

Sibling Prefixes: Identifying Similarities in IPv4 and IPv6 Prefixes

Fariba Osali, K. Zubair Sediqi and Oliver Gasser

{fosali, zsediqi}@mpi-inf.mpg.de
oliver@ipinfo.io

ACM Internet Measurement Conference (IMC) 2025
Madison, Wisconsin, USA



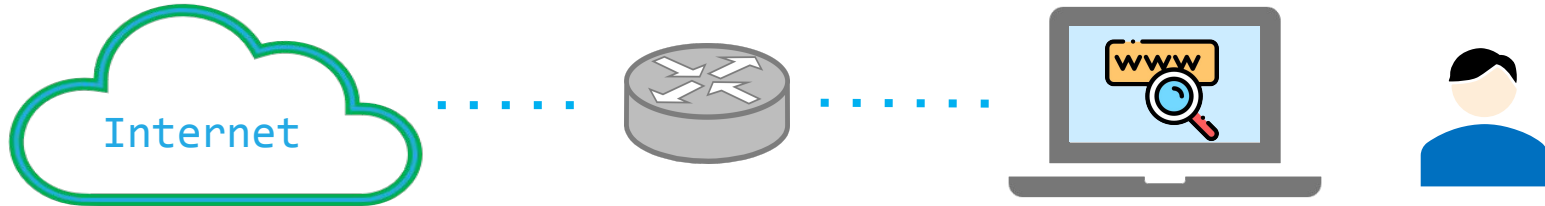
UNIVERSITÄT
DES
SAARLANDES



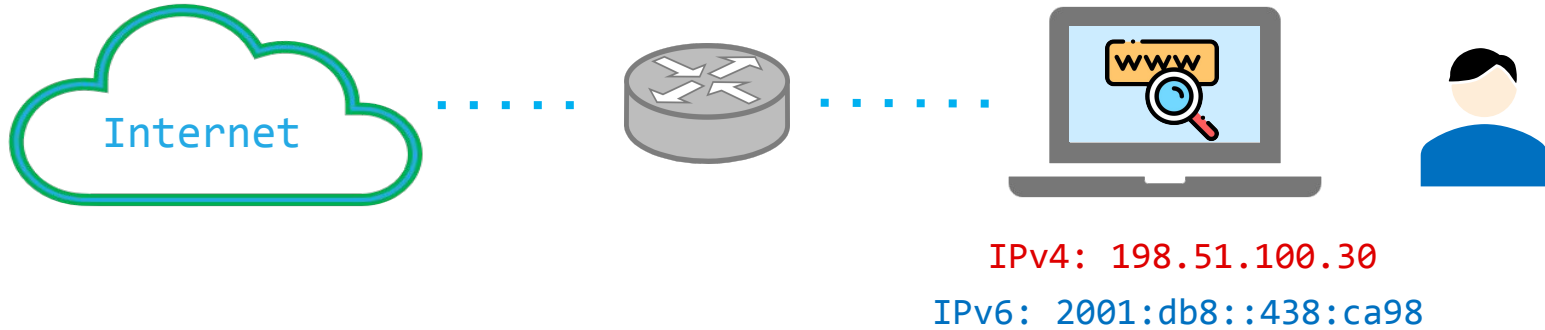
MAX PLANCK INSTITUTE
FOR INFORMATICS



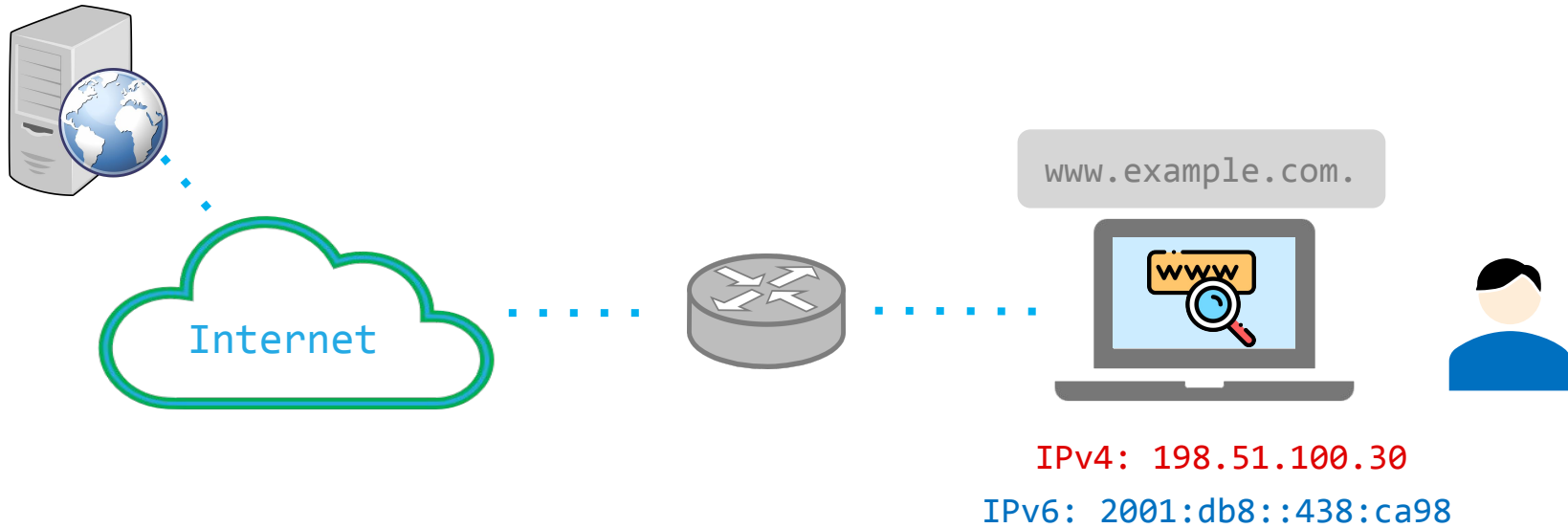
Introduction



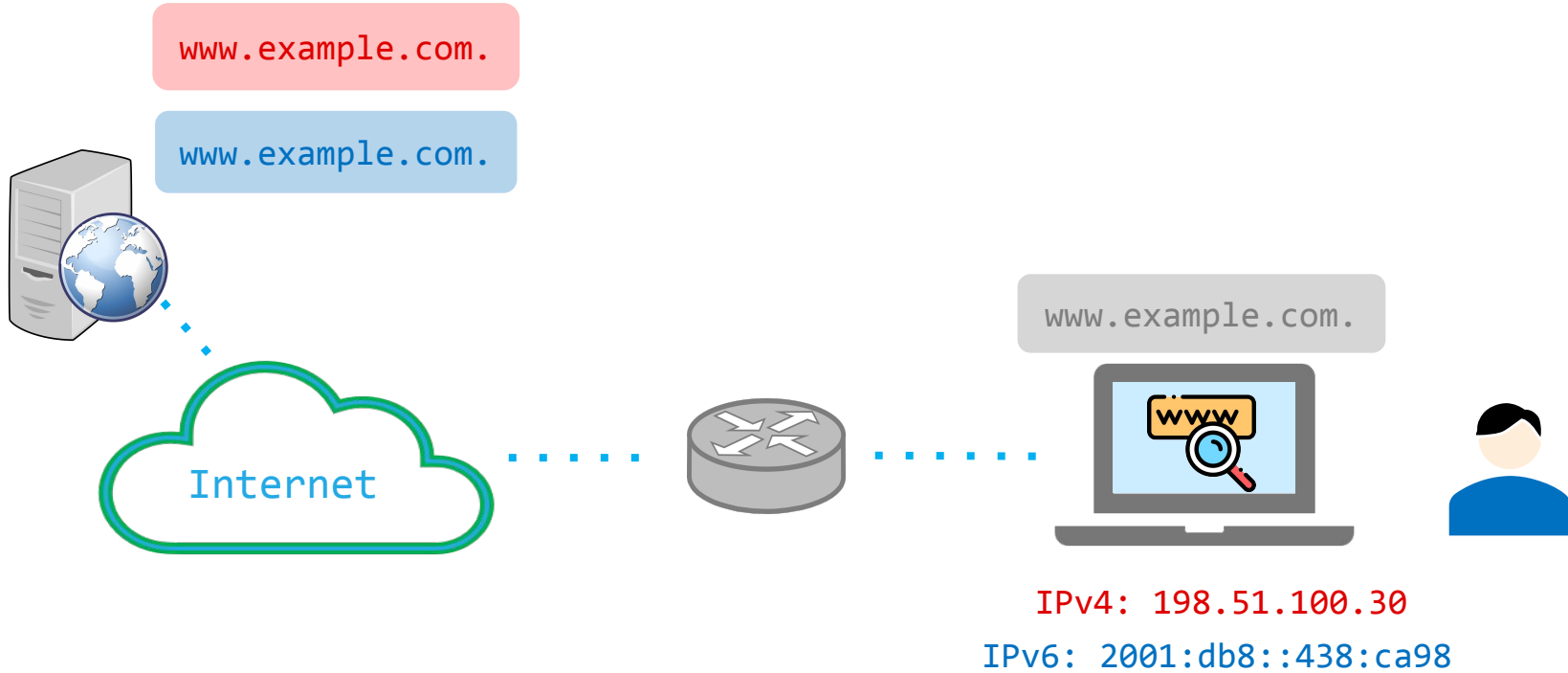
Introduction



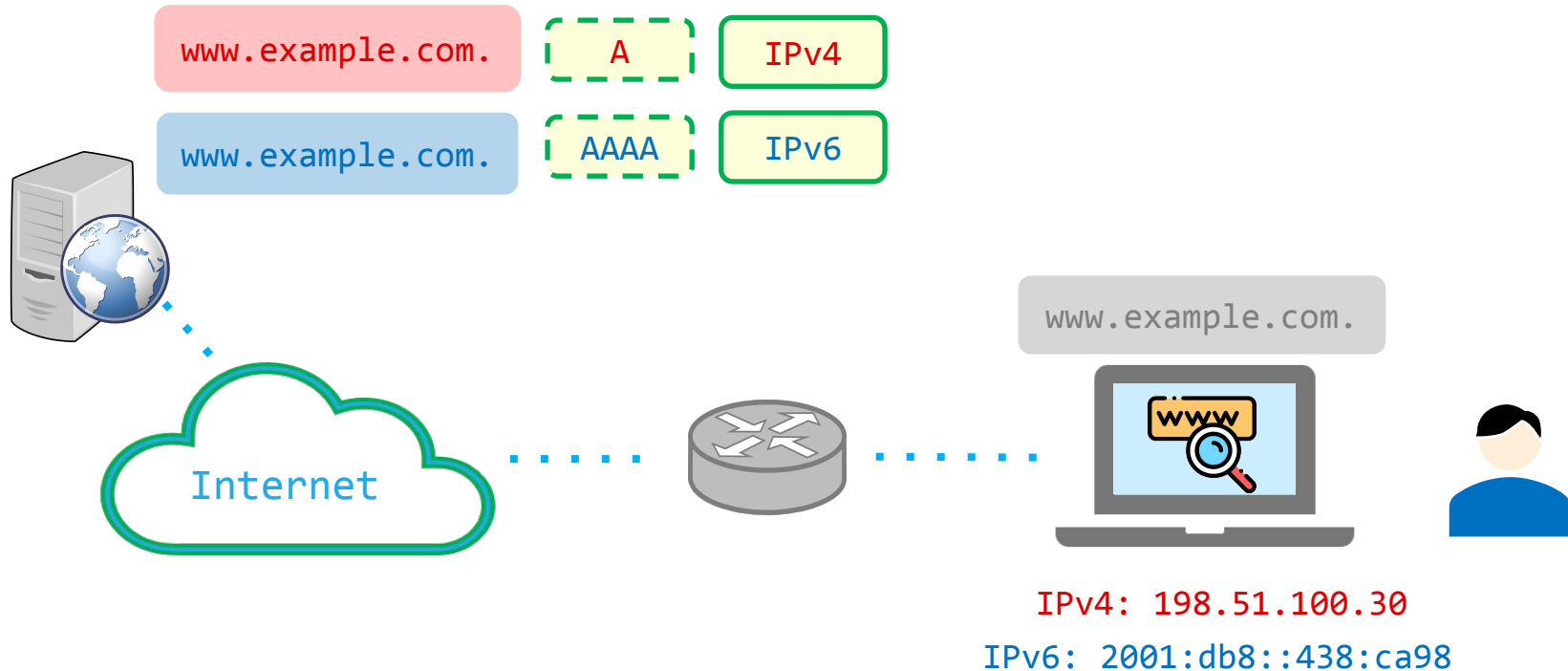
Introduction



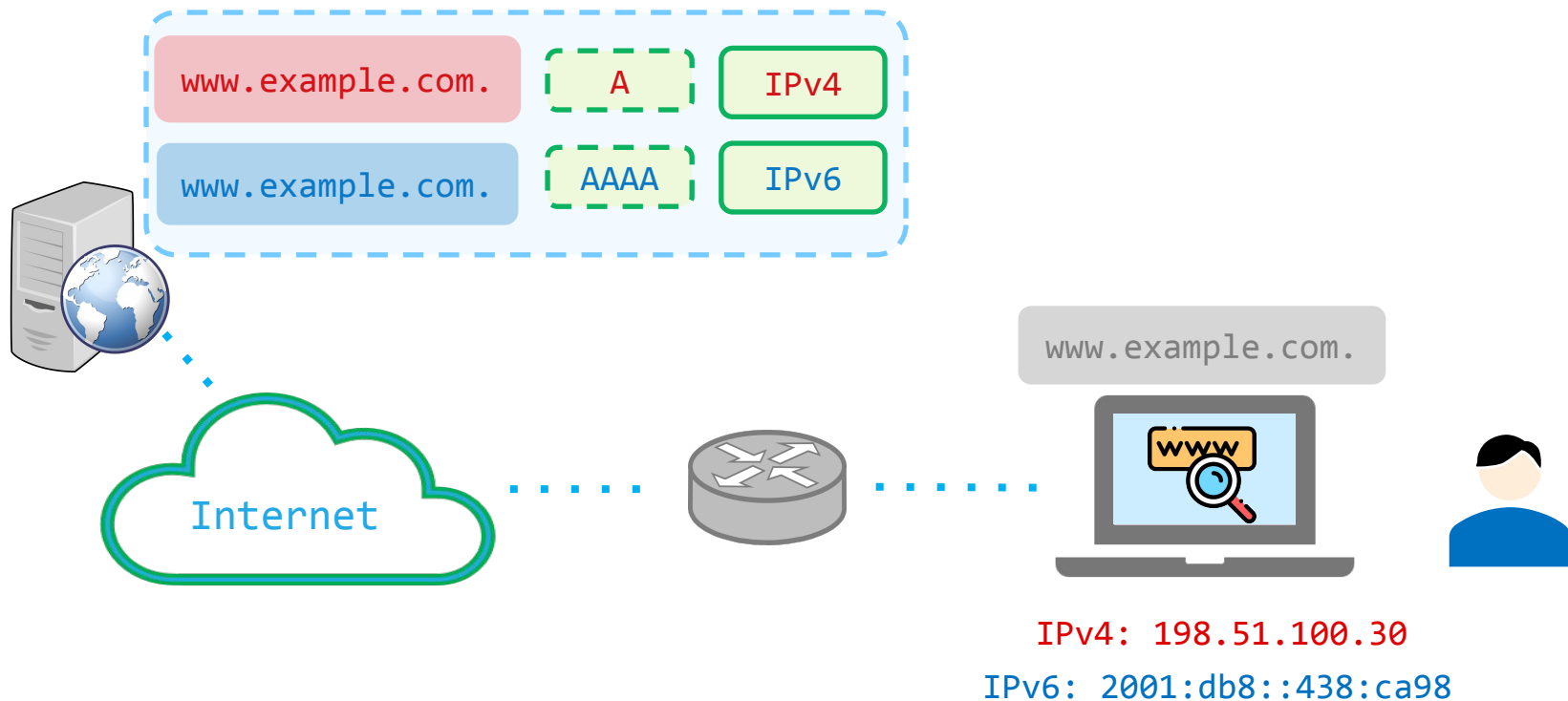
Introduction



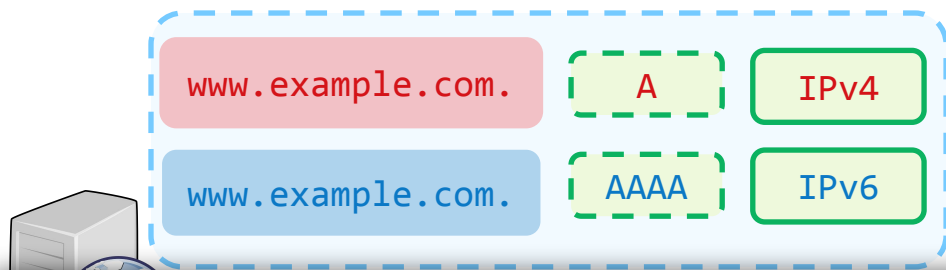
Introduction



Introduction: Sibling Address



Introduction: Sibling Address



Internet Nameserver IPv4 and IPv6 Address Relationships

Arthur Berger
MIT CSAIL / Akamai
awberger@csail.mit.edu

Nicholas Weaver
ICSI / UCSD
nweaver@icsi.berkeley.edu

Robert Beverly
Naval Postgraduate School
rbeverly@nps.edu

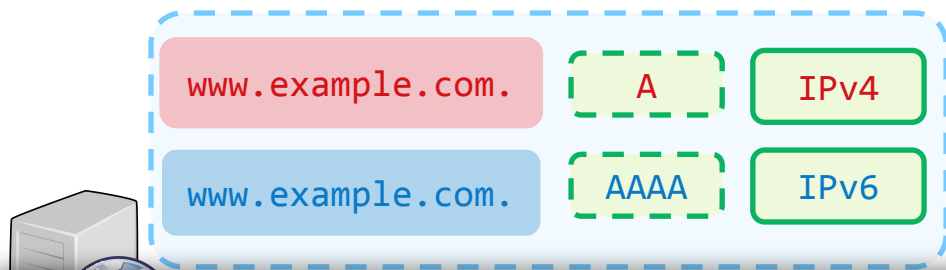
Larry Campbell
Akamai
lcampbel@akamai.com



IPv4: 198.51.100.30

IPv6: 2001:db8::438:ca98

Introduction: Sibling Address



Internet Nameserver IPv4 and IPv6 Address Relationships

