

Design features vs. potential benefits of id / loc separation

Internet area meeting, Thu 23 2007

IETF 68, Prague

Pekka Nikander

Presentation outline

- Design features
 - Deployment scenarios
 - Implementation loci
 - Identifier structure and properties
 - Resolution models
- Obvious benefits
- Potential benefits
- Conclusions

Design features

- Deployment scenarios
- Implementation loci
- Identifier structure and properties
- Resolution models

Deployment scenarios

(stolen from the LISP draft)

- S1: identifiers fully conventionally routable
- S1.5: identifiers routable over another infra
 - e.g. IPv6 identifiers vs. IPv4 locators
- S2: identifier–locator mapping from the DNS
- S3: advanced: new id-based routing / query infra
 - e.g. based on compact name-independent routing, such as [Abraham et al 2004].

Implementation loci

- Architectural
- Vertical locus
 1. Within app / library
 2. In IP stack proper
 3. Below IP
- Reflects primary, designed trust model
- Implementational
- Horizontal locus
 1. Within host
 2. First hop router
 3. Site border router
 4. ISP
 5. Tier I ISP
- Reflects deployment
 - incl. trust model there

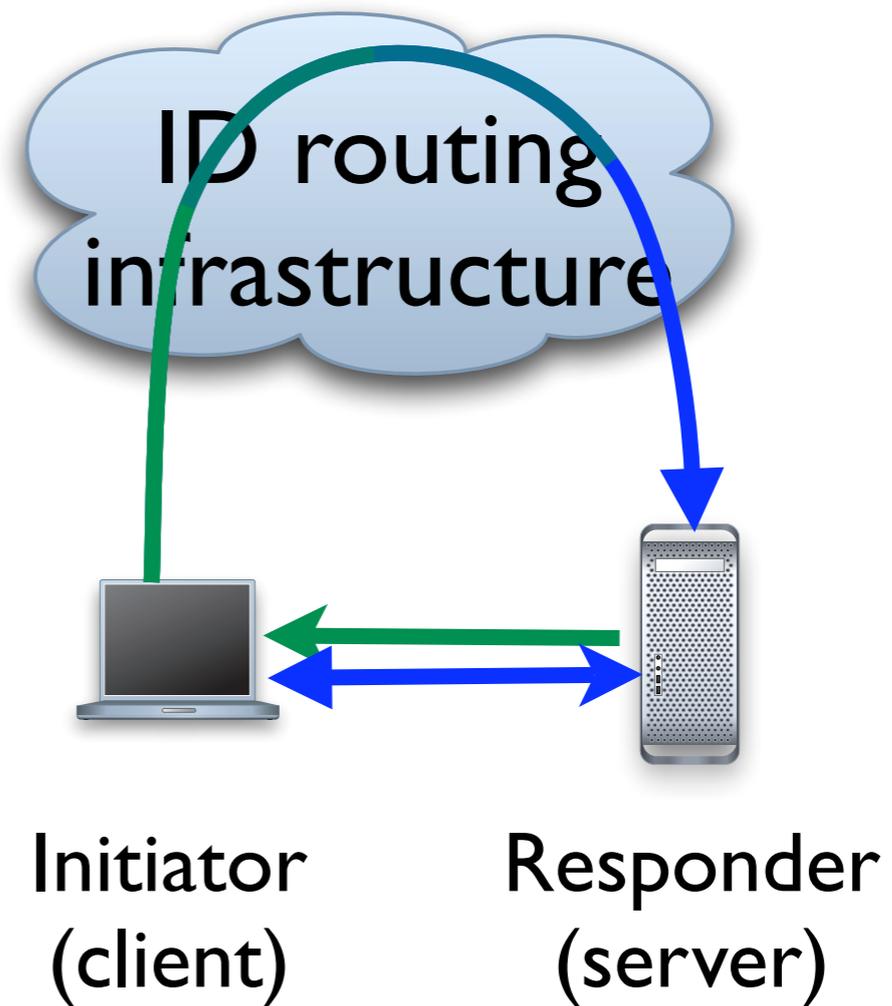
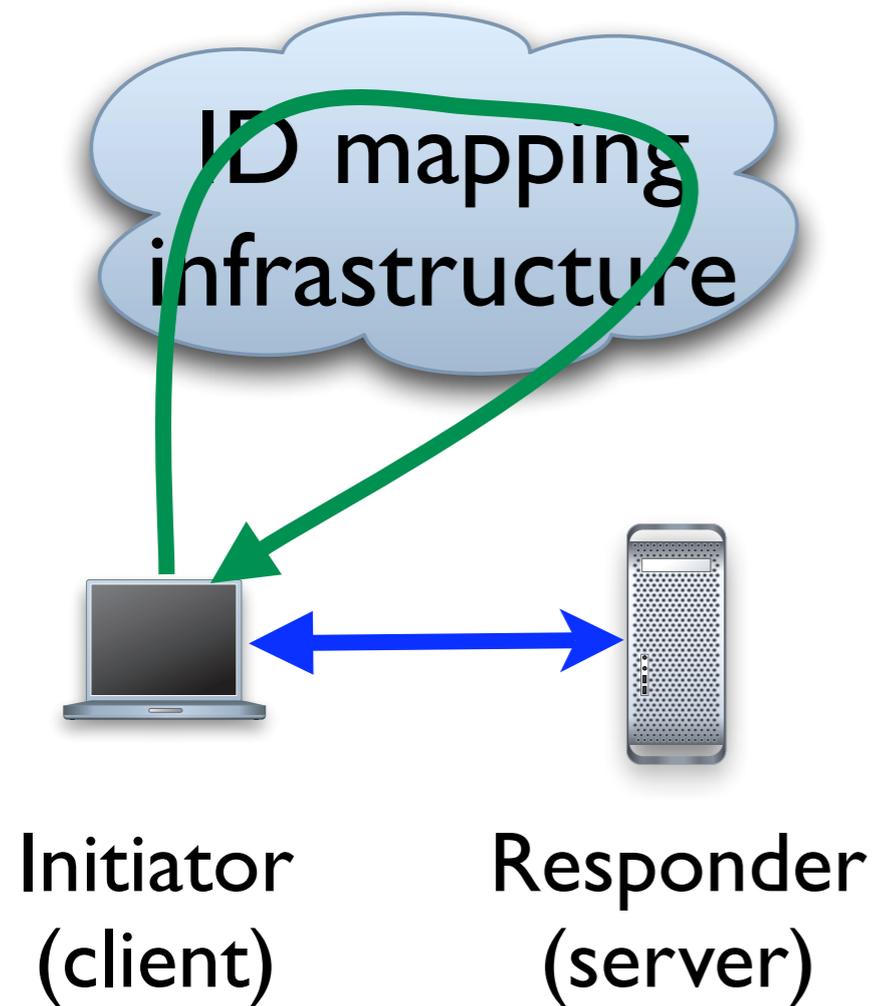
Identifiers: Structure and Properties

