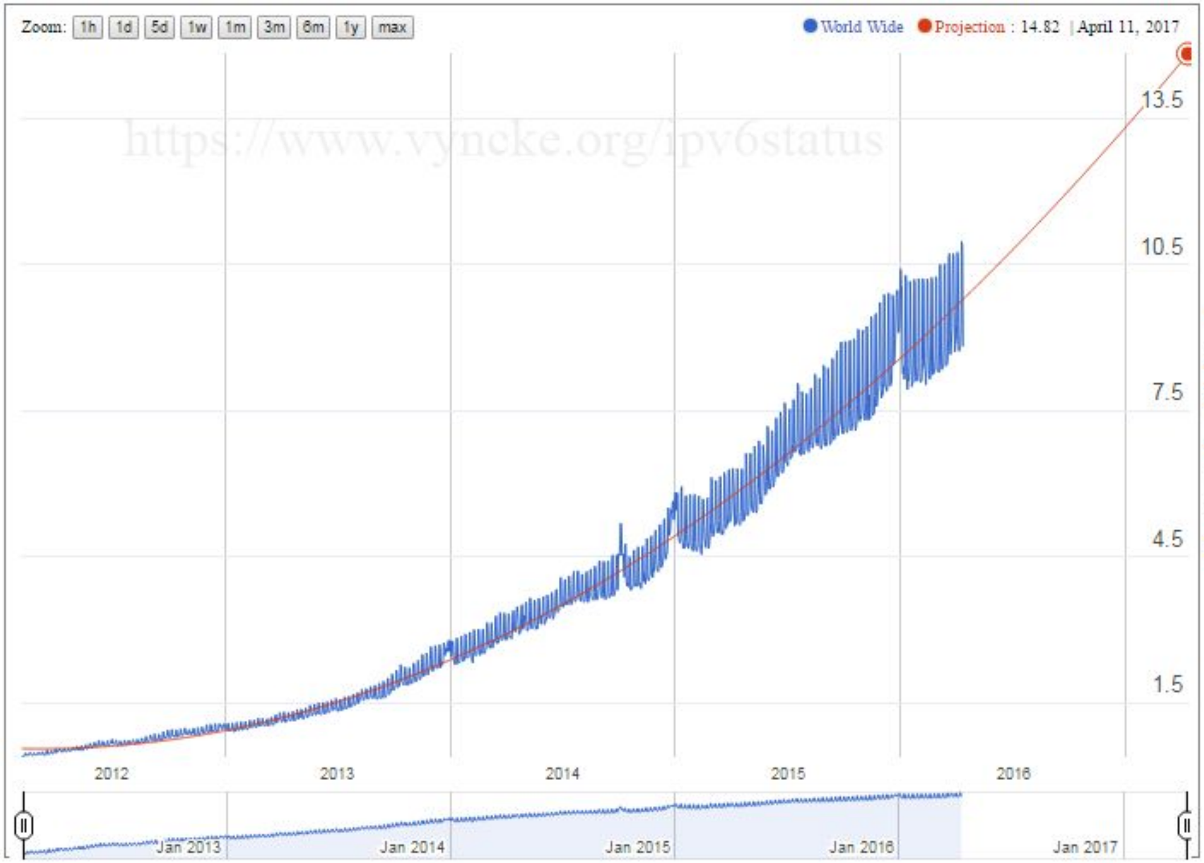
IPv6-трафик в сетях IPv4

Д.О. Ковалёв, Г.В. Хазанкин

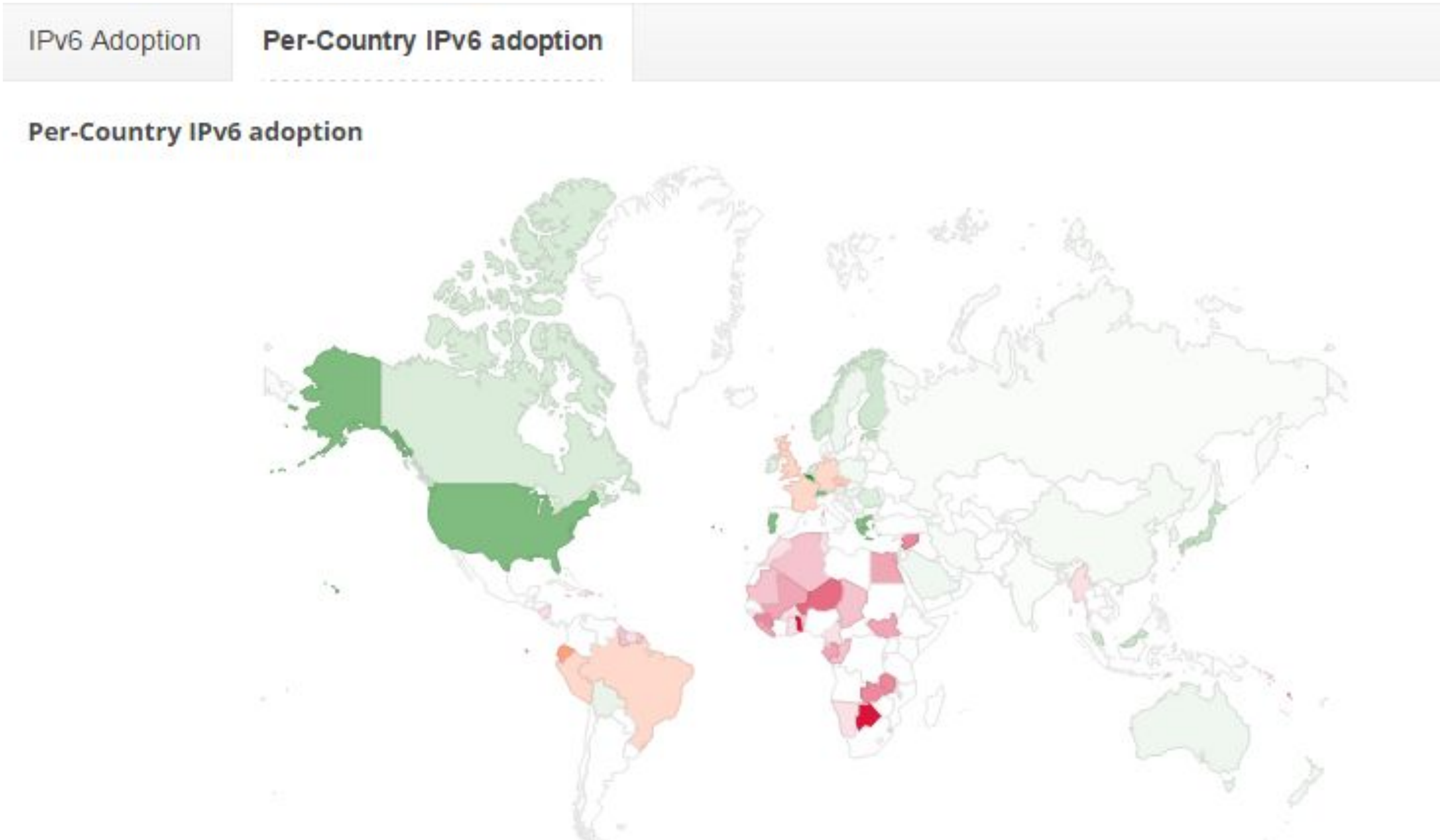
Celebrating New Year 2016 with 10% IPv6!



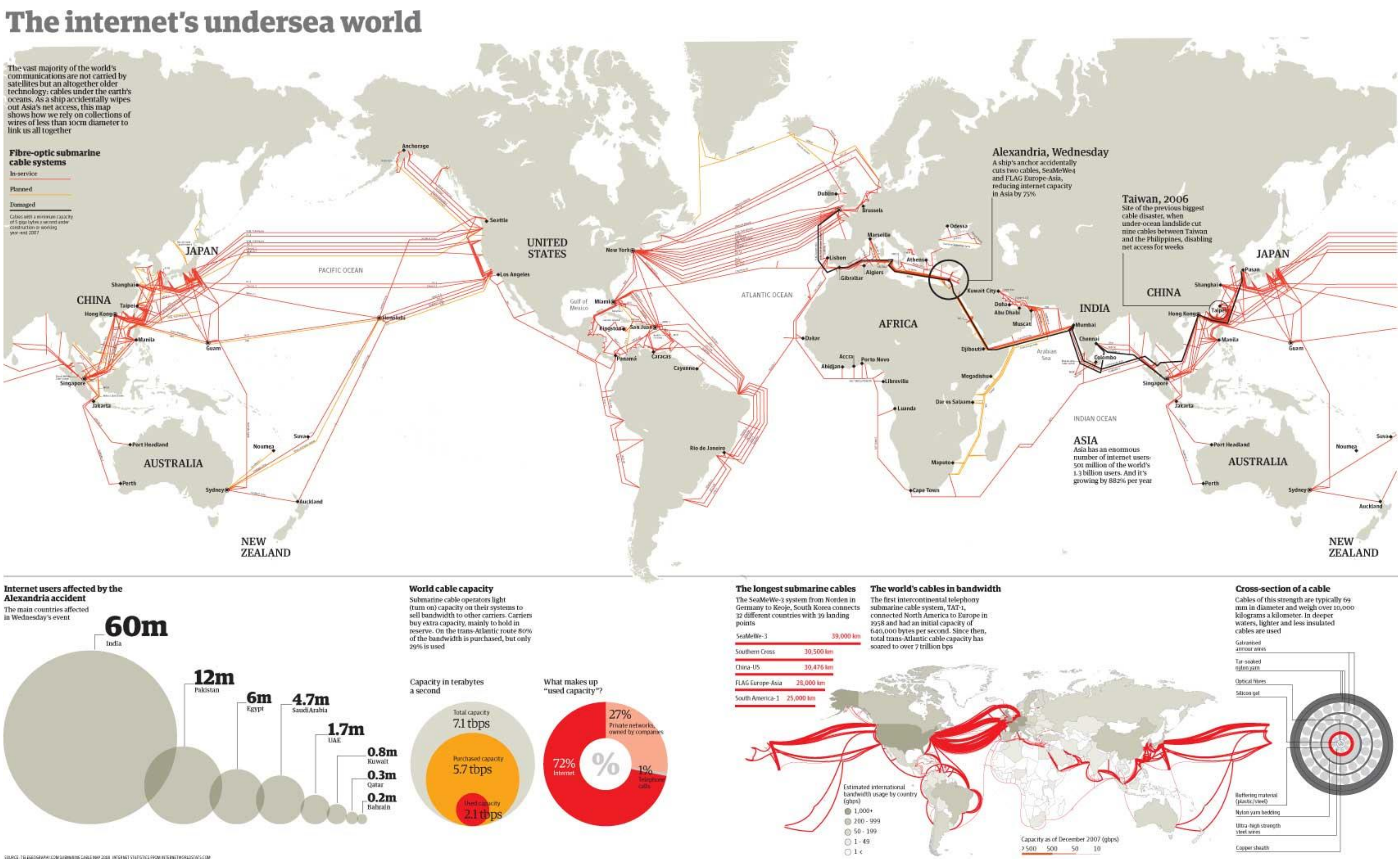
Projection of IPv6 %-age of IPv6-Enabled Web Browsers (courtesy Google) in World Wide