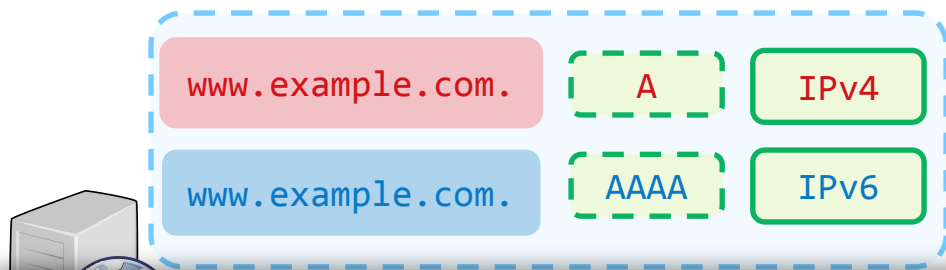
Server Siblings: Identifying Shared IPv4/IPv6 Infrastructure via Active Fingerprinting

Robert Beverly¹, Arthur Berger²

¹ Naval Postgraduate School, Monterey, CA

² MIT CSAIL / Akamai, Cambridge, MA
rbeverly@nps.edu, awberger@csail.mit.edu

Introduction: Sibling Address



Internet Nameserver IPv4 and IPv6 Address Relationships

Server Siblings: Identifying Shared IPv4/IPv6
Infrastructure via Active Fingerprinting

Large-Scale Classification of IPv6-IPv4 Siblings with Variable Clock Skew

Quirin Scheitle, Oliver Gasser, Minoo Rouhi, Georg Carle
Chair of Network Architectures and Services
Technical University of Munich (TUM)
Email: {scheitle,gasser,rouhi,carle}@net.in.tum.de

Introduction: Sibling Address



IP address → Prefix

Infrastructure via Active Fingerprinting

Large-Scale Classification of IPv6-IPv4 Siblings with Variable Clock Skew

Quirin Scheitle, Oliver Gasser, Minoo Rouhi, Georg Carle
Chair of Network Architectures and Services
Technical University of Munich (TUM)
Email: {scheitle,gasser,rouhi,carle}@net.in.tum.de

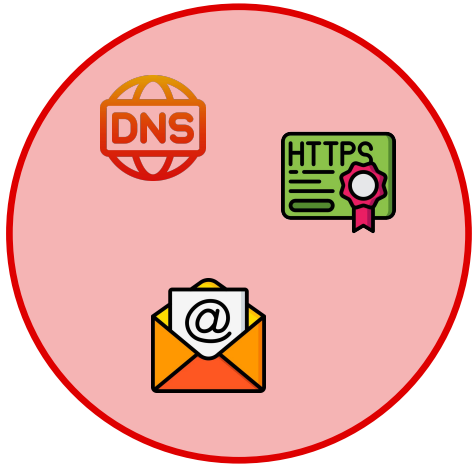
What is a Sibling Prefix?