- Structure: hierarchical / flat / other
- Uniqueness: statistical / managed
- API backwards compatibility:
 - IPv4 / IPv6 / both
- Routability: global / local / none
- Security: self-authenticating / not

Resolution models

Query based

ID-routing based



Presentation outline

- Design features
 - Deployment scenarios
 - Implementation loci
 - Identifier structure and properties
 - Resolution models
- ▶ Obvious benefits
- Potential benefits
- Summary

Obvious benefits

(not included in the comparison matrices)

- Stable identifiers for everyone
 - No need for provider independent locators
 - More freedom to change ISPs
 - Some NAT problems maybe alleviated
 - ... depends on details; see next slides

Potential benefits

- Eases pressure on the locator **routing table**
- Helps **traffic engineering** and **site multi-homing**
- Provides **end-node mobility** and **multi-access**
- Provides **sub-network mobility**
- Provides **interoperability** between **IPv4** and **IPv6**
- Makes **middle boxes** “first class citizens”
- Supports **delegable** application-level naming
- Provides for **DoS** and/or **DDoS** protection

Background for the next slide

(see the additional slides)

- Early evaluation of some solution proposals vs. the potential benefits
- Extraction of a number design options and considering them vs. the potential benefits
- Some recorded in additional slides
 - Too much to cover here
 - Too subjective at this point of time
- Going to be opened up in
`draft-nikander-ram-ilse-XX.txt`

Summary

- More features further down the road
 - Purely network-based solutions tend to limit what is possible in terms of overall features
- Surveyed above-IP approaches limit RIB / TE benefits
- Two distinct communities
 1. Jack-up / “routing” focus
 2. ID-loc split / “mobility” focus
 - A social or technical contrast?
- Is one solution possible? Do we need two separate ones?