World Wide Projection ~15% (April 11, 2017)

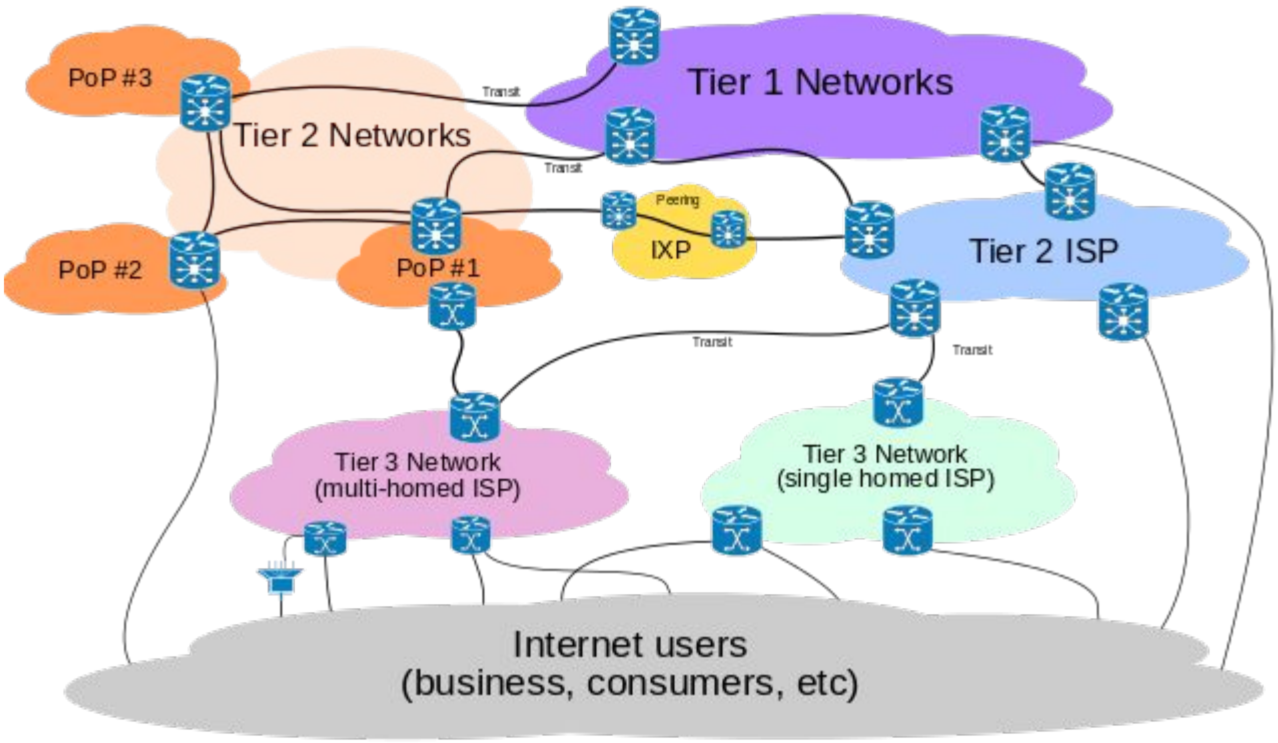
Celebrating New Year 2016 with 10% IPv6!



