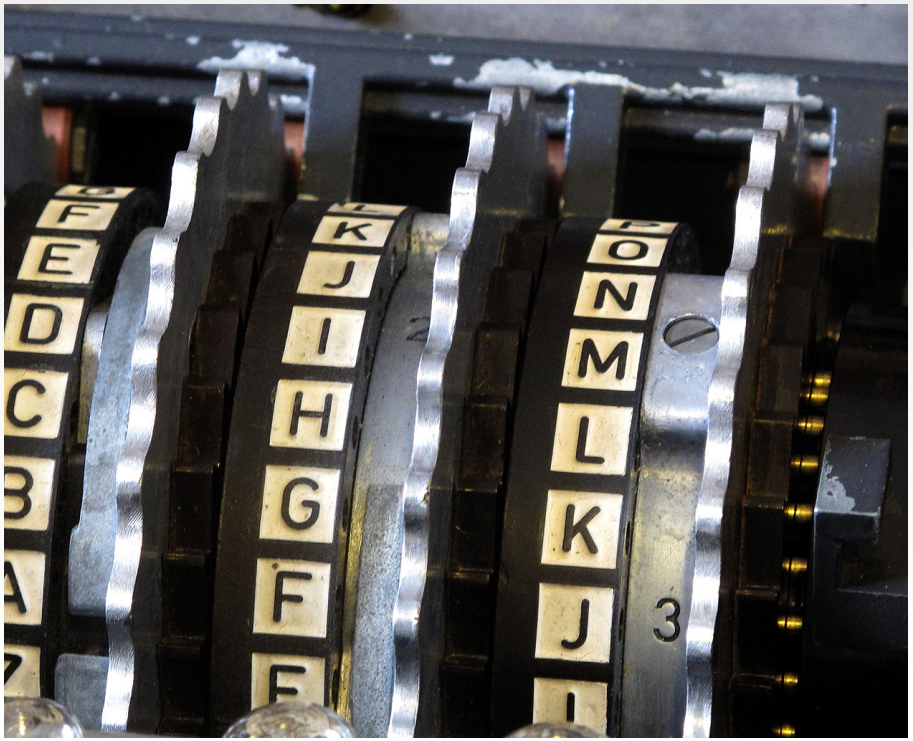

Photo: public domain

Adjust the rotors to their “ground setting”

