THE HOST IDENTITY INDIRECTION INFRASTRUCTURE (Hi^3): Analysis of the cost

Talk for Host Identity Protocol RG, IETF63

DMITRY KORZUN, ANDREI GURTOV, PEKKA NIKANDER

http://infrahip.hiit.fi
Helsinki Institute for Information Technology

August 2005

Outline

- Hi^3 **architecture**

  Separating control and data

  Requests to the control plane

- **What problems are important?**

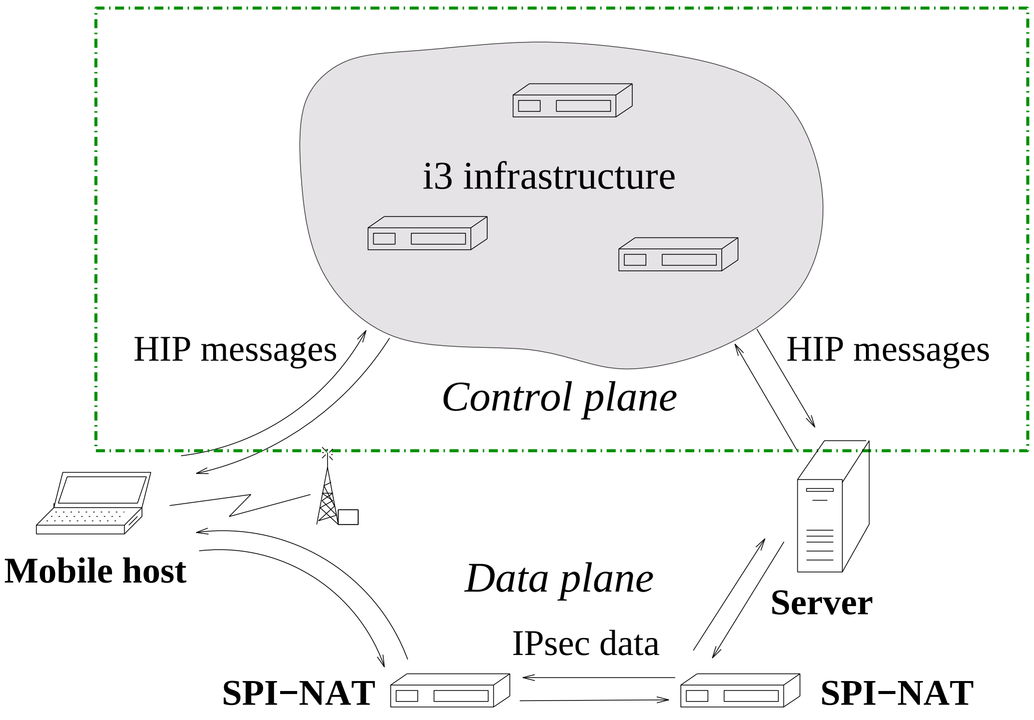
- **Analysis**

  Latency of requests

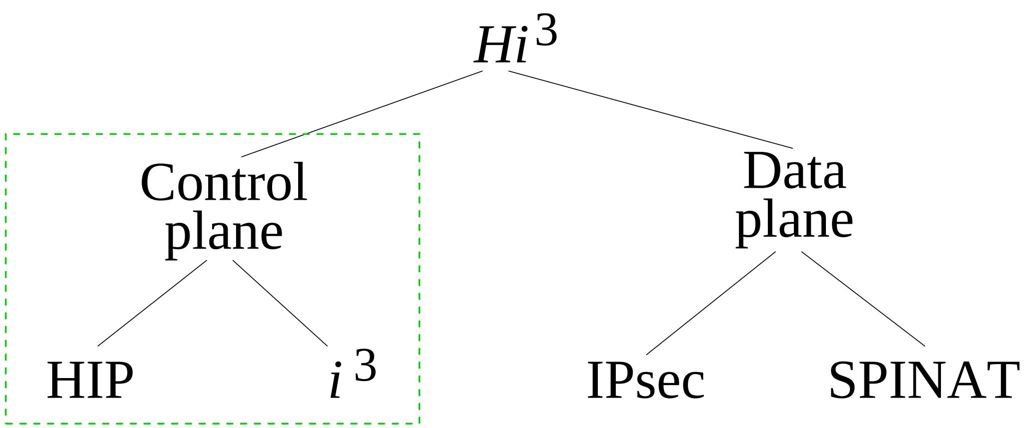
  Workload

- **Conclusion**

Hi^3 **architecture:** Rendezvous enhancement



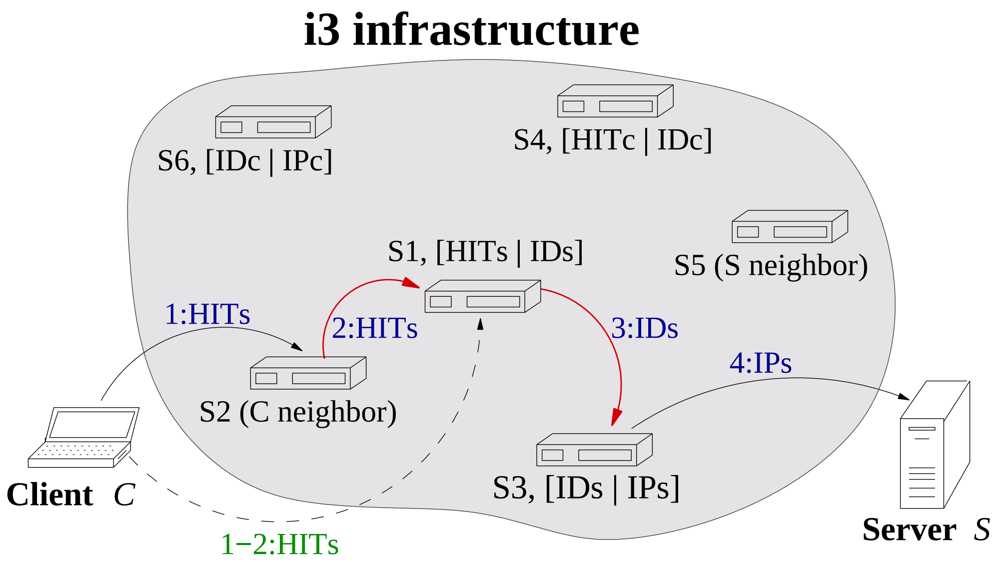
- HIP rendezvous server $\rightarrow$ overlay rendezvous infrastructure (distributed, decentralized)
- Trusted third-party for establishing and keeping the data plane connectivity



HIP messages (control plane):

- base exchange
- mobility exchange
- …

Hi^3 **architecture:** Naming implementation



- Public/private trigger pair to identify a host
- Public identifier is HIT-based
- Private identifier is constructed by the end-host
- Chord lookups in $O(\log N)$ time

	public		private
To host S:	$[\mathrm{HIT}_S \vert \mathrm{ID}_S]$	$\rightarrow$	$[\mathrm{ID}_S \vert \mathrm{IP}_S]$
To host C:	$[\mathrm{HIT}_C \vert \mathrm{ID}_C]$	$\rightarrow$	$[\mathrm{ID}_C \vert \mathrm{IP}_C]$

What problems are important?

- **Utilization**

  $U_{\mathrm{CPU}} = ?, \quad U_{\mathrm{COM}} = ?$

- **Scalability**

  i^3 size estimation: $N = N(\mathrm{workload}, \mathrm{latency})$

- **Resilience to zombie attacks**

  proportion $\quad \#\mathrm{zombies} : N$?