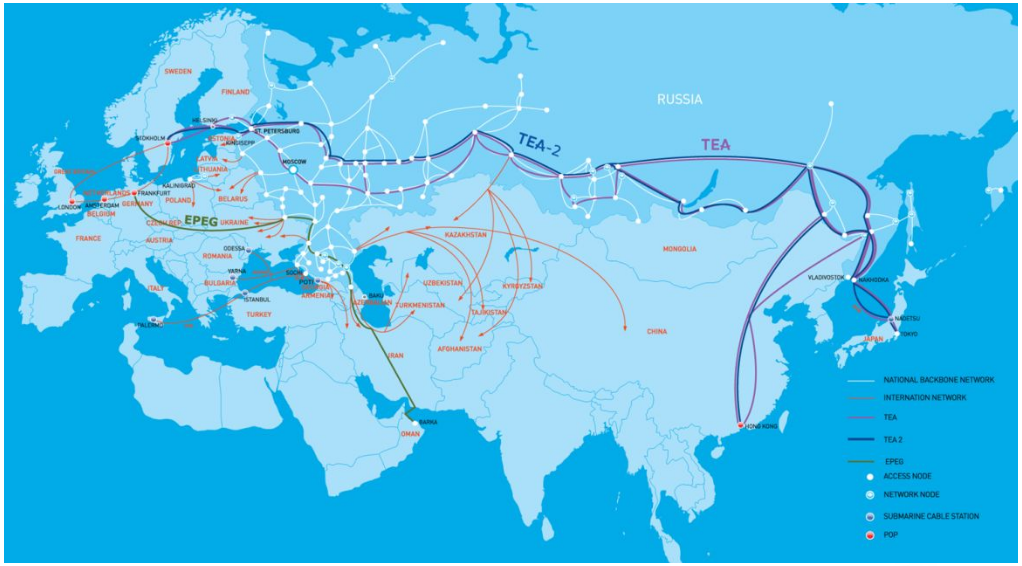
Internet Peering Ecosystem



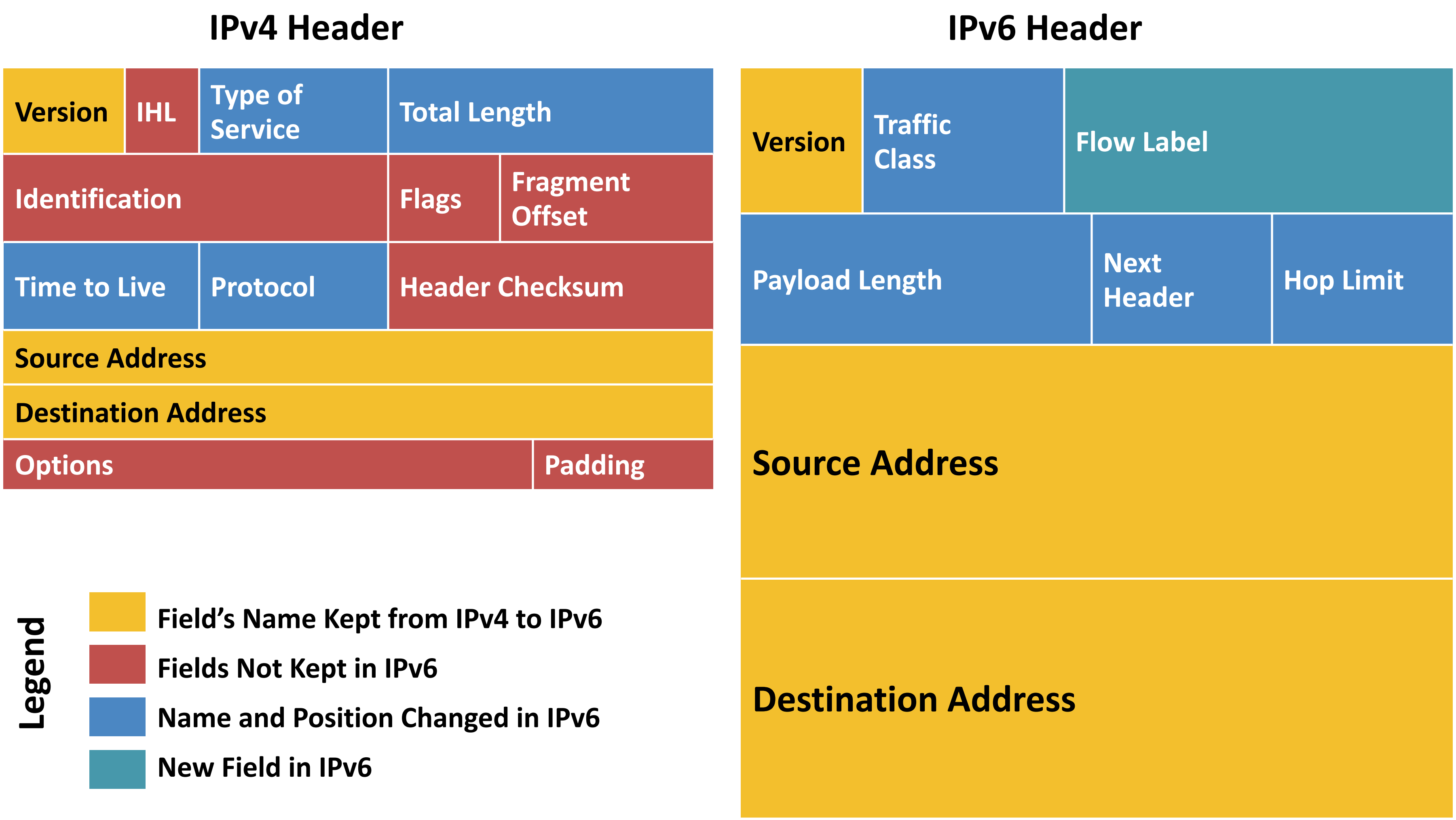
Internet Peering Ecosystem



Runet Peering Ecosystem



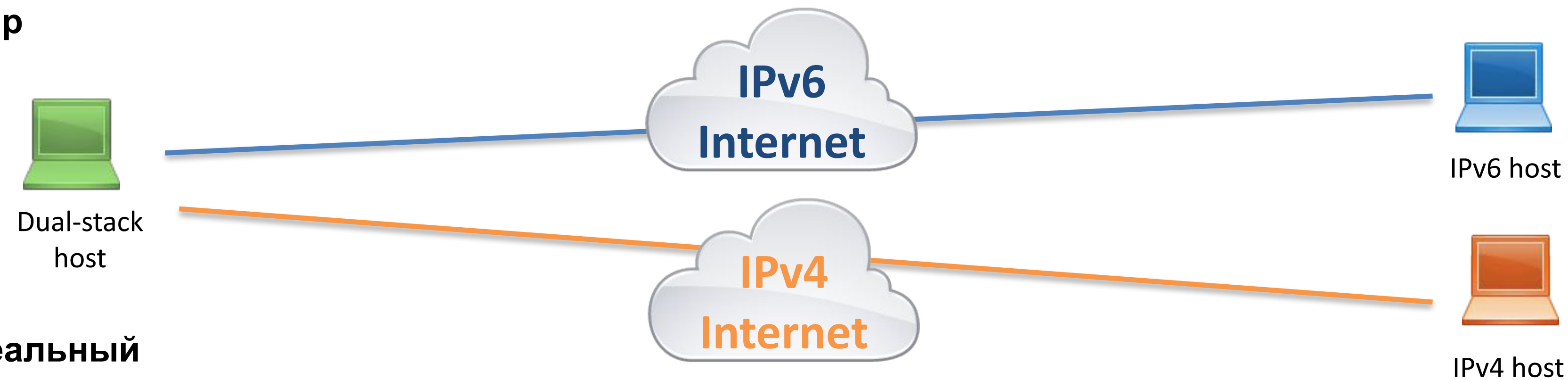
Каким образом IPv6-трафик присутствует в IPv4-



Каким образом IPv6-трафик присутствует в IPv4-

Dual-stack

Идеальный мир



Реальный мир

Еще более реальный мир

Каким образом IPv6-трафик присутствует в IPv4-

Redefining Flow Label in IPv6 and MPLS Headers for End to End QoS in Virtual Networking for Thin Client

Mohammad Aazam[1], Adeel M. Syed[2], Eui-Nam Huh[3]
Innovative Cloud and Security Lab, Department of Computer Engineering
Kyung Hee University, Suwon, South Korea.
[2]Bahria University, Islamabad, Pakistan
[1]aazam@ieee.org
[2]adeelmuzaffar@gmail.com
[3]johnhuh@khu.ac.kr

***Abstract*— Applications that require timeliness such as; video conferencing, Voice over IP (VoIP), Video-on-Demand (VoD) etc., require quantified and well managed Quality of Service (QoS). For this, the flow label field in IPv6 header and the Label field in MPLS header should be used for efficient QoS provisioning. In this paper, flow label specifications were first investigated and then a new structure is proposed for QoS provisioning. The 20-bit field in IPv6 as well as MPLS is meant to provide QoS, other than labeling or identification purpose. But its usage is not standardized and defined that how these 20-bits must effectively be used to provide maximum possible QoS in a better way. This paper discusses it by proposing portions for QoS by explicitly specifying bandwidth, delay, and packet loss. By keeping the QoS parameters open in this way, it would be easy for a flow to decide what to reserve and how much to reserve. After that, mapping of flow label with class fields of IPv6 and MPLS is presented. In virtual networks, specially for Thin client, service quality degradation is a major issue, when IPv4-IPv6 virtual networks co-exist. This is also where this redefinition can come very handy.**

Keywords*—Flow label; IPv6; MPLS; QoS; virtual networking; Thin client

is considered to be the best available and most efficient combination of protocols on layer 2 and layer 3 for routing of packets. So, the same specification of label field can be used with both these protocols to allow the stability in QoS for a particular flow, throughout the communication, from MAC to the network layer. A 'flow' can be defined as a sequence of packets sent from a particular source to a particular (unicast or multicast) destination for which the source desires special handling for QoS by the intermediate routers [8]. The flow is uniquely identified within the network by the flow label and its source address. All packets belonging to one flow should be treated within the network in the same way. The usage of the flow label is not completely specified within the RFCs [8] and [9]. However the flow label field can be defined as a 20-bit field that may be used by a source to label sequences of packets for which it requests special handling by the routers, such as non-default quality of service or real-time service [8]. During our studies [15, 16], it was realized that in virtual networking environment, when IPv4 and IPv6 coexist, it incurs some reasonable amount of signaling overhead, due to which, service quality degrades. Since, redefinition of flow label for end to end QoS is the main concern of this paper and

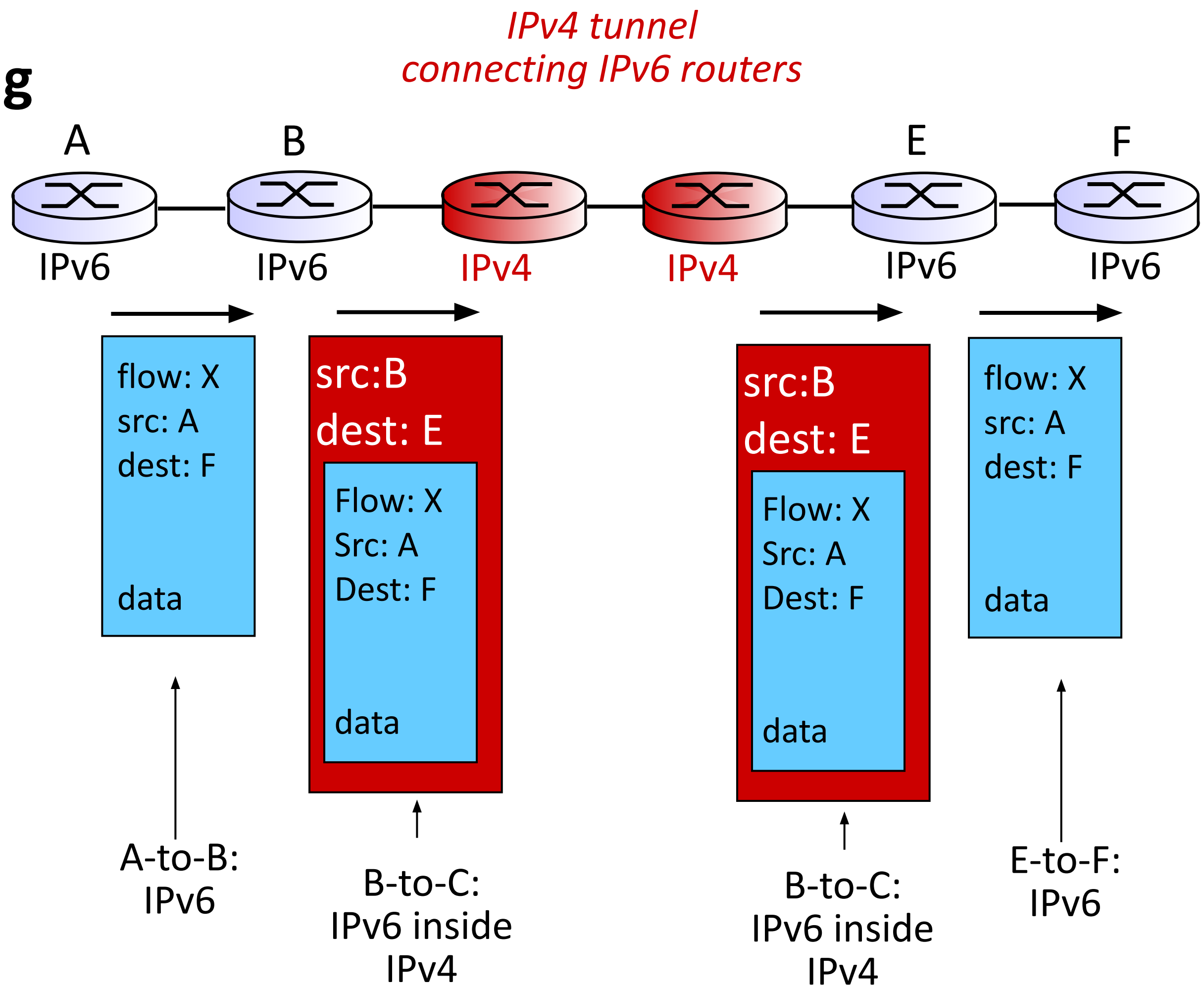
Многообразие механизмов

Dual-stack	Tunneling	Translation
Dual-stack	**6in4**	**NAT64**
Dual-stack lite	**6to4**	
	6a44	
	GRE tunneling	
	ISATAP	
	Teredo	
	6rd	
	LISP	
	IPv6 over MPLS	
	и другие	

