Anonymity

Slides from

- Michelle Mazurek 414-fall2016 with material from Dave Levin

- What is anonymity?
- Dining cryptographers
- Mixnets and Tor
- Web/device fingerprinting

What is anonymity?

- An observer/attacker cannot determine who is communicating
- **Sender K-anonymity**: Cannot distinguish sender from set of **K** potential senders
- **Receiver K-anonymity**: Cannot distinguish receiver from set of **K** potential receivers

Sender anonymity

- Ransom note
- Pass a note when teacher is not looking
- Hang fliers / chalk messages late at night
- etc

Receiver anonymity

- Dedicate a book/song/etc to “you know who”
- Codes in classified ads
- Cold war spies: Number stations
- etc

Dining cryptographers

[David Chaum]

Problem setup

- Three cryptographers having dinner
  - Waiter says someone has paid
  - Was it one of them? Or a third party?
- Can one of them admit to paying without the others knowing which one it was?

How to do it

- Each pair of cryptographers flips one coin, hidden from the 3rd person
- Everyone reports “same” or “different” for the two coins they can see
- **Except:** person who paid reports the wrong answer

Why does this work?

```
A : (b_AB XOR b_AC) XOR m
B : (b_AB XOR b_BC)
C : (b_AC XOR b_BC)

All messages:
 (b_AB XOR b_AB) XOR
 (b_AC XOR b_AC) XOR
 (b_BC XOR b_BC) XOR m
  = m
```

Why is this secure?

- Suppose you did not pay
- If the result is 1 (odd “diff”)
  - You can tell one of the others is lying
  - But without coin they share, can’t tell which
- If result is 0 (even “diff”) then no anonymity issue
  - We all know the third party paid

