Nissenbaum and Howe describe obfuscation in today's reading. Obfuscation is a technique that can be used to achieve various privacy and security properties. You might put it alongside cryptography and steganography.

Small mention of interesting things

- Ian Goldberg's simple attack on the WoT (@17:34 - 20:02)
- Janez Janša (@02:08 - 03:19, 50:52 - 51:52, 52:18 - 52:44)
- schedule changes: anonymity is next. No speaker on Thursday, we'll learn about networking instead.
- generate your keys locally for the assignment

"hilarity ensues"

“Obfuscation, at its most abstract, is the production of noise modeled on an existing signal in order to make a collection of data more ambiguous, confusing, harder to exploit, more difficult to act on, and therefore less valuable.”

From Nissenbaum and Howe.

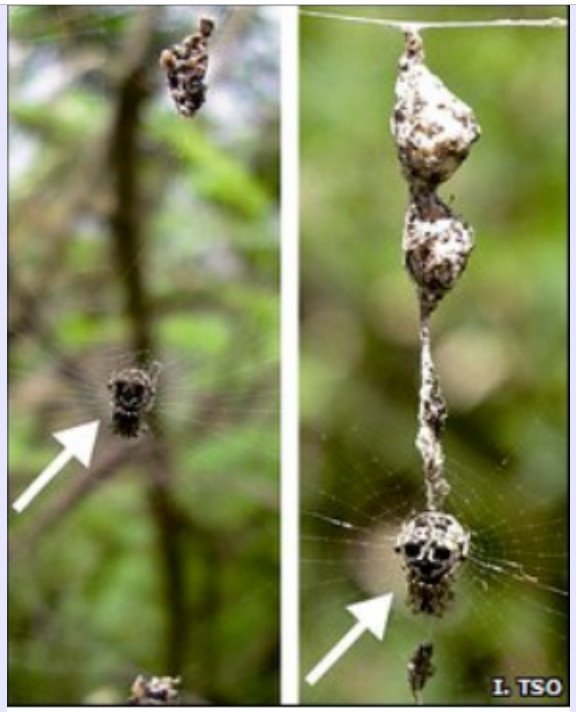
How can a strategy of obfuscation help in achieving anonymity---in a network, in a database?

Obfuscation is especially useful for people on the wrong end of information and power asymmetries, as we’ll see. It is a form of offensive privacy technology, designed to undermine data analysis by undermining its precision (rather than hiding in order to undermine its recall, for example).

“If privacy both as impossible-to-observe and impossible-to-identify is dead, then what might be an alternative? If you’re an optimist or an apparatchik, your answer will tend toward rules of procedure administered by a government you trust or control. If you’re a pessimist or a hacker/maker, your answer will tend toward the operational, and your definition of a state of privacy will be mine: the effective capacity to misrepresent yourself.”

Geer, ‘Identity as Privacy’

