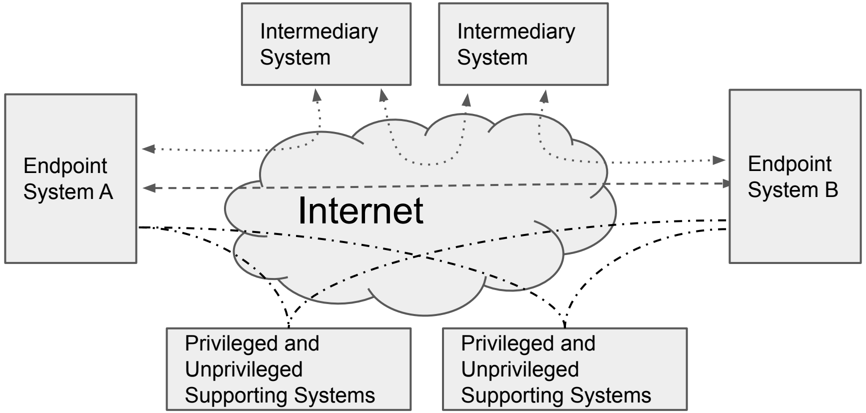
ITDP Trust Spanning & SCAP

Wenjing Chu - June 26, 2023

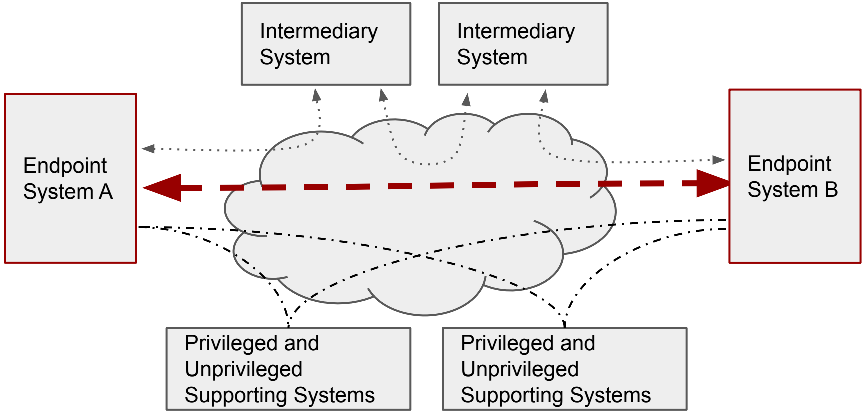
Review ITDP

The proposed Reference Architecture (RA)


Intermediary System
Intermediary System
Endpoint System A
Internet
Endpoint System B
Privileged and Unprivileged Supporting Systems
Privileged and Unprivileged Supporting Systems

- Subsystems are delineated by **locus of control (domain)**
- They interact through **a suite of protocols**.
- Each type of subsystems uses different parts of the suite.

Trust spanning between Endpoints is the goal


Intermediary System
Intermediary System
Endpoint System A
Endpoint System B
Privileged and Unprivileged Supporting Systems
Privileged and Unprivileged Supporting Systems

Review Settings (1)

- We model the main class of devices as "Endpoints". They are the main speakers of the protocols being designed.
- The endpoints wish to create trust relationships among themselves, and communicate through a "channel" to share sensitive information. This channel is message based and is carried over Internet (and other local networks) - considered a "trustless" medium in the trust model. The Internet is already end-to-end routable, but it does not have the principle trust & engineering properties we are looking for.
- Our objective is to design a protocol that enables trust relationships and such channels over the Internet (and other local networks) AND leaves as much freedom as possible to both layers above and below.
  - Goal category I: principle of trust properties (verifiability/authenticity, confidentiality, human/privacy, decentralization, relational, directional, limits, contextual)
  - Goal category II: principle of engineering properties (end-to-end, connectivity, hourglass, edge, design for simplicity, design for constant change)

Review Settings (2)

