2023-04-13 TSPTF Special Workshop Meeting Notes

Meeting Date & Time

13 Apr 2023 This is a special 2-hour meeting of the ToIP Trust Spanning Protocol Task Force in order to have a deep-dive workshop to begin the consolidation stage of our work on the trust spanning protocol (TSP).

See the **Calendar of ToIP Meetings** for exact meeting dates, times and Zoom links.

Zoom Meeting Recording

- https://zoom.us/rec/share/P-tT9DrWgtyIZEKewBYxAYUye_5IGCpvafILv3InV6XlPOZl7TDsMnQszsxxuEDr.Qcr5j_UUxCd02Xok

Attendees

- Drummond Reed
- Daniel Hardman
- Wenjing Chu
- Samuel Smith
- Darrell O'Donnell
- Antti Kettunen
- Mathieu Glaude
- Neil Thomson
- Subhasis Ojha
- Willem de Kok
- Jo Spencer
- Mark Scott
- Vladimir Simjanoski
- Keerthi Thomas
- Dima Postnikov

Agenda Items and Notes (including all relevant links)

Time	Agenda Item	Lead	Notes
5 min	• Start recording • Welcome & antitrust notice • Agenda review	Leads	• **Antitrust Policy Notice:** *Attendees are reminded to adhere to the meeting agenda and not participate in activities prohibited under antitrust and competition laws. Only members of ToIP who have signed the necessary agreements are permitted to participate in this activity beyond an observer role.* • New Members:
5 mins	Urgency vs. Completeness Poll Results	Sam Curren	See the poll and results here. If you have not voted yet, please do. So far (18 votes submitted), the majority are willing to do the work to complete the TSP within a year.
80 mins	Considerations raised by the Two-Layer Design Model Proposal	All	Drummond Reed began by explaining that the purpose of the Two-Layer Design Model Proposal was to see if we could achieve alignment by compartmentalizing which problems were solved where. He pointed out the following comments on the proposal so far: • Samuel Smith raised questions in this context about different approaches to supporting composability. • Darrell O'Donnell made some points about how TCP/IP required both TCP and IP to gain adoption. • Jo Spencer spoke in favor of the "logical separation" of the proposal. • Sam Curren suggested a specific set of five work items that could be a path forward. Daniel Hardman then kicked off discussion with some thoughts about points he has made in the discussion by sharing screenshots #1 through #7 below from this slide presentation (which is also here: https://bit.ly/3oaR27Y). Wenjing Chu understood Daniel's argument for a number of his proposed features but felt that a different test should be used for whether a feature MUST be in the trust spanning layer, "**If a feature is NOT included in Layer 2, it would be hard to implement in a higher layer**." If a feature can be implemented in a higher layer and is not strongly required at the spanning layer, then it should not be in the spanning layer. This may leave some "ambiguity" at the spanning layer with regard to those types of features, but "sometimes ambiguity is useful in language".

Daniel Hardman felt strongly that the properties he has been describing cannot be at any higher layers without losing key qualities.

Antti Kettunen is open to the possibility that there are n layers above the TSP layer, and believes that is where the abuse is likely to happen. He suggests we should focus on the features that are needed by those higher layer to achieve Daniel's properties. "So… the L2 TSP is the one that spans (i.e. every single trust task uses it the same). And the Layer 3 TTMP includes overlays that are composable. On top of TTMP we create Trust Tasks that define the properties they choose to use from TTMP?"

Samuel Smith replied "+1 to Should use the layer above".

Neil Thomson "Suggest a new term definition for what requires trust tasks vs. non-trust - interparty trust boundary."

Wenjing Chu feels that is important that we move in a way that we define the two layers at the same time. So that way, we can consider both sides of the problem will be considered at the same time.

Drummond Reed shared that the question of whether we can accomplish our objectives with two layers or whether it must be one is precisely why he posted the Two-Layer Design Model Proposal. So far it is eliciting exactly the different perspectives needed to decide about this question.

Samuel Smith made the distinction between **trust issues** and **reliability issues**. He feels the latter are important, but the core problem is **trust-across-domain-boundary** problems. So he would prefer to focus on keeping the TSP layer as thin as possible to address the trust-across-domain-boundary problems, while at the same time keeping Daniel's other considerations/properties in mind as we design the minimal TSP layer because we do need to pay attention to those problems.

Jo Spencer returned to the original question of whether the TSP needs to be designed as two layers. He felt that Samuel Smith and Wenjing Chu are focused on bottom-up design. He feels it is only necessary to separate the TSP layer from the TTMP layer if there is a good reason for the TSP to be agnostic about the layers above it and leave open the possibility of additional protocols at the next higher layer. The separation can be either a **logical separation** or a **physical one**.

He asked whether a trust task is only a trust task because **it must communicate across trust boundaries**. He gave an example of a trust task that may not need to cross trust boundaries: biometric verification within an edge device like a smartphone (but then admitted that the trust application requiring biometric verification may need to cross trust boundaries *within the phone*).