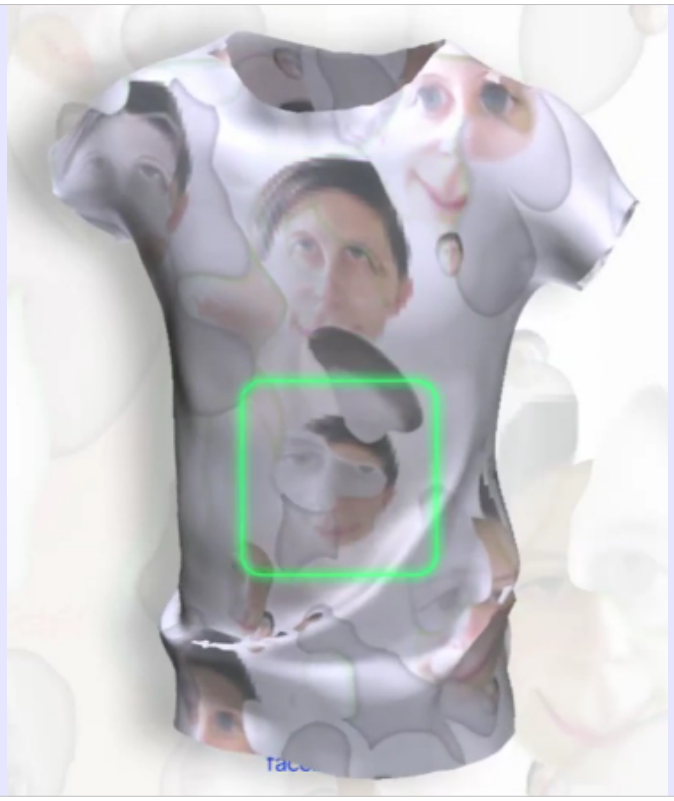
Dan Geer, former head of In-Q-Tel



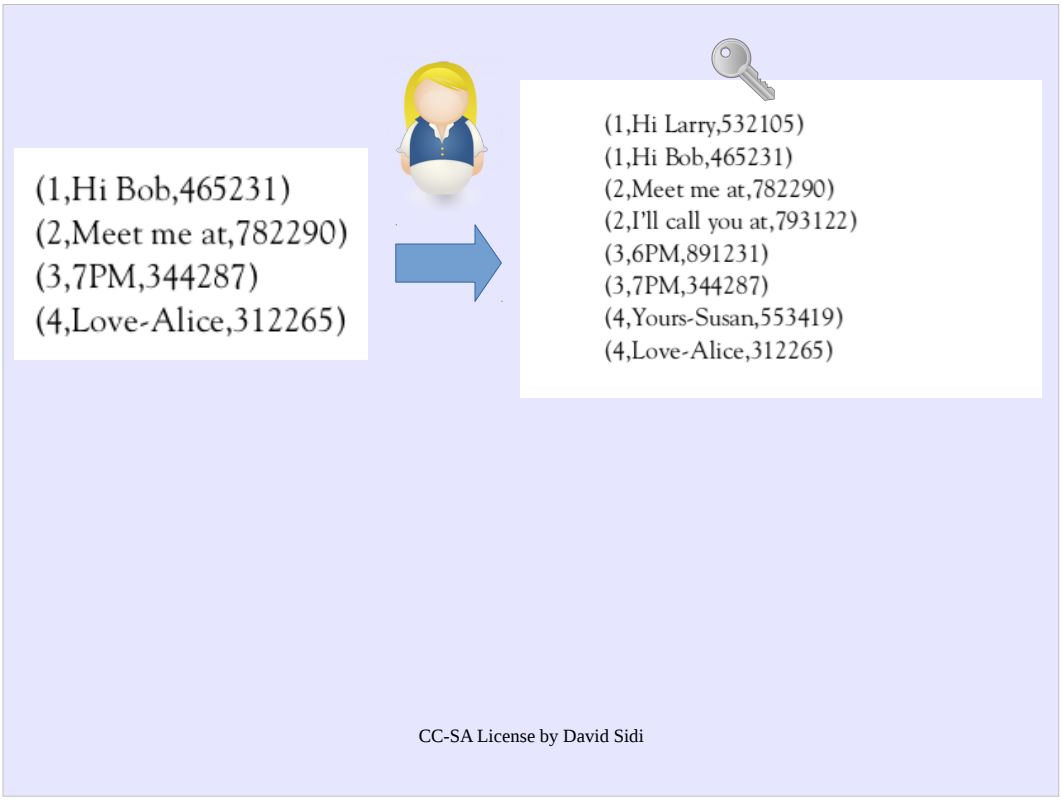
An adult *C. mulmeinensis* alongside decoy prey pellets (L) and decoy egg sacs (R)

