Pervasive Monitoring - what research topics emerge?

stephen.farrell@cs.tcd.ie
NDSS Conference February 2015

Talk Outline

- Me: Trinity College Dublin, working on security/privacy/DTN; IETF security area director (but not speaking for IETF)
  - (W) https://www.cs.tcd.ie/Stephen.Farrell/me/public-resume.html
- It's a keynote, you're expecting wisdom? Vision? Apologies:-)
- A quick reminder of how we got here...
- Stuff people are doing or have done
- Some research questions arising
  - Including a bit of work-in-progress
- What should we be doing about this?
- These slides (W) https://down.dsg.cs.tcd.ie/ndss/ more refs at end

It's an attack

- The actions of NSA and their partners (nation-state or corporate, coerced or not) are a multi-faceted form of attack, or are indistinguishable from that
- Not unique, others are likely doing the same... or will
- The scale arguably makes this an example of a new pervasive monitoring threat model that is neither purely passive nor a classic Man-in-the-Middle and that we have not normally considered in protocol design, implementation or deployment
- A purely technical response will not “solve the problem” but we should treat an attack as we usually do and try mitigate it

Nov 2013 IETF Technical Plenary
(W) https://www.ietf.org/proceedings/88/technical-plenary.html

What have we learned?

- We've mostly learned the unexpectedly broad scope and scale of Intelligence agency snooping
- If there is a way to get at data or meta-data, then they are trying or doing it, including “offensively”
- “Offensive” weapon foot-gun offends common sense
  - Send the bytes of your “offensive” weapon out and they will be used against your customers
- If there is >1 way, they'll try/do them all
- “Collection” fallacy (and others) happily trotted out

What else have we learned?

- Partial timelines:
  - (W) https://en.wikipedia.org/wiki/Global_surveillance_disclosure
  - (S) https://www.theguardian.com/us-news/nsa
- My favourite:
  - (S) https://www.theguardian.com/world/2014/feb/27/gchq-nsa-webcam-images-internet-yahoo
- My most interesting (politically):
  - (F) http://www.scmagazineuk.com/gchq-faces-new-belgacom-hack-allegations/article/388531/
- My most interesting (technically):
  - The short-range radar thing (W) https://en.wikipedia.org/wiki/NSA_ANT_catalog

A Definition

From RFC7258/BCP188: “Pervasive Monitoring is an Attack”

> Pervasive Monitoring (PM) is widespread (and often covert) surveillance through intrusive gathering of protocol artefacts, including application content, or protocol meta-data such as headers. Active or passive wiretaps and traffic analysis, (e.g., correlation, timing or measuring packet sizes), or subverting the cryptographic keys used to secure protocols can also be used as part of pervasive monitoring. PM is distinguished by being indiscriminate and very large-scale, rather than by introducing new types of technical compromise.

PM is not everything

- PM is far from the only security or privacy issue on which we need to work
  - Spam, malware, DDoS, …
  - But mitigations for PM can also help a lot with other problems
- Hypothesis: If we work to address PM, and prioritise services and mechanisms that mitigate PM and that are also effective against other attacks then we will be doing the "right thing"

IETF (Re)Action

- Overall: snowdonia has re-energised folks to do better on security and privacy in general (and not solely in response to PM)
  - Side meeting in Berlin @ IETF-87 (July 2013)
  - Tech plenary, major discussion @ IETF-88 (Nov 2013)
  - STRINT workshop before IETF-89 (Feb 2014)
    - (W) https://tools.ietf.org/html/draft-iab-strint-report
  - Topic at many meetings/BoFs @ IETF-89 (July 2014)
  - Starting to see results from IETF-90 (Nov 2014) onwards...
- Unsurprisingly this is similar to the more broad technical community reaction

IETF work related to PM

- RFC 7258/BCP188 published after major IETF LC debate – sets the basis for further actions
- RFC 7435 defines “Opportunistic Security” - less gold-plating, more deployment
- IAB Statement on Internet Confidentiality: basically: encrypt everything!
- New working groups established:
  - UTA: update BCPs on how to “Use TLS in Applications”
  - DPRIVE: “DNS Privacy”- unthinkable before snowdonia
  - TCPINC: “TCP INCreased security”: tcpcrypt proposed two years earlier but rejected
    - Mistakenly, including by me, as ack'd at mic @ IETF-88, bummer
- IAB re-factored security and privacy programme
  - Developing PM threat model document
- Stuff not going so well (yet!)
  - Old-RFC privacy/PM review team – go back and see what needs fixing, sadly moribund
  - Endymail email list for discussion of ways IETF can help those working on new e2e interpersonal messaging solutions

PM is an Attack RFC 7258

- RFC7258/BCP188 says that all IETF work will consider PM as an attack to be mitigated as part of our normal design processes for all protocol development
  - Note: this does not mean PM is always relevant nor that it's always practical to mitigate PM via protocol mechanisms, but if you can't, you need to be able to say why
- Took ~1000 emails to get rough consensus on that since countering PM is not free
    - Impacts on network management
    - Some folks scared of unreasonable security/privacy nerd dominance

Opportunistic Security RFC 7435

- IETF modus operandi has (in practice) been to define mandatory to implement security that works for higher security environments

  => often hard/expensive to deploy => often not used => cleartext often sent even when better options exist

- Opportunistic Security (OS) aims to evaluate these trade-offs on a connection-by-connection basis, explicitly allowing for e.g. unauthenticated endpoints for confidentiality (open-channel key exchange) as an option that is better than cleartext
- I (personally) hope that this concept is followed very often and is fleshed out to the point where we end up with a new security development approach that is based around OS
  - Not there yet: TLS deprecation of RC4 was interesting because of differing perspectives from web and mail folks about what conclusion to draw when following the OS approach