Historical Example: The World War II Enigma Machine

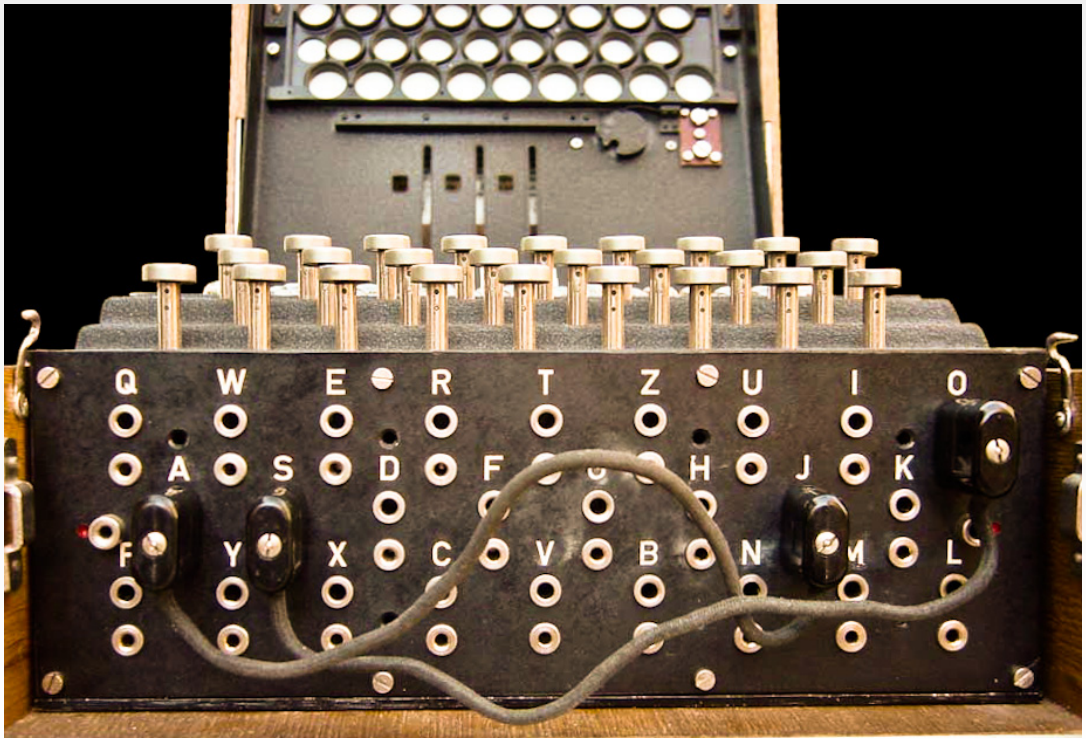

Photo: Bob Lord, via WikiMedia Commons

Set the plugboard

Historical Example: The World War II Enigma Machine

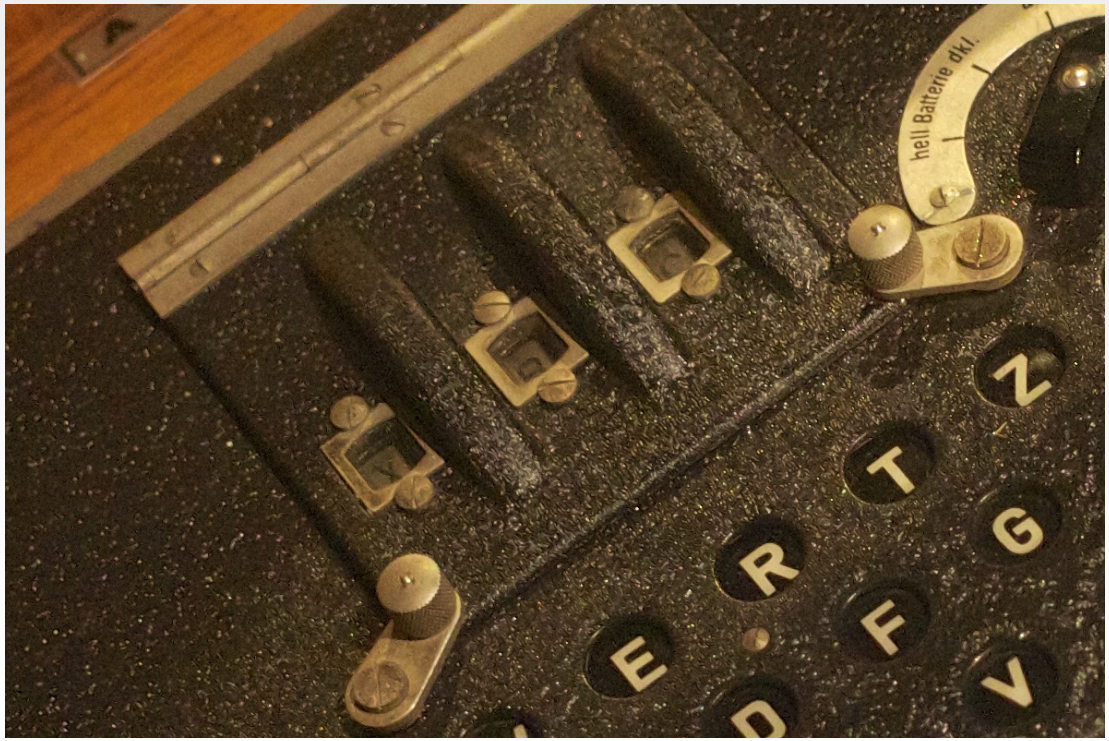

Photo: Paul Hudson, via Flickr

- Pick three random letters and encrypt them twice, and send those six letters as the start of the encrypted message
- Reset the rotors to those three letters

What Could Go Wrong?

- Sending the same, simple message every day was a fatal flaw
- Picking non-random letters was a fatal flaw
- Sending a message consisting of nothing but the letter “L” was a fatal flaw
- Encrypting the three letters *twice* was a fatal flaw

The Three Letters

- Imagine that "XJM" was encrypted to "AMRDTJ"
- The cryptanalysts realized that A and D represented the same letter, M and T were the same, and R and J were the same
- This gave away valuable clues to the rotor wiring and the rotor order!

