Feasibility of ILA as Network Virtualisation Overlay in multi-domain, multi-tenant Cloud

Tako Marks Bsc.

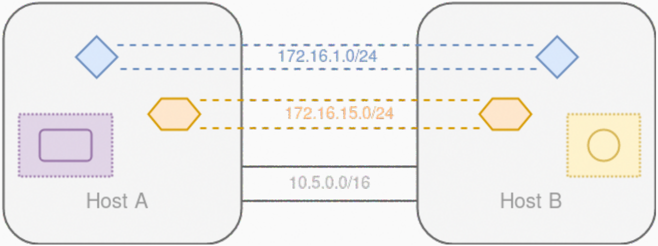
04-07-2017

System and Network Engineering
University of Amsterdam

Introduction

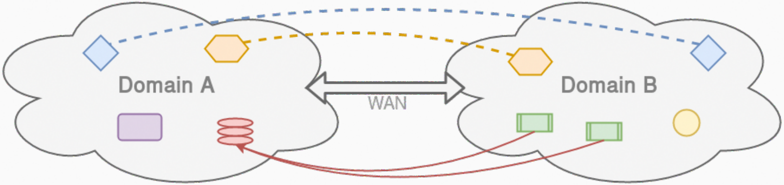
A **Network Virtualisation Overlay (NVO)** is a collection of virtual nodes and virtual links comprised of a subset of the underlying physical network resources.

This allows for multiple customers (tenants) to share the same physical network infrastructure. Similar to how Virtual Machines are used to share Physical Servers.



Introduction

- Multi-domain
  - **Instead of single infrastructure provider there are multiple**
  - Each provider wants control over own subset of the overlay
  - Overlay should be able to traverse WAN links
  - **Containers should be able to move between domains**
- Multi-tenant
  - Multiple tenants share physical infrastructure
  - By default tenants need to be isolated from each other
  - **Allow custom ACL between tenants with minimal friction**



Problem Statement

How to build shared virtual cloud based that spans multiple domains where a large number of organisations can work together on data processing and warehousing using containers.

Possible solution

- **ILA**: Identifier Locator Addressing
- ILA is a **Network Virtualisation Overlay** based on IPv6
    - Developed at Facebook, first draft RFC July 2015
    - Initial inclusion in Linux kernel 4.3 (August 2015)
    - Only provides layer 3 connectivity
- ILA advantages:
    - Each process, container or virtual machine can have an unique IPv6 address
    - Overlay addresses provide **mobility using Identifier/Locator split**

Research Questions

Is it feasible to use ILA in as a Network Virtualisation Overlay for a multi-domain, multi-tenant Cloud?

- What are the requirements for Network Virtualisation Overlay in a multi-domain, multi-tenant environment?
- What are possible ILA configurations that satisfy the multi-domain, multi-tenant environment requirements?
- What would be a suitable control plane for ILA when used as a NVO in a multi-domain, multi-tenant environment?

Related Work

Identifier/Locator split concept

Protocol Level	Current identifier	Identifier/Locator split
Application	FQDN / IP Address	Identifier
Transport	IP Address (+port)	Locator (+port)
Network	IP Address	Locator
Interface	IP Address	Locator

- Impossible to replace IP addresses in protocols because of backwards compatibility.

Identifier/Locator split concept

Protocol Level	Current identifier	Identifier/Locator split
Application	FQDN / IP Address	Identifier
Transport	IP Address (+port)	Locator (+port)
Network	IP Address	Locator
Interface	IP Address	Locator

- Impossible to replace IP addresses in protocols because of backwards compatibility.
- Solution: Use transparent translation layer to give application layer the illusion that an identifier still has the same IP addresses semantics

Previous Identifier/Locator split systems

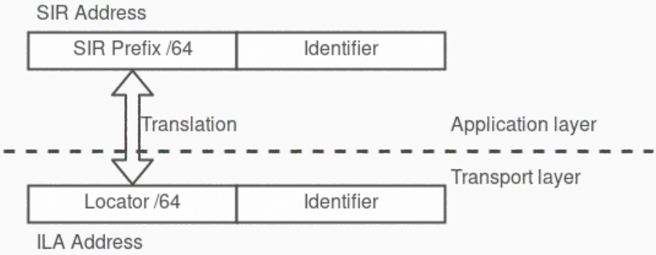
Throughout the years the discussion of overloaded IP address semantics has surfaced many times over[1].