OS example: Deprecating RC4

- RC4 past sell-by date: agreed by all
- For the web ~15% of https sites were using TLS/RC4 (FF 2014 measurement)
  - When RC4 zapped 99% of those just picked a better option (AES, 3DES)
- SMTP+STARTTLS between MTAs
  - There is a widely deployed MTA that only does RC4, 3DES is buggy and won't work (so I'm told)
  - Zapping RC4 means emails will be sent in clear between MTAs when one is the buggy one
- So – which is better: deprecate RC4 entirely or add this and possibly other caveats?
  - IETF rough consensus was to deprecate entirely, but some mail folks were in the rough
- Interesting example implying conclusion from following OS protocol design pattern will depend on scope
  - OS requires us each to figure out some kind of utility or objective function and where those differ enough, different well meaning folks will reach different conclusions
  - Any way to produce evidence as to what's the best thing to do for things like this?
- It is OK that it is harder to figure out what to do when following the OS approach

IAB Statement

“We recommend that encryption be deployed throughout the protocol stack since there is not a single place within the stack where all kinds of communication can be protected.

The IAB urges protocol designers to design for confidential operation by default. We strongly encourage developers to include encryption in their implementations, and to make them encrypted by default. We similarly encourage network and service operators to deploy encryption where it is not yet deployed, and we urge firewall policy administrators to permit encrypted traffic.”

(W) https://www.iab.org/2014/11/14/iab-statement-on-internet-confidentiality/

Other relevant IETF Things

- TLS 1.3 aiming for better handshake encryption properties and learning from previous TLS problems in various ways
- HTTP/2.0, the major deployment model for which seems to be to run much much more HTTP traffic over TLS
- Extension to HTTP/2.0 defining opportunistic security way of sending http URI schemed content over TLS
- DHCP privacy considerations work
- DNS Query Minimisation
- Negatively: deprecate RC4 in TLS, SSL3, ...
- And since all this is IETF stuff, you can (and please do) join in and help if you're willing and able – that's how to make it better!
  - Even a small amount of good researcher input is hugely valuable (but you need to be able to deal with a noisy environment;-)

Non-IETF Things Relevant to IETF

- Internet Research Task Force (IRTF) CFRG work on additional elliptic curves for TLS that perform better and have better side-channel resistance
  - Basically: consider last decade of academic crypto work on ECC
  - Has adopted Curve25519, more tbd
- IEEE 802 have started work on privacy and are considering e.g. MAC address randomisation
  - Collaborating with IETF
- W3C TAG statement on “Securing the Web”
  - Builds on RFC7258 and IAB statement
  - (S) https://www.w3.org/2001/tag/doc/web-https
- Cryptech – (W) https://cryptech.is/
  - Aiming to build open-source h/w crypto module to help increase confidence
- … there are loads more

Longer Term Factors at Play

- Spooks will be spooks (whether govt. or private sector)
- Privacy invasive commerce (legitimate and not)
- Legal accountability mechanisms (courts of various kinds)
- Small+good things can transition to (big+bad, dead or living-dead)
- Badly-informed decision makers/commentators/twits
- Government regulation of business (e.g. Data Protection Agencies)
- Commercial reaction to user privacy requirements (even evil corporate behemoths have many good folks working for 'em)
- NGOs working to enhance privacy (and get attention)
- Constantly refreshed naivety of yet another generation of clean-slaters (producing occasional good ideas)
- Guilt-by-association is a fallacy no matter who makes the error
- Technical privacy enhancing/enforcement mechanisms (when those work)

So what else?

- We've outlined the problem
- We've seen there's work ongoing
- But most of that is addressing relatively low-hanging fruit in a sense
  - A lot of it is hard to get agreed/done/finished/deployed
    - Esp. deployed, which is REQUIRED for this to be at all useful – fantasy is of no use here
  - But almost all of these are fairly obvious things to do
    - Encrypt more, do more security, yeh!

A Few Research Questions Arising

- OS again
  - Adversary Detection
- Crypto
- Managing mainly-ciphertext networks
- Business Models
- “Consent”
- Privacy Metrics

Opportunistic Security (OS)

- We need to figure out HOWTOs
  - Priority: better, useful deployment of mitigations for security and privacy problems; Anti-priority: Perfection first
- Research questions:
  - Does this work in practice?
  - How can we evaluate ahead of time which of N options will really be more likely to be deployed and useful?
  - Can someone go back and look and tell us what happened?
  - When evaluating OS mechanisms, how can we determine the correct/best scope within which to evaluate the pros and cons?

Post-facto MitM Detection

- Any exchange that uses D-H means Alice and Bob end up with different shared secrets if a MitM attack has happened
  - In OS cases with unauthenticated endpoints, it can happen
- In principle, Alice and Bob could (later) deposit a “witness” value derived from the shared secret to a DB and detect mismatches
  - Could we make this work in practice?
- Adversary can try to disrupt us in many ways
  - Crucial thing is that Alice and Bob need a good session identifier
- Note: this is proposed as a research topic, not as being something near ready to deploy!