Cyclosa mulmeinensis, a decoy-building spider

Glamouflage is related

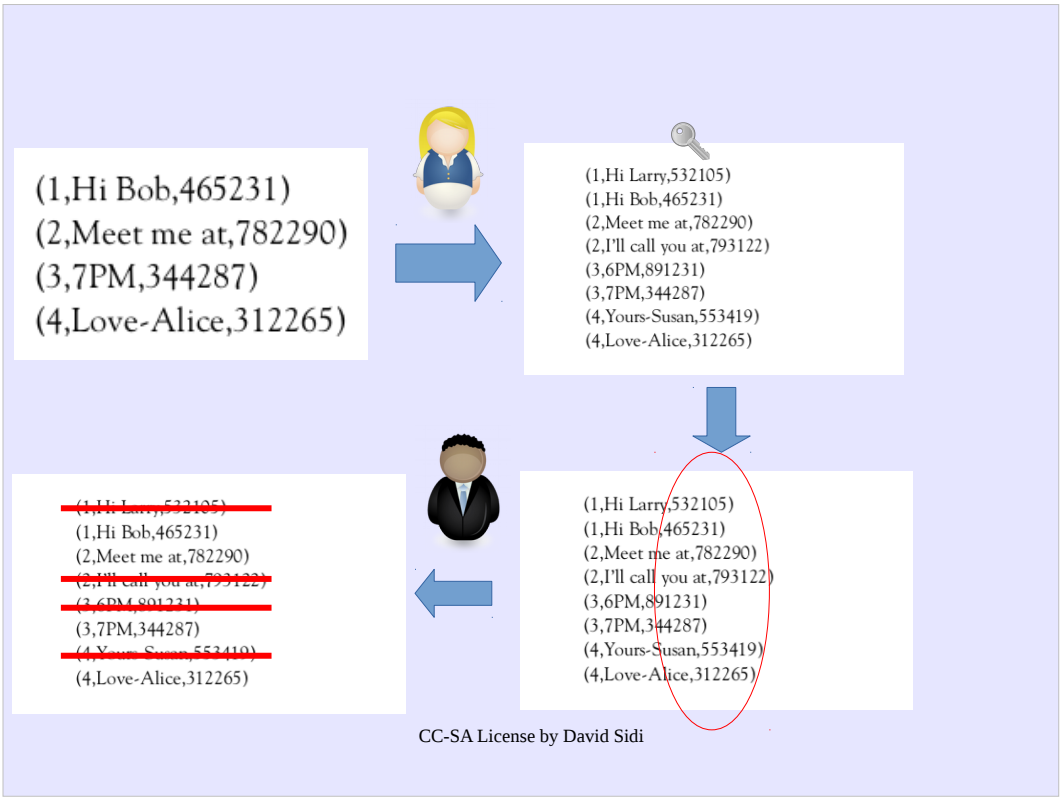


- Ron Rivest's Chaffing and Winnowing (C&W) arose amid concerns about key escrow, clipper chips, and other backdoors for encryption.
- Simple idea: Don't use encryption, use integrity checking to achieve confidentiality over an insecure channel!