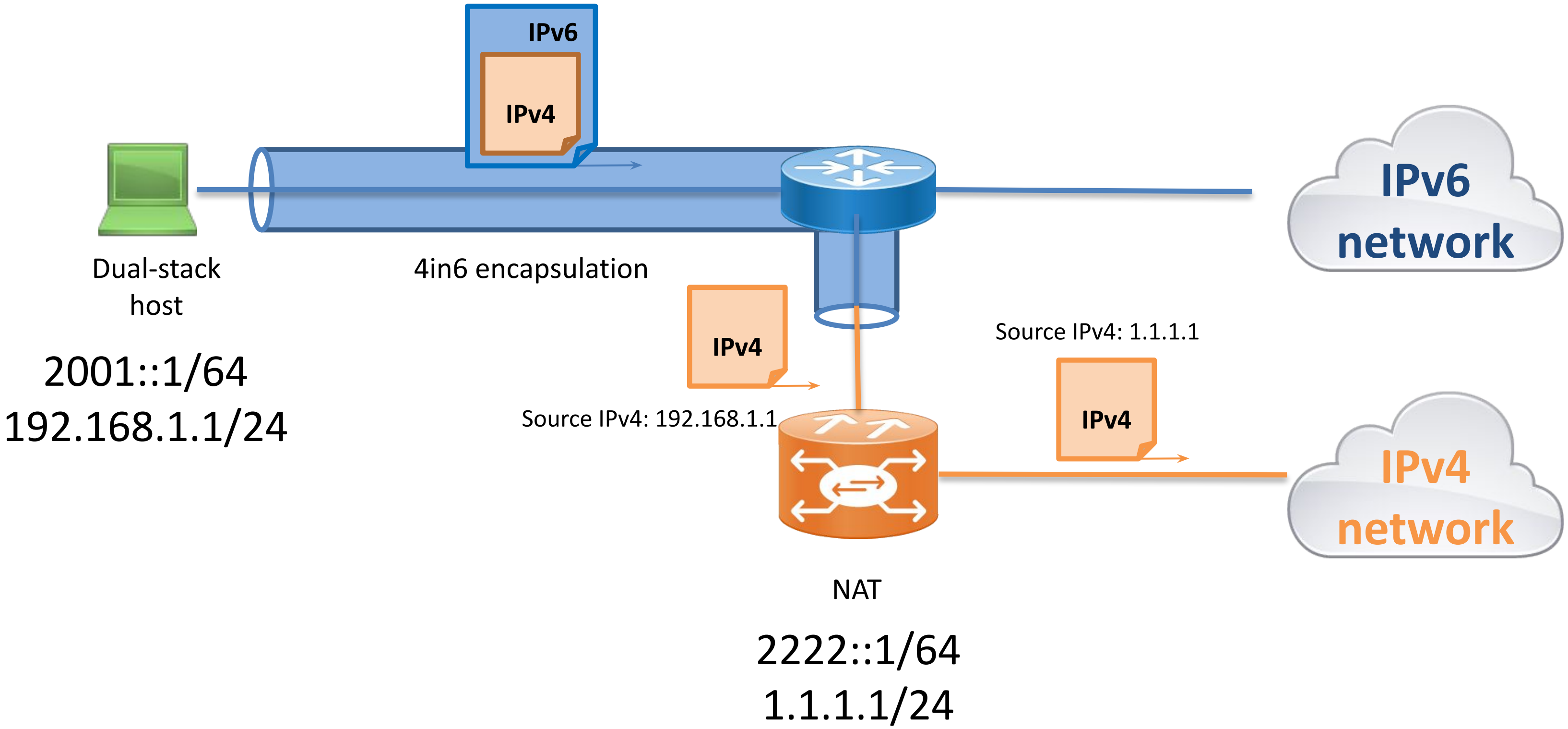
Некоторые из этих механизмов инкапсулируют IPv6-пакеты внутрь IPv4-пакетов.
Есть желание посмотреть, что на самом деле передается внутри IPv4-пакета (вложенный внутрь IPv6-пакет – тоже часть IPv6-трафика, который необходимо учитывать при анализе). Каким образом может быть инкапсулирован IPv6 в IPv4?

Каким образом IPv6-трафик присутствует в IPv4-

Tunneling



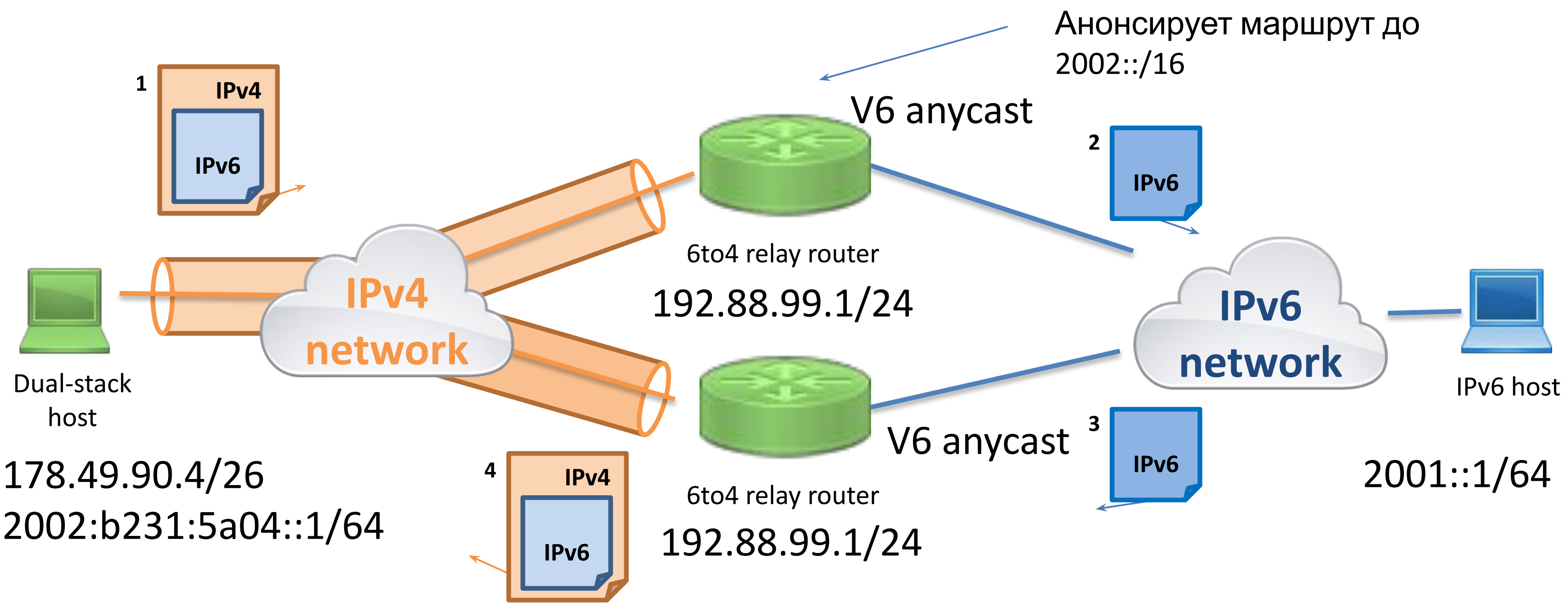
Обзор Dual-stack lite



Заполняется таблица трансляции на NAT:

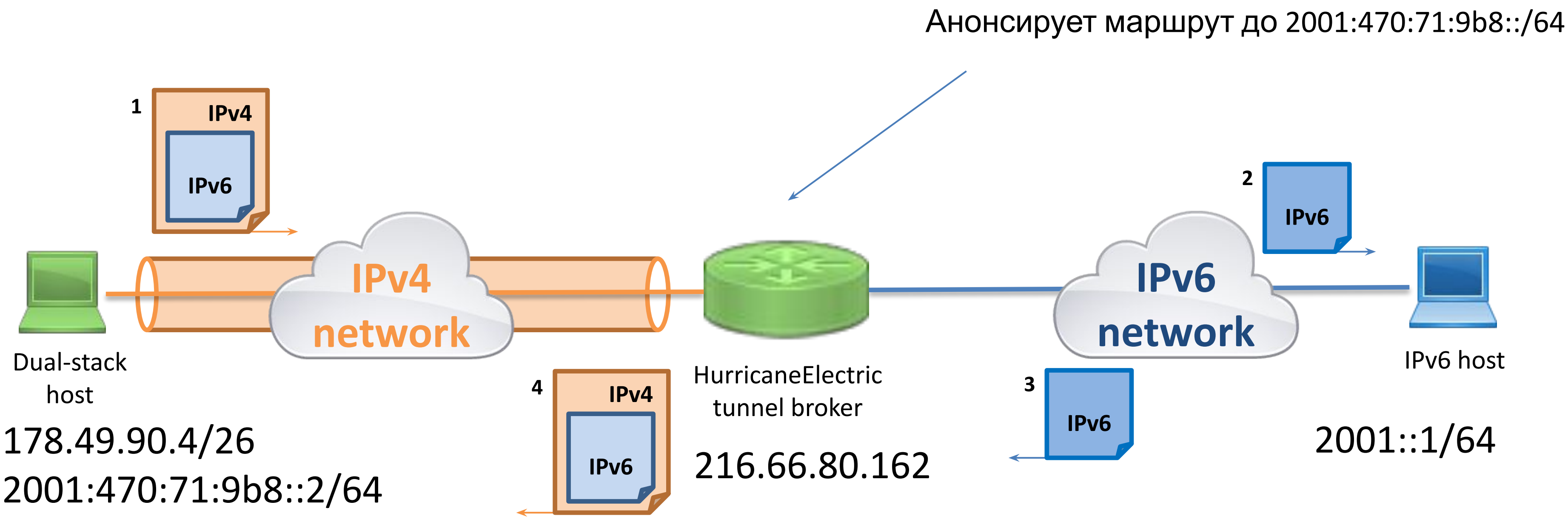
192.168.1.1:XXXX == 1.1.1.1:YYYY (классический NAT44)
IPv4-пакет с IP 192.168.1.1 пришел от 2001::1