- Endpoints wish these channels to observe a set of trust principles
  - Authenticity, more precisely
    - The sender is authentic (this is about identity & depends on the identity schemes)
    - The information being shared has not been tempered (a prerequisite)
  - Confidentiality
    - Optionally, if chosen, the information is not leaked to non-participants
  - Privacy, more precisely
    - Since the common medium on which this channel is carried is the Internet & higher layer protocols over the Internet, the channel usually leaks non-content information, aka metadata. This is generally true in other common media, e.g. Bluetooth, NFC with varying degrees. The privacy should also include all other necessary facilities a solution may rely on, e.g. correlation, out-of-band channels, biometrics, social engineering. It should be understood as “open ended”. Stronger AI algorithms, for instance, make privacy harder to protect both by utilizing more modes of information and by discovering deeper correlations that are harder to see.
    - Legitimate participants can also disclose information. We generally choose not to tackle this problem in the TSP layer, however, we should consider that the TSP layer has the necessary facility for other layers to solve/remediate this type of privacy breach.
    - As an open ended definition of privacy, we aim to maximize privacy but the protection is never perfect. In addition, privacy may also conflict with other properties that a given ecosystem may also want. For both reasons, whenever possible, we should leave this tradeoff decision to higher layer protocols, applications, or end users.
  - Others omitted here for simplicity for this discussion

Review Settings (3)

- The endpoints utilize additional services (in addition to the base Internet services) to assist them in achieving these principle properties (1) ensuring authenticity, (2) optionally ensuring confidentiality, (3) maximizing or making a local optimal tradeoff wrt privacy.
    - Intermediary Systems (or simply Intermediaries): Intermediaries handle messages, provide storage, notification, scaling, and other features of the channel. They make the channel work better.
    - Supporting Systems: They provide source of trust in authenticity, e.g. DPKI, DID methods, KERI-AID.
- The trust relationship and the communication channel are bootstrapped together in a cyclical fashion and so they are recursively dependent in a spiral.

Endpoint Definition

- An Endpoint is a computing element, usually software element combined with storage, communication, execution environment, under the effective control of a competent party.
    - Have a unique transport address (e.g. TCP/UDP port + IP address in the context of the Internet)
    - Can be a software program in a physical or virtual machine, or container, etc. which provides secure storage, communication, execution
    - Locus of control: a competent party has effective control of secrets, keys and generation and verification methods that reside in the Endpoint and substantiate the party's trustworthiness to others.
    - An Endpoint is a delineation of issues within the Endpoint and issues outside of the Endpoint.
- An Equivalent Endpoint Group (EEG)
    - A group of Endpoints that are equivalent in the eyes of other parties. For example, a mobile wallet app, desktop app, cloud app by the same user. An outside party assumes that they share secrets.

Type of Endpoints

Type	**OS**	**Security features**	**Notes**
Mobile phones			
Personal computers			
Cloud virtual machines, containers			
XR Headsets			
Hardware tokens			
Internet of Thing devices (sensors, cameras)			