- LISP was introduced first, primarily focused on IPv4 and routing table size
- Recent efforts focus on IPv6, because the address space is so large it is easier to experiment with
  - ILNP, HIPv2
- ILA differs from these earlier approaches because it focuses on a single domain instead of global system

[1] Examples are IEN1, RFC1498 and RFC2956

ILA addressing

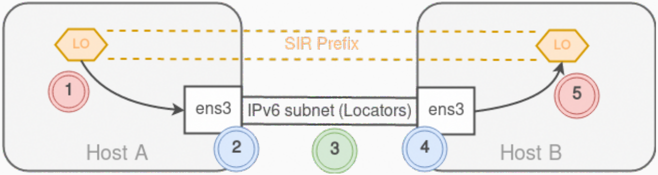
- ILA's overlay does not use any encapsulation but instead (ab)uses[2] Network Address Translation (NAT) to do this.
    - No overhead at transport layer, only minimal stateless translation at ILA hosts.
    - The NAT scheme does not break the re-instated end-to-end principle of IPv6. Phew!



[2] Sorry Karst, we're breaking the Internet again. . .

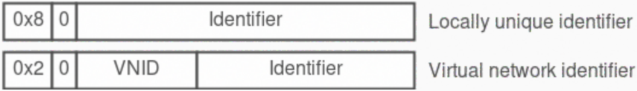
ILA addressing

- Translation done before packet is transmitted on the wire
- Network only sees ILA addresses as destination
- Locator is translated back to overlay SIR prefix at host



ILA Identifier Types

- Identifiers can be pseudo-random addresses assigned to new applications that need network access. But can also be part hierarchically assigned.
- To support this multiple identifier types have been defined[3]:
- Virtual Networking Identifiers that support this have a Virtual Network ID (VNID) encoded in the identifier space.



0x8	0	Identifier		Locally unique identifier
0x2	0	VNID	Identifier	Virtual network identifier

[3] https://tools.ietf.org/html/draft-herbert-nvo3-ila-04

ILA control plane

- To do correct address translations from SIR address to the current locator of a identifier ILA hosts need to have a mechanism to exchange mappings.
- ILA draft RFC leaves control plane unspecified. It is up to the implementors to find suitable choice.

Approach

Approach

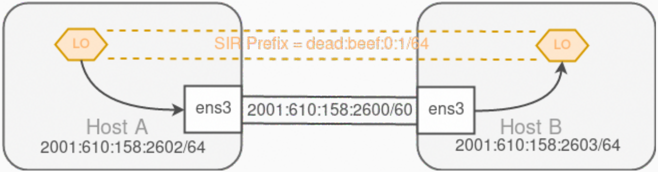
- Create minimal ILA overlay to test on.
- Find possible ILA configurations that meet requirements
  - What are possible addressing schemes for envisioned overlay
- Find suitable control plane that meets requirements
  - Comparison of existing control planes that handle mappings
  - Adapt/Extend most favourable option to suit ILA

Results

Minimal ILA overlay setup

- SIR addresses are inserted as destinations into routing table with Locator as next-hop.
- Translation of address is done by the kernel by encap ila

```
modprobe ila
ip addr add dead:beef:0:1:8000::2 dev lo
ip route add table local local 2001:610:158:2602:8000::2/128
  encap ila dead:beef:0:1 dev lo
ip route add dead:beef:0:1:8000::3 encap ila
  2001:610:158:2603 via 2001:610:158:2601 dev ens3
```



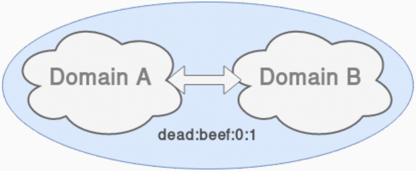
ILA Configuration

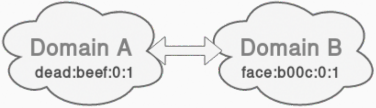
Two different scenarios.