Potential issues

- Unfair coins
- Not executing the protocol honestly

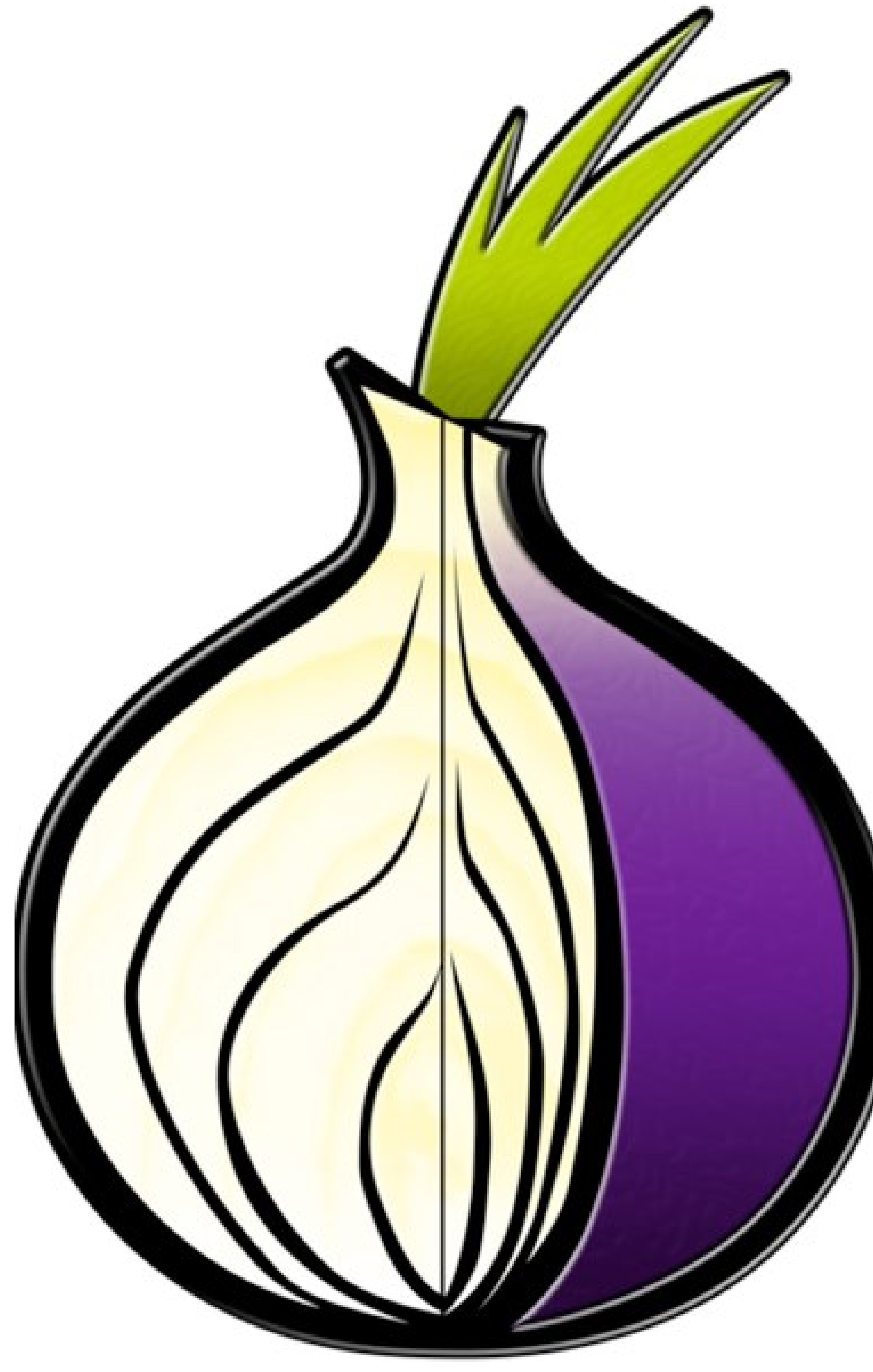
Generalizing the protocol

- More than 3 people:
  - Fine with one shared bit per pair of users
- More than 1 bit of data
  - Proceed in rounds, one bit per round
  - Now we need a shared **key** (one bit per round)
- What about collisions?

Pros and Cons

- Pro: Not interactive
  - After key establishment, no crosstalk by users
  - Make systems simpler, proofs easier
- Pro: Collusion is hard
  - Generally need everyone conspiring against you
- Cons:
  - Collisions / Jamming
  - N^2 shared keys

Mixnets



Problem setup

- One mail server, M