So he's asking the question of whether it is helpful to separate the **bottom up** from the **top down** considerations.

Wenjing Chu said that the trust tasks at Layer 3 are going to be very numerous, and there will be requirements for types of trust tasks that we are not going to be able to anticipate. This suggests that there will be at least two higher layers, and maybe more. That again would focus the TSP on the "Inter Trust Domain Protocol (ITDP)" problem domain.

The reason for the focus of the TSP to be **across trust boundaries** is that there are many solutions within a trust boundary.

He clarified that the "upper part" in this discussion should belong to Layer 3.

Drummond Reed requested and received consensus with the whole group on the following axiom:

DECISION: There MUST be one and only one protocol at Layer 2 to serves as the trust spanning protocol.

Daniel Hardman said that the two layer design model might be acceptable if the second layer is either required or highly expected.

Antti Kettunen felt that more features are available at the lower layer, the less negotiation is required. So he wants to look at the trade offs, but to do that, it would be more helpful to have more information about the business problems at Layer 3.

Wenjing Chu first wants to address the question about commonality. He used the examples of the two different host-to-host protocols — TCP and UDP — that evolved on top of IP. But that could (and did) change over time as the computing world changes — for example most Internet traffic used to be over TCP, but now less so. So **layering allows for the evolution of those protocols**. Whatever assumptions we make today about Layer 3 protocols may change. Other designers may have in mind other target contexts (such as IoT or factory floors), but they could use TSP too. So we should be humble about Layer 3 protocol design.

His second point is that, if we have no **compelling** reason to include a feature in the base layer, we should leave it to a higher layer so that the TSP protocol can be as thin as possible. That will result in the most durable TSP.

Drummond Reed made the point that adoption will actually work **from the top of the stack down**. For example, adoption of the TCP/IP stack did not start with developers adopting IP. Rather they adopted TCP (or protocols above TCP that used TCP) because that was the easiest path for them. And TCP used IP. The same will be true here, which is why if we agree with the Two-Layer Design Model Proposal, we should ensure that the TTMP is developed and delivered at the same time as the TSP.

Jo Spencer feels that we decide on **a baseline set of expectations for the TSP**. One example is **time-to-live** or **message-expiry** as potential base capabilities. If we have a "flex layer" above the base layer, the flex layer gives us the ability to understand what the base layer needs.

Wenjing Chu felt that if we are going to design the two protocols at the same time, **it would be best to do it in separate task forces**.

Jo Spencer thought it would be better to have "alignment".

Wenjing Chu said "feedback" would be more important than "alignment" because the latter would mean dependencies in both directions, whereas the TTMP should depend on the TSP but not vice versa.

Daniel Hardman shared screenshot #8 below and explained how it illustrated the choices we have in this discussion. "This visual is now the last slide in the deck that I linked to above (https://bit.ly/3oaR27Y)".

Drummond Reed loved Daniel Hardman's picture. So did many of the attendees in the chat.

Jim St.Clair "I really couldn’t follow things til this visual."

Willem de Kok agreed. "This visual was exactly what I somehow had in mind."

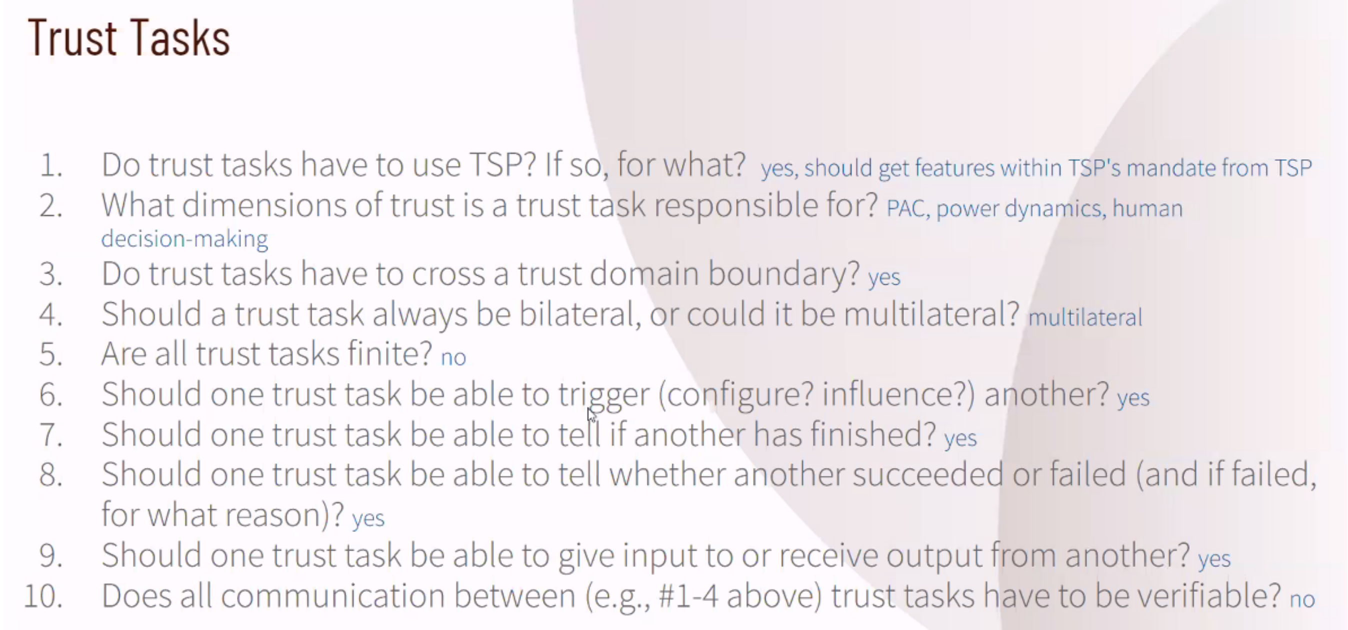
Jo Spencer said that it is an interesting option for trust tasks to "skip" the TTMP layer, but that most would not.