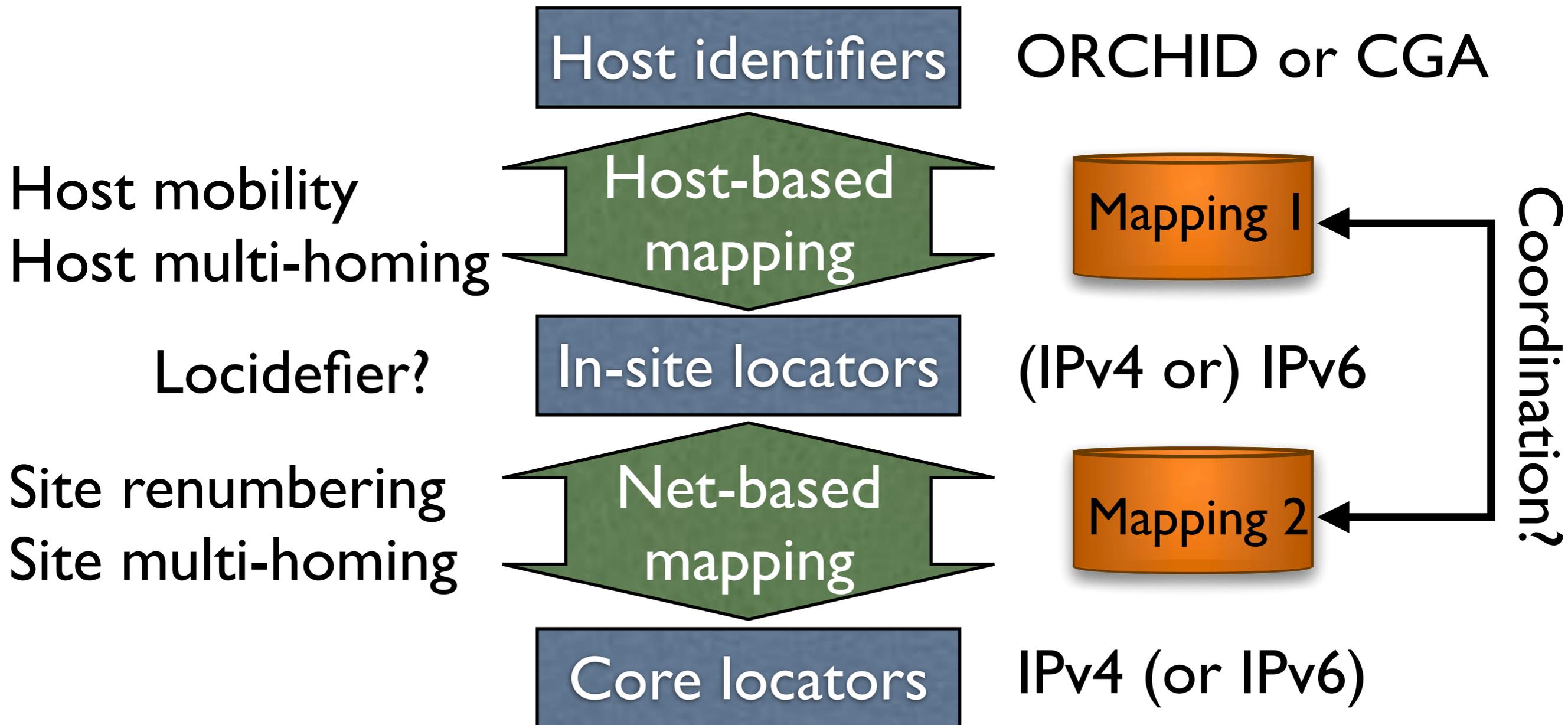
Additional slides

(for reference)