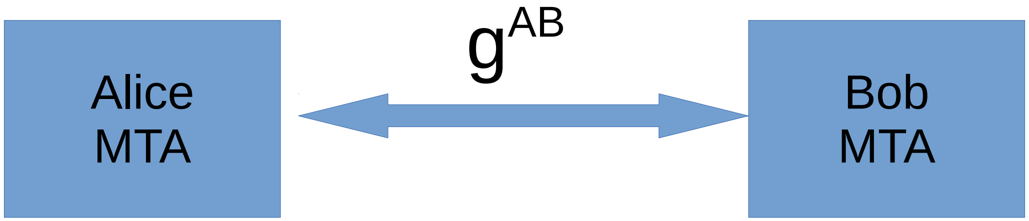
Post-facto MitM Detection

- Works if both have a good sessionID that the attacker cannot play with
  - MPLS LSR-IDs: (W) https://tools.ietf.org/html/draft-farrelll-mpls-opportunistic-encrypt
- Won't work for TCP in general (NAT) or where Alice and Bob don't already share a session-specific value the adversary can't muck with
  - Meaning not for the web:-(

Post-facto MitM Detection via Higher Layer Unbreakables

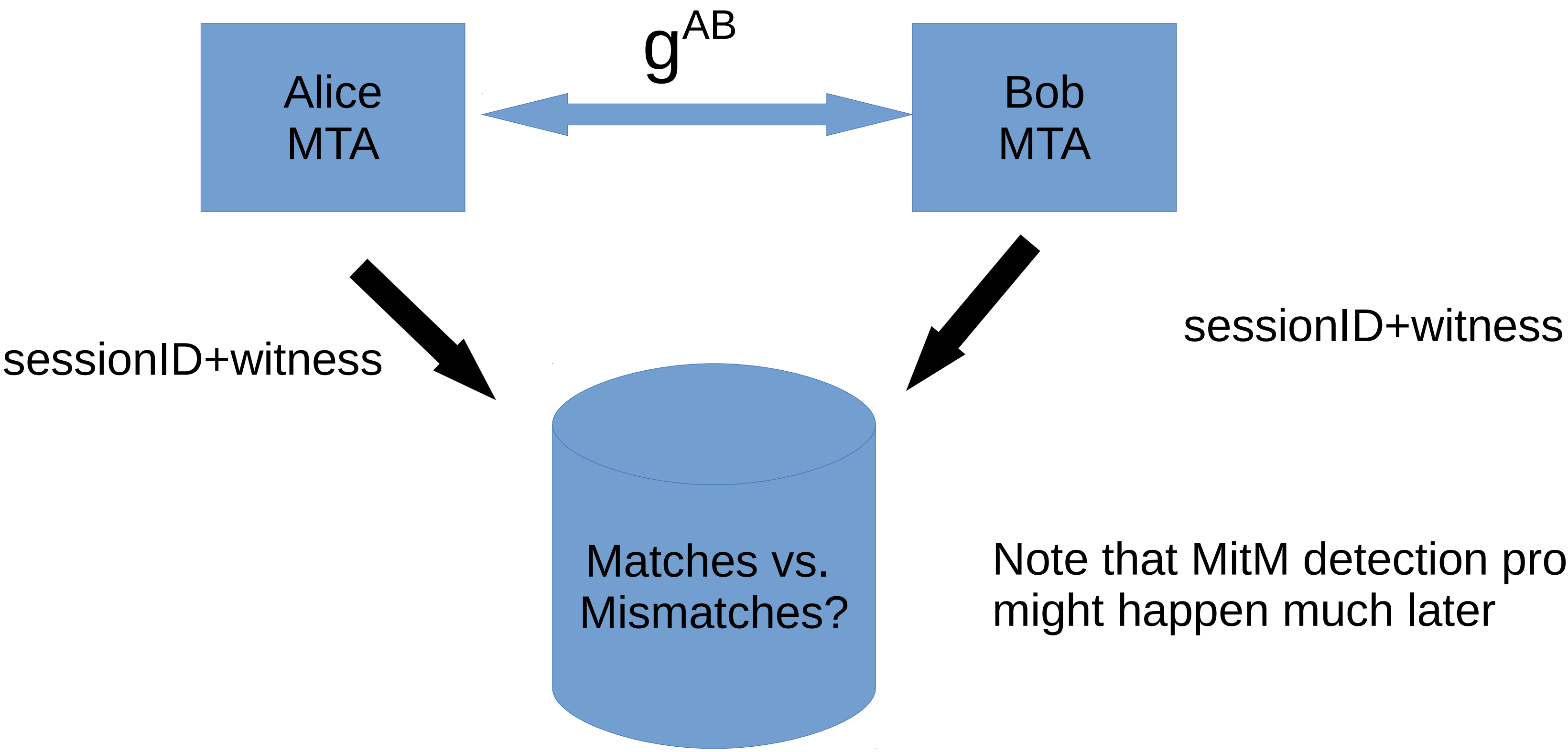
- But... A higher-layer unbreakable might help...
- “unbreakable” ~= “can't break without P(getting noticed)>1%”
- Examples
  - **SMTP+STARTTLS + DKIM-signature bits** => good sessionID if attacker can't delete/modify messages?
  - DPRIVE – say if there were a short-TTL RR that is DNSSEC signed, and both sides use those signature bits in place of DKIM above, seems to work but we've added a wrinkle
    - Implementations MUST do this “on the side” and without signalling (MitM can control signals)
  - TCPInc: Same short-lived RR thing, same wrinkle
  - And the web? (W) https://tools.ietf.org/html/draft-ietf-httpbis-http2-encryption
    - Short-lived RR thing could work for a browser/web-server (maybe) but now we've modified the implementations and probably the protocol, could possibly signal along with pinning?
    - Or maybe leverage signed-code used for s/w update?