Wenjing Chu said that, with a protocol stack, **developers always have a choice of the protocols to use and the levels**. The only thing that we can insist on is that **if a trust task crosses trust boundaries, it should use the TSP**.

			Daniel Hardman made the point that he was arguing for common semantics to be pushed down to the lowest layer. But he can accept that some properties can be at a higher layer. Jo Spencer "+1 Daniel - We need the base construct to be catered for in the TSP. How that is used and what it means to the higher layers is determined at the TTMP+ protocols." Willem de Kok "I hope that If Daniel (and we together with him ;)) can put all our passion in the TTMP and it becomes a very good one, then it could become the canonical one." Darrell O'Donnell "This is what happened pre-TCP - there were oodles of inconsistent approaches right on top of IP. Thankfully things calmed down and we ended up with two - TCP and UDP. TCP had its abilities that drove many things into it. The UDP realm is where the things that didn't fit with TCP landed in the UDP realm. Having 2 places where just about everything could fit was amazingly powerful. Having dozens or more is not empowering." Antti Kettunen "We still need to understand the implications of layering design choice in regard to adoption & interoperability and what responsibilities are given to higher layers (e.g. application-level / Trust Tasks). Applications (e.g. websites in traditional web) grew in adoption because they didn't need to understand lower levels or interior. Browser wars (JS) were a terrible example of interoperability problems due to poor layering choice. Adoptability of TSP Suite is critical and a wrong layer design choice affects adoption & usefulness on the application levels. Too much complexity, choice & ambiguity on application-level, and adoption probably becomes harder to achieve." Jo Spencer "We just need to decide what goes in TSP and what isn't in TSP (and in TTMP, alpha, beta etc.)" Neil Thomson agreed. "+1 to understanding the "trust task" primitives that are the compossible elements." Wenjing Chu said that the right side of Daniel's diagram (screenshot #8) represents what we want to see happen in the market, i.e., that developers will choose to use the TTMP layer because it gives them what they need, just as they chose to use TCP rather than raw IP. Drummond Reed spoke about DIDComm being a potential candidate for what the Two-Layer Design Model Proposal calls (temporarily) TTMP. Wenjing Chu also liked DIDComm at this layer just above TSP, and also suggested there could be others over time, both because they improve upon DIDComm, or because they solve different problems than DIDComm (just as UDP solved different problems than TCP). Drummond Reed summarized that today's workshop appears to have produced a general consensus that: **1.** The two-layer design model makes sense because it partitions the problem spaces. **2.** Development of both the TSP protocol at Layer 2 and the TTMP protocol at Layer 3 should proceed in parallel (likely in separate task forces). **3.** As a result, we need to start executing on a **communication strategy** (and a **community strategy**) to see who is attracted to work on each of these two layers respectively. Jo Spencer "We also need a communications strategy!" Wenjing Chu said he would be happy in the next TSP Workshop to prepare a set of slides proposing specific features of the TSP. Jo Spencer seconded that step because if we follow this strategy, it is essential to decide which capabilities are required at the TSP layer and which are essential at the TTMP layer. Drummond Reed shared that BOTH of these protocol names ("TSP" and "TTMP") are strictly temporary for purposes of these discussions, and that we should make it a separate effort — at the appropriate time — to decide on the actual protocol names. Mathieu Glaude "Thank you for the great productive discussion! Fascinating to listen to."
5 mins	• Review decisions /action items • Planning for next meeting	Leads	Drummond Reed took a quick poll and it was agreed that due to Internet Identity Workshop, we would **cancel all TSPTF meetings next week (April 17-21)**. ACTION: Darrell O'Donnell to request that Michelle Janata cancel TSPTF meetings next week, i.e., both the NA/EU and APAC meetings on Wednesday April 19. The week after that is the Technology Stack WG Plenary meeting. So our next meeting will be the Terminology Design Training Workshop in the NA/EU slot on Wednesday May 3. We agreed to hold the next TSP Workshop at the same time as this one on **Thursday May 4th**.