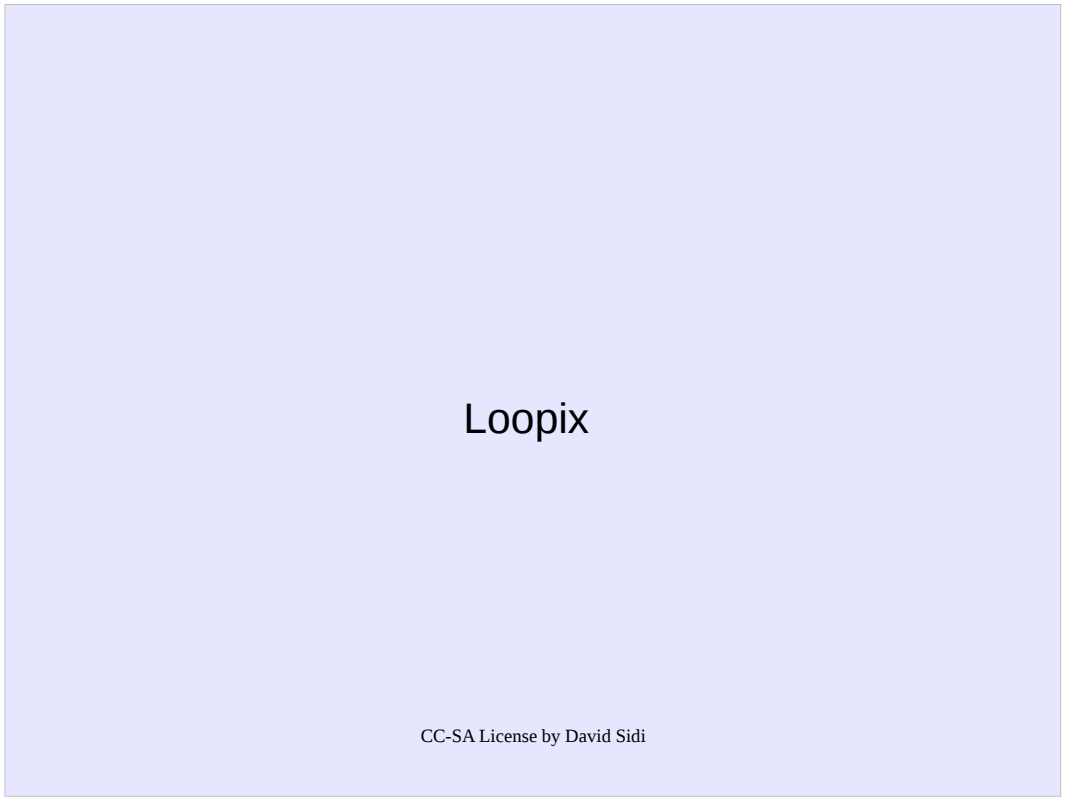


Sending a message has a few parts

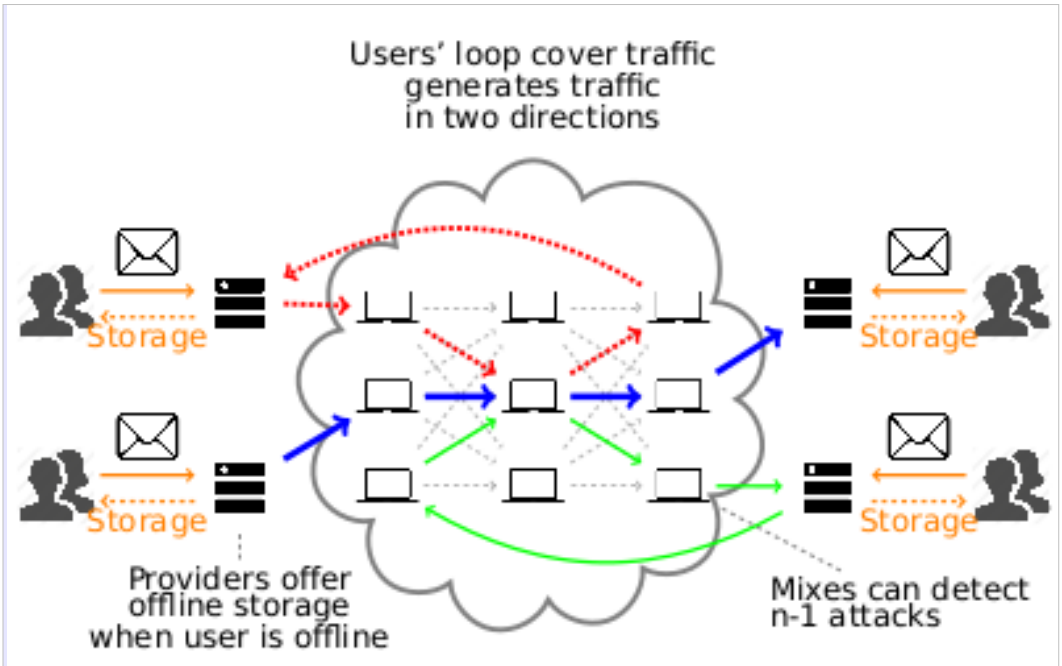
- Message is broken into packets by the sender
  - authenticating (adding MACs)
    - in particular, secure MACs, computed from (a hash of) the packet content (plus ordering information), and a secret key
  - adding “chaff”
    - Chaff packets are not part of the real message



