

# Project IPv6 only

Tore Anderson  
CG Security and Networking  
Redpill Linpro  
RL Gathering, Sunne, October 2011

# Main goals

- Set up and operate a standard MS service delivery platform (an external or internal customer) without any use of IPv4
- Discover IPv4-dependent parts of our infrastructure and routines
  - Get them dual-stacked
- Evaluate various translation techniques to support IPv4 users
- Maintain RL's position as the IPv6 leader in our markets

# Things we'll look at

- Server deployment and management
  - PXE booting/network install, ILO/IPMI cards, software repos
- Managed Services' supporting infrastructure services
  - DNS, NTP, SMTP servers
  - Bacula, Icinga/Nagios, Munin, Puppet, Syslog
  - iSCSI/NFS storage systems
- Network components (firewalls, switches, routers)
  - Use IPv6 auto-configuration or not?
- MS' most commonly used applications
  - LAMP, PostgreSQL, HA suites, Escenic (help!), etc.

# Motivation: IPv4 depletion

- RL currently has 8k IPv4 addresses, ~65% utilisation
- A typical MS customer needs anywhere between 8 and 256 addrs:
  - The servers themselves
  - Public service addresses (e.g. for web. and mail.cust.com etc.)
  - Internal service addresses (e.g. for DB and NFS services)
  - Unused addresses due to the power-of-2 boundary overhead
- The RIPE NCC looks on track to deplete Q1 2012
  - We **might** be able to get another allocation by then (likely 2k)
  - We're guaranteed a single 1k allocation post depletion
- Continuing with today's practise will render us **unable to accept new customers** sometime within the next 12-36 months!