IPv4 Prefix

IPv6 Prefix

What is a Sibling Prefix?

IPv4 Prefix

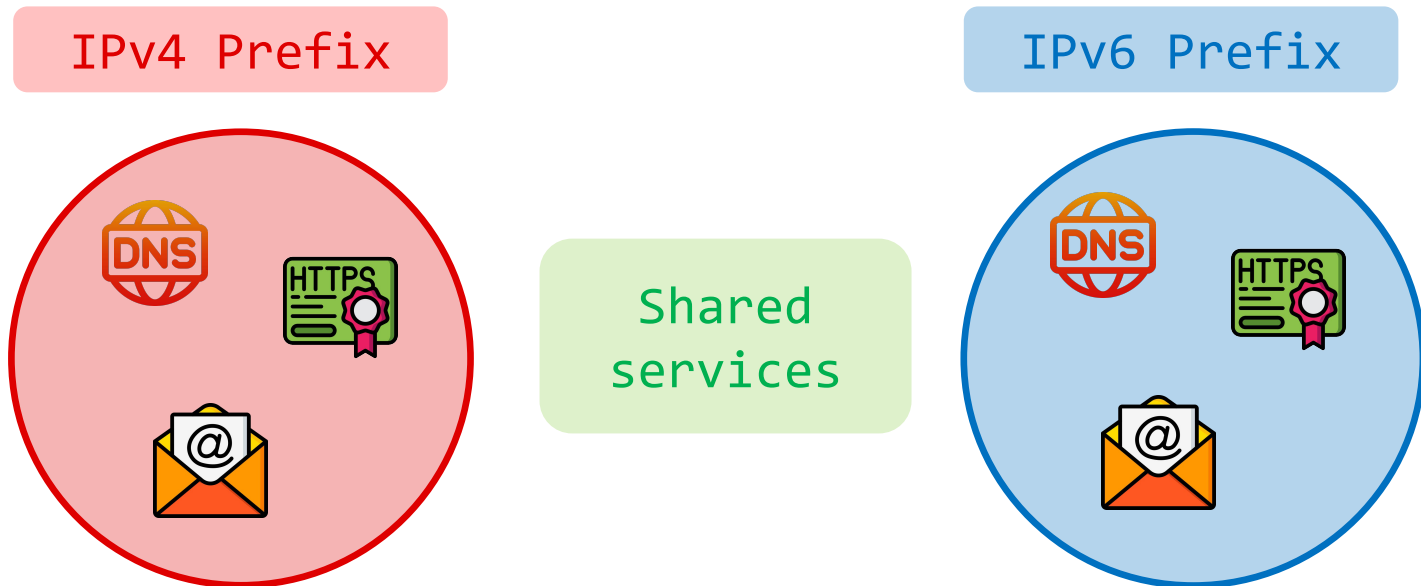


IPv6 Prefix



What is a Sibling Prefix?

A Sibling Prefix is an **IPv4** and **IPv6** prefix pair with similar services.



Why Identify Sibling Prefixes?

Why Identify Sibling Prefixes?

- Relation between IPv4 and IPv6 prefixes

Why Identify Sibling Prefixes?

- Relation between IPv4 and IPv6 prefixes
- Reveal IPv4-IPv6 interdependencies and shared infrastructures

Why Identify Sibling Prefixes?

- Relation between IPv4 and IPv6 prefixes
- Reveal IPv4-IPv6 interdependencies and shared infrastructures
- Apply consistent policies across both IP address families

Why Identify Sibling Prefixes?

- Relation between IPv4 and IPv6 prefixes
- Reveal IPv4-IPv6 interdependencies and shared infrastructures
- Apply consistent policies across both IP address families
- Provide a consistent basis to compare performance, geolocation, security, etc.