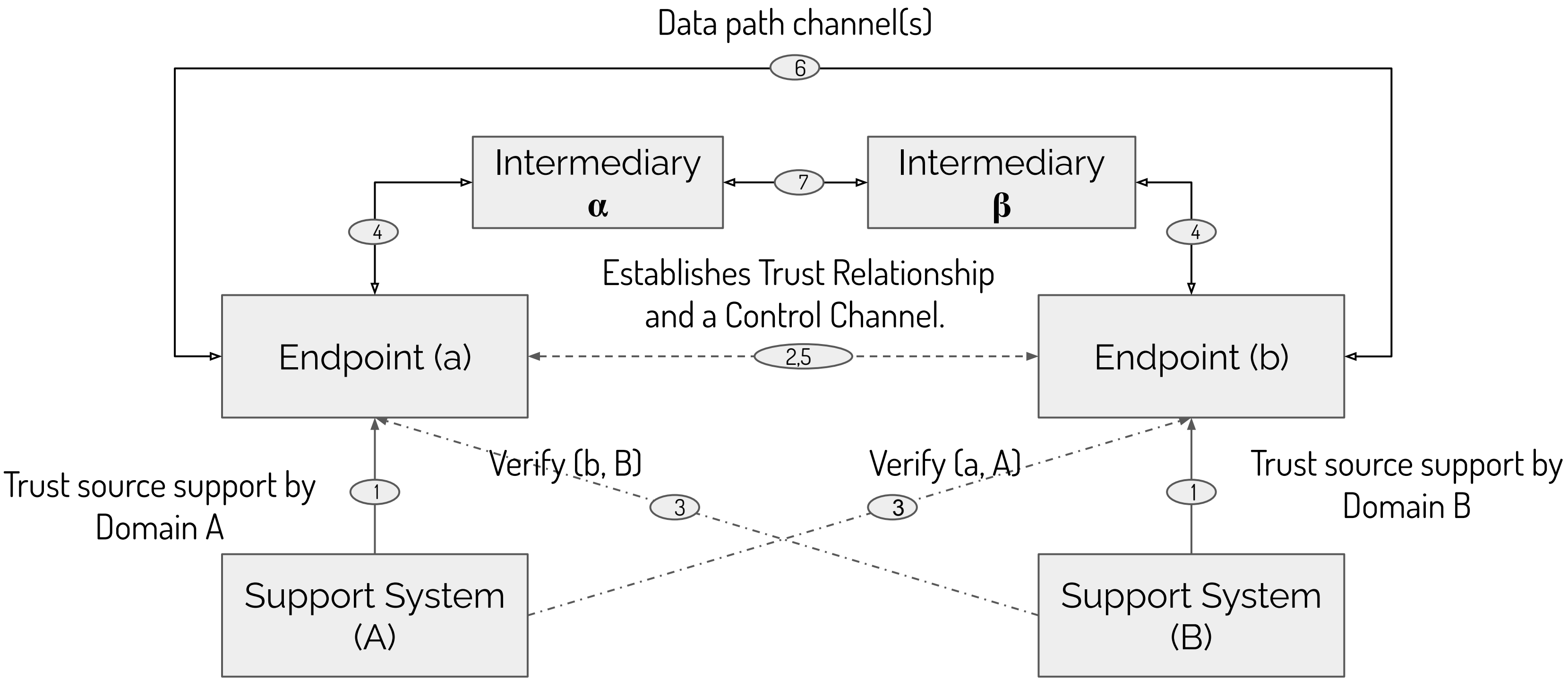
Type of communication patterns

1.Pairwise (Point to Point, unicast)	2 Endpoints	Endpoints are identified to each other in a pairing (Relationship protocol). Verification of this pairing protocol is aided by their respective Supporting Systems. They optionally utilize Intermediaries for performance etc.	Example: Secure & Private IM, e.g. Signal, unicast-ish.
2.Star (Point to a central point, unicast)	2 Endpoints, one of them is the central point	It is a special case of the Pairwise mode where one endpoint is always the same central point. This central point could also act as an intermediary (but may not be as a Supporting System for other endpoints).	Example: a bank app, a streaming app, web2 app, enterprise apps
3.Group (Point to multipoints, multicast)	N Endpoints with known membership	Members in the group are identified to each other. Each can verify every other member. A Membership protocol is required.	Example: Group IM, e.g. Signal, multicast-ish.
4.Broadcast (Point to *wildcard, anycast)	1 Endpoint to a wildcard group without membership	Each speaking endpoint is identified. Everyone can verify a speaker for authenticity. Non-speakers are not necessarily identified. There is no known membership (can not be verified).	Example: Various social media modes, broadcast-ish.
Others	TBD	TBD	TBD

Trust Spanning as defined by ITDP

1. TRUST: We **need** to form "source of trust" from a "trustless" environment. This source of trust is between an Endpoint and a Supporting System.
2. SPANNING (A): We **need** to enable Endpoints to communicate over "trustless" environment to establish Trust Relationship & a Control Channel between Endpoints.
    a. This enables Endpoints to THEN create any kinds of communications channels which are numerous based on various communication needs (aka data path protocols).)
3. SPANNING (B): We **want** to include as many Endpoints as possible.
    a. Eliminate non-essential elements from mandatory part of the SPANNING (A), allow optimal flexibility
        i. Easier accommodation to existing systems' migration.
        ii. Flexibility in future developments.
    b. Allow maximum types of Endpoints - diversity in how source of trust is obtained, in types of computing elements, in transport mechanisms.
    c. Allow maximum freedom in communication patterns: p2p, p2c, group, broadcast...

Trust Model in ITDP (only P2P case illustrated)



Threat model: Other parties may observe all arrowed lines. They may or may not collude.