Screenshots/Diagrams (numbered for reference in notes above)

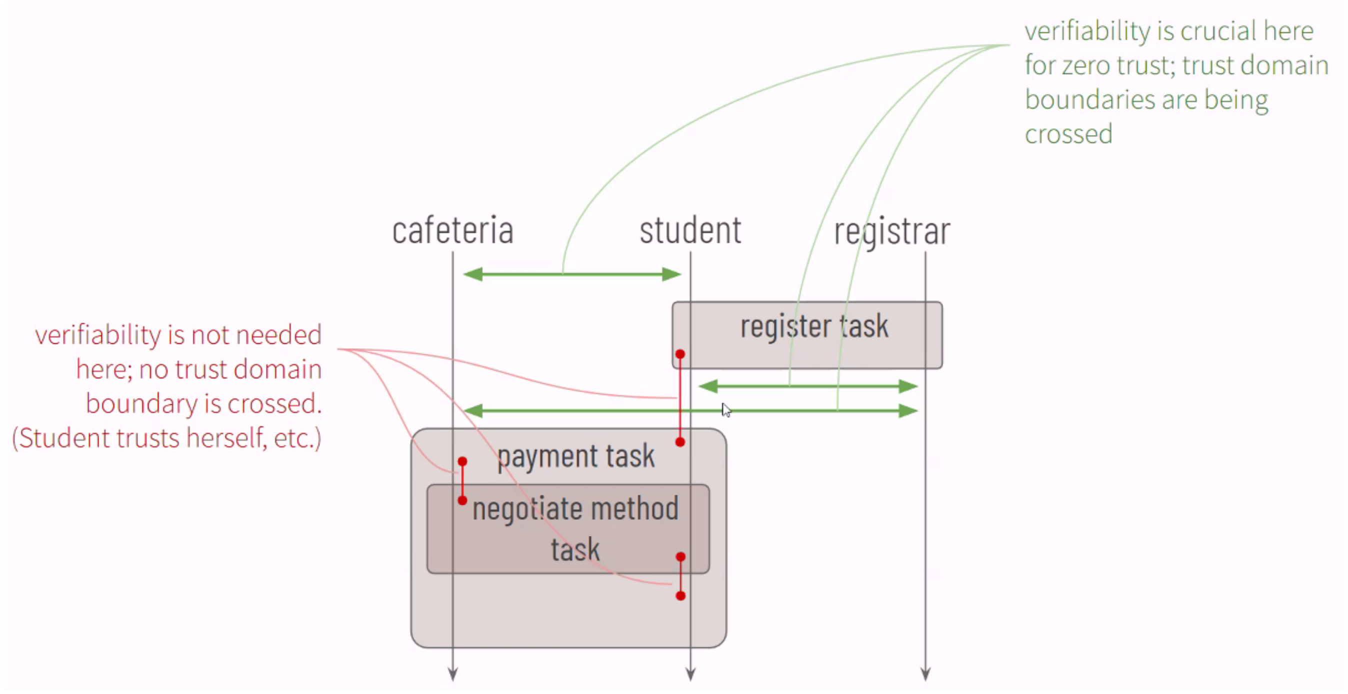
#1



Trust Tasks

1. Do trust tasks have to use TSP? If so, for what? yes, should get features within TSP's mandate from TSP
2. What dimensions of trust is a trust task responsible for? PAC, power dynamics, human decision-making
3. Do trust tasks have to cross a trust domain boundary? yes
4. Should a trust task always be bilateral, or could it be multilateral? multilateral
5. Are all trust tasks finite? no
6. Should one trust task be able to trigger (configure? influence?) another? yes
7. Should one trust task be able to tell if another has finished? yes
8. Should one trust task be able to tell whether another succeeded or failed (and if failed, for what reason)? yes
9. Should one trust task be able to give input to or receive output from another? yes
10. Does all communication between (e.g., #1-4 above) trust tasks have to be verifiable? no