D-H



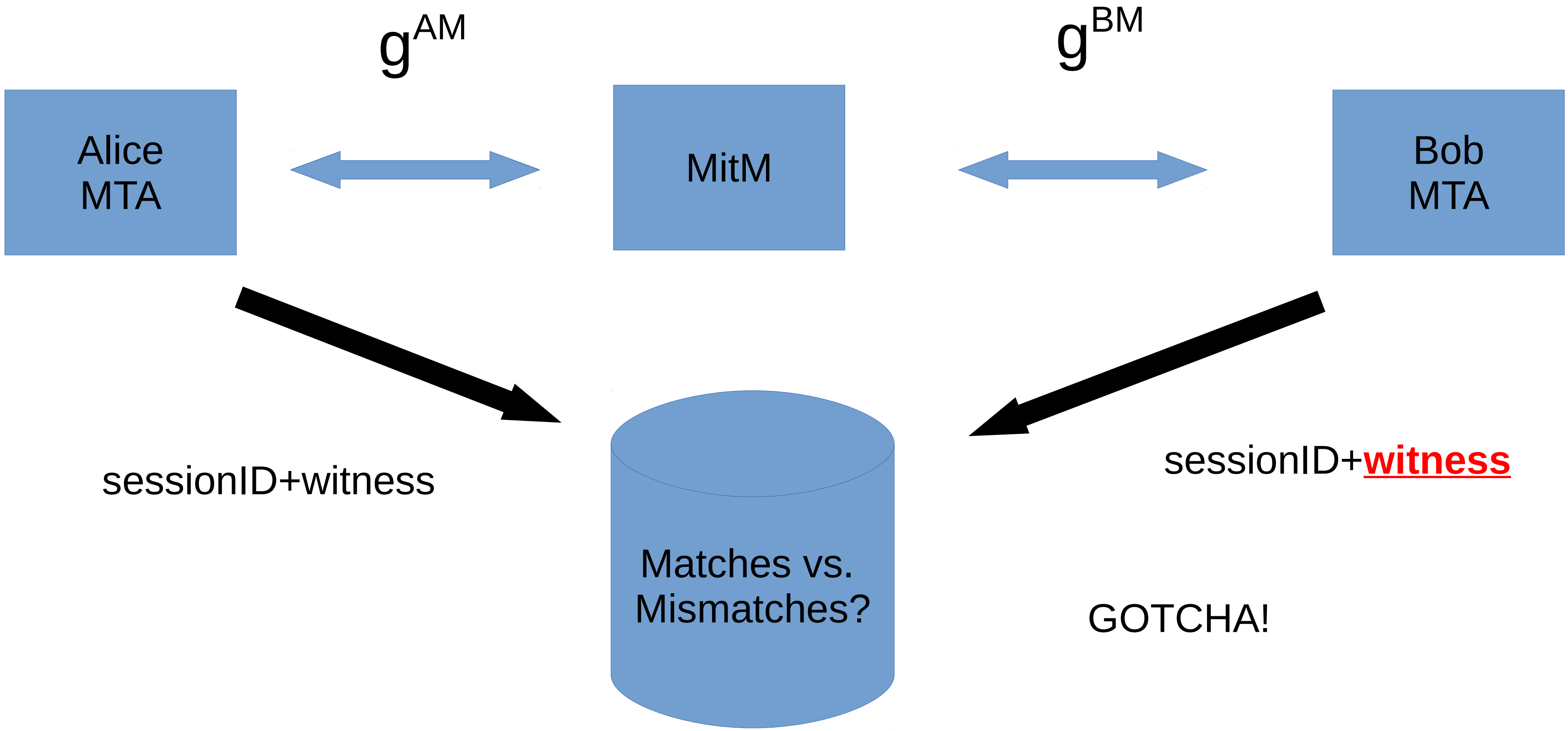
D-H with MitM Detection Protocol

witness = KDF(g^{AB})
sessionID = time:Alice:Bob



Note that MitM detection protocol might happen much later

D-H MitM'd by stupid



g^{AM}

g^{BM}

Alice
MTA

MitM

Bob
MTA

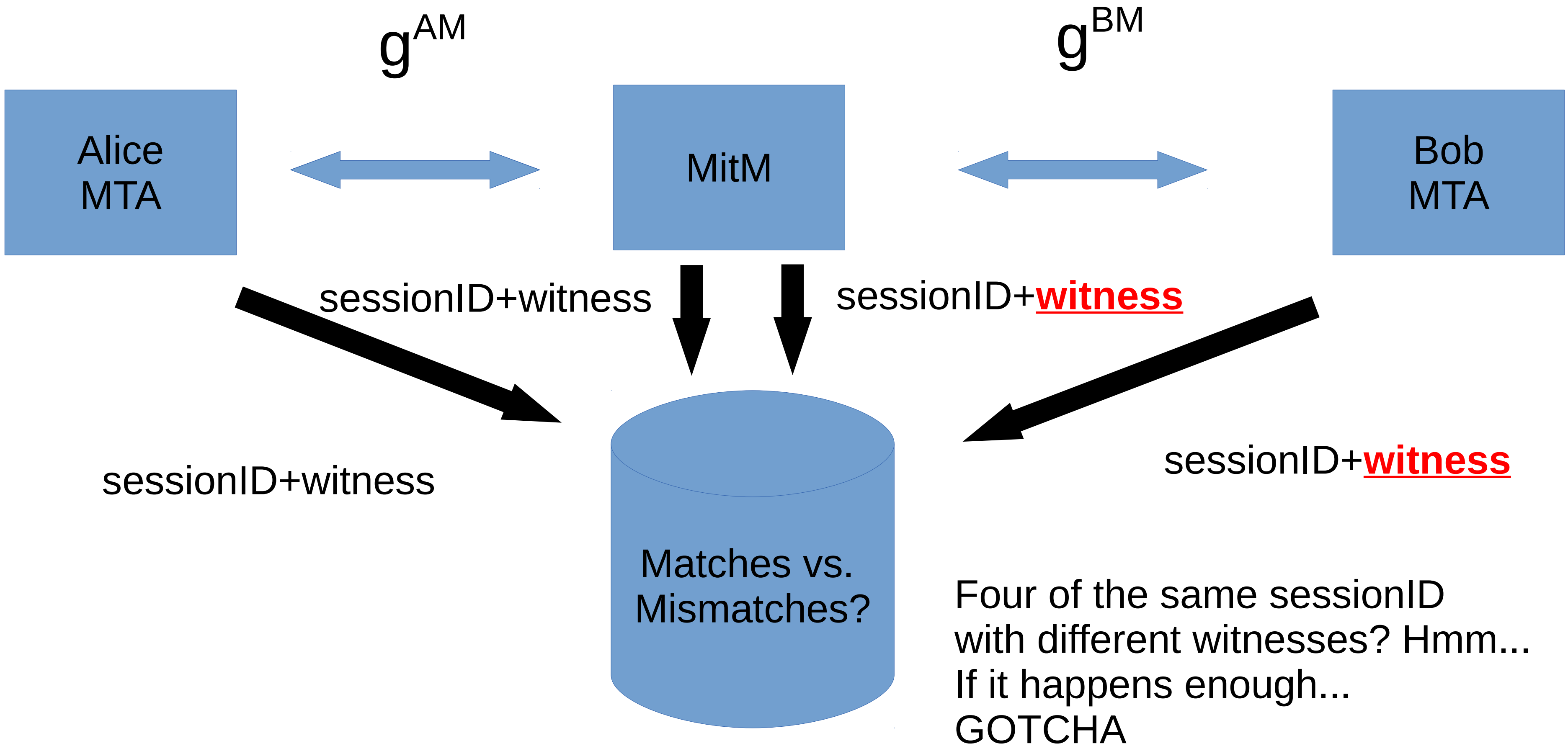
sessionID+witness

Matches vs.
Mismatches?

sessionID+**witness**

GOTCHA!