Cryptography is hard…

A Proposed Compromise: Additional Decryption Keys

- Generic name: “exceptional access”
- (Avoids the value judgment implicit in calling it a “back door”, a “front door”, a “golden key”)
- One proposal: Any encryption system should provide an *additional decryption key*, accessible under proper legal safeguards
- First instantiated in the *Clipper Chip* (1993), special hardware that implemented a then-classified encryption algorithm (Skipjack)
  - It had an unexpected flaw in the exceptional access mechanism…

System and Policy Problems

- How do you protect the secret key necessary to use this feature?
- How do you protect it against a major intelligence agency?
- How do you protect the *process* against routinization of access?
  - Manhattan alone has 200 phones the DA wants to decrypt; Sacramento County has 80
  - There are undoubtedly thousands more across the country *today*
  - Will people do the right thing when it's something they do every day, repeatedly? Hint: "rulebook slowdowns" work because normally, people don't follow every last rule…

Which Countries Can Decrypt?

- Who has the right to the decryption key?
- Where the device was sold?
- Where the device is now?
  - Does a new key get installed at the border? How can that be done securely?
  - Twice, I've been in one country but my phone was talking to a cell tower in another across the border
- The citizenship of the owner? How does the encryption code know?
- Will countries trust each other? Not likely…

International Economics

- What about foreign-made cryptography?
  - The majority of encryption products are developed abroad
  - The last time crypto was an issue, in the 1990s, the loss of business to non-US companies was a major factor in loosening export restrictions
- What non-US buyers will want American software if the crypto has an exceptional access facility accessible to the FBI and the NSA?
  - In 1997, the Swedish parliament was *not* amused to learn that they'd purchased a system to which the NSA had the keys
- What will the State Department say to China when it wants its own access?

The Cost of Compliance

- If breaking encryption is too cheap, it is bad for society: “the ordinary checks that constrain abusive law enforcement practices [are]: ‘limited police resources and community hostility.’” (*US v. Jones*, 615 F. 3d 544 (2012), Sotomayor, concurring)
- If it‘s too expensive for the vendor, it inhibits innovation
- Code complexity is also a cost and security problem
- (As forecast, CALEA compliance indeed led to security problems)

Apple versus the FBI: San Bernadino

- When Syed Farook died in a shootout, the FBI found a county-owned iPhone in his car
- The county gave consent to a search, the FBI had a warrant—but the phone was locked (with some data encrypted) and *might* erase everything if the PIN was entered incorrectly 10 times
- Magistrate Judge Pym ordered Apple to produce software that would allow unlimited guesses, with a provision to enter them rapidly
- Apple objected

It’s Not About This One Phone

- There is good reason to believe the FBI will find nothing of interest on this phone
- Building the infrastructure to unlock this single phone is time-consuming and expensive—but once the code exists, it becomes easy to unlock others
- Apple and the FBI both know this.
  - The FBI wants a precedent set in what seems like an ideal case
  - Apple is afraid of exactly that happening

Cost

- Apple estimates that it would take 3-10 person-months to produce the code
  - My own, independent estimate is quite compatible with theirs
  - All iPhone code must be “digitally signed”, using a cryptographic key possessed by Apple
- This, though, is the cost to produce the *first* copy of the software, for this one phone. Each subsequent version would be very cheap
- If the software is not locked to one phone, it *will* become a target of other governments
- If it is locked to one phone, you have the routinization problem

Compelled Speech?

- Is computer code "speech" under the First Amendment, or is it purely functional?
  - The 2nd, 6th, and 9th Circuits have said code can be speech (9th Circuit opinion withdrawn)
  - In all three cases, the code was linked to an political issue
- Apple has *expressed an opinion* that back doors are ethically wrong. Can they be compelled to "say" something they don't believe?
- What about the digital signature?
  - Is that merely a functional access control mechanism?
  - Or is it Apple's attestation that the code meets their standards?
  - Their app store policies and signed apps have been a major reason why iOS has much better security than Android

Subpoenaing the Code and Signing Key

- The FBI has indicated that if Apple won't help it unlock the phone, it will subpoena the code and signing key
- Can the code be subpoenaed? Probably, but producing a usable copy of the code base and build environment is far from easy
- The signing key?
  - There's still the compelled speech issue
  - Apple may not be able to turn it over—best practices dictate keeping such keys in a "Hardware Security Module" (HSM)
  - The whole point of an HSM is to prevent disclosure of a major signing key!