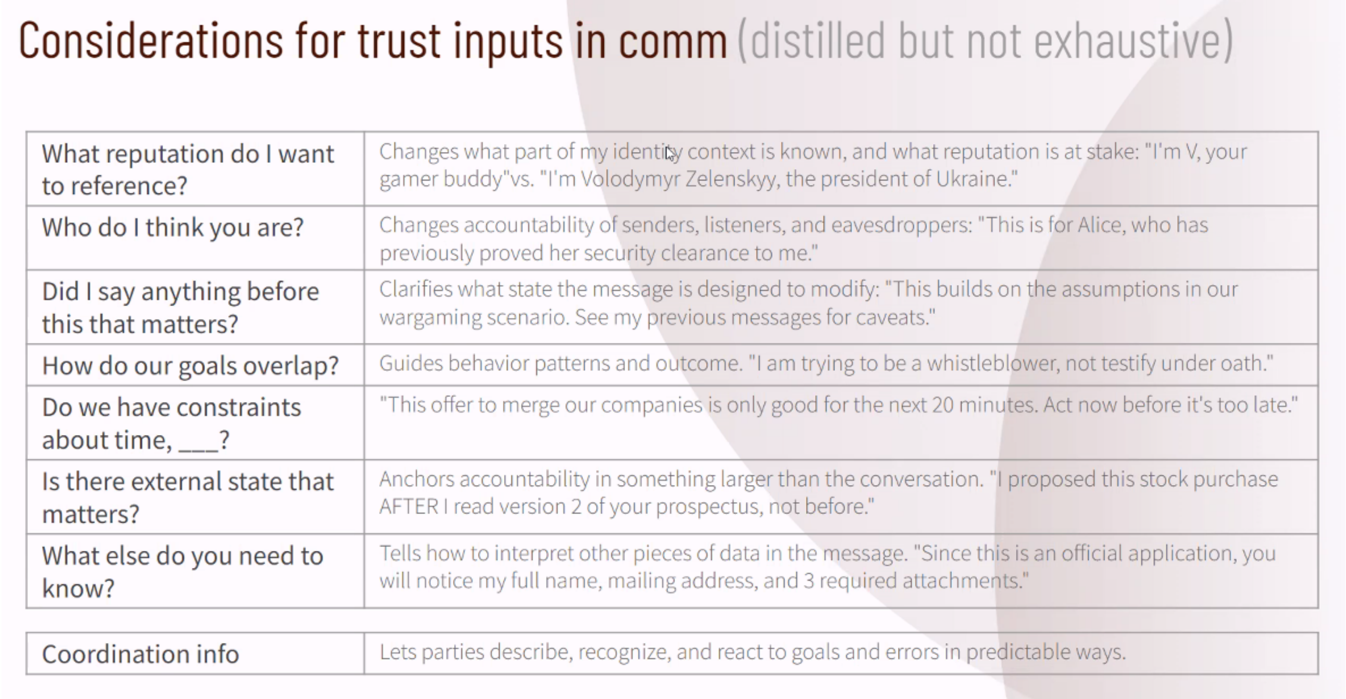
#2



verifiability is crucial here for zero trust; trust domain boundaries are being crossed

verifiability is not needed here; no trust domain boundary is crossed. (Student trusts herself, etc.)

#3

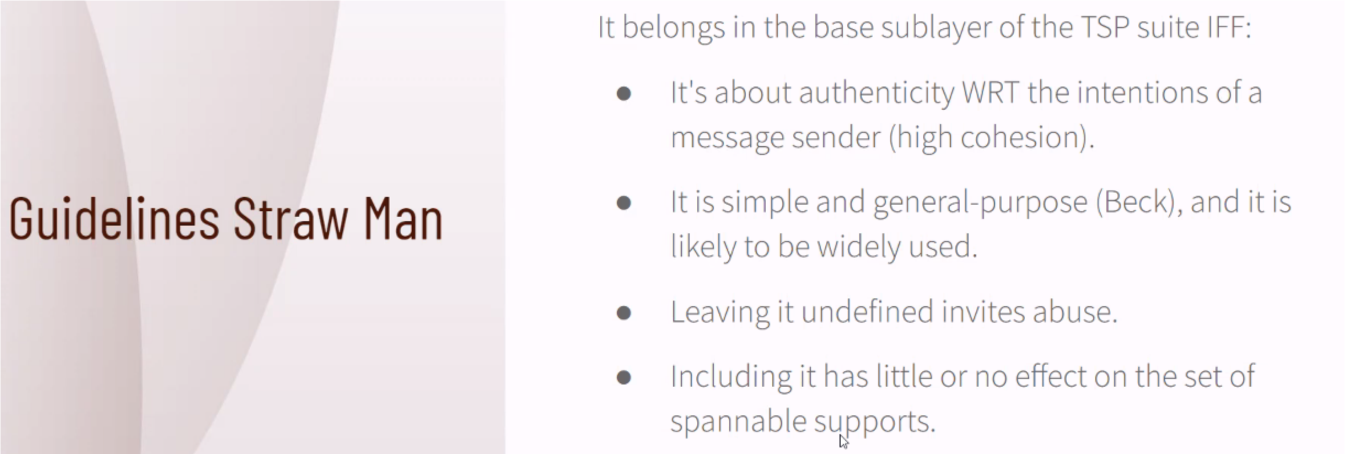


Considerations for trust inputs in comm (distilled but not exhaustive)

What reputation do I want to reference?	Changes what part of my identity context is known, and what reputation is at stake: "I'm V, your gamer buddy"vs. "I'm Volodymyr Zelenskyy, the president of Ukraine."
Who do I think you are?	Changes accountability of senders, listeners, and eavesdroppers: "This is for Alice, who has previously proved her security clearance to me."
Did I say anything before this that matters?	Clarifies what state the message is designed to modify: "This builds on the assumptions in our wargaming scenario. See my previous messages for caveats."
How do our goals overlap?	Guides behavior patterns and outcome. "I am trying to be a whistleblower, not testify under oath."
Do we have constraints about time, ___?	"This offer to merge our companies is only good for the next 20 minutes. Act now before it's too late."
Is there external state that matters?	Anchors accountability in something larger than the conversation. "I proposed this stock purchase AFTER I read version 2 of your prospectus, not before."
What else do you need to know?	Tells how to interpret other pieces of data in the message. "Since this is an official application, you will notice my full name, mailing address, and 3 required attachments."

