NUTSS: An End-Middle-End Approach to Connection Establishment

Saikat Guha and Paul Francis

Cornell University

SIGCOMM 2007
End-Middle-End: Why?

Originally, Internet supposed to provide:

1. User-friendly naming of hosts (DNS)
2. Network level identification of hosts (IP address) and best-effort delivery
3. Identification of application on host (port)
Implicit assumption:

- Application can defend itself. Competent to look inside packet.
- Wrong. (DoS, software bugs, . . .)
- Resulted in firewalls
  - Compromised end-only control
  - Cannot identify application. Or hosts behind NAT.
  - Resort to deep-packet inspection
  - Endhost unaware
- Made network brittle
- Often legitimate connections fail!!!
End-Middle-End: Why?

Required additional Internet services

4. Block unwanted packets before they reach application

5. Explicit negotiation of middlebox usage.
   - Need not be on data path

---

End-Middle-End

These services, along with original three, represent the minimum requirements for the Internet.
NUTSS is an architecture and protocol that instantiates End-Middle-End

**Primary Goal**

Allow connection establishment that honors access control policy of all stakeholders (ends and middle).

Also, middlebox steering, host mobility, anycast, redirection, multi-homing, multicast, protocol negotiation
End-Middle-End and End-To-End

- E2E broken by middleboxes
  - Middlebox control in the middle
  - Endpoints oblivious of middle, cannot adapt
- EME exposes functionality in the middle
- Allows ends and middle to cooperate in middlebox control
  - Explicit two-way negotiation between ends and middle
  - firewall policy, NAT ports, protocol stack
Names vs. Identifiers

Names or identifiers?

- Identifiers are scalable, efficient, can be self-certifying BUT not for the middle
- Middle needs (user-friendly) names for policy
- Must be aggregatable
  - Identifiers (HIP, i3, DONA) don’t allow for this
  - Need additional reverse name resolution
- Internet-wide shared namespace
Where is policy applied?

- **On-path (on the data path)**
  - Privacy (for address-based paths\(^1\))
  - Constraining (name-resolvers on-path)
  - Intrusive (routers route by name)

- **Off-path (separate control plane)**
  - Replicate, deploy far from endpoint (DoS, scalability)
  - But data path is address-based

---

\(^1\) “Identity Trail: Covert Surveillance Using DNS” in PET ’07
Address-routed path, off by default
Name-routed path, on by default
Overlay of stakeholders.
Address-routed path, off by default
Name-routed path, on by default
Overlay of stakeholders.
- Address-routed path, off by default
- Name-routed path, on by default
- Overlay of stakeholders.
Address-routed path, off by default
Name-routed path, on by default
Overlay of stakeholders.
- Address-routed path, off by default
- Name-routed path, on by default
- Overlay of stakeholders.
Turning on data path

How to determine impending data path?

- Control plane fixes path
  - Constraining (virtual circuit)
- Control plane guesses path
  - Recovers from incorrect guess
User-friendly, long-term stable, aggregatable names

Off-path signaling
- Name-based overlay
- Applies policy
- Authorization token

On-path signaling (of token)
- Verify data-path works
- Referral back to off-path if fail
NUTSS: Components

- P-Box/M-Box associated, possibly same device
- Also in-host
- P-Box overlay (parent-child, fan-in, fan-out)
P-Box/M-Box associated, possibly same device
Also in-host
P-Box overlay (parent-child, fan-in, fan-out)
P-Box/M-Box associated, possibly same device
Also in-host
P-Box overlay (parent-child, fan-in, fan-out)
Endpoints register with P-Box chain in front. DNS has outermost P-Box address.
NUTSS: Name-Routing

Up (config./discovery), Across (DNS), Down (registration)
NUTSS: Name-Routing (Tokens)

- P-Box gives token \( \langle \text{nonce, next-hop} \rangle \) to M-Box via endpoint.
- Set of tokens. One for each P-Box/M-Box pair.
- Exchange effective addresses (may be of M-Box)
Once endpoint has effective address and tokens
NUTSS: Referral

What if address and name routed paths differ
What if address and name routed paths differ
What if address and name routed paths differ
What if address and name routed paths differ
NUTSS: Referral

Referral from M-Box to P-Box
Resumes name-routed signaling for more tokens
Resumes name-routed signaling for more tokens
NUTSS: Some Use Cases

- Mobility
  - Register new address with P-Box overlay.
  - Renegotiate flows.
- NAT Traversal
  - Exchange hole-punched address and port over name-routing
- Anycast, Multicast
  - Multiple endpoints share same name
  - P-Box forwards to one (to all for multicast).
  - Address routed path negotiated (possibly application multicast or IP multicast)
- Protocol negotiation
  - Endpoints advertise software stack (transport, security, network etc.)
  - P-Box filter out unsupported stacks
1. Update applications to perform dual-signaling. 3-rd party P-Box service.
   - Implemented as a userspace library. Works with legacy apps.
   - P-Box service on nutss.net
   - NAT traversal helper M-Box on Planetlab

2. Networks deploy P-Boxes. Only weak access control (but better than firewalls today).

Summary and Future Work

- End-Middle-End requirements, NUTSS architecture and protocol.
- Need for dual-signaling: Name-routed and address-routed signaling.
- Coupling between the two can solve a broad range of Internet problems:
  - Network ACL, mobility, multihoming, steering, protocol negotiation, ...
- Pursued in the E-M-E RG in the IRTF.
- Investigate non-FQDN based naming, non-DNS “across” routing, multipath connections, secure P-Box discovery.

http://nutss.net/
Related Work

- Endpoint-only control
  - TRIAD, i3, IPNL, HIP, SHIM6
- Middle involved only in name resolution
  - Metanet, Plutarch, UIA, DONA, AVES

- Off-path only
  - SIP
- On-path only
  - i3, HIP, RSVP
NUTSS: Optimizations

- Lower latency
  - Piggyback application-data in signaling messages
- Faster authorization
  - Use self-certifying ID’s
NUTSS: Dual-Signaling

Name-routed
- \langle \text{user@domain, app} \rangle
- P-Boxes (overlay)
- Path always exists (Default on)
- Policy decision
- Tokens

Address-routed
- IP address\(^2\) and port
- M-Boxes (on IP path)
- Initially, does not exist or blocked (Default off)
- Policy enforcement
- Referral

\(^2\)or other address e.g. i3, HIP, etc.