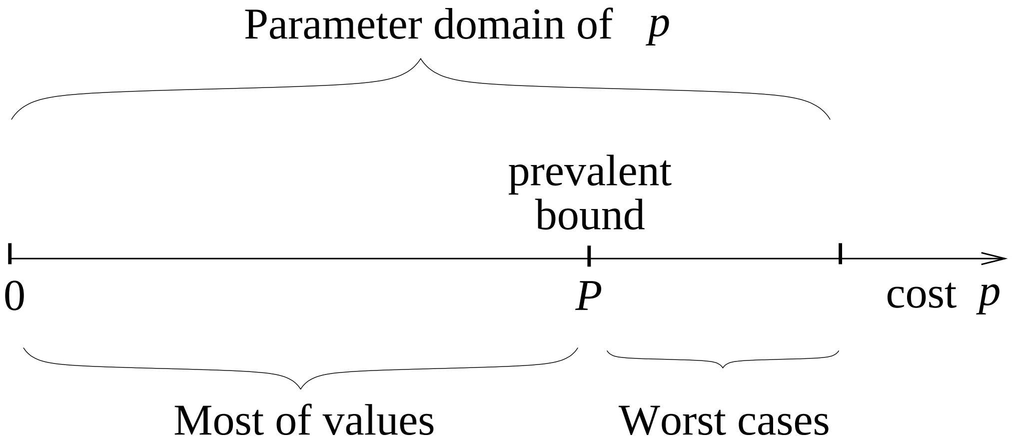
- **Short-term and long-term performance**

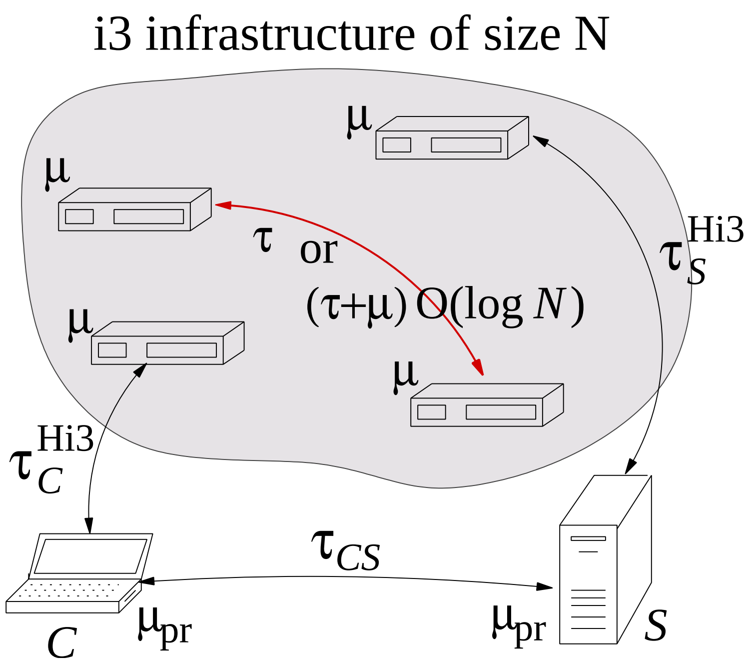
  Balance, stable state, small changes, capacity

- **Forwarding performance**

- **Consistency/availability/stability balance**

Analysis: Basic costs





- Transmission cost:
  - τ: node-to-node trip time
  - τ_A^{Hi3}: host-to-node or node-to-host trip time
  - τ_{CS}: one-way trip time
- Processing cost:
  - μ: forwarding cost
  - μ_{pr}: HIP cryptography cost
- Chord lookup:
  - $(\tau + \mu)O(\log N)$: with high probability
  - $\alpha(\tau + \mu)\log N$: upper bound (const $\alpha > 0$)