D-H MitM'd by less-stupid MitM



g^{AM}

g^{BM}

Alice MTA

MitM

Bob MTA

sessionID+witness

sessionID+**witness**

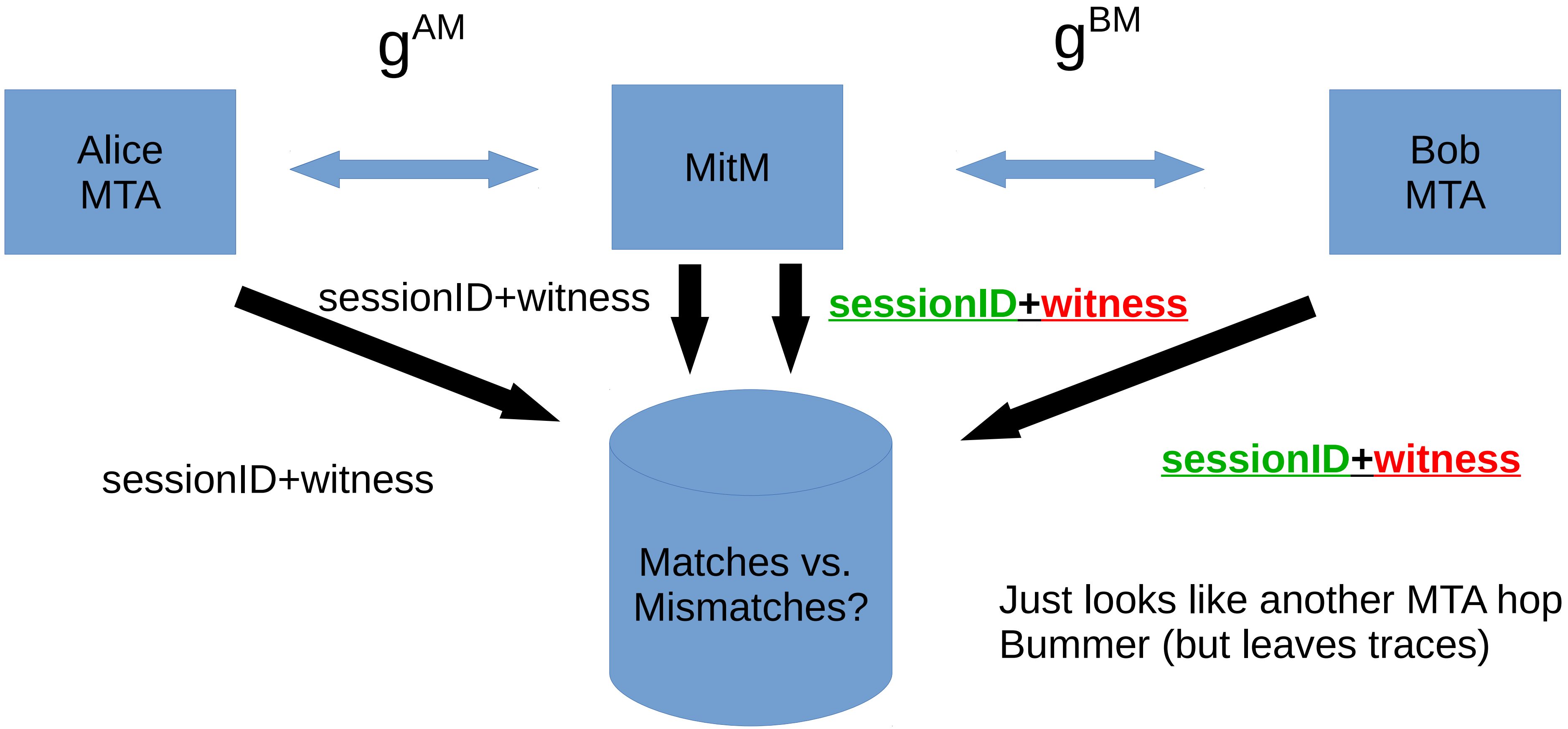
sessionID+witness