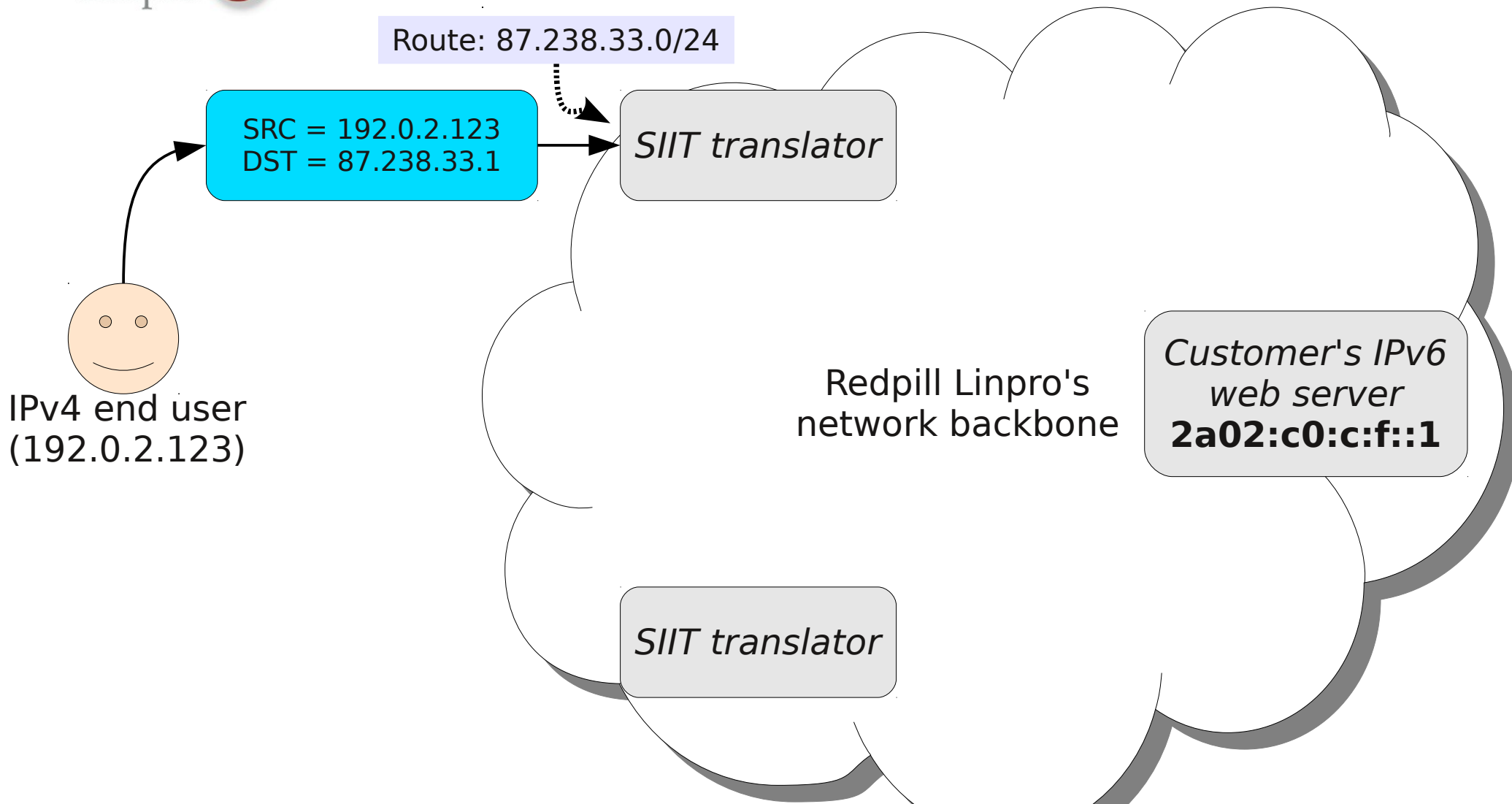
# Some IPv6-less solutions

- Buy IPv4 addresses from other organisations
  - Risky: price and availability are highly uncertain factors
- Use RFC 1918 private addresses instead
  - The good: Familiar technology, the dinosaurs feel at home ;-)
  - Requires NAT44 or proxies to connect to the public IPv4 internet
    - Stateful devices, obvious [D]DoS targets
    - NAT44 devices must be located on-path, which significantly constrains the network architecture and adds complexity
    - Proxies suffer from issues with port starvation and possible loss of information (the original IPv4 address of the user)
- Dual-stack required to simultaneously support IPv6
  - Adds complexity and operational overhead

# Communicating with IPv4 users

- Proxies?
  - Application-mode (Varnish) or transport-mode (HAProxy/TCP)
  - Same advantages/disadvantages as when using private IPv4
- Preferred solution: Translator-In-The-Cloud
  - Should (cloud) ensure (cloud) huge (cloud) budget (did I mention cloud?)
  - **Stateless IP/ICMP Translation** (SIIT; RFC 6145)
  - Maps the entire IPv4 internet into an IPv6 /96 prefix