Обзор 6to4



1	IPv4 HDR	Src: 178.49.90.4	Dst: 192.88.99.1	IPv6 HDR	Src: 2002:b231:5a04::1	Dst: 2001::1	Payload
2				IPv6 HDR	Src: 2002:b231:5a04::1	Dst: 2001::1	Payload
3				IPv6 HDR	Src: 2001::1	Dst: 2002:b231:5a04::1	Payload
4	IPv4 HDR	Src: 192.88.99.1	Dst: 178.49.90.4	IPv6 HDR	Src: 2001::1	Dst: 2002:b231:5a04::1	Payload

Обзор



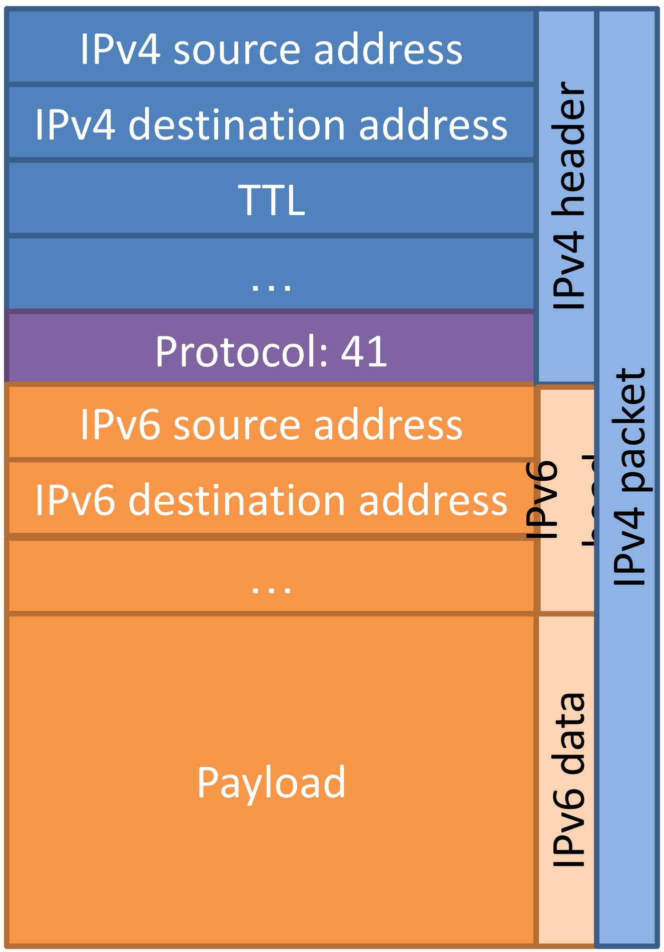
1	IPv4 HDR	Src: 178.49.90.4	Dst: 216.66.80.162	IPv6 HDR	Src: 2001:470:71:9b8::2	Dst: 2001::1	Payload
2				IPv6 HDR	Src: 2001:470:71:9b8::2	Dst: 2001::1	Payload
3				IPv6 HDR	Src: 2001::1	Dst: 2001:470:71:9b8::2	Payload
4	IPv4 HDR	Src: 216.66.80.162	Dst: 178.49.90.4	IPv6 HDR	Src: 2001::1	Dst: 2001:470:71:9b8::2	Payload