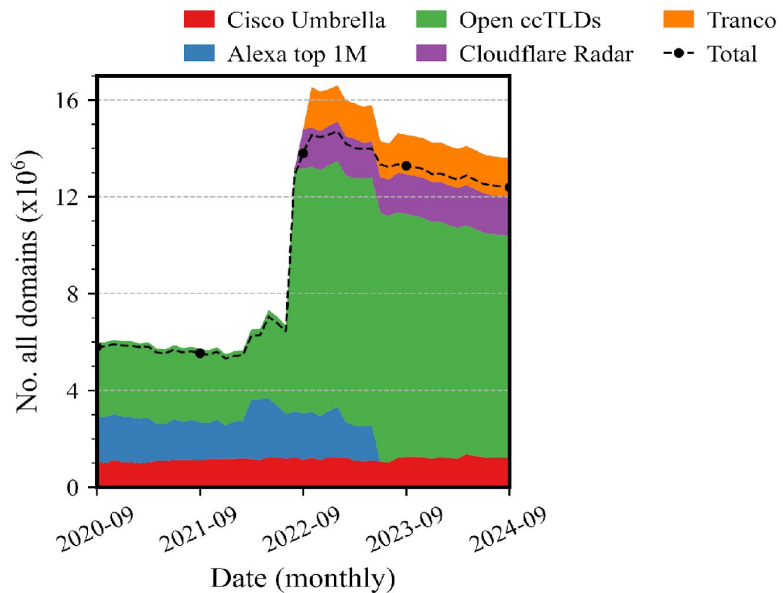
Dataset

DNS dataset from Open 

 Cisco Umbrella  Open ccTLDs  Tranco
 Alexa top 1M  Cloudflare Radar

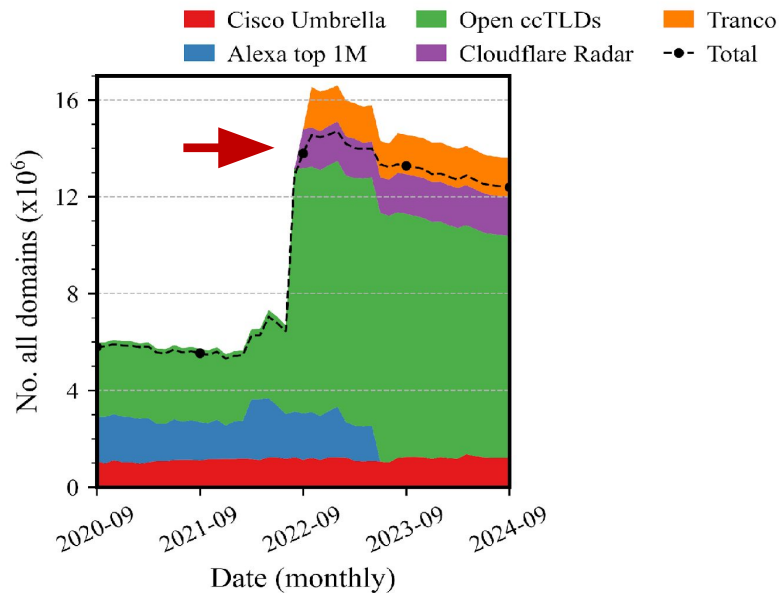
Dataset

DNS dataset from Open INTEL



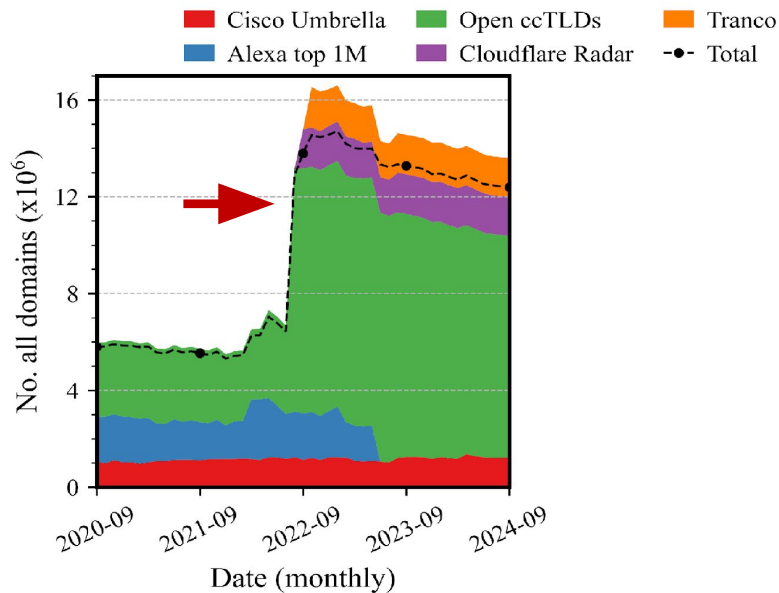
Dataset

DNS dataset from Open INTEL



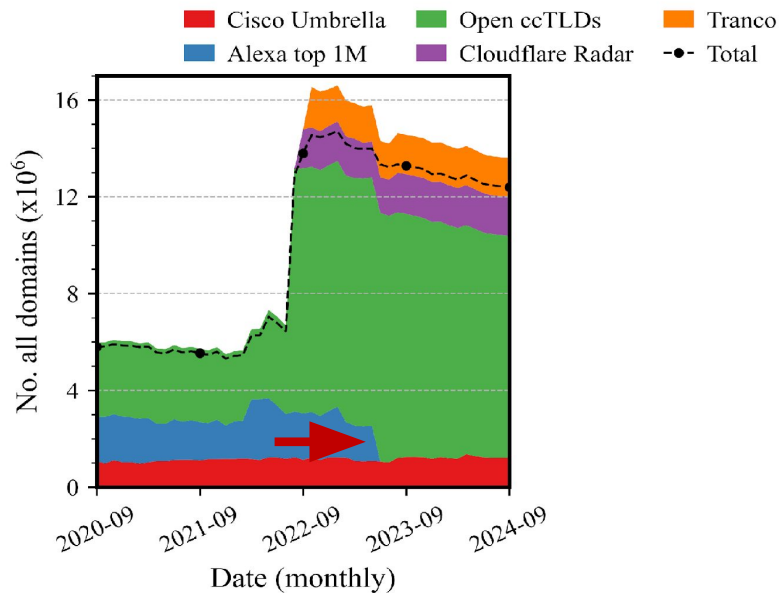
Dataset

DNS dataset from Open INTEL



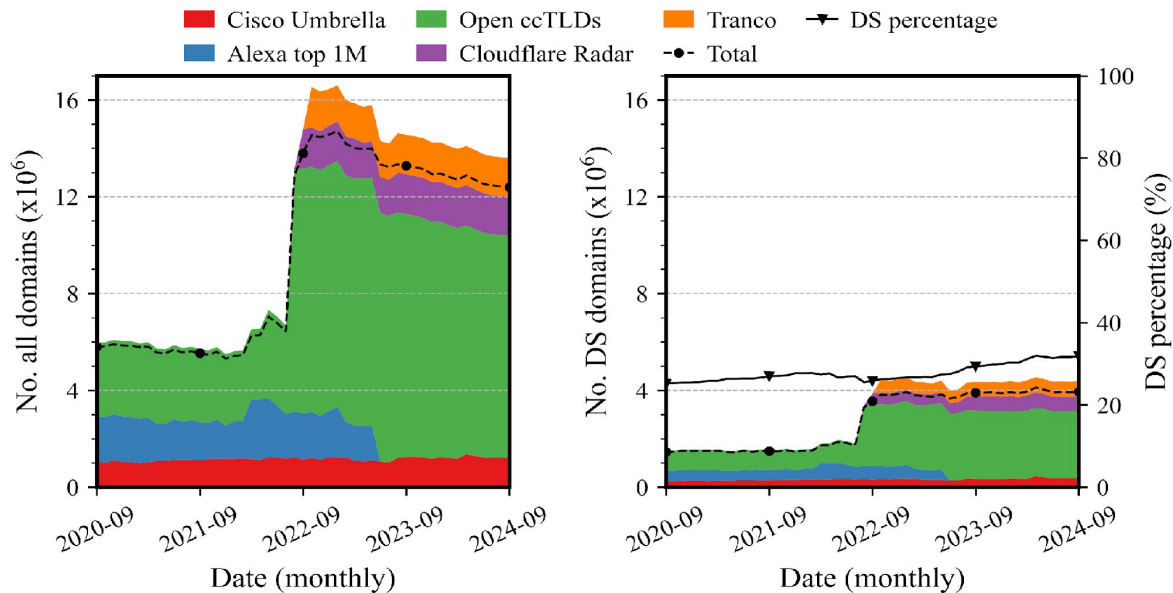
Dataset

DNS dataset from Open INTEL



Dataset

DNS dataset from Open INTEL

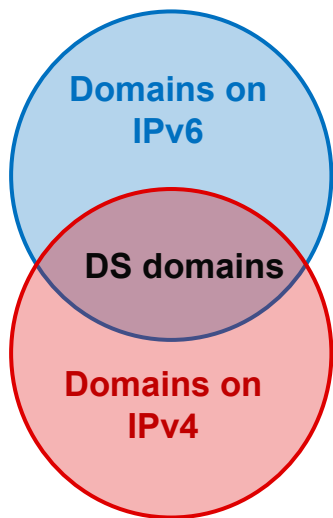


Identifying Sibling Prefixes

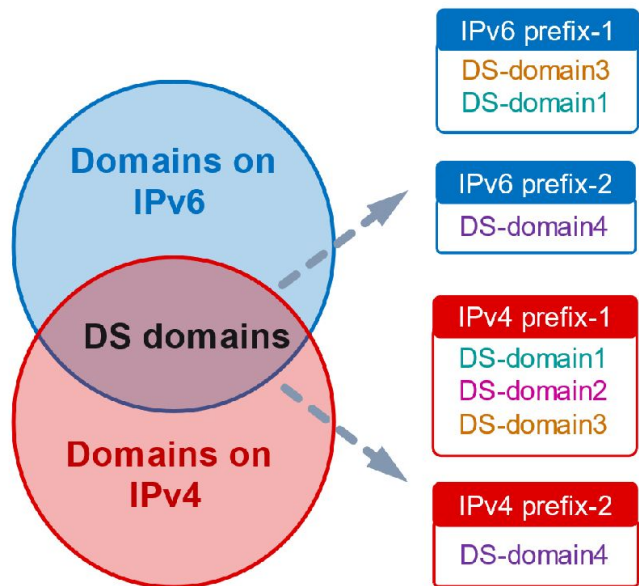
Identifying Sibling Prefixes

Step 1

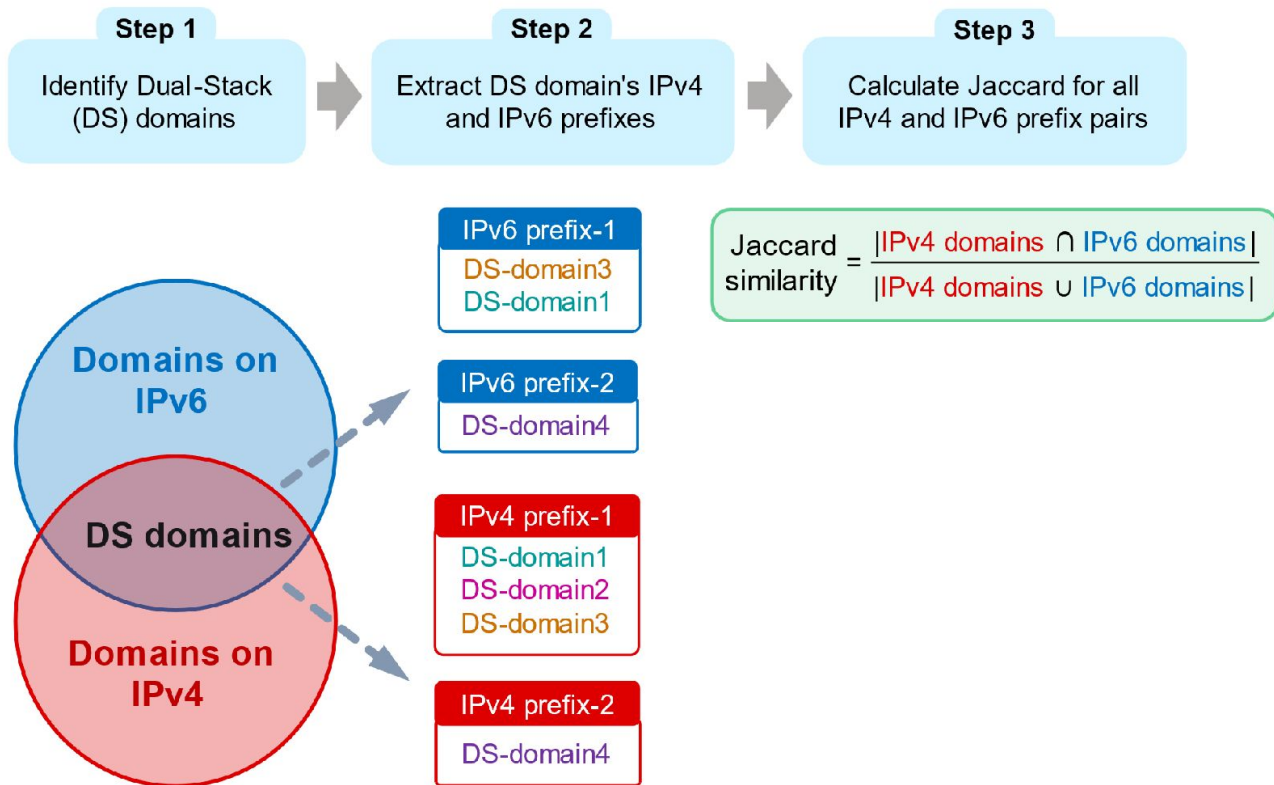
Identify Dual-Stack
(DS) domains



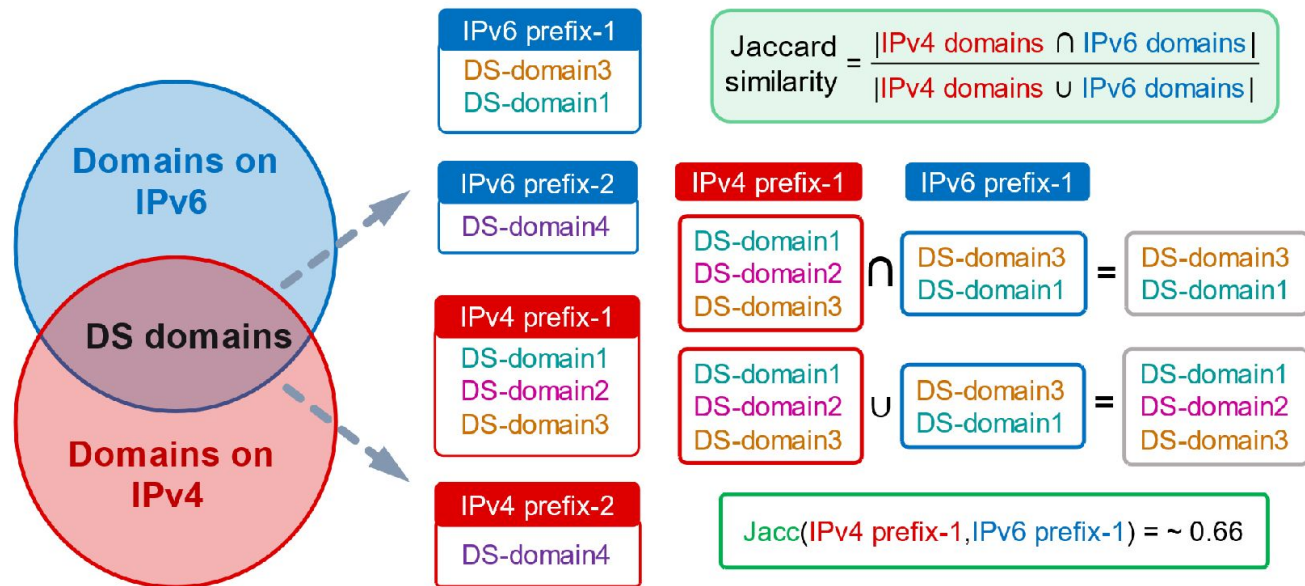
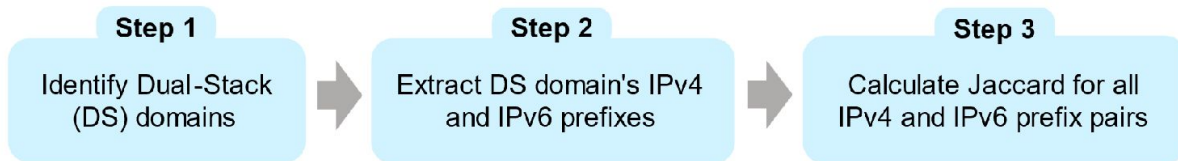
Identifying Sibling Prefixes