sessionID+**witness**

Matches vs. Mismatches?

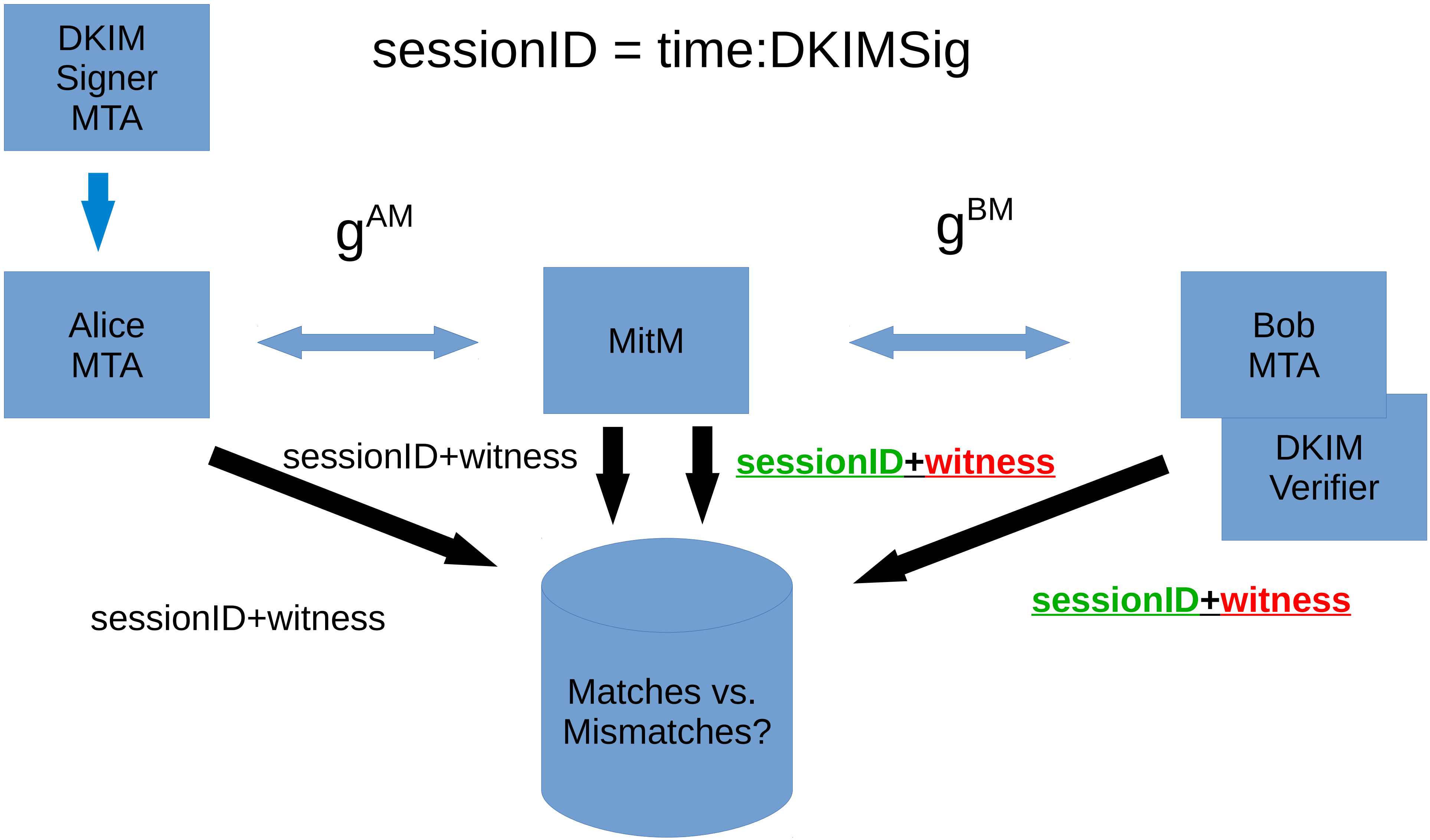
Four of the same sessionID
with different witnesses? Hmm...
If it happens enough...
GOTCHA