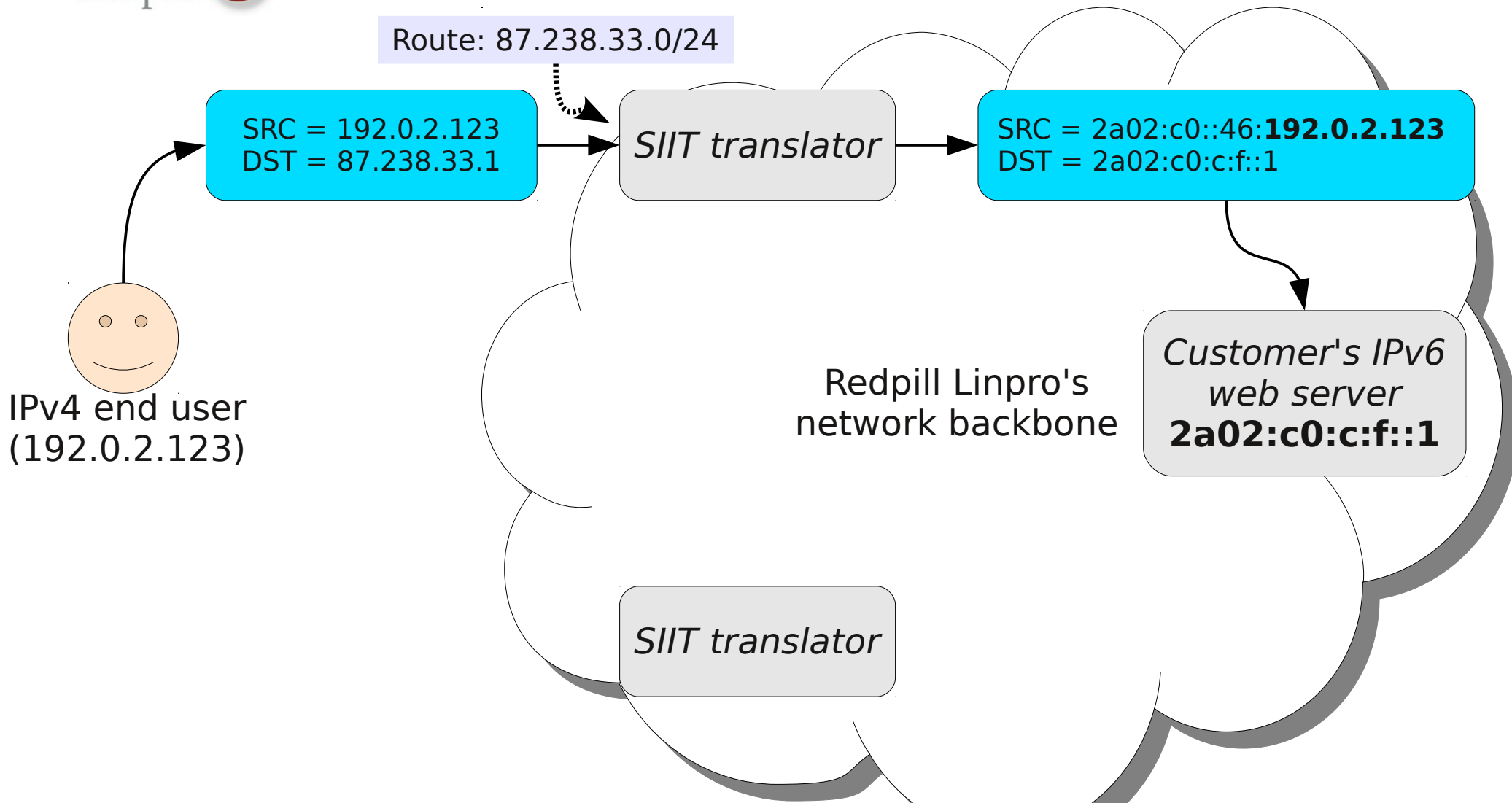
# SIIT explained



The IPv4-only end user sends a packet to the public IPv4 address of the customer. This is routed to the nearest translator using anycast.

```
DNS setup:  
www.cust.no. IN AAAA 2a02:c0:c:f::1  
www.cust.no. IN A 87.238.33.1
```