The iCloud Backup

- Farook's phone was backed up to Apple's iCloud about six weeks before the shooting
- iCloud backups are *not* encrypted
  - Customers want to recover their data, even if they've forgotten their PIN
  - Apple's threat model is loss of a device, not hacking of iCloud
- What was done with the phone during those six weeks?
  - An FBI error prevented them from forcing a new backup
- Some apps have data that is (deliberately) not backed up
- But—Apple knows exactly which apps are on the phone, and hence what they can do, where the metadata might be, etc. Statements by law enforcement suggest they think the odds on finding useful information are low.

Apple and Privacy

- Ideological: Tim Cook strongly believes in privacy
  - He also believes in speaking out in the face of injustice—as a child, he tried to intervene in a Klan cross-burning
- People store lots of sensitive data on their phones ("Modern cell phones are not just another technological convenience. With all they contain and all they may reveal, they hold for many Americans "the privacies of life." *Riley v. California*, 134 S. Ct. 2473 (2014))
- Marketing: Privacy is a distinguisher from Google, which earns its revenue from users' personal data
- All of the above? Probably.

It's Not Privacy, It's Security

- Phones hold a lot of sensitive information (passwords, bank account numbers, email account access, etc.)
- The decline of Blackberry and the rise of "Bring Your Own Device" (BYOD) means that corporate data is on phones, too
- Phones are are used as authenticators for network login, sometimes in place of hardware tokens
- Imagine an American business executive crossing the border into a country with an oppressive government—and that government can unlock the phone…

Where Are We?

- This case may be moot, but the issue will arise again
  - News reports suggest that Apple is going to strengthen their security mechanisms
- There's been no thorough, public discussion of the extent to which law enforcement access to metadata can substitute for access to content
  - Some have called this "the golden age of surveillance"
- The debate has often been lawyers and policy makers versus technologists—and they talk past each other
  - We need people who speak both languages!

Further Reading

- Harold Abelson, Ross Anderson, Steven M. Bellovin, Josh Benaloh, Matt Blaze, Whitfield Diffie, John Gilmore, Matthew Green, Susan Landau, Peter G. Neumann, Ronald L. Rivest, Jeffrey I. Schiller, Bruce Schneier, Michael A. Specter, and Daniel J. Weitzner. Keys under doormats: Mandating insecurity by requiring government access to all data and communications. *Journal of Cybersecurity*, 1(1), September 2015. http://cybersecurity.oxfordjournals.org/content/early/2015/11/17/cybsec.tyv009

- Hal Abelson, Ross Anderson, Steven M. Bellovin, Josh Benaloh, Matt Blaze, Whitfield Diffie, John Gilmore, Peter G. Neumann, Ronald L. Rivest, Jeffrey I. Schiller, and Bruce Schneier. The risks of key recovery, key escrow, and trusted third-party encryption, May 1997. https://www.cs.columbia.edu/~smb/papers/paper-key-escrow.pdf

- Susan Landau, Testimony, Hearing on "The Encryption Tightrope: Balancing Americans' Security and Privacy", Judiciary Committee, United States House of Representatives, March 1, 2016. https://judiciary.house.gov/wp-content/uploads/2016/02/Landau-Written-Testimony.pdf

How iPhone Encryption Works

- A random, 256-bit number (the “UUID”) is manufactured into the phone’s processor, and isn’t easily retrievable from outside
- When a PIN is entered, the PIN and the UUID are combined to form a “key-encrypting key” (KEK) via a process that *must* take about 80 milliseconds
- The KEK is used to encrypt the “data-encrypting key” (DEK)
- The DEK is used to encrypt (certain) data on the phone
- The DEKs are useless without the KEK, but the KEK can only be calculated (a) using the PIN, and (b) using the UUID not visible externally
- Newer iPhones do key-handling in a special, secure area of the processor