One possible solution architecture

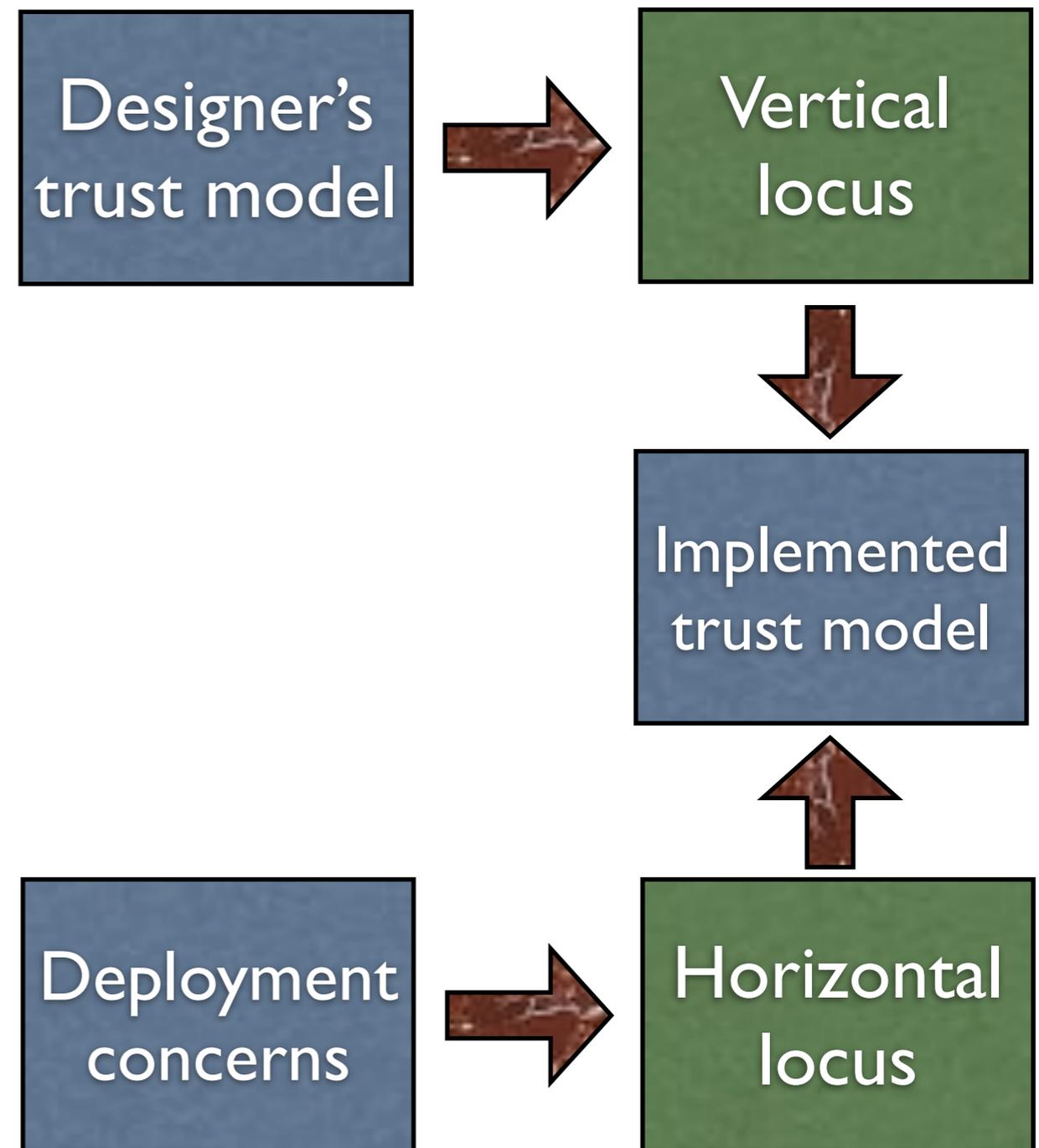
Privacy?

Security?

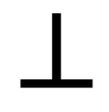


Trust models

- Whom to trust?
 - Host Owning a problem?
 - Choice between ISPs?
- Recall: vertical vs. horizontal locus quite independent
 - You can delegate...
- Real question: the designed, built-in trust model
 - Deployment model can wait; flexibility there



Correlation matrices: notation

	Feature <u>implemented</u> *
	Feature <u>designed</u> but not implemented
	Loose <u>specification</u> exists
	Back-of-the- <u>envelope</u> design exists
	Thought to be <u>possible</u> ; no design exists
	<u>Orthogonal</u> or mutually neutral issues
	Not analysed; open interactions
	<u>Incompatible</u> ; suspect architectural conflict

* the icon is a running imp

This slide is subjective and contains mistakes

	A	C	Deployment				Horizontal			Vertical			Resol.	
	Self-authenticating identifiers	Both IPv4 and IPv6 representation	1 Conv. routable identifiers	1.5 Separ. routable identifiers	2 DNS-based id→loc lookup	3 New mapping mechanism	At hosts	At site border	Deeper in network	Above IP	At ~IP	Below IP	Query-based	ID-routing
RT	⊥	⊥	🚩	✎	✎	✎	✉	✎	✎	🚩	✉	✎	⊥	?
TE	🌿	⊥	🚩	✉	✉	✉	?	✉	🌿	🚩	?	✉	⊥	⊥
Mobility and multi-homing	👤	⊥	🚩	?	📡	📡	👤	✉	🚩	👤	👤	🌿	⊥	✎
Mobile subnetworks	👤	⊥	🌿	?	👤	✎	👤	✉	🚩	🚩	👤	🌿	⊥	⊥
IPv[46] interop	👤	👤	🚩	🚩	👤	✎	👤	🚩	🚩	👤	👤	🌿	⊥	⊥
1 st class middlebox	👤	⊥	🚩	🌿	👤	✎	👤	🌿	🚩	👤	👤	?	🚩	✉
Delegative names	👤	⊥	⊥	⊥	⊥	⊥	👤	🌿	🚩	✎	👤	?	⊥	⊥
DoS resistance	👤	⊥	🚩	?	🚩	✎	👤	🌿	?	🚩	👤	✉	🚩	✎

This slide is subjective and may contain mistakes

	LISP	PASH	SHIM6	Unmodified HIP	Plain jack-up (e.g. MPLS)
RT					
TE					
Mobility and multi-homing			 & 		
Mobile subnetworks					
IPv[46] interop					
1 st class middlebox					
Delegative names					
DoS resistance					