- Receiving a message requires removing the “chaff” by checking MACs
  - The MAC of chaff doesn’t check, so intended recipients can discard them
- reliability for connection-based protocols becomes a basis for confidentiality
- confidentiality depends on how hard it is to distinguish chaff from wheat, not breaking an encryption scheme (there is no encryption, contents are in the clear)
  - MAC should be indistinguishable from a random function to an adversary



A variety of mix network, used for asynchronous (relatively high-latency) communications. Now there is Katzenpost as well.



Cover traffic can help to resist traffic analysis.
This is Loopix, which I've mentioned before.
(Walk through).

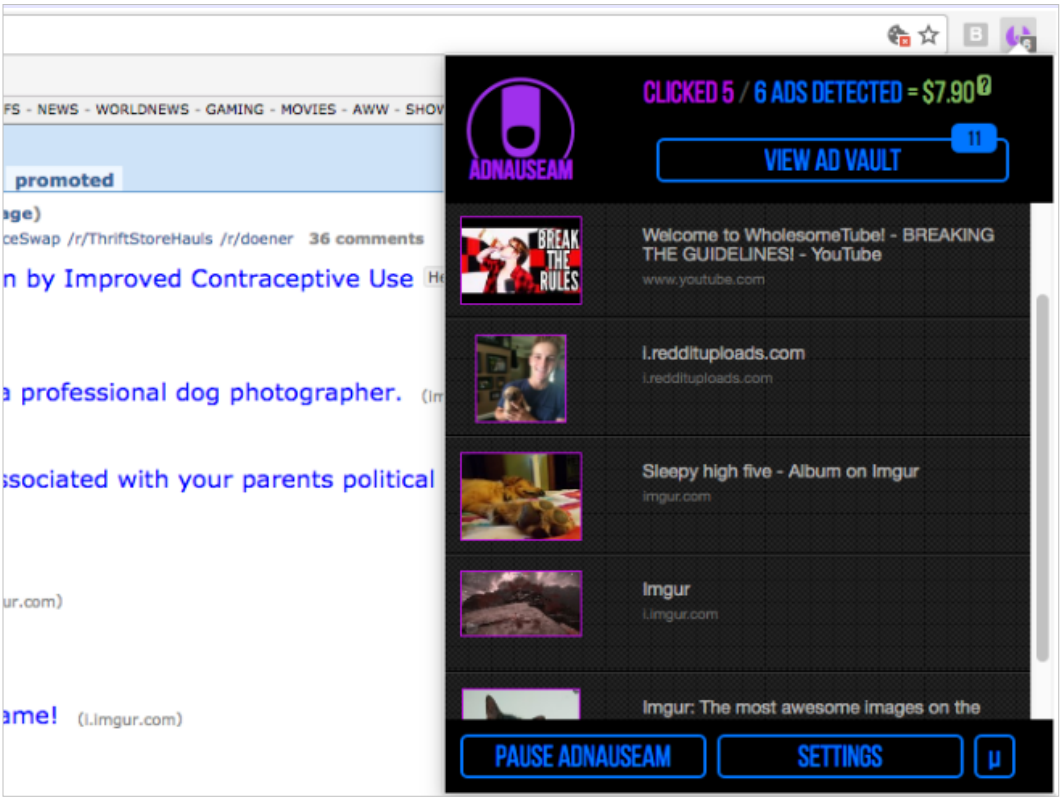
Again, the security properties of the system depend on an adversary's inability to distinguish cover traffic from payloads. (Example: loop traffic).