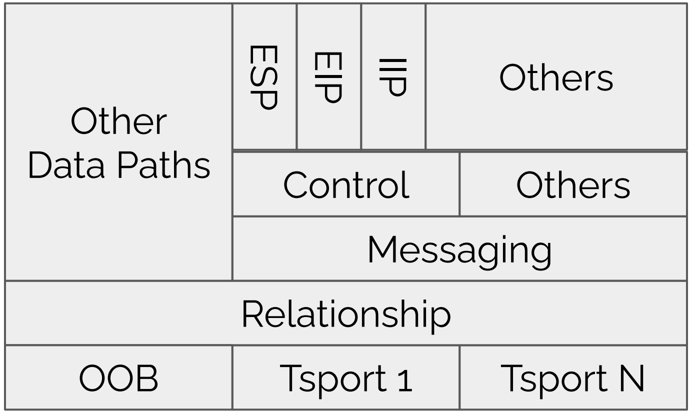
Notes

The 3-party model in SCAP is very similar, although I prefer multi-party formulation over 3-parties.

Other cases 2, 3, 4 are skipped here in the interest of time & simplicity

ITDP Spec sections	What it covers	Notes
Introduction	Introduce motivations and the overall structure	
Use cases	Use canonical cases to help delineate scope of work	
Reference architecture	Illustrate general architecture approach, define key terms, concepts, abstractions, methods, …	
Trust source support abstraction	Abstract interface between an Endpoint and its Support System	(1)
Trust relationship	Specify trust relationship among endpoints	(2)
Messaging	Specify the required messaging protocol & relationship protocol	(2)
Transport abstraction	Abstract interface to any acceptable transport mechanisms	(2), (1), (4)
Trust verification abstraction	Abstract interface between an Endpoint and another endpoint's Support System	(3)
Trust intermediary abstraction	Abstract interface between an Endpoint and an intermediary	(4)
Trust control	Specify a control protocol that endpoints can use for other extended operations	(5)
Data path by control channel	Specify methods to use the control channel to establish another data path	(6)
Intermediary routing	Abstract interface between two intermediaries	(7)

ITDP sub-layering



Other Data Paths

ESP

EIP

IIP

Others

Control

Others

Messaging

Relationship

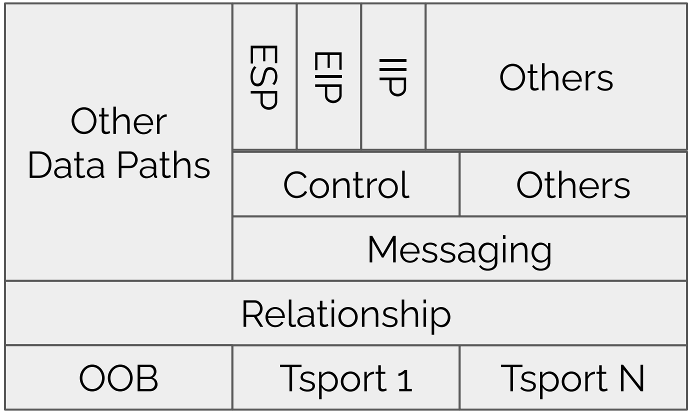
OOB

Tsport 1

Tsport N

ITDP with SCAP

ITDP with SCAP



SCAP provides relationship to messaging to new relationship bootstrap cycle & the eventual trust messaging channel.

ITDP and SCAP MAY be complementary...

- Agreements include: Both aligned with the Reference Architecture
  - Compatible definitions of: Endpoints, Intermediaries, Support Systems
  - Same definition of authenticity, confidentiality and privacy
- Trust/Threat models are aligned
- SCAP specifies ONE solution to (1): AID. ITDP wants to allow other VIDs.
- SCAP specifies ONE solution to (2): Relationship and Messaging. We only need ONE for the control channel - so if SCAP works out, then it may also meet this key requirement.

1. But potential open issues include: SCAP is AID specific. Can it accommodate other VIDs with abstraction?
2. ITDP supports many patterns of communications which are critically important. Can SCAP support other patterns: p2c, group, broadcast etc.? Privacy tradeoffs may be different in each case.
3. In SCAP, messaging is the channel for all trusted communications. In ITDP, it is considered as the control channel among potentially many other possible channels of communication, esp data path channels. This is a different approach but I think it can be compatible with general relationship formation which leads to channel formation.
4. SCAP analysis. E.g. is scalability sufficient for messaging & control channel purposes? Privacy models?
5. As noted in #2, ITDP intends to have flexible privacy tradeoffs based on patterns of use & with regard to intermediaries. Can SCAP support the variations with one protocol? (I think yes)
6. IP addr should be replaced with transport addr abstraction.

Feedbacks ? Next steps.