Способы инкапсуляции IPv6 в

Protocol 41

6to4, 6in4, ISATAP

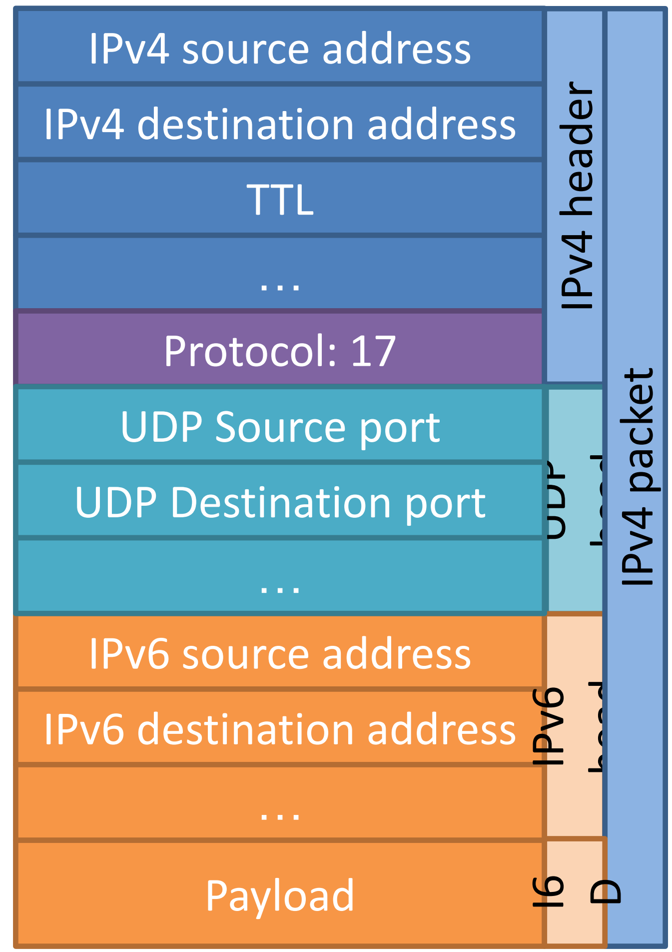
Easily filtered



UDP

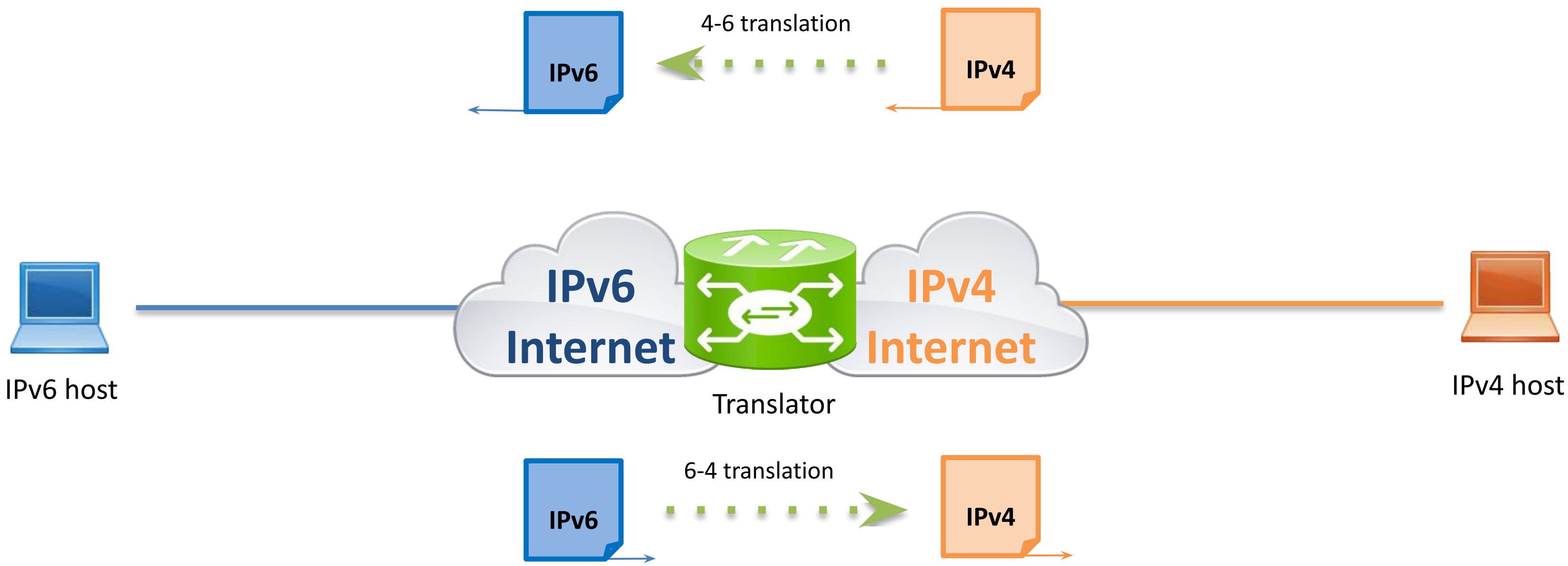
Teredo, 6a44

Less susceptible to filtering

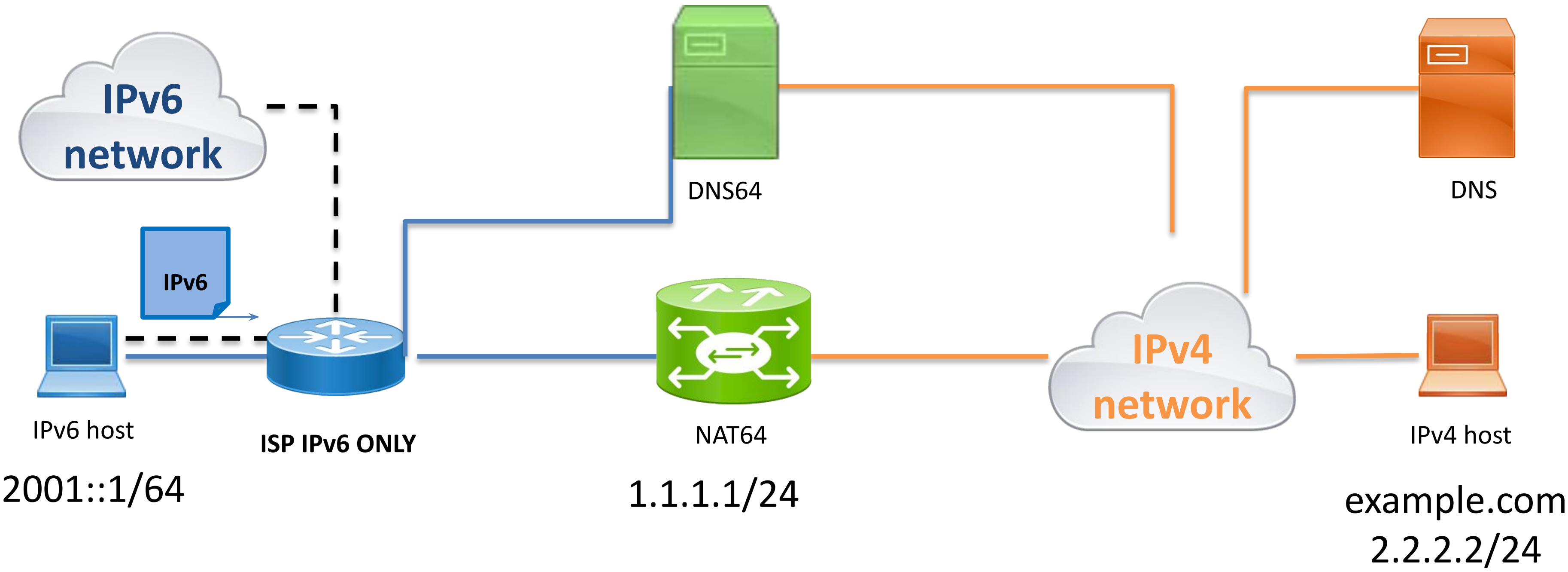


Каким образом IPv6-трафик присутствует в IPv4-

Translation



Обзор



1

DNS64 ask IPv4 DNS:
example.com A: 2.2.2.2
example.com AAAA: none

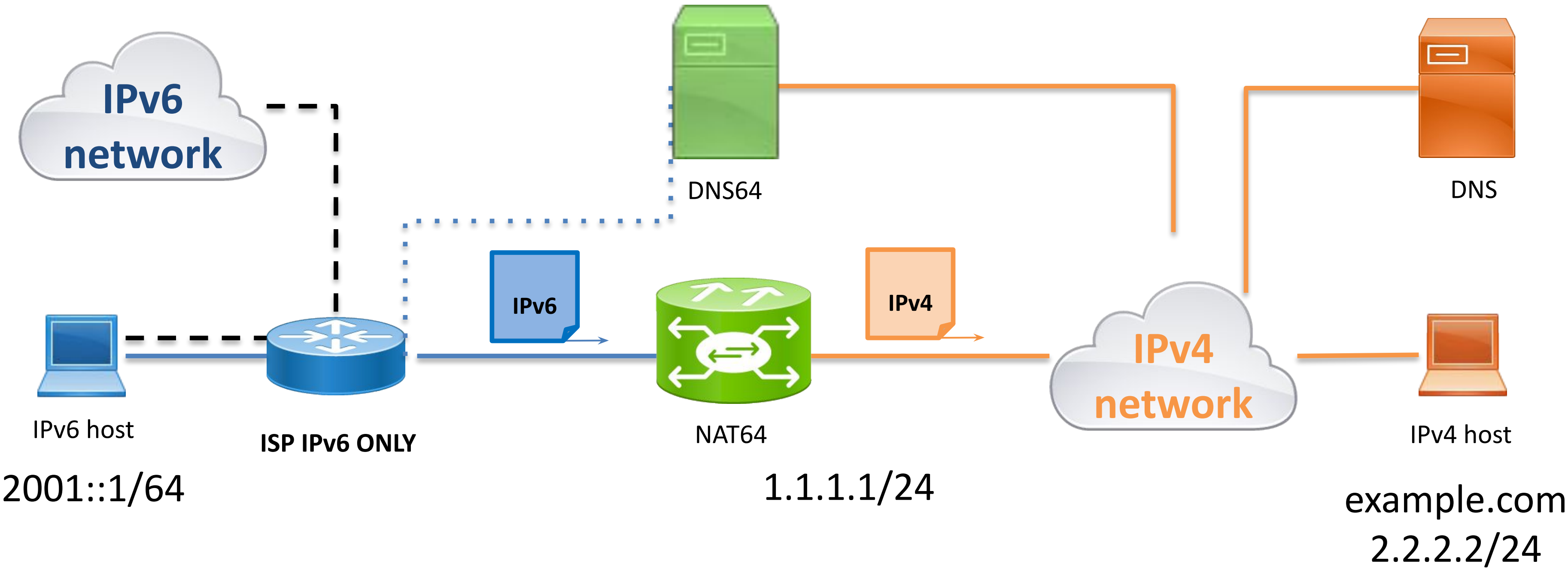
DNS64 answer IPv6 host:
example.com A: 2.2.2.2
example.com AAAA: 64:ff9b::**2:2:2:2**

2

IPv6 host 2001::1 отправляет пакет для example.com:

IPv6 HDR	Src: 2001::1	Dst: 64:ff9b::2:2:2:2	Payload

Обзор



3

Заполняется таблица трансляции на NAT64:

[2001::1]:8080 == 1.1.1.1:8080

4

Согласно RFC6145 заголовок IPv6 конвертируется в IPv4. Пакет IPv6 превращается в IPv4 и высылается через IPv4 интерфейс NAT64:

IPv4 HDR	Src: 1.1.1.1	Dst: 2.2.2.2	Payload

Обзор

Monitoring of Tunneled IPv6 Traffic
Using Packet Decapsulation and IPFIX

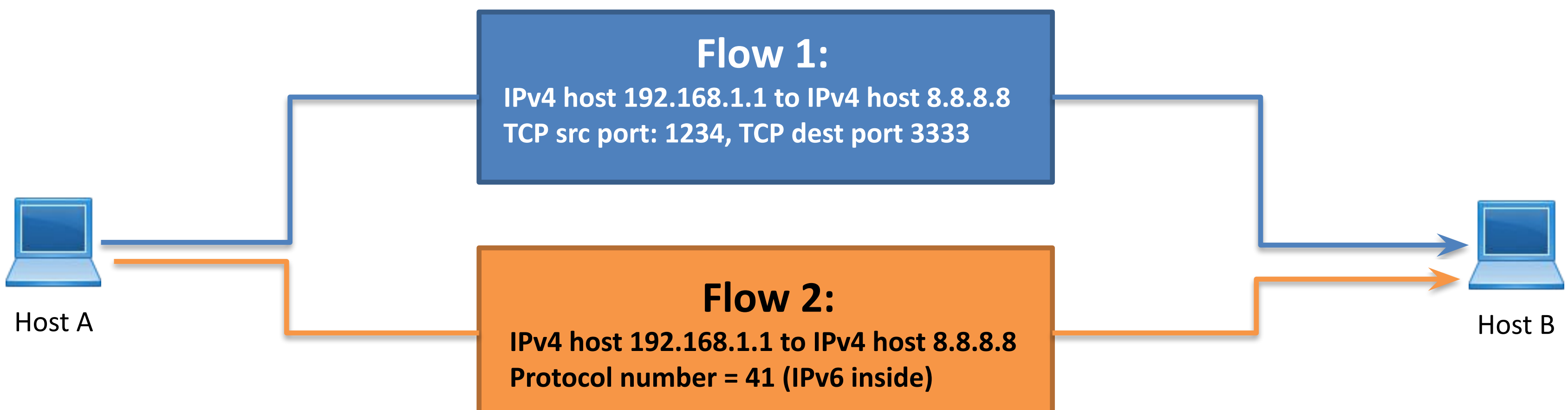
By Martin Elich, Matej Gregr and Pavel Celeda