Ad Nauseum

Ad nauseum. Built on top of uBlock.

Why not just hide ads? Why click on them?
privacy is a social value. What does that mean for privacy technologies?

Idea is: PETs should help everyone (not just individual users of the technology) by damaging ad-based business models. A form of privacy protest. Related: https://heinonline.org/hol-cgi-bin/get_pdf.cgi?handle=hein.journals/arz55&section=32.



adversarial view of advertising: ad networks surreptitiously track, and intentionally avoid alerting you, since you might otherwise object. https://www.youtube.com/watch?v=GAXLHM-1Psk @ 13:56 - 15:00

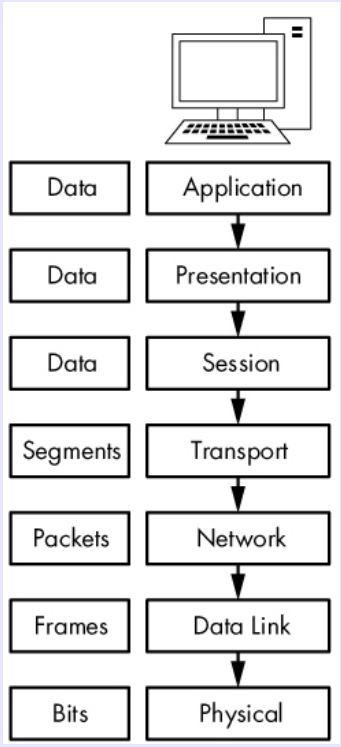

D.C.'09

“Computers” use protocols to communicate

- networking protocol: rules for synchronization, (encryption,) data formatting, error correction
- Examples: TCP, UDP, IP, ICMP, ARP, DHCP, SOCKS, HTTP, FTP, SMTP, DMTP, DMAP ...

Local processes communicate with remote ones (their so-called “peers”).

The Seven-Layer OSI Reference Model classifies protocols by function



We'll mostly stick with a simplification of this model, the TCP/IP protocol stack (book diagram). A stack is composed of layers, which only interact with their immediate neighbor layers, via encapsulation (book diagram).

The TCP/IP networking model has four layers.

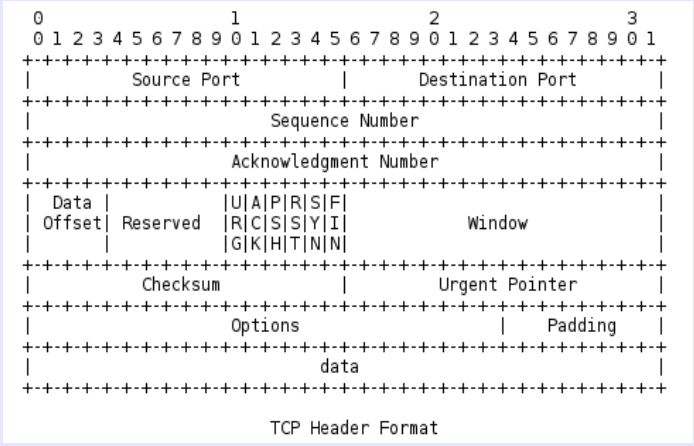
At its beginnings in the application layer, data is generated carrying information in a form useful to a particular network service. The data is a *payload* that progresses "downward" through the other layers in a fixed order, accumulating control information as it goes in the form of layer-specific headers.

Let's look at some concrete details

HTTP

- You can use telnet to manually be the client in an HTTP connection
- try it for yourself!

  $ telnet www.arizona.edu 80
- “A client sends an HTTP request to a server in the form of a request message,
  - beginning with a request-line that includes a method, URI, and protocol version (Section 3.1.1), followed by
  - header fields containing request modifiers, client information, and representation metadata (Section 3.2),
  - an empty line to indicate the end of the header section,
  - and finally a message body containing the payload body (if any, Section 3.3).” (RFC7231)
- The header information becomes data for the next layers



TCP Header Format

show IP header next (book diagram)