Last lecture!

The New York Times

Facebook Tells Barr It Won't Open Up Encrypted Messages

Dec. 10, 2019, 10:07 a.m. ET

WASHINGTON — Facebook executives told Attorney General William P. Barr on Monday that they would not open up the company's encrypted messaging products to law enforcement, escalating a standoff with the government over privacy and policing.

In a letter from the company, the executives overseeing Facebook's WhatsApp and Messenger, Will Cathcart and Stan Chudnovsky, wrote that creating a so-called backdoor into their services would make their users less safe.

"The 'backdoor' access you are demanding for law enforcement would be a gift to criminals, hackers and repressive regimes, creating a way for them to enter our systems and leaving every person on our platforms more vulnerable to real-life harm," they said in the letter, which was obtained by The New York Times. "It is simply impossible to create such a backdoor for one purpose and not expect others to try and open it."

Hardware

- logical/functional/architectural structure
  - bus connects CPU, RAM, disks, other devices
  - caching
  - CPU cycle: fetch-decode-execute; kinds of instructions
    - toy machine as an example
    - different processor families are incompatible at the instruction level
  - von Neumann: architecture; Turing: equivalence of all machines
- physical implementation; sizes and capacities
  - chips; Moore's law, exponential growth
- analog vs digital
- representation of information
  - bits, bytes, numbers, characters, instructions
  - powers of 2; binary and hexadecimal numbers
  - interpretation determined by context
- it's all bits at the bottom

Software

- algorithms: sequence of defined steps that eventually stops
  - complexity: how number of steps is related to amount of data
    - linear: searching, counting, …
    - quadratic: simple sorting
    - logarithmic: binary search (logarithm = number of bits needed to store)
    - n log n: quicksort
    - exponential: towers of Hanoi, traveling salesman problem, …
- programs and programming languages:
  - evolution, language levels: machine, assembly, higher-level
  - translation/compilation; interpretation
  - a program can simulate a machine or another program
- basic programming, enough to figure out what some code is doing
  - variables, constants, expressions, statements, loops & branches (if-else, while), functions, libraries, components
- operating systems: run programs, manage file system & devices
  - file systems: logical: directories and files; physical: disk blocks
- application programs, interfaces to operating system, APIs

Communications

- local area networks, Ethernet, wireless, broadcast media
- Internet: IP addresses, names & DNS, routing; packets
  - bandwidth
- protocols: IP, TCP, higher-level; layering
  - synthesis of reliable services out of unreliable ones

- Web: URLs, HTTP, HTML, browser
  - caching
- security & privacy: viruses, cookies, spyware, …
  - active content: Javascript, plugins, addons

- cryptography
  - secret key; public key; digital signatures; secure hashes
- compression; error detection & correction

- wireless, cell phones, GPS, …

Real world issues

- **legal**
  - intellectual property: patents, copyrights, contracts, licenses
  - jurisdiction, especially international
- **social**
  - privacy, security
- **economic**
  - open source vs proprietary
  - who owns what
- **political**
  - policy issues
  - balancing individual, commercial and societal rights and concerns

Things to take away

- **some skills, some specific technical knowledge**
  - how computers and communications work today
  - what's ephemeral, what's likely to still be true in the future
- **improved numeracy / quantitative reasoning**
  - what makes sense, what can't possibly make sense, and why
    - plausible estimates, engineering judgment, enlightened skepticism
- **another way of thinking**
  - how do things work?
  - how *might* something work?
  - you can often figure it out
- **some appreciation of tradeoffs & alternatives**
  - you never get something for nothing
- **some historical perspective**
  - everything derives from what came before

- **informed opinions about the role of technology**

Final exam (watch the web page!!!)

- **Wednesday January 22, 9:00 am, Peyton 145**
  - Q/A session sometime (Sunday Jan 19?)
  - watch the web page for schedule
- **similar to midterm but twice as long**
- **open notes (on paper), book, problem sets, labs, old exams, …**
- **bring a calculator if you can — it might make something easier**
- **hints**
  - I'm usually looking for something brief that shows that you understand or can reason
  - if you're writing or calculating a lot, you're likely on the wrong track
  - questions meant to test understanding of basic ideas and critical distinctions
    - meant to be simple and straightforward, not complicated, if you understand
    - not meant to be tricky or rely on obscure facts
  - think about plausibility and where I'm likely coming from
  - if it still seems ambiguous, say "I'm assuming this..." and carry on


THE STUDENT VIEW:
AHH! I HAVE TO TAKE AN EXAM TOMORROW!
THE PROFESSOR VIEW:
AHH! I HAVE TO GIVE AN EXAM TOMORROW!
JORGE CHAM © 2018
WWW.PHDCOMICS.COM

What should be different next time?

- faster or slower?
- more topics or fewer?
- broader or deeper?
- different topics?
  - like what?

- how did the problem sets work out?
  - how would you improve them?
- how did the labs work out?
  - how would you improve them?

- what else would make it better next time?