- Lots of senders (S_i) and receivers (R_i)
- One global observer G
- Goal: Send messages without G being able to determine which sender sent to which receiver

Strawman protocol

- Every sender sends a message to M
  - msg indicates intended receiver
  - msg encrypted with M’s pub key
- M waits for all messages; shuffles the order
- Send each msg encrypted for recipient
- Why is this a strawman?

Fixing this protocol (1)

- Problem: M reads all messages
- Solution: Encryption layers
  - $E(k_M, R_i \,||\, E(k_{R_i}, m))$

Fixing this protocol (2)

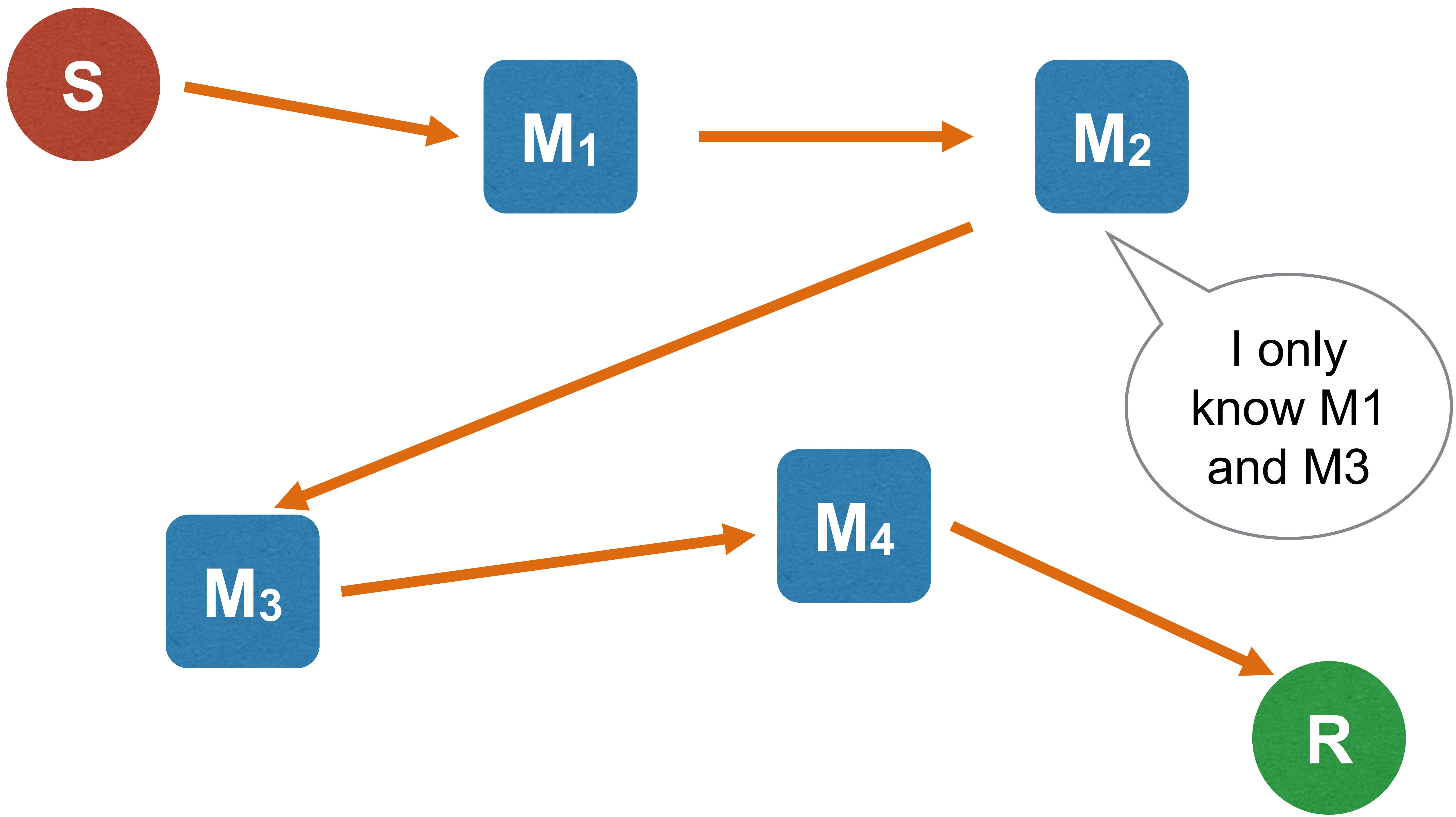
- Problem: What if not everyone has a message
  - Mail server might wait forever!