Coordination info	Lets parties describe, recognize, and react to goals and errors in predictable ways.

#4

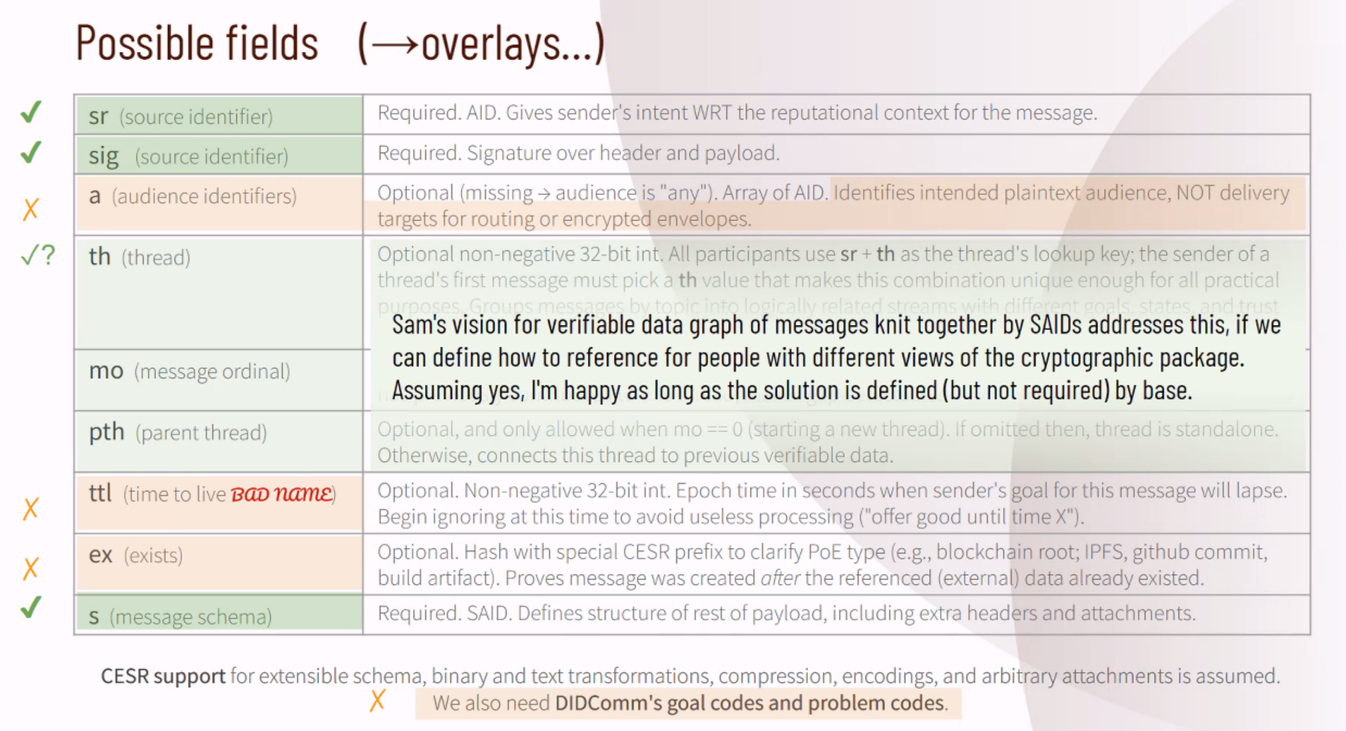


Guidelines Straw Man

It belongs in the base sublayer of the TSP suite IFF:

- It's about authenticity WRT the intentions of a message sender (high cohesion).
- It is simple and general-purpose (Beck), and it is likely to be widely used.
- Leaving it undefined invites abuse.
- Including it has little or no effect on the set of spannable supports.

#5



Possible fields (→overlays...)