D-H MitM'd by non-stupid MitM

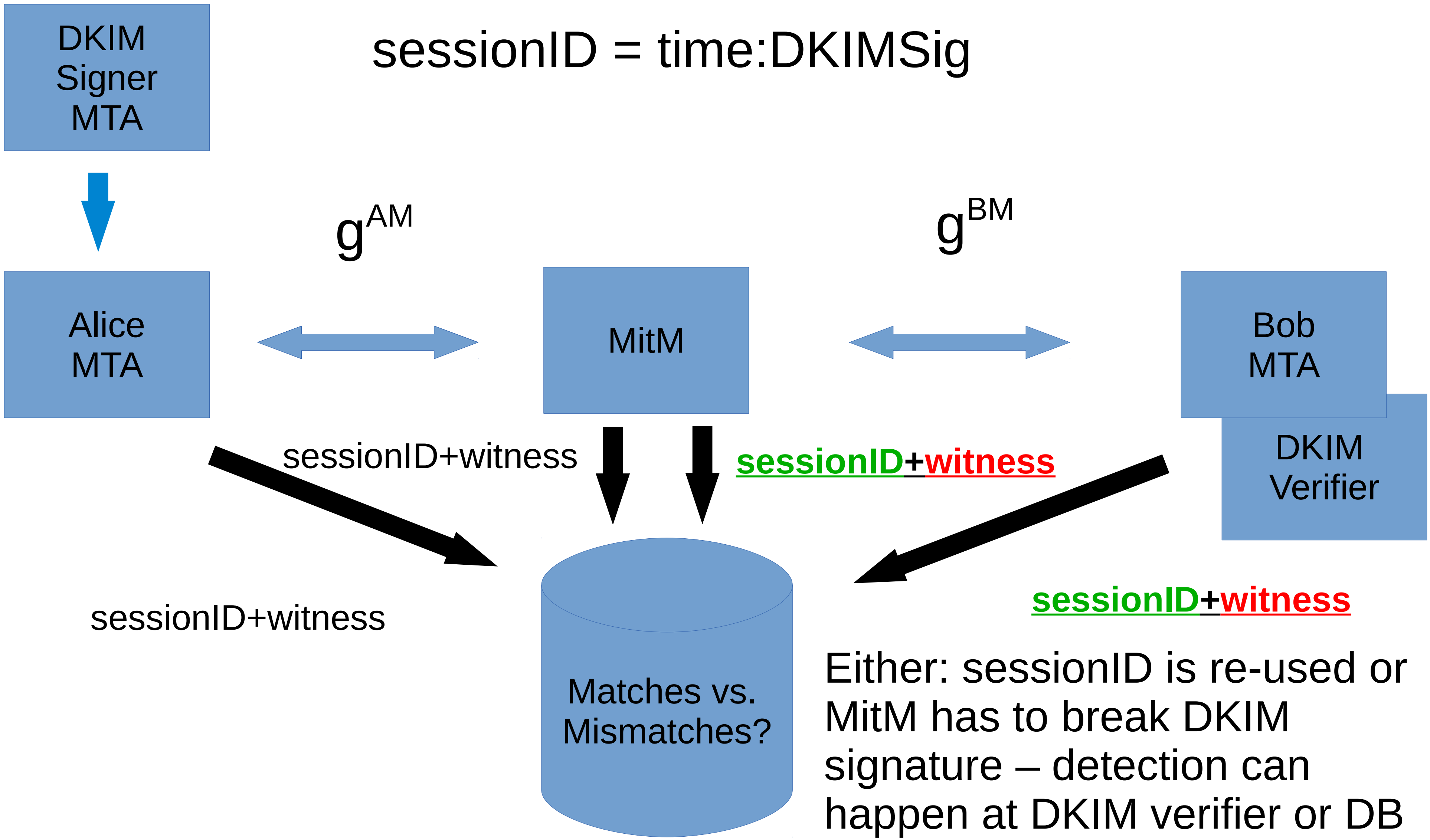


Just looks like another MTA hop
Bummer (but leaves traces)

D-H with “unbreakable”



D-H with “unbreakable” - GOTCHA?



Either: sessionID is re-used or MitM has to break DKIM signature – detection can happen at DKIM verifier or DB

Issues for Post-facto MitM Detection via Higher Layer Unbreakables

- Threat model – does it really work usefully enough?
- Could it expose meta-data that the adversary likes?
  - Would in-band protocol changes help against that? e.g. sessionID
- How to deal with imperfection and not allow the adversary to hide?
  - Various write-to-DB steps will fail from time to time
- Is there a way to make (the infrastructure for) a thing like this scale to cover multiple protocols and many many users?
- Who'd operate the append-only-logs/databases/services that are likely needed? What are we trusting them for?
- What'd happen when we discover instances of MitM attack?
  - Hopefully we don't get inured to them and accept it, but we might

Some Crypto Challenges

- Practical FHE could significantly change the relationships between data subject/owner and processor/controller
  - But in practice, only if unencumbered and efficient
- Ubiquituous, usable, scalable, portable, simple, cheap, … secret-key/private-key storage tokens could help
  - Been done dozens of times, but never scaled up
  - Could be a pre-requisite for progress elsewhere
  - Possibly severe privacy issues – privacy friendly keyID?
- Code/system integrity
  - Not just, but especially, open-source crypto code/system

Managing Ciphertext Networks

- Say if we do get to encrypt ~everything
- What is the impact on current network management?
  - Current network managment is probably overly intrusive (because we've always been able to look at plaintext)
- How do we manage future ciphertext networks without re-enabling the adversary?
  - Tricky: Spam, inbound malware detection, maybe more
- Beware “problems” intended to prefer the status quo!
  - The status quo benefits many

New Business Models

- The advertising-driven Internet is not that old
- The advertising-driven Internet is horribly privacy unfriendly
  - Will it always will be so?
- Today:
  - No motive to really fix that?
- Even privacy unfriendly service providers do have lots of staff genuinely very concerned with privacy
  - This is not about good/bad companies really but about business models
- What new business models might be sustainable and privacy friendly and less vulnerable to PM even in the face of (possibly co-erced) collusion?