- Solution: Everyone sends every round
  - Some is labeled as junk
  - Wastes bandwidth/resources on junk

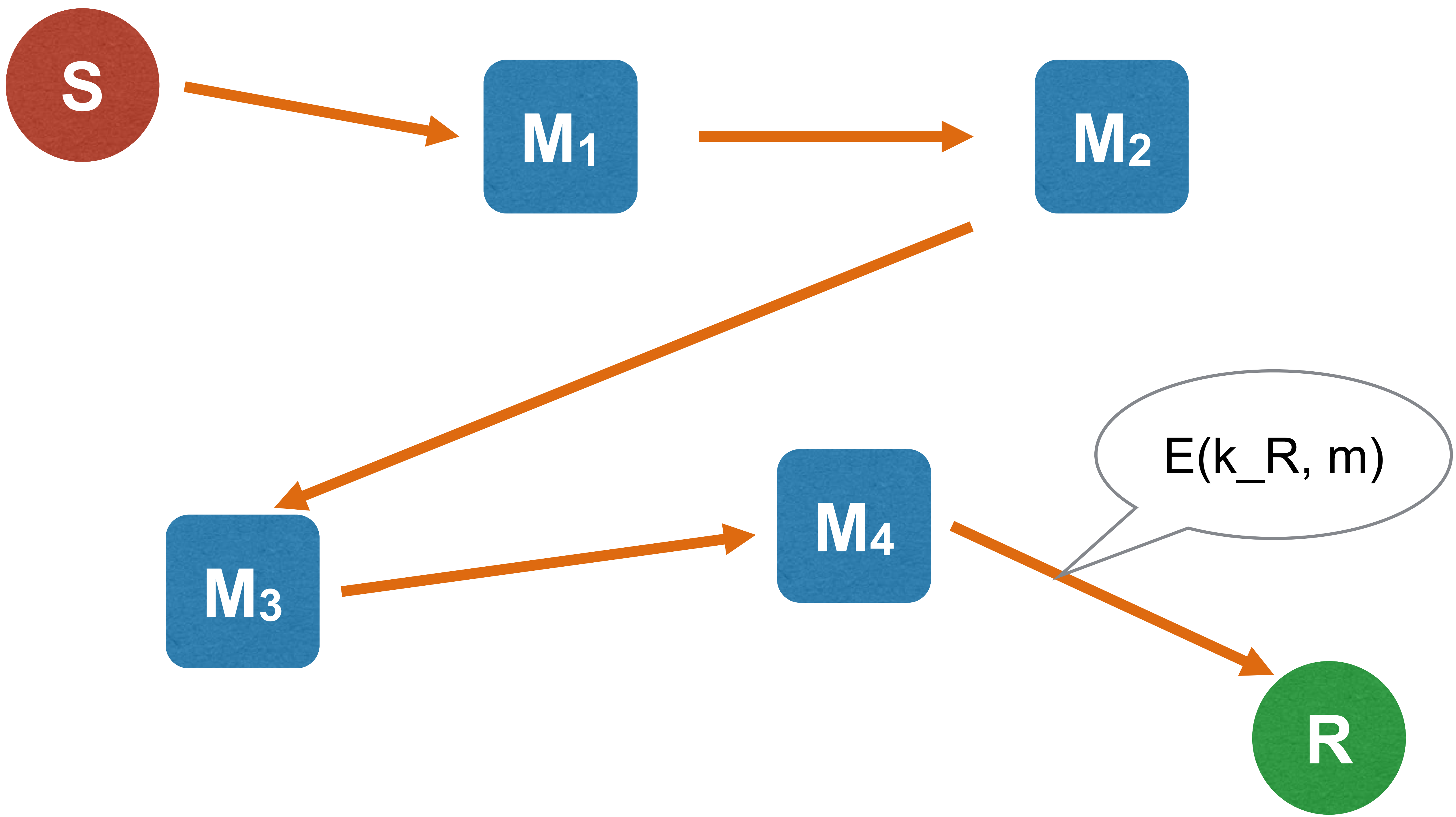
Fixing this protocol (3)