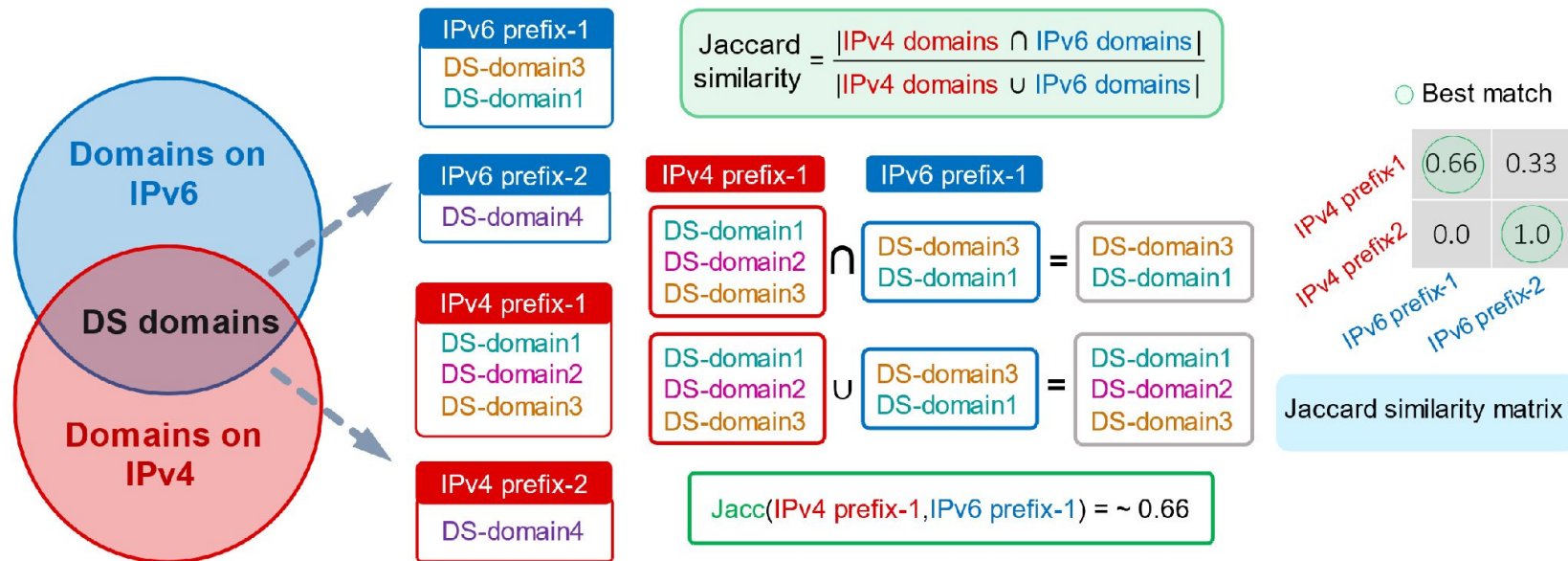
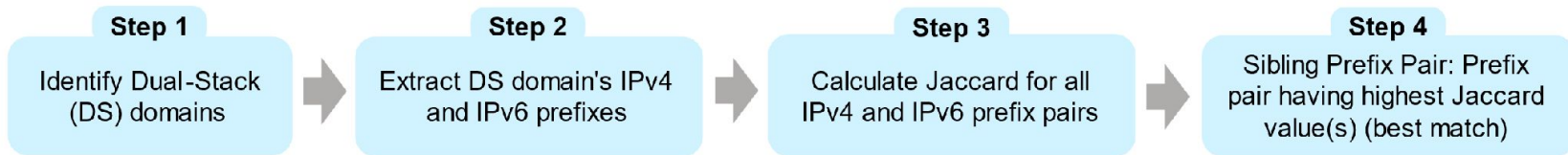
Identifying Sibling Prefixes



Identifying Sibling Prefixes

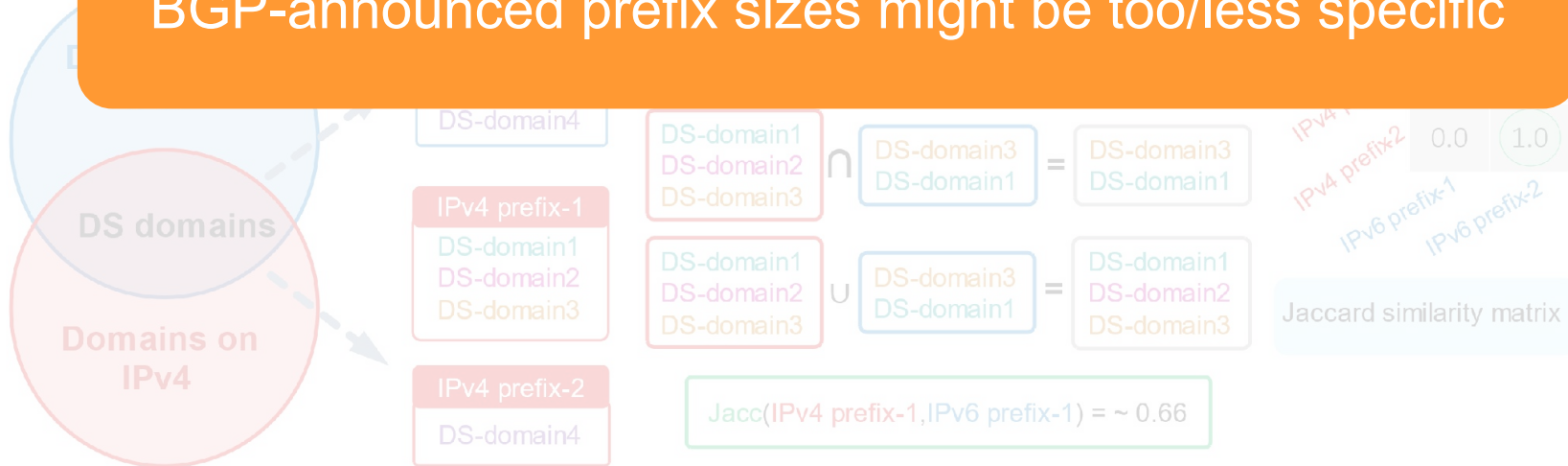


Identifying Sibling Prefixes



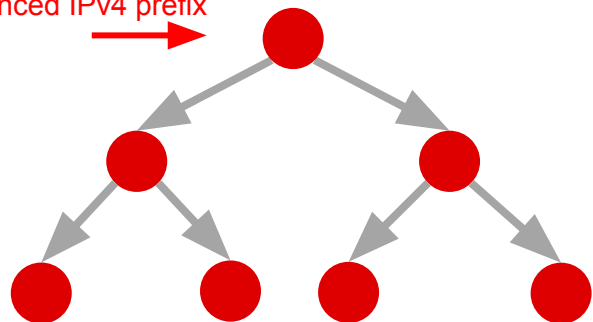
Identifying Sibling Prefixes

BGP-announced prefix sizes might be too/less specific



Sibling Prefix Tuner Algorithm (SP-Tuner)

BGP-announced IPv4 prefix



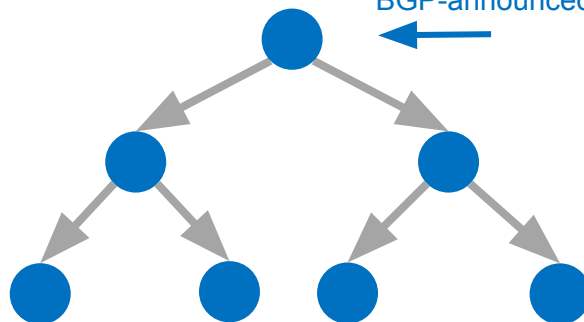
⋮

⋮



IPv4 addresses and dual-stack domains at each leaf

BGP-announced IPv6 prefix



⋮

⋮



IPv6 addresses and dual-stack domains at each leaf

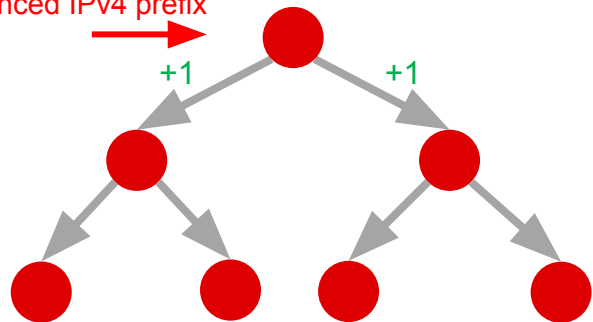
Sibling Prefix Tuner Algorithm (SP-Tuner)