Analysis: Latency

Request type	k	$T^{\mathrm{Hi3}} = k\tau^{\mathrm{Hi3}}$	$T^{\mathrm{out}} = k\tau^{\mathrm{out}}$
Pure association setup	4	$6\alpha(\tau + \mu)\log N$	$4\tau_C^{\mathrm{Hi3}} + 2\mu_{\mathrm{pr}} + 4\tau_S^{\mathrm{Hi3}}$
Opt. association setup	4	$2\alpha(\tau + \mu)\log N$	$3\tau_C^{\mathrm{Hi3}} + 2\mu_{\mathrm{pr}} + \tau_S^{\mathrm{Hi3}} + \tau_{SC}$
Loc. update, $A \in \{C, S\}$	2	$\tau + \mu$	$2\tau_A^{\mathrm{Hi3}}$
Double-jump	2	$\alpha(\tau + \mu)\log N$	$\tau_C^{\mathrm{Hi3}} + \tau_S^{\mathrm{Hi3}} + \tau_{SC}$
HIT insertion, $A \in \{C, S\}$	2	$2\alpha(\tau + \mu)\log N$	$2\tau_A^{\mathrm{Hi3}}$
HIT refreshment, $A \in \{C, S\}$	4	$2(\tau + \mu)$	$4\tau_A^{\mathrm{Hi3}}$

k**:** packets in a request

τ^{Hi3}**:** internal latency of a packet

τ^{out}**:** external latency of a packet

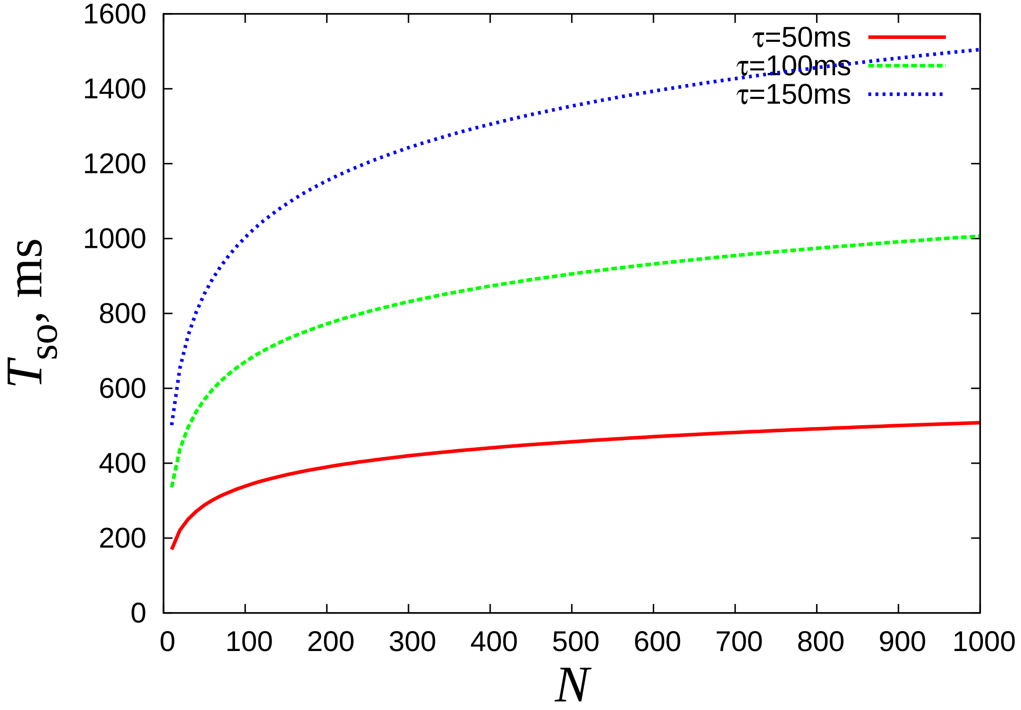
T^{Hi3}**:** internal latency of a request

T^{out}**:** external latency of a request

$L = T^{\mathrm{Hi3}} + T^{\mathrm{out}}$**:** request latency

Analysis: Internal latency

Optimized association setup, $T_{\mathrm{so}}^{\mathrm{Hi3}}$
$\mu = 1\text{ms}, \quad \alpha = 1/2$