- Problem: M knows who talks to who
- Solution: Chain of mail servers
- …. wrapped in layers
- …. like an ***onion***

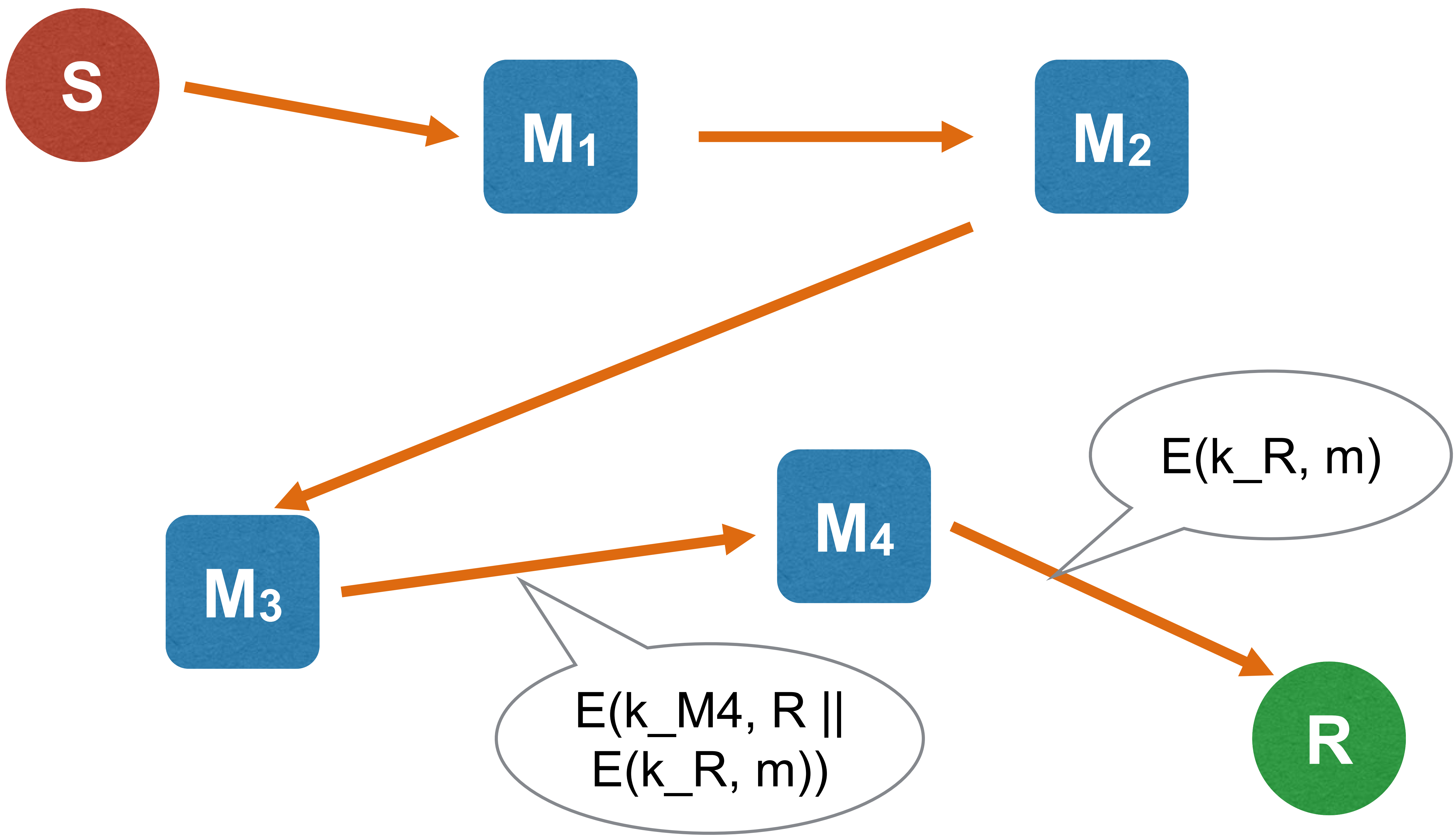
Only know your links



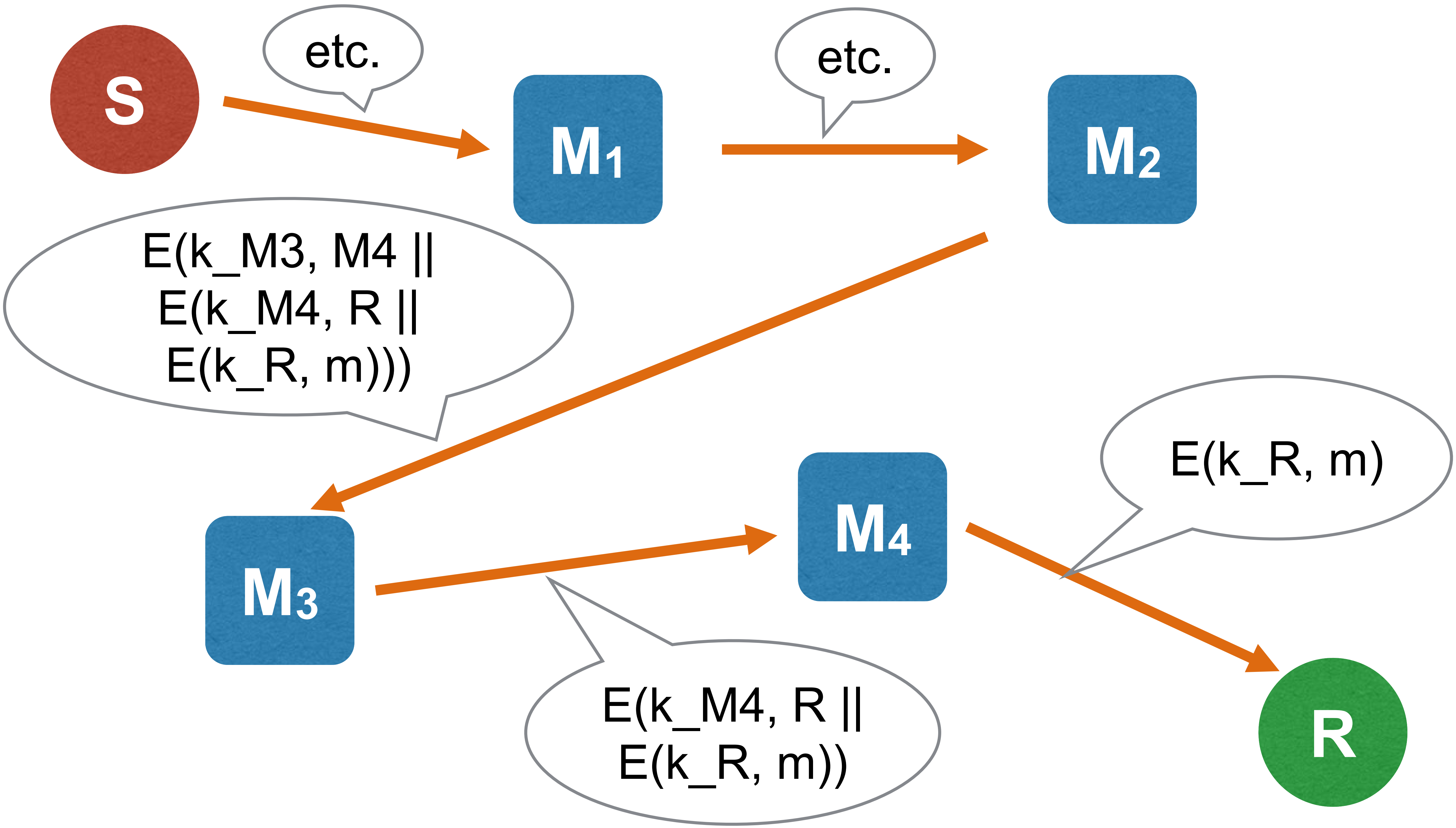
Encryption layers



Encryption layers



Encryption layers



Tor: The Onion Router

- This layering is the basis for Tor
- End-to-end path = circuit
    - Default = 3-hop circuits
    - Download a big list of available peers
- ***Exit node:*** last hop before destination
    - Looks like it connects to all receivers
    - Nodes decide whether to be exit, for where