BGP-announced IPv4 prefix



+1

+1



⋮

⋮



IPv4 addresses and dual-stack domains at each leaf

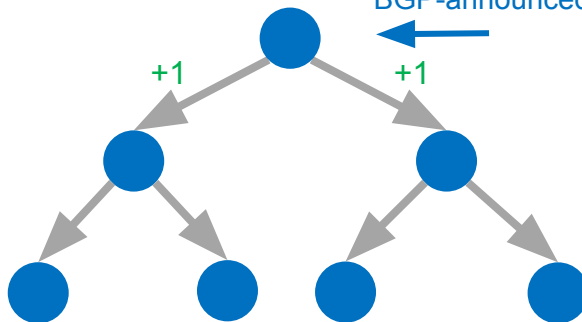


BGP-announced IPv6 prefix



+1

+1



⋮

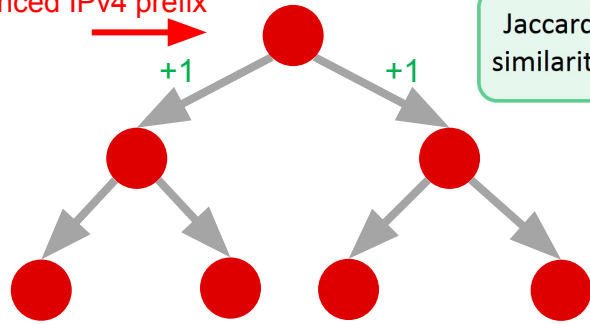
⋮



IPv6 addresses and dual-stack domains at each leaf

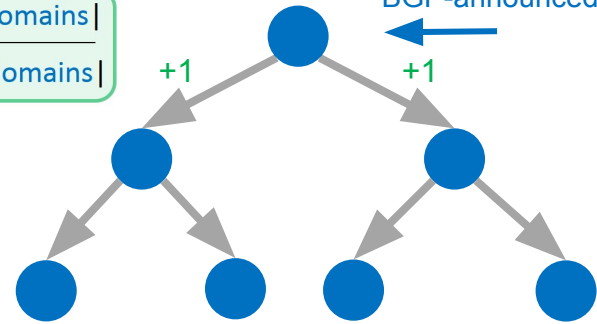
Sibling Prefix Tuner Algorithm (SP-Tuner)

BGP-announced IPv4 prefix



$$\text{Jaccard similarity} = \frac{|\text{IPv4 domains} \cap \text{IPv6 domains}|}{|\text{IPv4 domains} \cup \text{IPv6 domains}|}$$

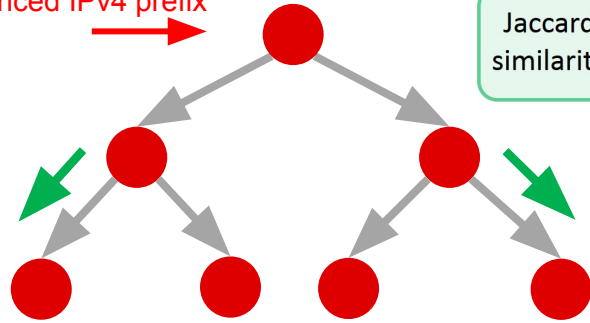
BGP-announced IPv6 prefix



- Calculate Jaccard similarity for all possible prefix pairs in each level

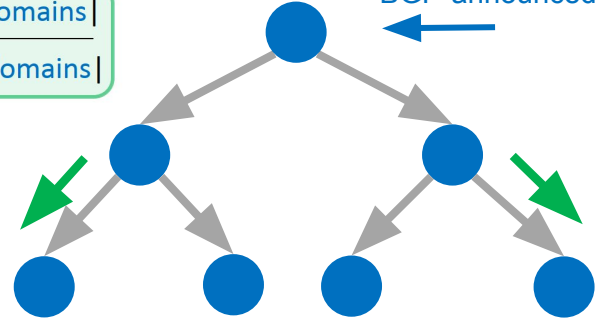
Sibling Prefix Tuner Algorithm (SP-Tuner)

BGP-announced IPv4 prefix



$$\text{Jaccard similarity} = \frac{|\text{IPv4 domains} \cap \text{IPv6 domains}|}{|\text{IPv4 domains} \cup \text{IPv6 domains}|}$$

BGP-announced IPv6 prefix



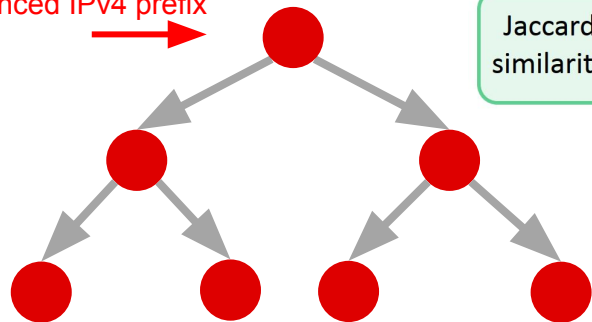
- Calculate Jaccard similarity for all possible prefix pairs in each level
- Continue, if Jaccard similarity values are better/same as before

IPv4 ad

af

Sibling Prefix Tuner Algorithm (SP-Tuner)

BGP-announced IPv4 prefix



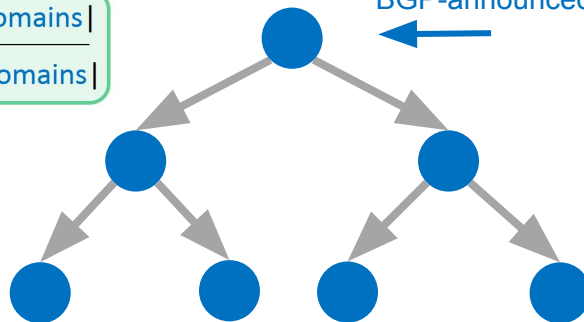
$$\text{Jaccard similarity} = \frac{|\text{IPv4 domains} \cap \text{IPv6 domains}|}{|\text{IPv4 domains} \cup \text{IPv6 domains}|}$$

Thresholds:

IPv4 CIDR: /28

IPv6 CIDR: /96

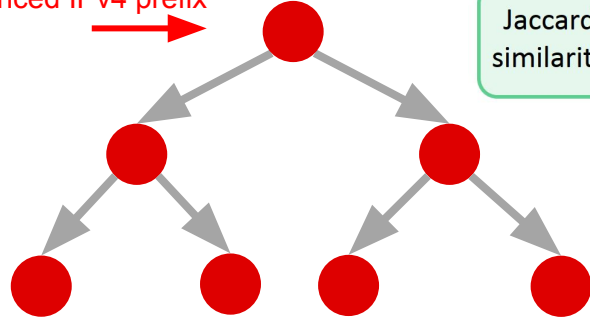
BGP-announced IPv6 prefix



- Calculate Jaccard similarity for all possible prefix pairs in each level
- Continue, if Jaccard similarity values are better/same as before
- Stop when prefix sizes exceeds the prefix-length thresholds

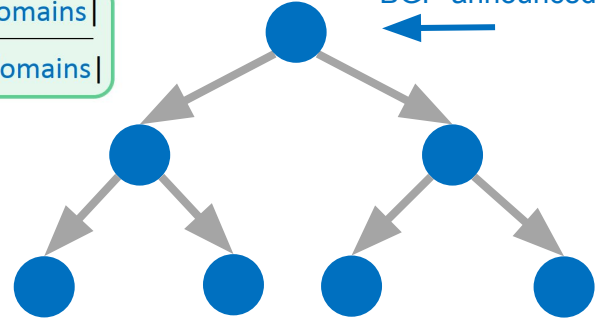
Sibling Prefix Tuner Algorithm (SP-Tuner)

BGP-announced IPv4 prefix



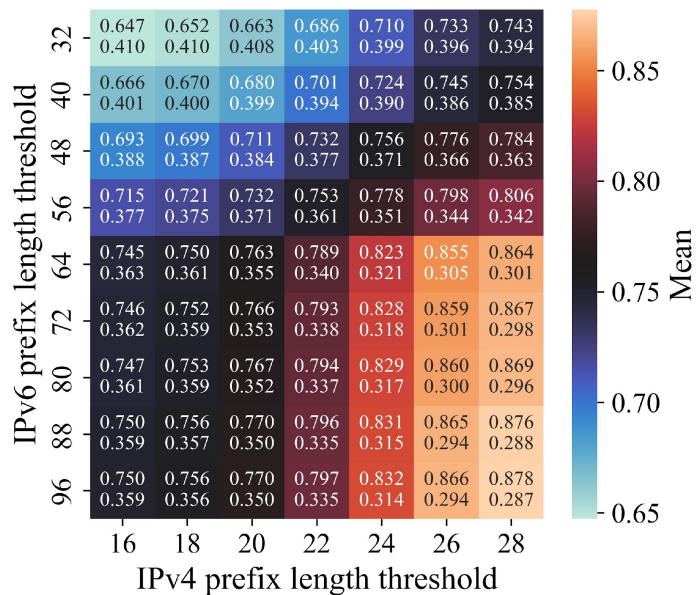
$$\text{Jaccard similarity} = \frac{|\text{IPv4 domains} \cap \text{IPv6 domains}|}{|\text{IPv4 domains} \cup \text{IPv6 domains}|}$$

BGP-announced IPv6 prefix

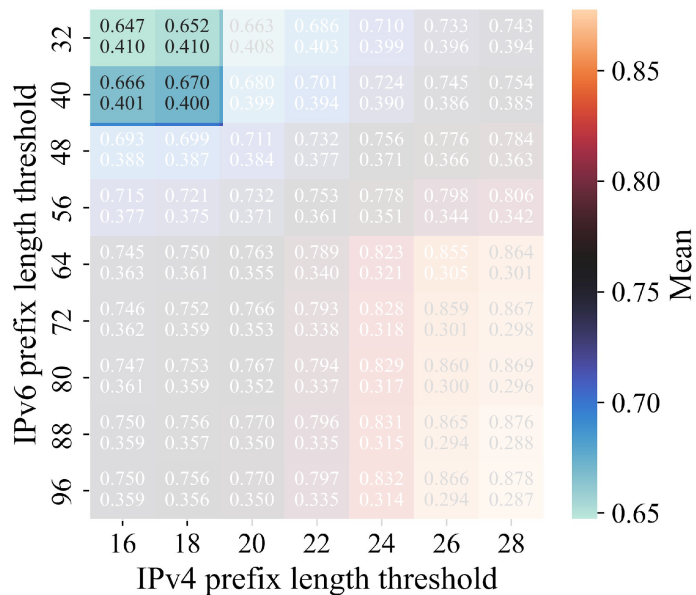


- Calculate Jaccard similarity for all possible prefix pairs in each level
- Continue, if Jaccard similarity values are better/same as before
- Stop when prefix sizes exceeds the prefix-length thresholds
- Keep branches and treat them as new candidate sibling prefix pairs