Кратко о

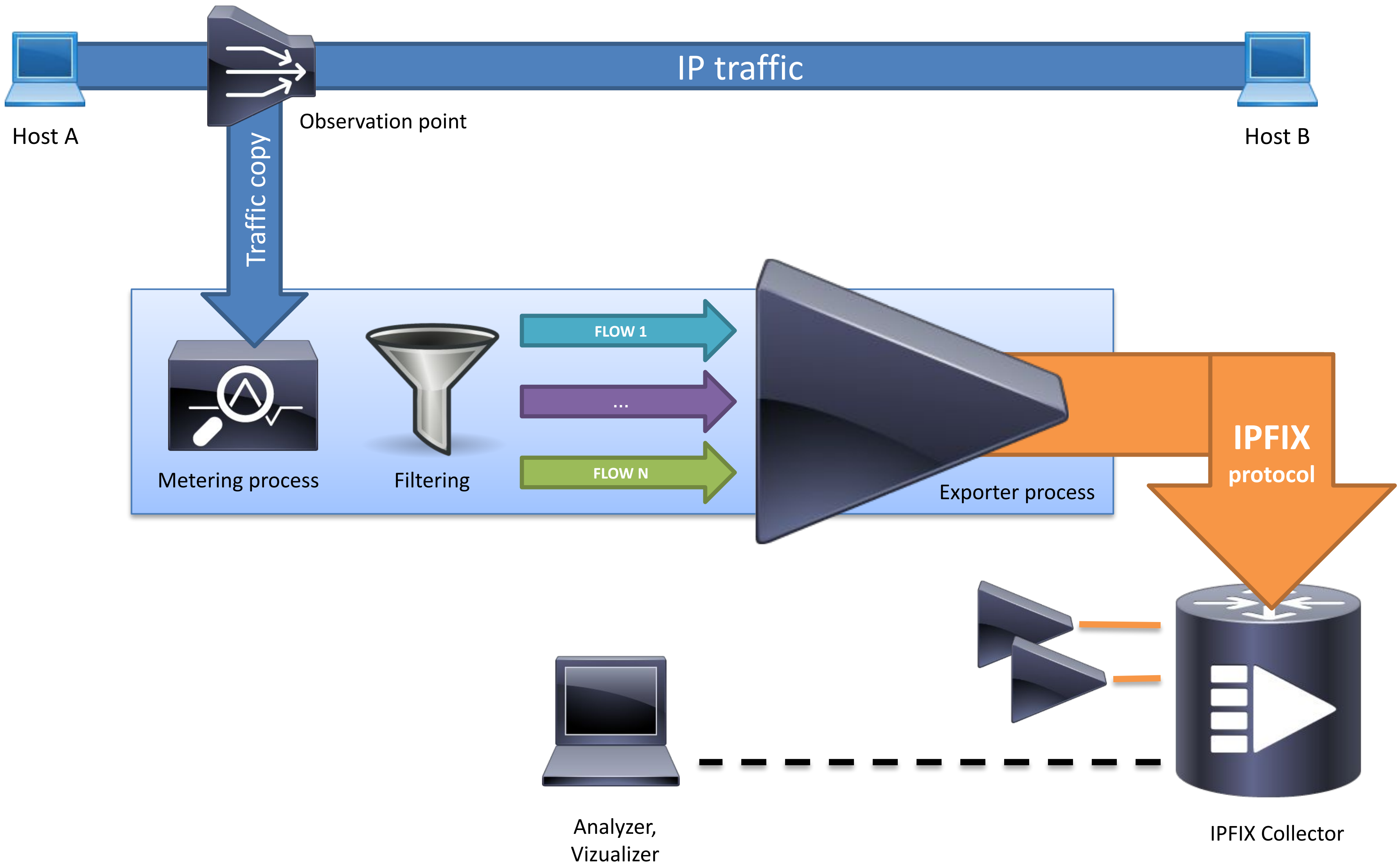
Internet **P**rotocol **F**low **I**nformation E**x**port

Протокол IETF, определяющий стандарт экспорта информации о **потоках данных** с различных сетевых устройств.
Основан на протоколе Cisco NetFlow v9.

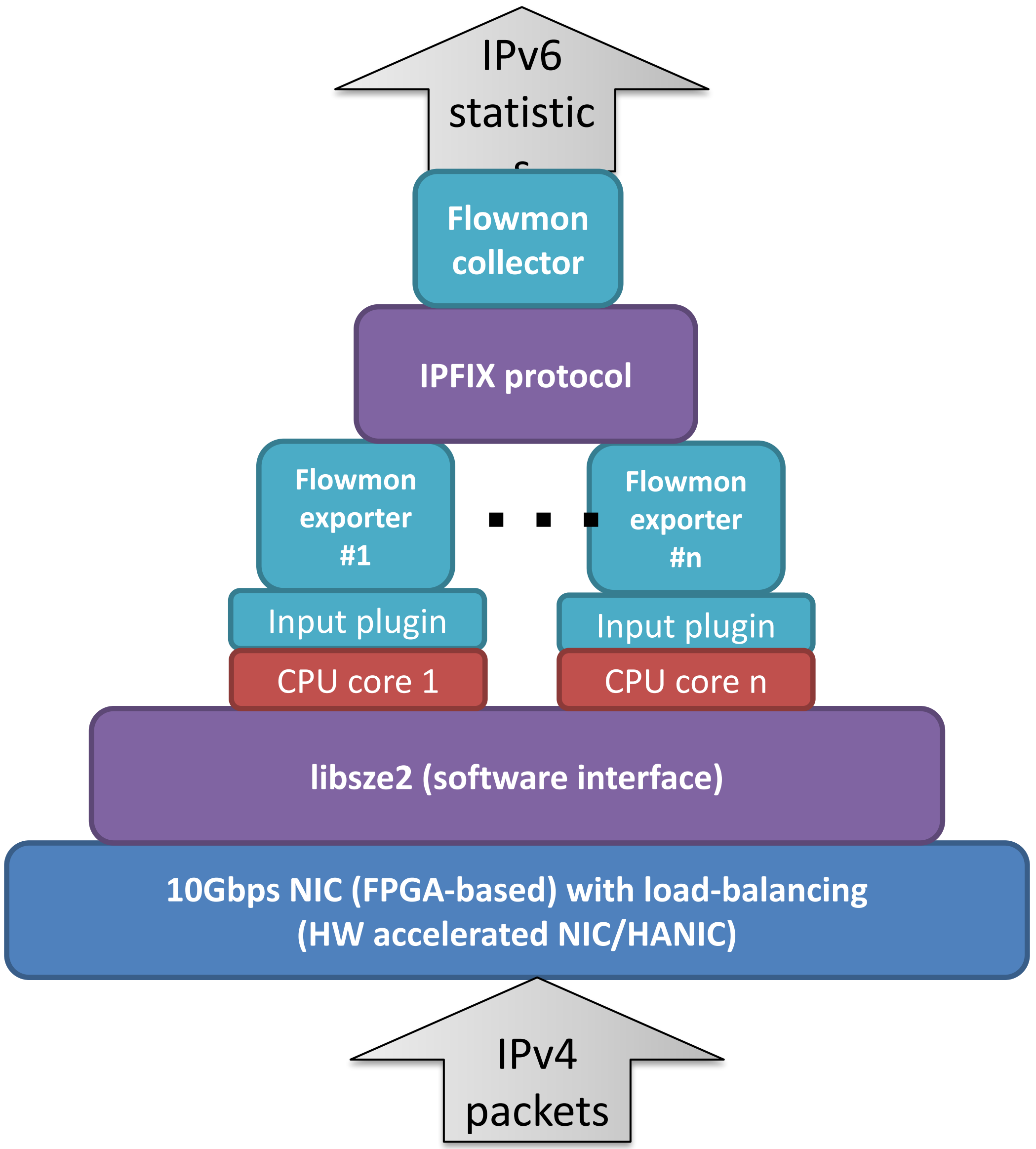
Информация о потоках данных может использоваться в таких задачах, как:
- Мониторинг трафика, учет использования сетевых ресурсов
- Анализ трафика
- Обнаружение/предотвращение вторжений



Кратко о


IP traffic
Host A
Observation point
Host B
Traffic copy
FLOW 1
...
FLOW N
Metering process
Filtering
Exporter process
IPFIX
protocol
Analyzer,
Vizualizer
IPFIX Collector

Суть



Получение статистики по IPv6-трафику, инкапсулированному в IPv4 (разбор некоторых туннелей)

Сборщик IPFIX (в данном случае Flowmon) собирает итоговую статистику по потокам со всех экспортеров и презентует ее пользователю в том или ином виде

Информация о потоке передается сборщику при помощи протокола IPFIX.