Tor vs. Mix-nets

- Tor doesn't assume global observer
- Instead
  - some (small) fraction of Tor nodes may be malicious
  - eavesdroppers on a fraction of links
- As a result, does not batch/delay packets
  - Which would not be practical for many uses, eg, web browsing
- Relies on lots of ***cover traffic***!

Confirmation vs. analysis

- If you suspect Alice is talking to Bob
  - Watch both ends
  - ***Confirm*** via timing, volume

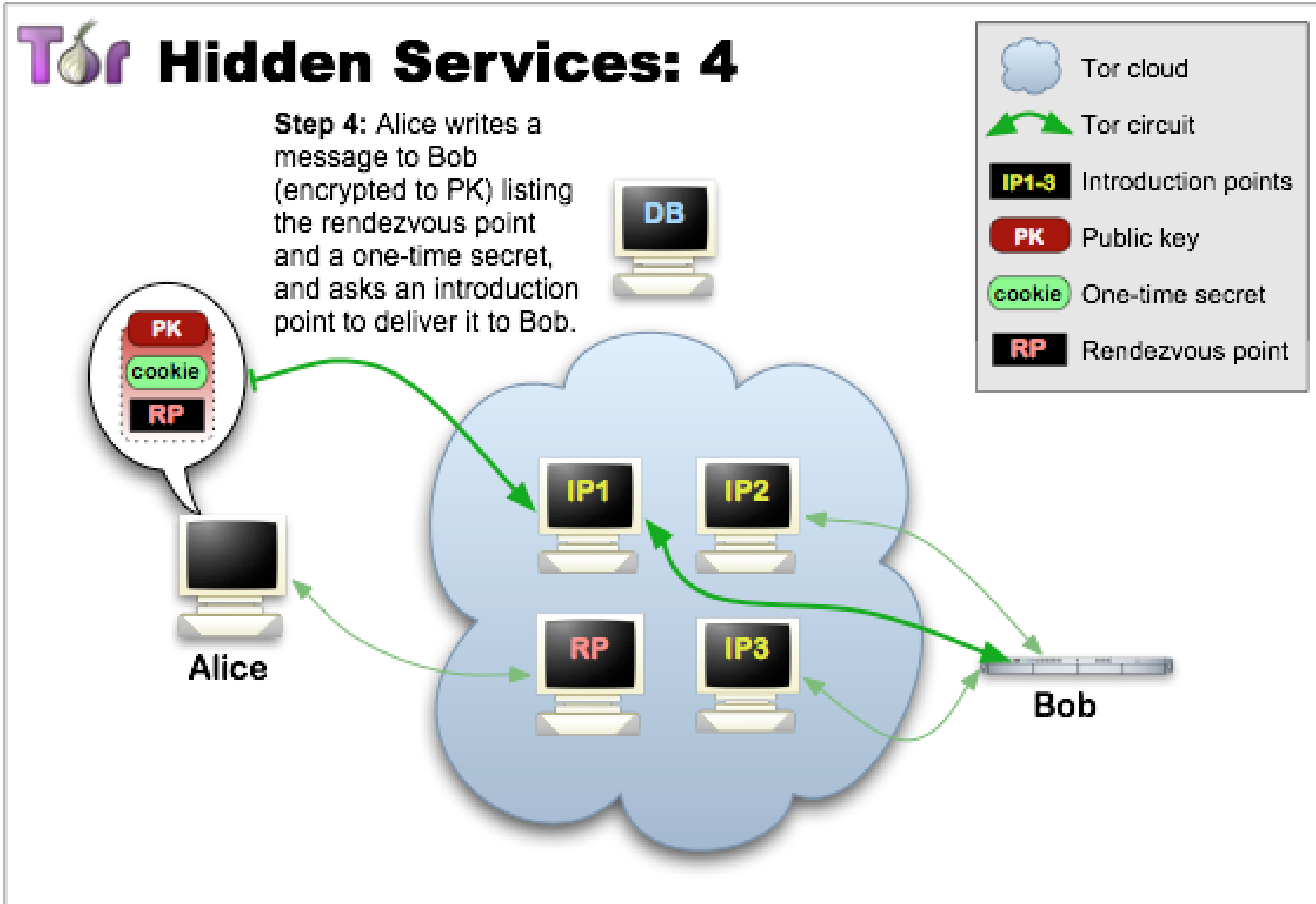
- Tor instead aims to prevent analysis attacks
  - Figure out who Alice is talking to