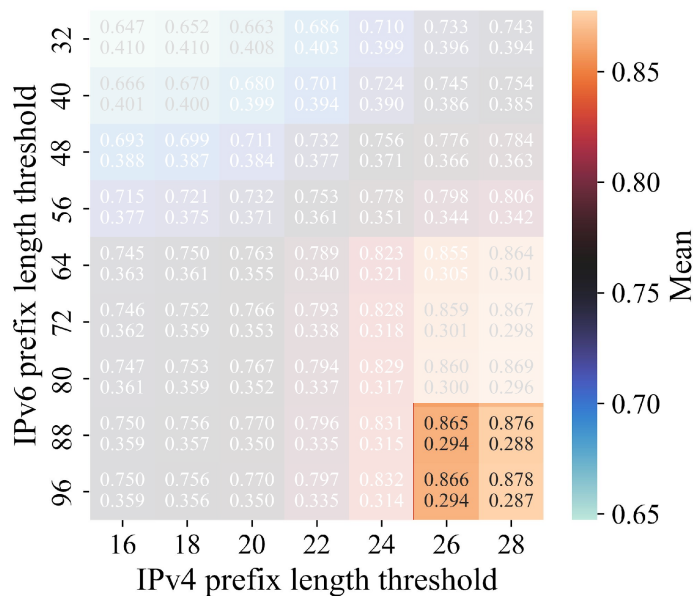
Effectiveness of SP-Tuner



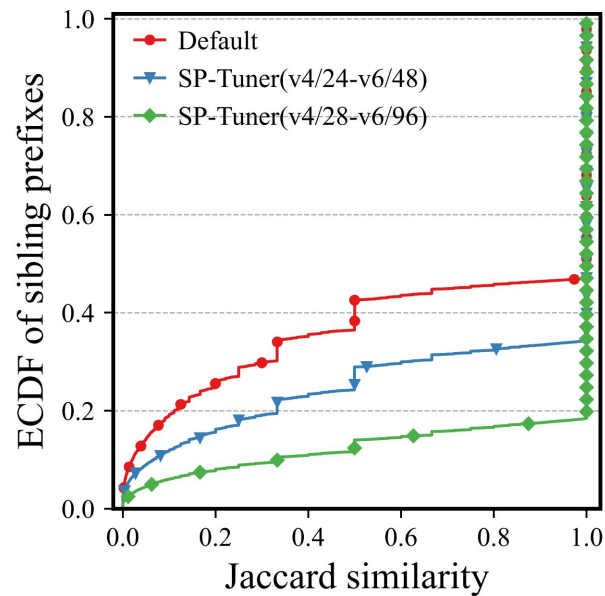
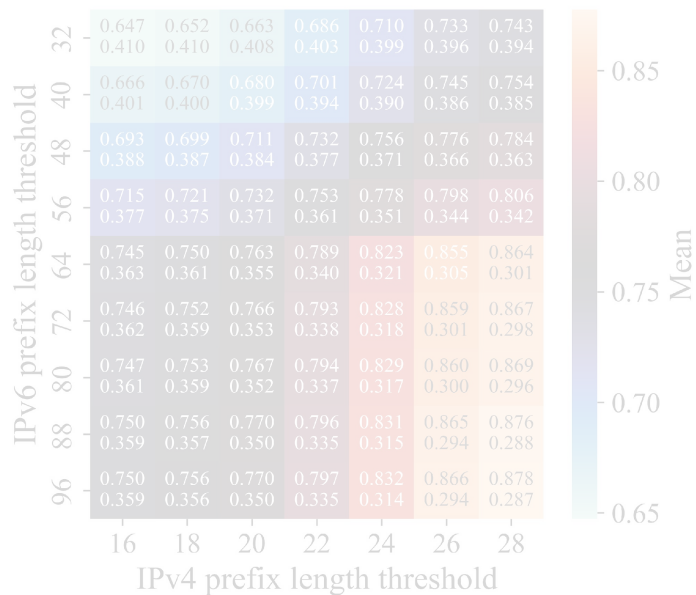
Effectiveness of SP-Tuner



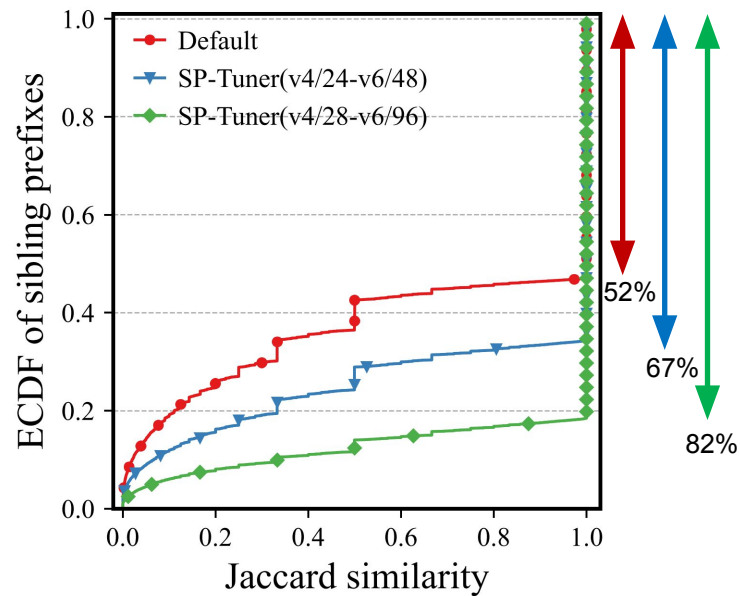
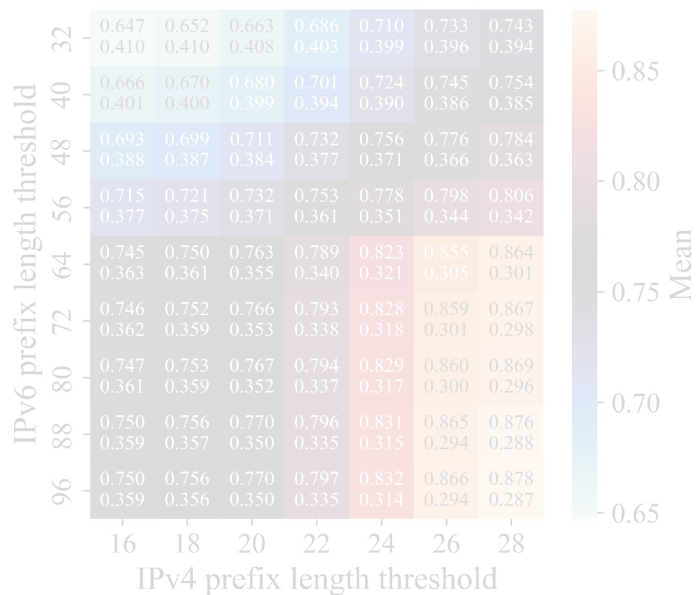
Effectiveness of SP-Tuner



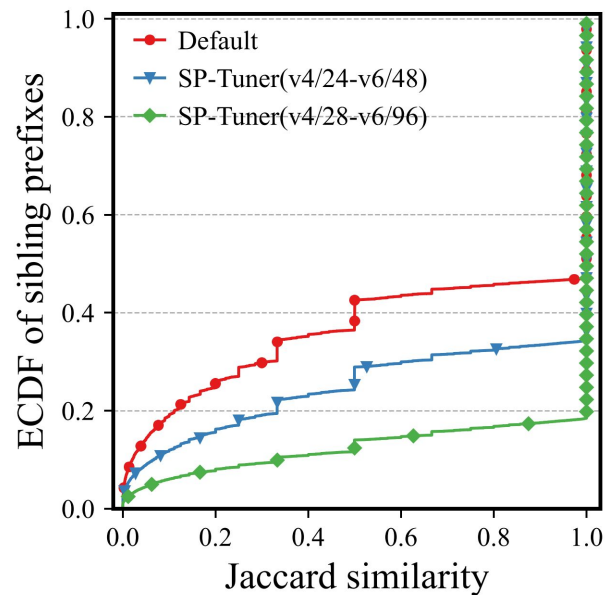
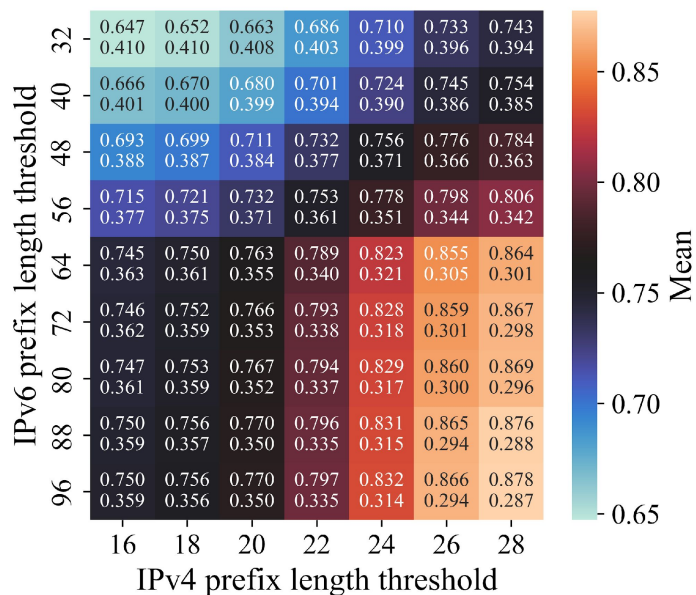
Effectiveness of SP-Tuner