“Informed” “Consent”

- Please click ok so I can do whatever I want with you and your content and your descendants and their content...
  - Clearly bogus, even if not legally bogus; Used to justify all sorts of commercial PM; Which is used in turn by nation-state PM actors
- But, what alternatives exist for good-actor service-providers?
  - And how do I know I'm dealing with one? And what happens when they get bought by a bad-actor?
- Examples:
  - How can a random user really give consent to e.g. their IMEI being included in an HTTP header?
  - With WebRTC when I say ok to camera/mic access for a site, what does a user really think that means?
  - IoT toilet flushing – medical analysis vs. privacy?
- Research into usable ways in which real people could really give (and revoke) real consent could be valuable

Privacy Metrics

- I'd like if an app store could (honestly) score applications based on how (un)intrusive they are
- Not easy perhaps, but the Torino Scale seems to work (slightly different target audience)
  - https://en.wikipedia.org/wiki/Torino_Scale
- Is it possible to develop some equivalently understandable metrics for privacy (un)friendliness?

What to do? (1)

- Measure what is being used from more places and how that changes
  - And try detect adversaries if you can
- Consider privacy issues in your experiments and the supplementary data you make available
  - Avoid logging potentially sensitive data if you can
  - That means more work! But you should do it.
- Publish on the Internet (incl. your work) via ciphertext
  - Maybe exclusively via ciphertext if you can

What to do? (2)

- When inventing stuff:
  - Turn on crypto – ciphertext should be base assumption for new things
  - Data minimisation – don't invent new protocols without considering privacy implications
    - E.g. DNS QNAME minimisation
    - More uncertain, more to learn here, how much is useful?
- Help with better implementations
  - https://cryptech.is/ and similar
- Encourage target diversity - Don't all use the same services all the time
  - Even if researchers aren't a huge population, you may start trends

What to do? (3)

- Discuss the issue openly
  - In whatever fora are relevant for you
- Try bring your work out of the lab
  - Or sucker someone else into doing that:-)
  - RFC 6417: “How to Contribute Research Results to Internet Standardization”
- Do not demand the impossible!
  - Do clean-slate work, but don't imagine it can all be deployed now – and only deployed things help
- Agitate (if that's your kind of thing:-)
- Educate
- Go and be responsible researchers and take the broader implications of your work into account before, while and after doing it

Summary

- IETF has consensus PM is an attack (RFC7258) and is working that problem, as are others
- We all should consider how we can work to make PM harder, since those doing it will not just stop
- When/if societies do decide that PM is as bad as it is, then the technical community should have in place the tools to effect that decision

Thanks

Offline questions welcome too
stephen.farrell@cs.tcd.ie

These slides:
(W) https://down.dsg.cs.tcd.ie/ndss/

Links in this slide deck

- (W) https://example.com/
  - Working HTTP/TLS, Hooray!
  - Includes self-signed (that I like) or un usual roots because those are not really that special
- (S) https://silly.example.com/
  - Works, but HTTP re-directs to cleartext
- (D) http://didnt-even-try.example.com/
  - Not even listening on port 443 – none of those today
- (F) http://wtf.example.com/
  - HTTP/TLS just hangs for me, who knows why? There is one
- http://i-just-missed-that.example.com/
  - I'm not perfect:-) All the links below here for example
- Apologies about accessibility, the URLs above are colour-coded, but also preceeded by a parenthesised letter that says which case applies

More References (1)

- General IETF stuff:
- https://www.ietf.org/
- https://www.ietf.org/newcomers.html
- Working group details for WG <foo>:
  - https://tools.ietf.org/wg/<foo> - links to charter, docs, mail archive etc
  - Suggested <foo> values:
    - tls, dprive, tcpinc, httpbis, uta

References (2)

- Relevant IETF non-wg lists:
  - All of them (loads): https://www.ietf.org/list/nonwg.html
  - Perpass – triage list for PM related stuff:
    - https://www.ietf.org/mailman/listinfo/perpass
  - Secruity area list (saag)
    - https://www.ietf.org/mailman/listinfo/perpass
  - Possible e2e interpersonal messaging discussion
    - https://www.ietf.org/mailman/listinfo/endymail
  - General privacy discussion
    - https://www.ietf.org/mailman/listinfo/ietf-privacy
- IRTF:
  - https://www.irtf.org/
  - IRTF Crypto Research Forum Goup: https://irtf.org/cfrg

References (3)

- Videos (ISOC hint:-)
  - IETF youtube stuff in general
  - https://www.youtube.com/user/ietf/videos?sort=p&view=0&flow=grid
- Nov'13 IETF technical plenary video
  - https://www.youtube.com/watch?v=oV71hhEpQ20
- Dan york videos 5 minute summaries of IETF meetings
  - There are loads but these are about PM
  - https://www.youtube.com/watch?v=HG54EsHYKr0
  - https://www.youtube.com/watch?v=fbjs_6Mz-6s
- STRINT workshop
  - Has all 66 position papers
  - https://www.w3.org/2014/strint/

References (4)

- IEEE Internet Computing “soapbox” column on why PM is bad:
  - http://www.computer.org/csdl/mags/ic/2014/04/mic2014040004.pdf
- Some Internet drafts not referenced above:
  - PM Threat model
    - https://tools.ietf.org/html/draft-iab-privsec-confidentiality-threat
  - DNS Privacy problem statement
    - https://tools.ietf.org/wg/dprive/draft-ietf-dprive-problem-statement/
  - “Modern” TLS best current practices
    - https://tools.ietf.org/html/draft-ietf-uta-tls-bcp