Something is still missing …

- We have disguised senders, what about receivers?
- Goal: Run service X on host D
  - Without anyone knowing D runs it
  - ***hidden service***
  - (aka, dark web)

Hidden services

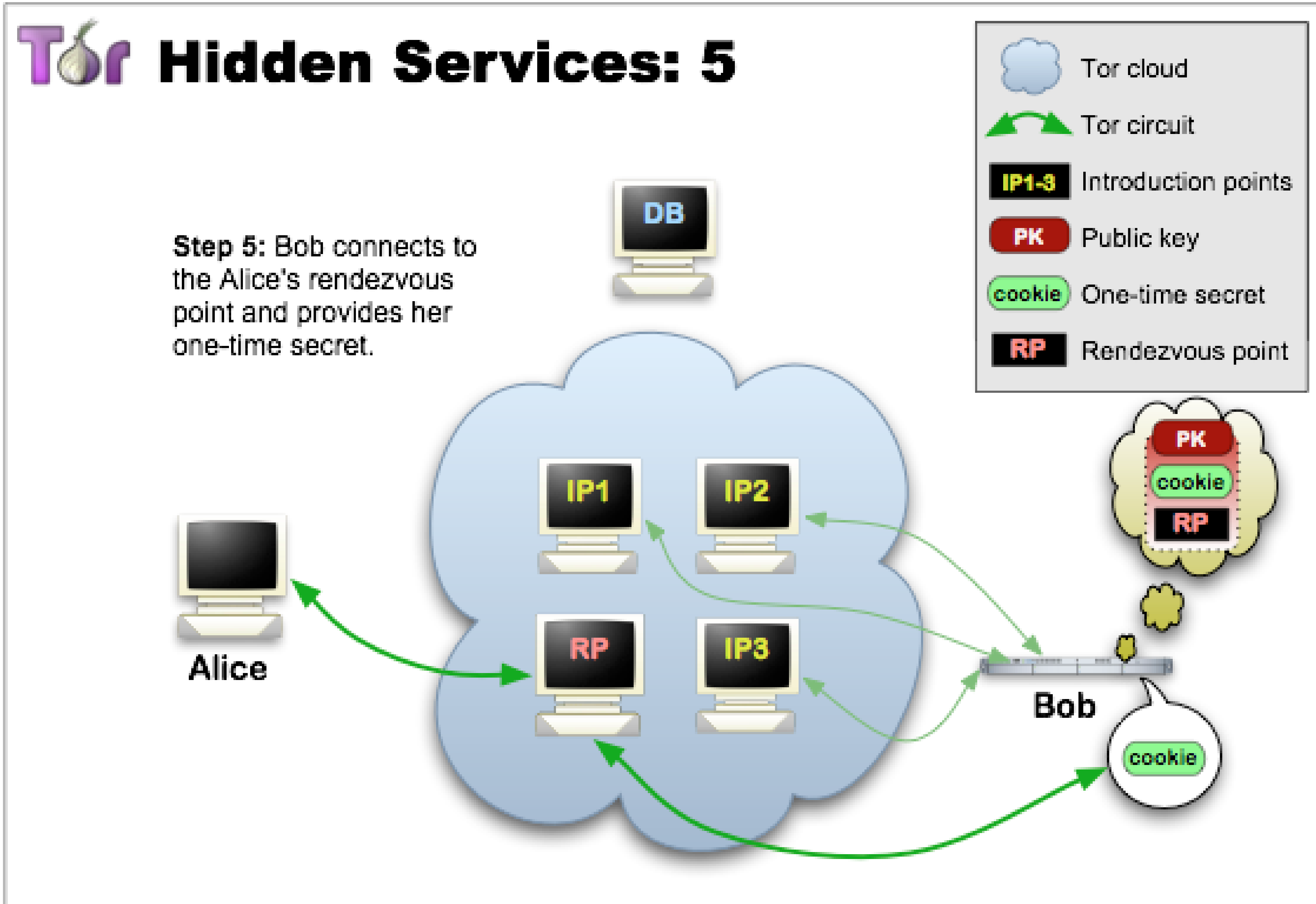
- Bob creates his service X
  - Set up circuits to *introduction points*
  - Posts a listing that maps X to intro points
- Alice wants to connect
  - Set up circuit to *rendezvous point* R
  - Associate with unique token I
  - Set up circuit to one of the intro points
  - Send message: Please forward [R, I] to X