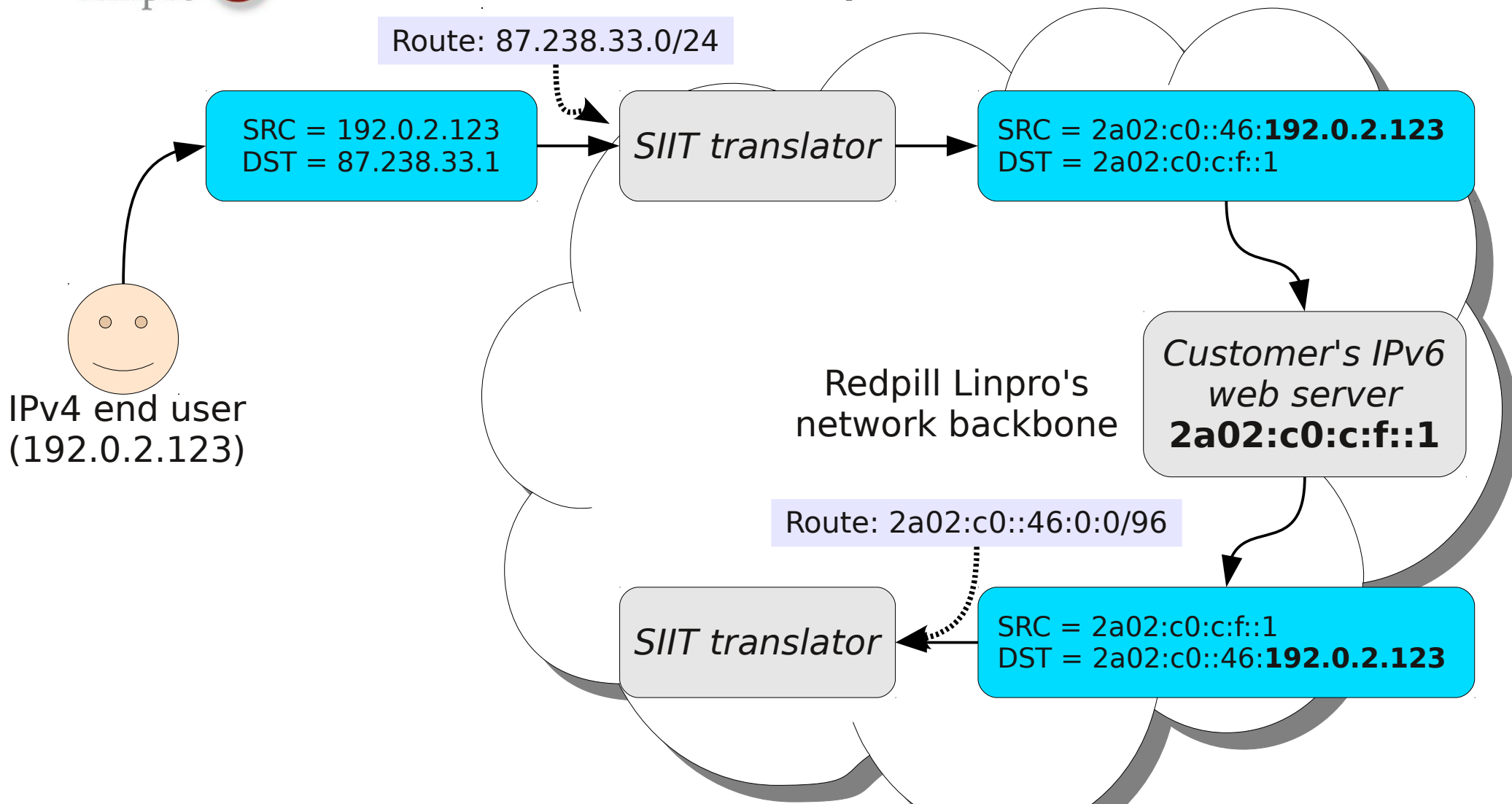
# SIIT explained



The translator prepends a 96-bit IPv6 prefix to the original IPv4 source address, and swaps the IPv4 destination address with the actual IPv6 one.

DNS setup:  
[www.cust.no](http://www.cust.no). IN AAAA 2a02:c0:c:f::1  
[www.cust.no](http://www.cust.no). IN A 87.238.33.1