- Single SIR for all domains
- Multiple SIR, one SIR for each domain/provider

For each scenario make sure ILA constraints are met.

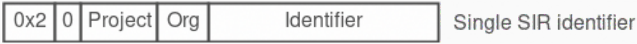
- Only one SIR can be used for each locator address
- Make sure that identifier address does not overlap





Single SIR prefix

- Use VNID space to differentiate between projects and participating organisations.
  - First 18 bits project ID, second 10 bits participating Organisation
    - Gives 262k projects each with 1024 possible organisations
  - ACL can be done on project basis
    - Able to match on prefixes[4]
    - Only coordination needed for project ID & Organisation ID
- Container mobility is guaranteed to work **inter-domain**

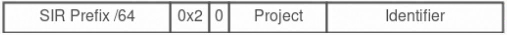


0x2	0	Project	Org	Identifier	Single SIR identifier

[4] Similar to `http://romana.io/`

Multiple SIR prefix

- Use SIR prefixes to differentiate organisations in overlay and VNID to differentiate between projects.
    - More space for project IDs (28 bits)
    - More ACL rules, separate rules for each organisation as they no longer share same prefix
- Container mobility only works **intra-domain** when control plane includes announcement of SIR prefix with identifier mapping.



SIR Prefix /64	0x2	0	Project	Identifier

Multi SIR address

Locator options

- Locators can either be Locally Unique IPv6 (private range) addresses or globally unique unicast addresses
  - Locators need to be global unicast addresses to support multi-domain over WAN links.
  - No need to create tunnels between domains, piggyback on regular IPv6 routing
- When SIR prefix is also global unicast subnet mobility also works for WAN connections.

Existing control planes

- Facebook uses distributed K/V store
- ILNP created new RRTYPES in DNS to mapping
- LISP+ALT creates a global overlay for exchanging mappings
- LISP-DDT re-creates disjoint DNS-like tree to store mappings
- MP-BGP enables to distribute VPN mappings over eBGP

Existing control planes

- Facebook uses distributed K/V store
- ILNP created new RRTYPES in DNS to mapping
- LISP+ALT creates a global overlay for exchanging mappings
- LISP-DDT re-creates disjoint DNS-like tree to store mappings
- MP-BGP enables to distribute VPN mappings over eBGP

Control Plane	**Scope**	**Organisation**	**ACL**
Distributed K/V store	Intra-Domain	Flat	Limited
ILNP	Global	Hierarchical	Limited
LISP+ALT	Global	Flat	Limited
LISP-DDT	Global	Hierarchical	Limited
MP-BGP extensions	Selective (global)	Hierarchical	Many

Table 1: Existing control plane characteristics

ILA BGP Control Plane

- Scales very well when used properly
  - Able to use iBGP internally with route reflector and eBGP to other organisations.
- Leveraging MP-BGP over eBGP to announce mappings gives each organisation control over which prefixes (for projects) to share.
- Can re-use existing methods to filter incoming/outgoing routes/mappings.
  - Combined with firewall ACL gives great security possibilities
- Already a draft extension to MP-BGP available[5]

[5] `https://tools.ietf.org/html/draft-lapukhov-bgp-ila-afi-02`

Conclusion

Conclusion

- ILA is very flexible and can meet multi-domain, multi-tenant network overlay requirements
- Initial addressing design complicated
    - Lot of implicit configuration in address choices
    - Single SIR configuration preferred
- BGP suits multi-domain, multi-tenant environment best as control plane
    - Gives each provider control and flexibility in announcements
    - BGP is known to scale well

Discussion / Future Work

- Middleware to orchestration software still needs to be implemented.
  - Started work on implementing ILA MP-BGP extension for exaBGP daemon.
  - Integration with container orchestration could be done by extending network plugin project Calico[6].
- Use other ILA implementations with increased performance.
  - VPP/DPDK
  - eBPF/XDP

[6] `https://www.projectcalico.org/`

Questions