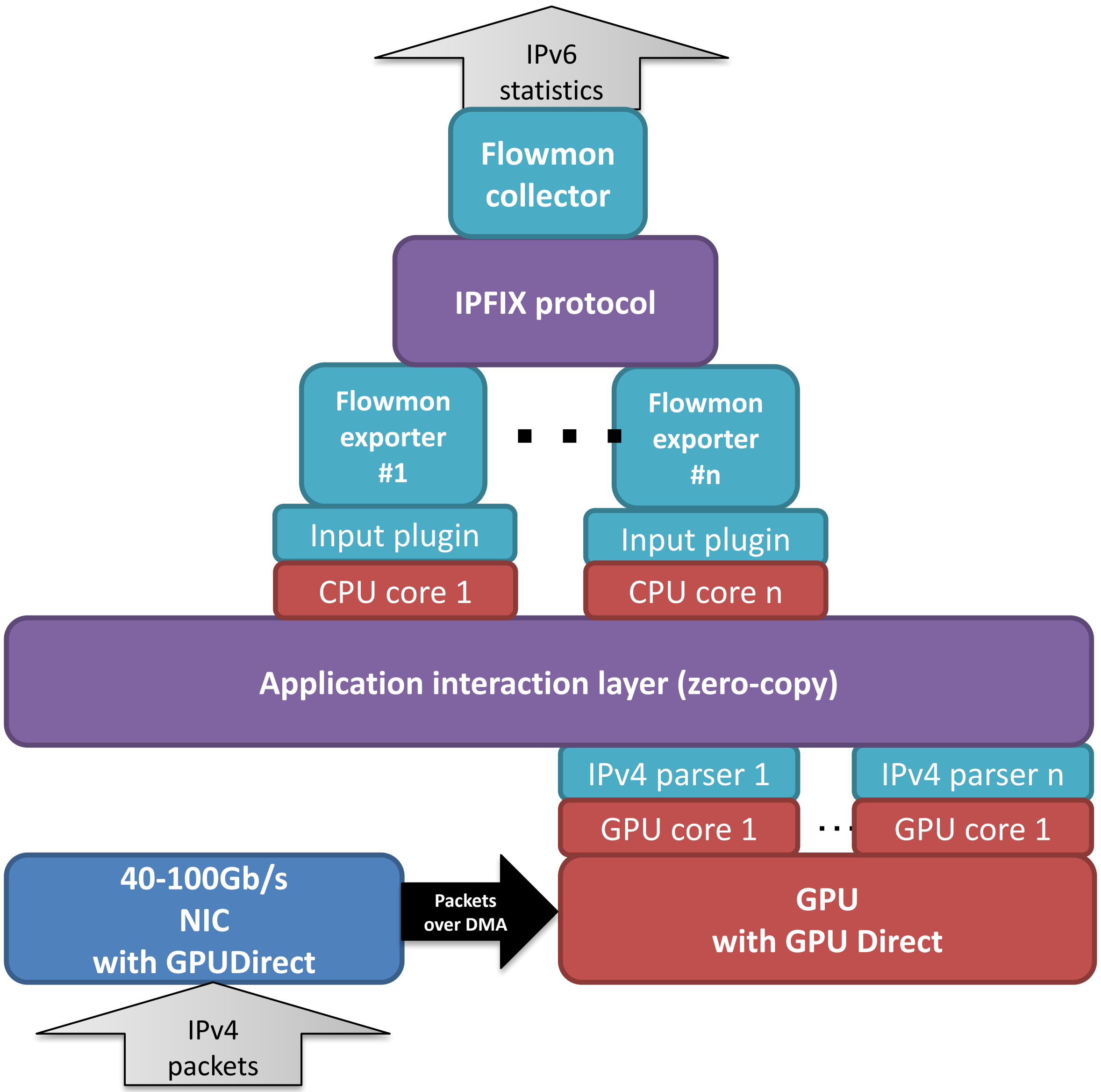
При обнаружении IPv6-пакета внутри IPv4, IPv6-пакет декапсулируется из IPv4, на основании информации в заголовке IPv6 пакет сопоставляется определенному потоку.

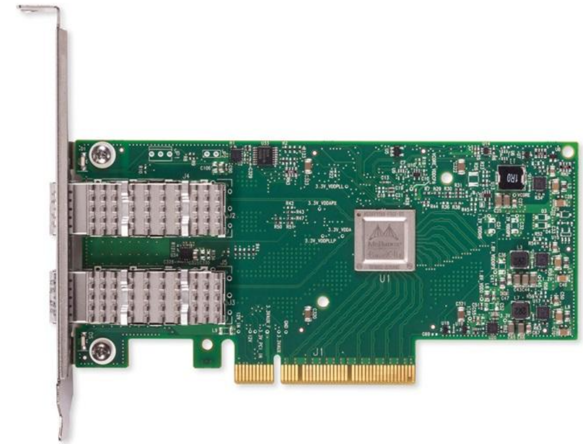
Пакеты распределяются по обработчикам
На каждом ядре сидит экспортер с самописным плагином для обнаружения IPv6 в v4
(реализовано обнаружение Teredo, ISATAP, 6to4)

В сети практически нет информации по библиотеке
Скорее всего, с ее помощью делается zero-copy

Для каждого принятого пакета ставит timestamp
Некоторым образом получает хэш заголовка пакета
Далее согласно хэшу пакеты распределяются по обработчикам

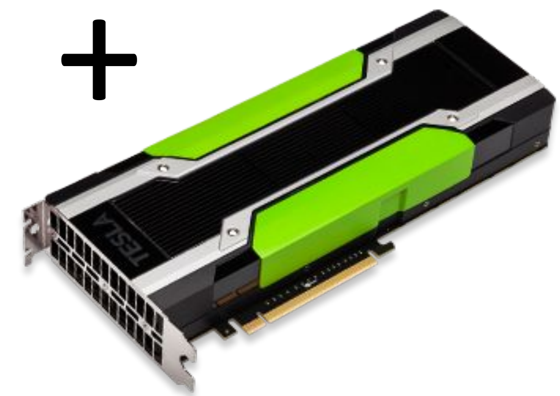
Варианты новой высокопроизводительной





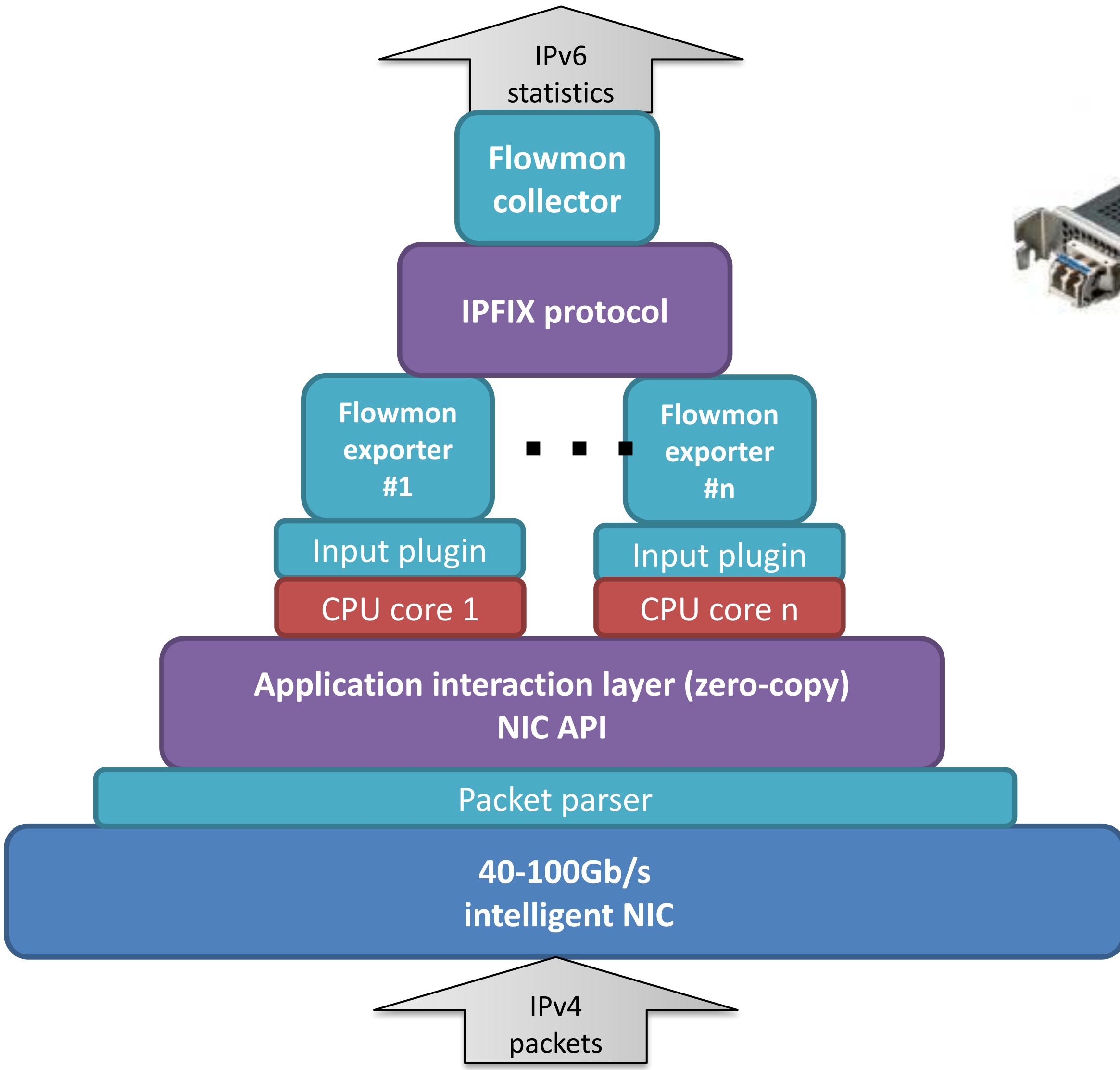
Mellanox ConnectX-4 LX

+

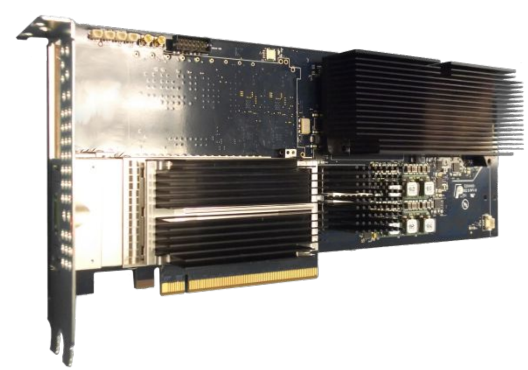


Nvidia Tesla Kepler GPU

Варианты новой высокопроизводительной



Napatech NT100-E3-1-PTP



Accolade ANIC-100K

Основные

- RFC 7059 - A Comparison of IPv6-over-IPv4 Tunnel Mechanisms
- RFC 4213 - Basic Transition Mechanisms for IPv6 Hosts and Routers
- RFC 6145 - IP/ICMP Translation Algorithm
- RFC 6146 - Stateful NAT64
- RFC6147 - DNS64

Спасибо за внимание