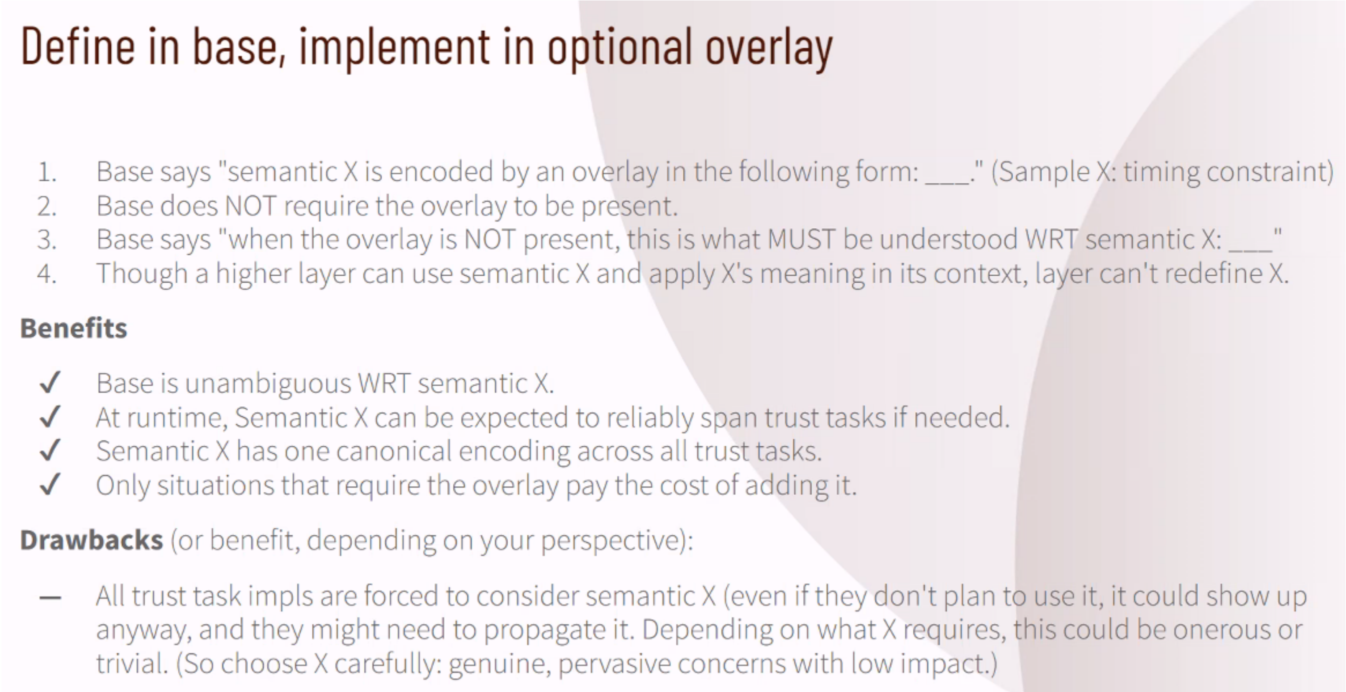
✓	**sr** (source identifier)	Required. AID. Gives sender's intent WRT the reputational context for the message.
✓	**sig** (source identifier)	Required. Signature over header and payload.
✗	**a** (audience identifiers)	Optional (missing → audience is "any"). Array of AID. Identifies intended plaintext audience, NOT delivery targets for routing or encrypted envelopes.
✓?	**th** (thread)	Optional non-negative 32-bit int. All participants use **sr** + **th** as the thread's lookup key; the sender of a thread's first message must pick a **th** value that makes this combination unique enough for all practical purposes. Groups messages by topic into logically related streams with different goals, states, and trust
	mo (message ordinal)	
	pth (parent thread)	Optional, and only allowed when mo == 0 (starting a new thread). If omitted then, thread is standalone. Otherwise, connects this thread to previous verifiable data.
✗	**ttl** (time to live *BAD NAME*)	Optional. Non-negative 32-bit int. Epoch time in seconds when sender's goal for this message will lapse. Begin ignoring at this time to avoid useless processing ("offer good until time X").
✗	**ex** (exists)	Optional. Hash with special CESR prefix to clarify PoE type (e.g., blockchain root; IPFS, github commit, build artifact). Proves message was created *after* the referenced (external) data already existed.
✓	**s** (message schema)	Required. SAID. Defines structure of rest of payload, including extra headers and attachments.

Sam's vision for verifiable data graph of messages knit together by SAIDs addresses this, if we can define how to reference for people with different views of the cryptographic package. Assuming yes, I'm happy as long as the solution is defined (but not required) by base.

CESR support for extensible schema, binary and text transformations, compression, encodings, and arbitrary attachments is assumed.

✗ We also need **DIDComm's goal codes and problem codes**.

#6



Define in base, implement in optional overlay

1. Base says "semantic X is encoded by an overlay in the following form: ___." (Sample X: timing constraint)
2. Base does NOT require the overlay to be present.
3. Base says "when the overlay is NOT present, this is what MUST be understood WRT semantic X: ___"
4. Though a higher layer can use semantic X and apply X's meaning in its context, layer can't redefine X.