https://www.torproject.org/docs/hidden-services.html.en

Hidden services (2)

- Connection via R
  - Bob sends message containing I to R
  - R links the two circuits together (forwarding)
  - Alice and Bob can now talk anonymously



https://www.torproject.org/docs/hidden-services.html.en

Who knows what?

- Only Bob knows he runs service X
- Intro point knows someone accessed X, but not who
- R knows someone accessed a hidden service, but not who or what
- Alice knows she accessed X, but not who/where X is

Potential Tor attacks

- Insert malicious relays into the network
  - Or compromise legitimate ones
  - Generally need multiple to be useful
- DOS on trustworthy routers
  - Drive traffic toward your relay
- DOS more generally
  - Force relay to do expensive crypto a lot

More Tor problems

- Exit nodes can be blamed for abusive actions
  - Limits desire to be an exit node
  - Monitor exit nodes for traffic analysis
- Option/configuration issues / fingerprinting

Fingerprinting vs. Anonymity

What is fingerprinting?

- Using browser characteristics (fonts, screen dimensions, clock skew etc.) to uniquely ID
- *Does not require* client-side storage
  - Unlike traditional cookies
  - Works fine even in private browsing mode
- In 2010, 83% (of almost 500k users) were unique!
  - panopticlick.eff.org

Eckersley, PETS 2010

Legimitate uses

- Preventing DOS
- Preventing fraud or account hijacking
- Identify content scrapers

- … but also tracking with no consent, no opt out

http://arstechnica.com/security/2013/10/top-sites-and-maybe-the-nsa-track-users-with-device-fingerprinting/

Font probing

- Using JavaSript, load fonts and measure
  - In 2013, 13 scripts on 404 sites in Alexa top 100k
- Using Flash, enumerate directly
- Mainly anti-fraud and analytics companies
  - Ad campaigns, newspaper paywalls
  - But also anonymizer.com, CoinBase

Acar, CCS 2013

Canvas fingerprinting

- Draw text on Canvas API
  - Varies w/ OS, font library, graphics card/driver, browser, rasterization, physical display …
  - Retrieve via *dataURL* – binary pixel data, then hash
- Like font probing, no local storage
- Estimate: No more than 1/1000 overlaps

Mowery + Shacham, 2012
Acar, CCS 2014

Canvas fingerprinting in the wild

- Survey of Alexa top 100k sites: home pages
  - See paper for interesting detection details
- More than 5.5% actively using
  - Vast majority via addthis.com
- Additional techniques
  - Draw in 2 different colors
  - Use fake font name to get default font
  - Cwm fjordbank glyphs vext quiz, 💻

Alexa range	% using
[1, 1k)	1.8
[1k, 10k)	4,9
[10k, 100k]	5.7

Acar, CCS 2014

Cookie abuse

- Cookie syncing: 3rd-party domains sharing IDs
  - e.g., via HTTP referer
- Evercookies: respawn cleared cookies via flash, HTML 5, canvas cache, etc. etc.

Cross-device targeting

- Explicit: Same account on multiple devices
- Implicit: Related searches from same geo. location
- Bizarre: Generate/listen for high-pitched sounds

Countermeasures & mitigations

- Canvas: ask on all data reads?
  - Can’t disable entirely without breaking functionality
- Evercookies
  - Clear lots of storage locations
  - Browser mechanisms are not straightforward
  - e.g., Flash across browsers
- Cookie syncing:
  - 3rd party cookie blocking
  - But only from fresh state!

Countermeasures & mitigations

- Tor browser
  - Fixed settings to prevent differentiation
  - Cap on font enumeration (fixed in 2013)
  - Return empty object from canvas reads
  - Clear huge list of storage caches
- Assorted research tools
  - e.g., Firegloves extension
- Having Tor (or a research extension) is kind of unique to start with, though!