# SIIT explained

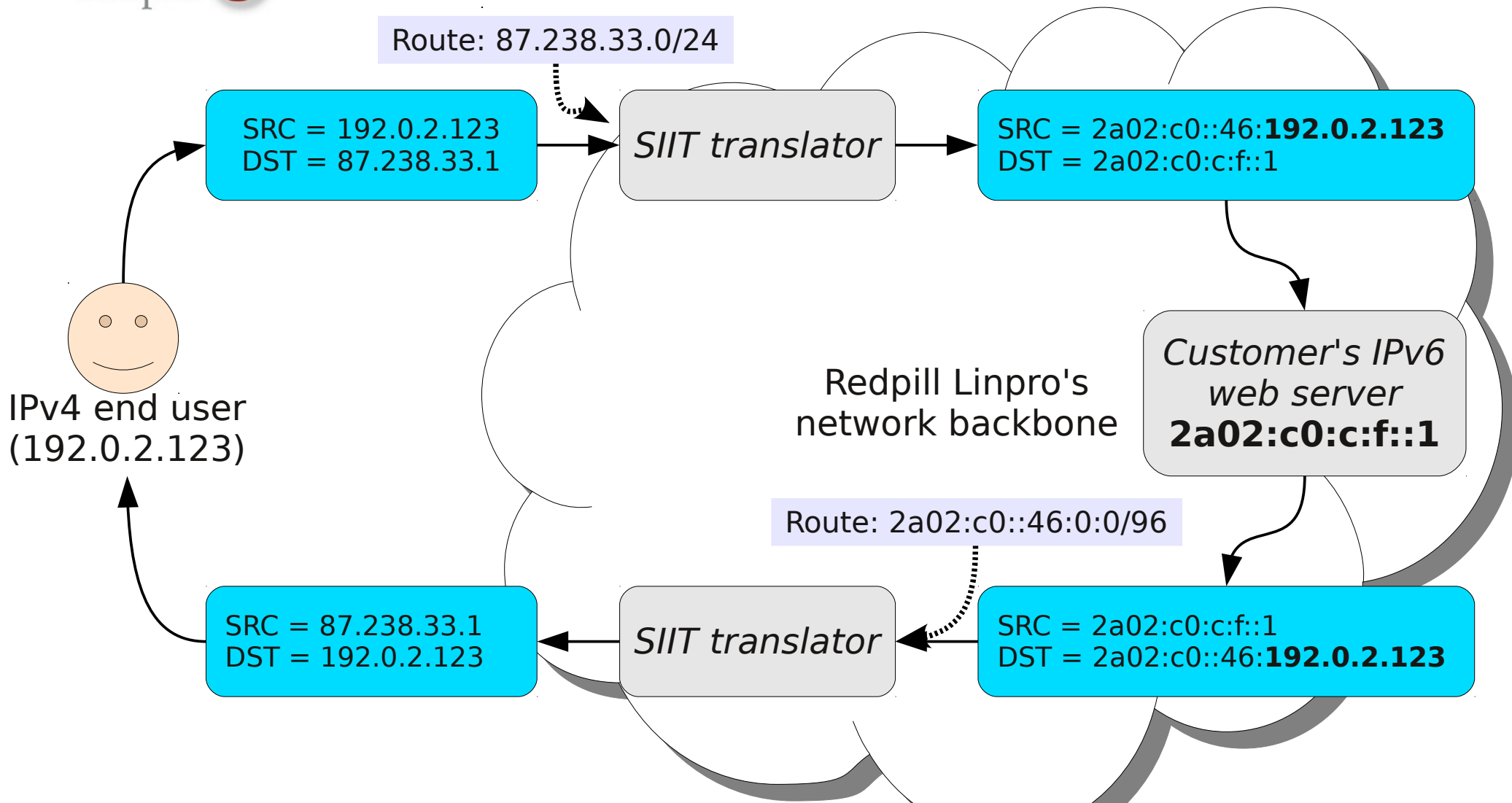


The web server responds normally to the request. It does not need to care that the packet was translated. The response is then routed to the closest translator.

DNS setup:

www.cust.no.	IN AAAA	2a02:c0:c:f::1
www.cust.no.	IN A	87.238.33.1

# SIIT explained



The first 96 bits of the IPv6 destination address is stripped to generate an IPv4 address. The IPv6 source address is swapped according to a static mapping.

DNS setup:

www.cust.no.	IN AAAA	2a02:c0:c:f::1
www.cust.no.	IN A	87.238.33.1

# Some advantages of SIIT

- Packet-by-packet stateless operation
  - Performs just like a normal IP router (fast!), no cps limitations
  - Place anywhere in the network, just route the IPv4 prefix with the service addresses and the special IPv6 translation prefix to it
  - HA and load balancing are trivial - anycast and multipath routing
- One IPv4 address per public service/customer
  - Minimal or no loss of addresses due to  $^2$  overhead
- The server/application does not need to care about the translation process, on the wire it looks exactly like a native IPv6 connection
  - However, it **can** easily recognise the special translation prefix, strip it, and perform IPv4 geolocation, logging, etc.
- No operational overhead due to running dual-stack – one service to monitor, one place to change firewall rules, and so on.

# Summary

- Preparing for IPv4 depletion is absolutely essential in order to maintain business continuity
- IPv6-only deployments plus translators (or proxies) are a likely a viable path forward
  - Requires IPv6 support by the customer's applications
  - Private, unrouted, IPv4 could be used for internal M2M communication if necessary – but adds complexity
- RFC 1918 + NAT44 (or proxies) is the only other option
  - Adds complexity and performance issues to the network design
  - If the customer also wants IPv6 connectivity, (partial) dual-stack is necessary – adds operational overhead and complexity
- Questions?