Effectiveness of SP-Tuner

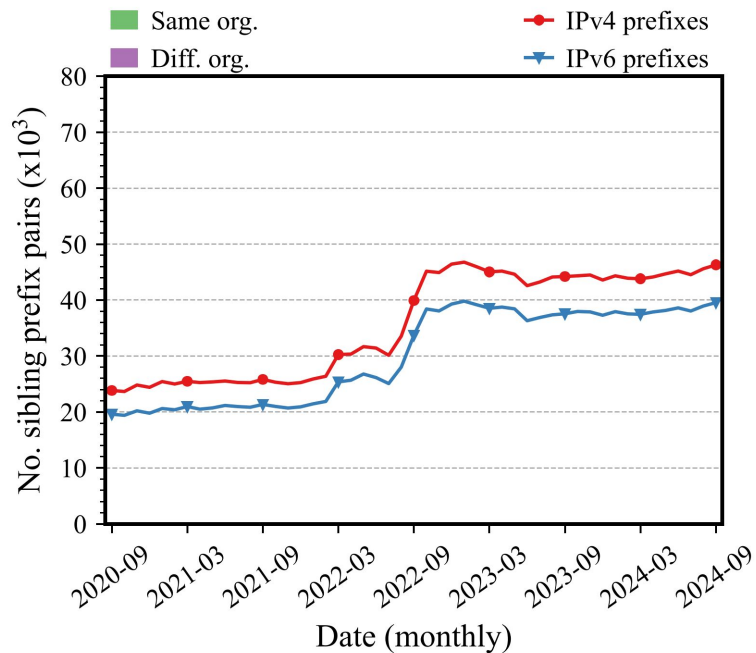


Effectiveness of SP-Tuner



SP-Tuner improves Jaccard similarity
and identifies more similar sibling prefixes

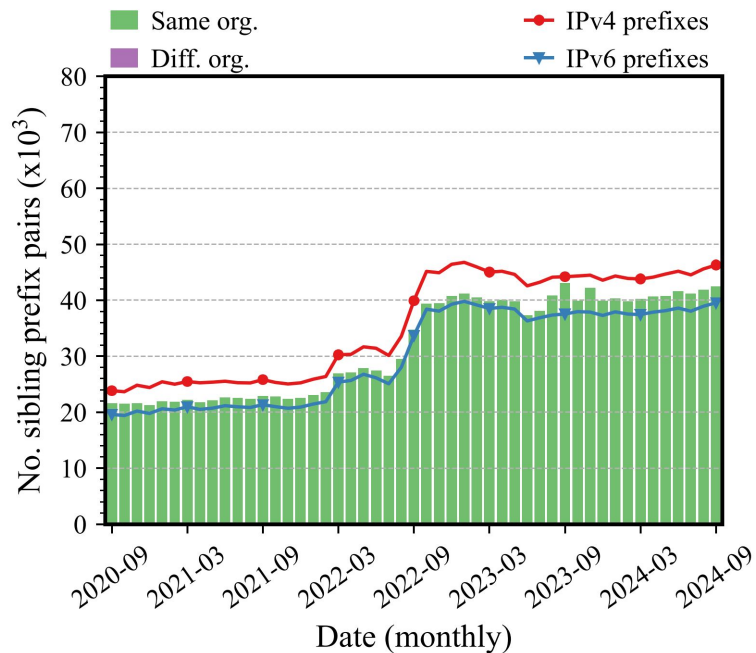
Sibling Prefixes Over Time



IPv4 Prefix

IPv6 Prefix

Sibling Prefixes Over Time



IPv4 Prefix

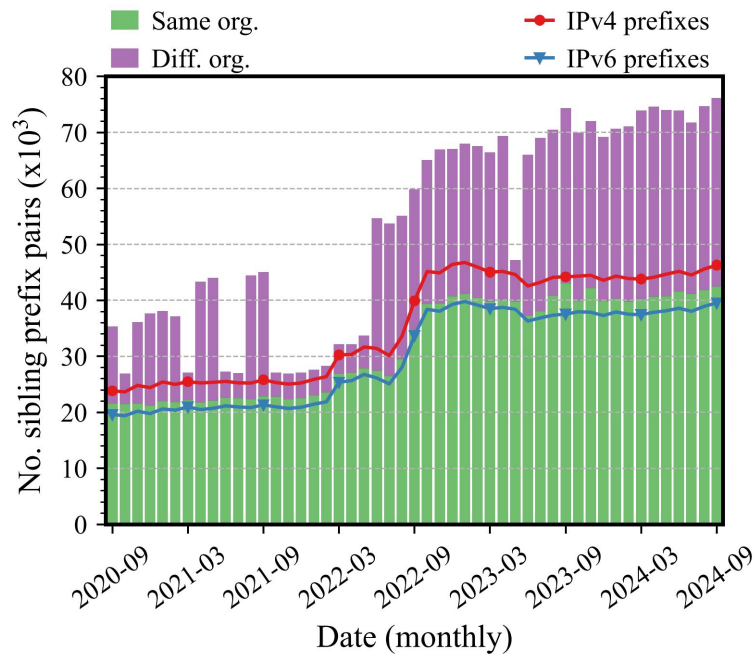
IPv6 Prefix

Organization

=

Organization

Sibling Prefixes Over Time



IPv4 Prefix

IPv6 Prefix

Organization

=

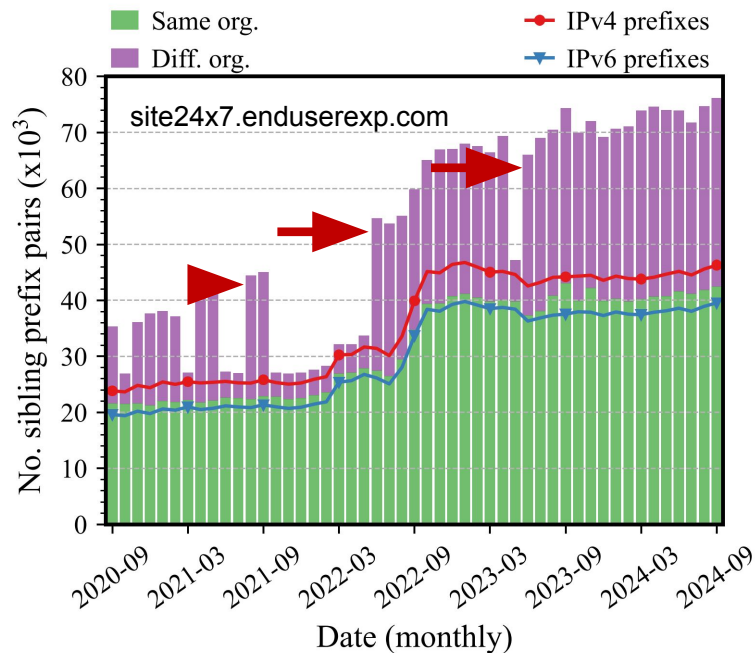
Organization

Organization

≠

Organization

Sibling Prefixes Over Time



IPv4 Prefix

IPv6 Prefix

Organization

=

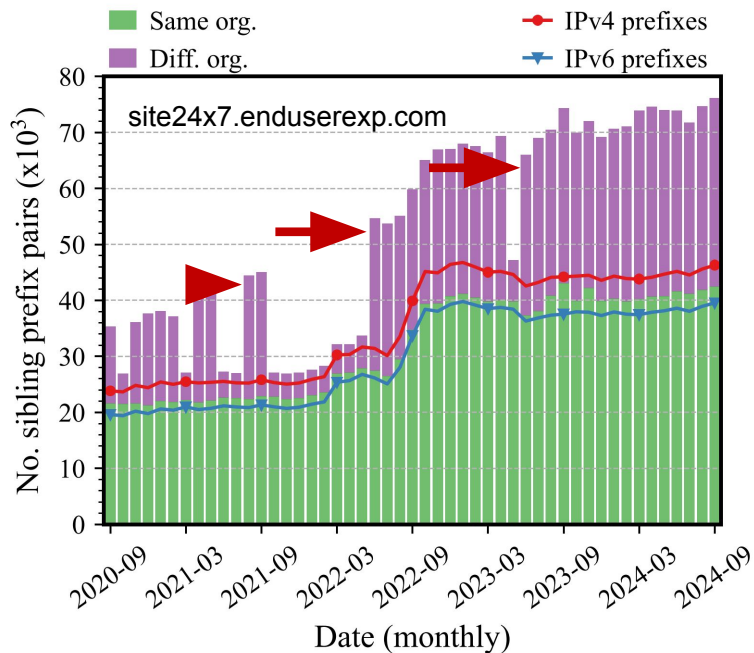
Organization

Organization

≠

Organization

Sibling Prefixes Over Time



IPv4 Prefix

IPv6 Prefix

Organization

=

Organization

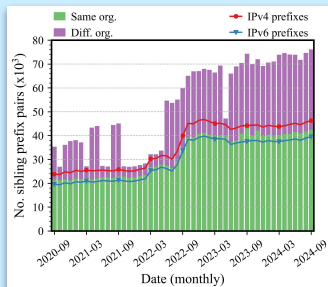
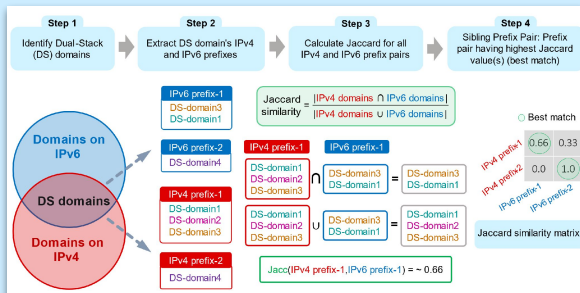
Organization

≠

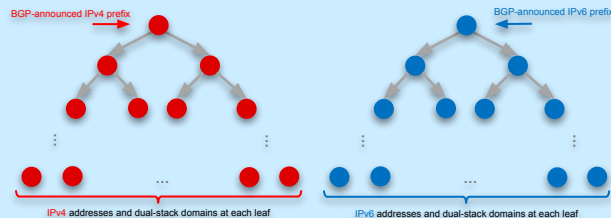
Organization

Identified 76k sibling prefixes in the latest snapshot

Conclusion



sibling-prefixes.
github.io



Sibling Prefixes: Identifying Similarities in IPv4 and IPv6 Prefixes

Fariba Osali
Max Planck Institute for Informatics
Saarbrücken, Germany
fosalai@mpi-inf.mpg.de

Khwaja Zubair Sediqi
Max Planck Institute for Informatics,
Saarland University
Saarbrücken, Germany
zsediqi@mpi-inf.mpg.de

Oliver Gasser
IPInfo
Seattle, USA
oliver@ipinfo.io