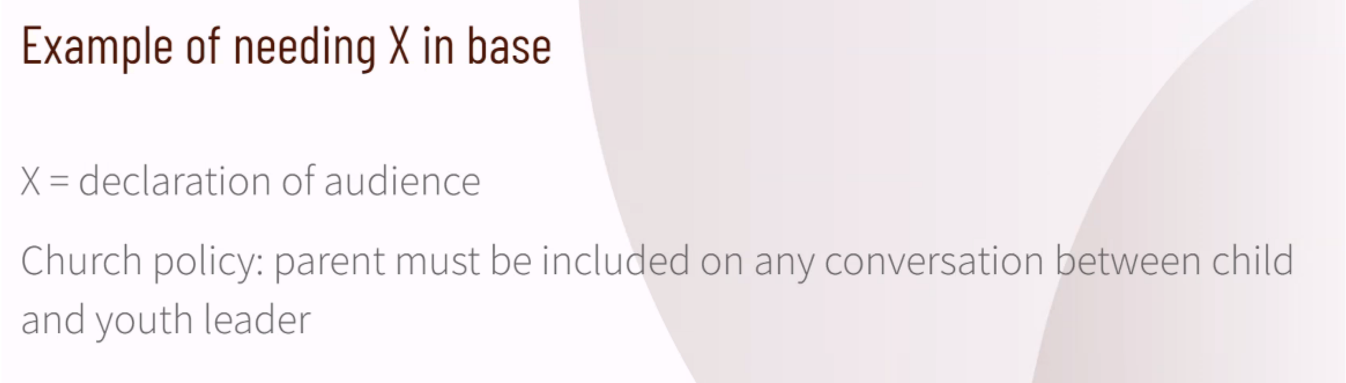
Benefits

✓ Base is unambiguous WRT semantic X.
✓ At runtime, Semantic X can be expected to reliably span trust tasks if needed.
✓ Semantic X has one canonical encoding across all trust tasks.
✓ Only situations that require the overlay pay the cost of adding it.

Drawbacks (or benefit, depending on your perspective):

– All trust task impls are forced to consider semantic X (even if they don't plan to use it, it could show up anyway, and they might need to propagate it. Depending on what X requires, this could be onerous or trivial. (So choose X carefully: genuine, pervasive concerns with low impact.)

#7



Example of needing X in base

X = declaration of audience

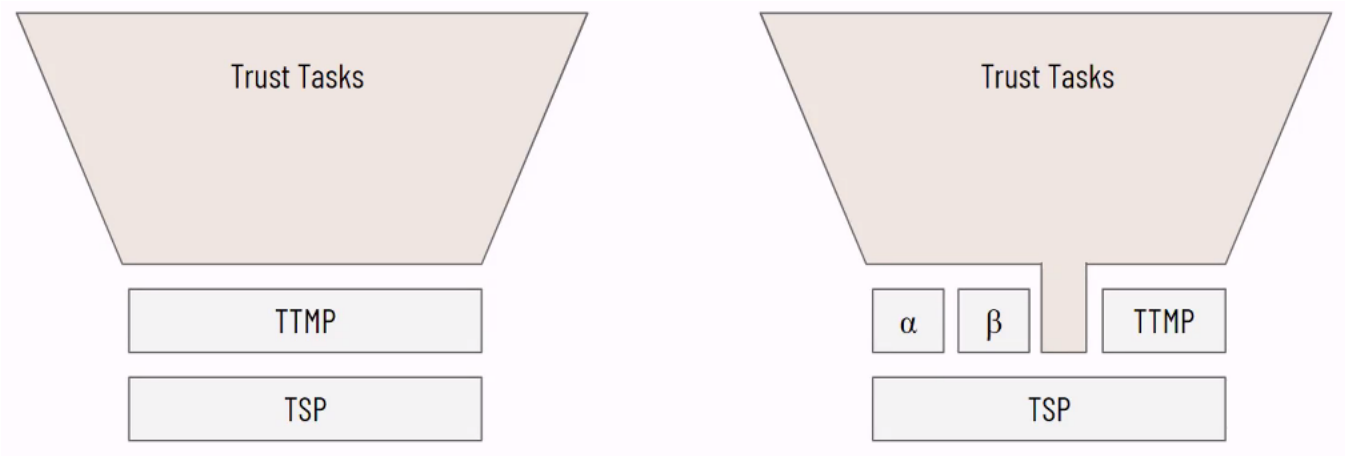
Church policy: parent must be included on any conversation between child and youth leader

If the message says nothing about the parent is involved in this conversation, it changes the accountability of the sender and the audience in that assertion.

That cannot be communicated via cryptography alone, as the messages to the child and to the parent will be cryptographically separate.

If the audience property is not included, then it cannot be consistently across trust tasks.

#8



Decisions

- **DECISION: There MUST be one and only one protocol at Layer 2 to serves as the trust spanning protocol. Any higher level protocol belongs in Layer 3 or above.**

Action Items

□ ACTION: Darrell O'Donnell to request that Michelle Janata cancel TSPTF meetings next week, i.e., both the NA/EU and APAC meetings on Wednesday April 19.