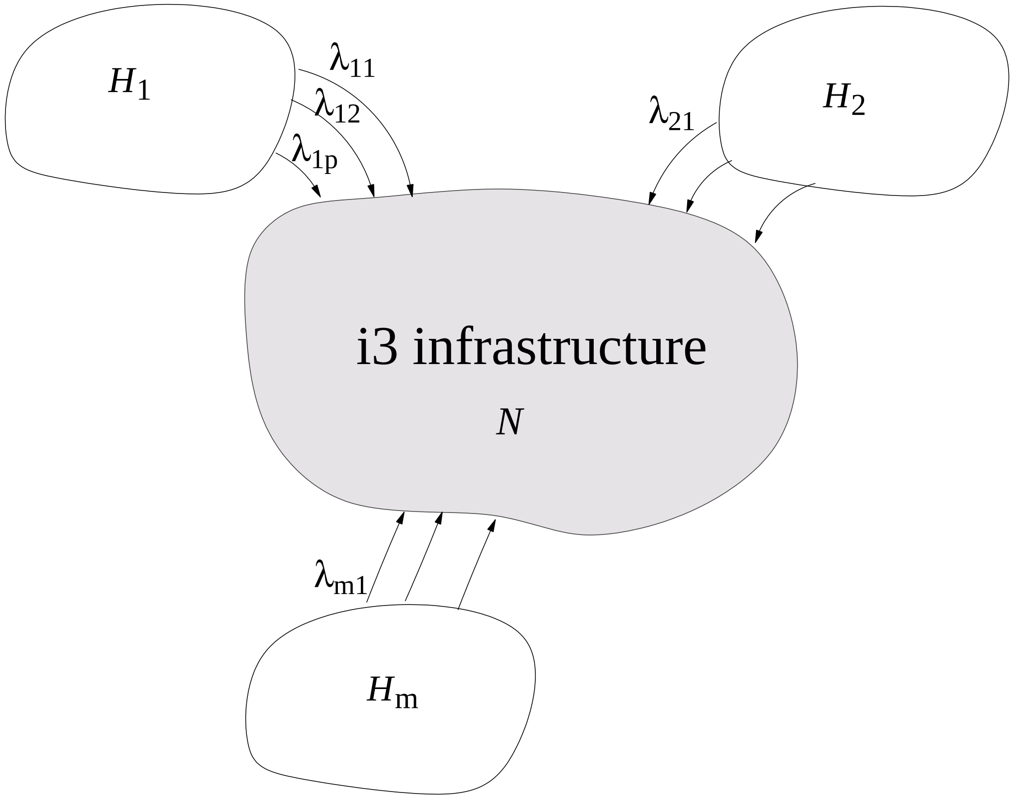


- **Slowly increasing latency** even in the worst case (with lookups)

$$T^{\mathrm{Hi3}} = (\tau + \mu)O(\log N)$$

- **Several seconds** for $O(\log N)$-requests
- **Primary factors:**
  - Lookup cost $O(\log N)$
  - node-to-node trip time τ for a lookup path
- **Design solutions:**
  - i^3 caching
  - trigger allocation

Analysis: Workload pattern



Parameters:

H**:** #end-hosts

λ**:** rate of a end-host

r**:** #nodes loaded by a request

Workload metric:

$$W = \frac{\lambda H r}{N}$$

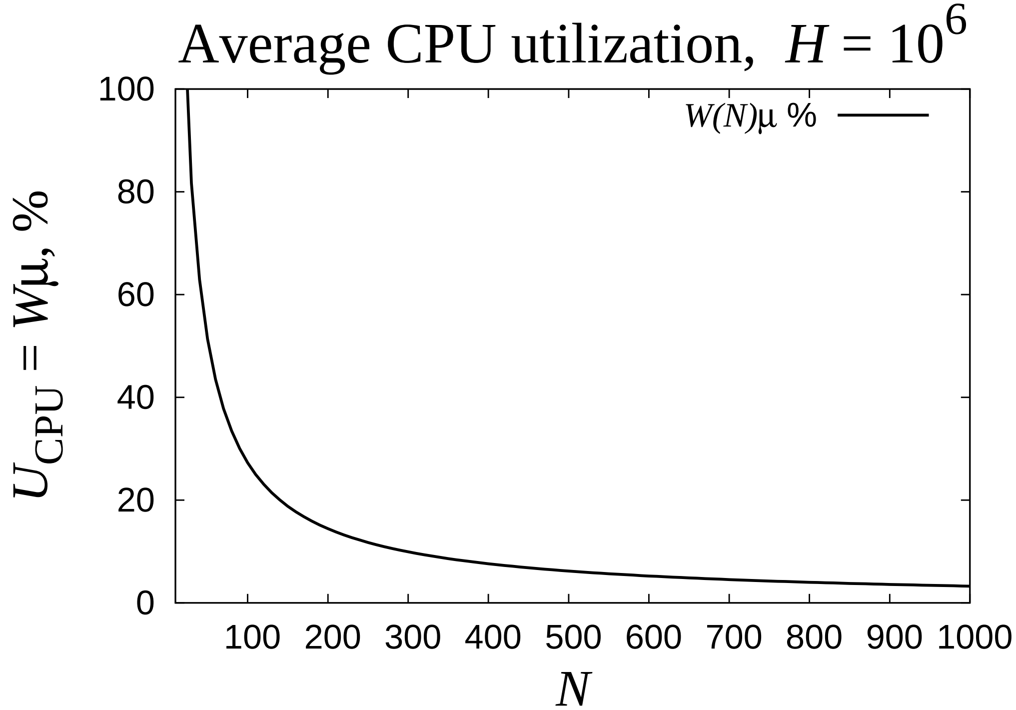
i.e., how many packets a node serves

Analysis: Workload estimates

Request type	**Rate,** λ	**#(**i^3 **nodes),** r	**Workload,** W
Pure association setup	λ_{s}	$6\alpha \log N$	$W_{\mathrm{s}} = \frac{6\alpha\lambda_{\mathrm{s}} H \log N}{N}$
Opt. association setup	λ_{so}	$2\alpha \log N$	$W_{\mathrm{so}} = \frac{2\alpha\lambda_{\mathrm{so}} H \log N}{N}$
Location update	λ_{u}	1	$W_{\mathrm{u}} = \frac{\lambda_{\mathrm{u}} H}{N}$
Double-jump	$\lambda_{\mathrm{u}} P_{\mathrm{us}}$	$\alpha \log N$	$W_{\mathrm{us}} = \frac{2\alpha\lambda_{\mathrm{u}} P_{\mathrm{us}} H \log N}{N}$
HIT insertion	λ_{i}	$2\alpha \log N$	$W_{\mathrm{i}} = \frac{2\alpha\lambda_{\mathrm{i}} H \log N}{N}$
HIT refreshment	λ_{r}	2	$W_{\mathrm{r}} = \frac{\lambda_{\mathrm{r}} H}{N}$

Analysis: Workload behavior

$\mu = 1\text{ms}, \quad \lambda_{\text{s}} = \lambda_{\text{so}} = 30\text{min}^{-1}$
$\lambda_{\text{u}} = 1\text{min}^{-1}, \quad P_{\text{su}} = 10^{-2}$



$H = \#\text{end-hosts}$

- **Rapidly decreasing workload:**
$$W = O\left(\frac{\log N}{N}\right) + O\left(\frac{1}{N}\right)$$
- **Reasonable proportion** between end-hosts and nodes:
$$H : N \ \sim \ 10^6 : 10^2$$
- **Workload/latency trade-off:**
$$N \approx C \cdot \frac{T^{\text{Hi3}}}{W}$$

Conclusion

- **Simple assumptions**

  i^3: τ, $O(\log N)$-requests

  end-hosts: λ, H

- **Coarse estimates**

  basic trends and their order

- **Reflection in design**

- **More accurate model?**
  - forwarding packets
  - heterogeneity
  - network flows approach