black hat EUROPE 2023

# TsuKing: Coordinating DNS Resolvers and Queries into Potent DDoS Amplifiers

Speaker: **Haixin Duan**

Slides Contributors: **Wei Xu** & **Xiang Li** & **Chaoyi Lu**

Tsinghua University, Dec. 2023

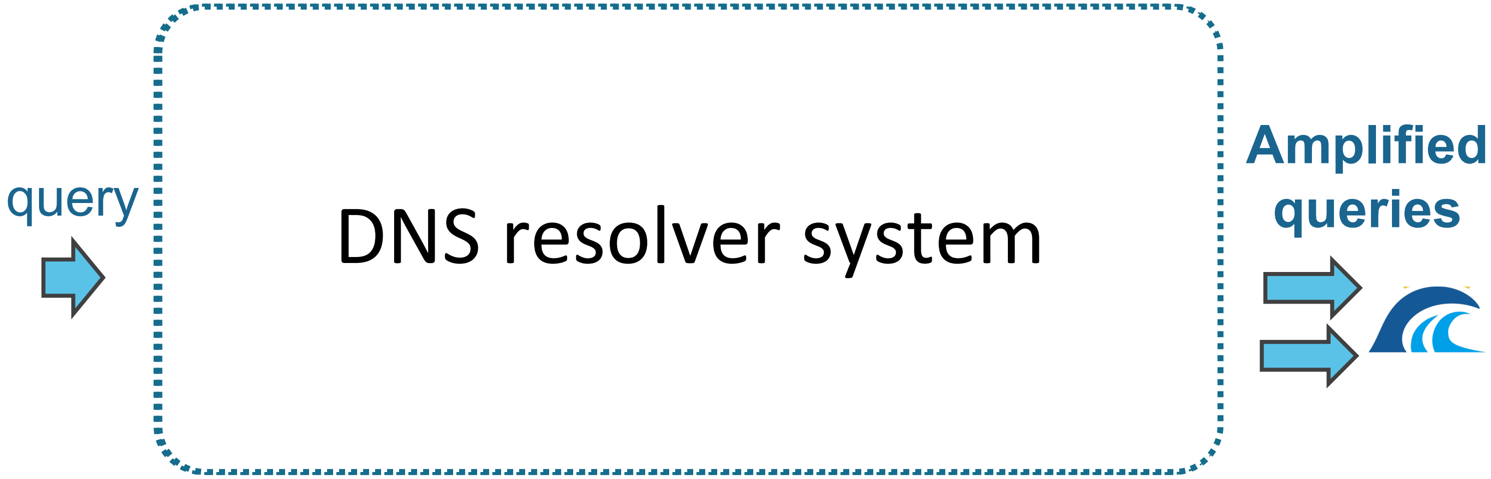
# TsuKing: Tsunami + King

?

# TsuKing: Tsunami + King

*(Traffic amplification ability)*

- Cause: DNS implementation choices & complex service infrastructure



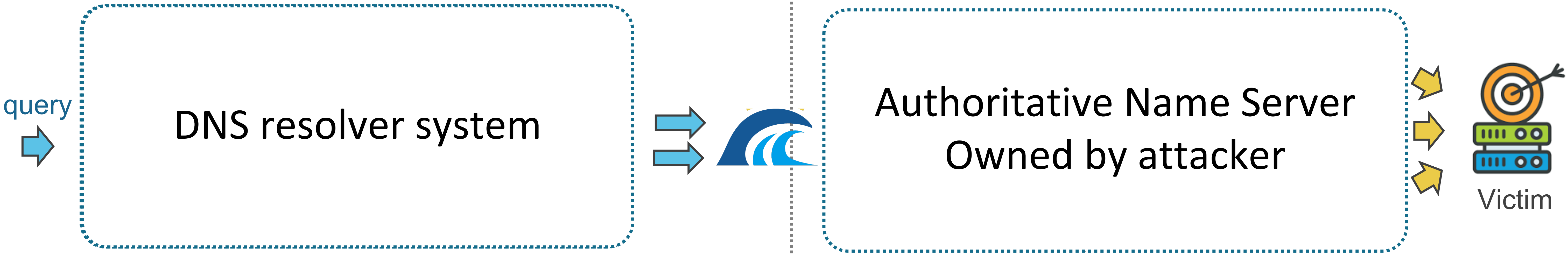
# TsuKing: Tsunami + King [1]

(Traffic amplification ability)

(Server coordination ability)

- ❖ Cause: DNS implementation choices & complex service infrastructure

- ❖ Coordinates DNS server systems -> **3,000+× amplification factor (*king* of DoS)**



query

DNS resolver system

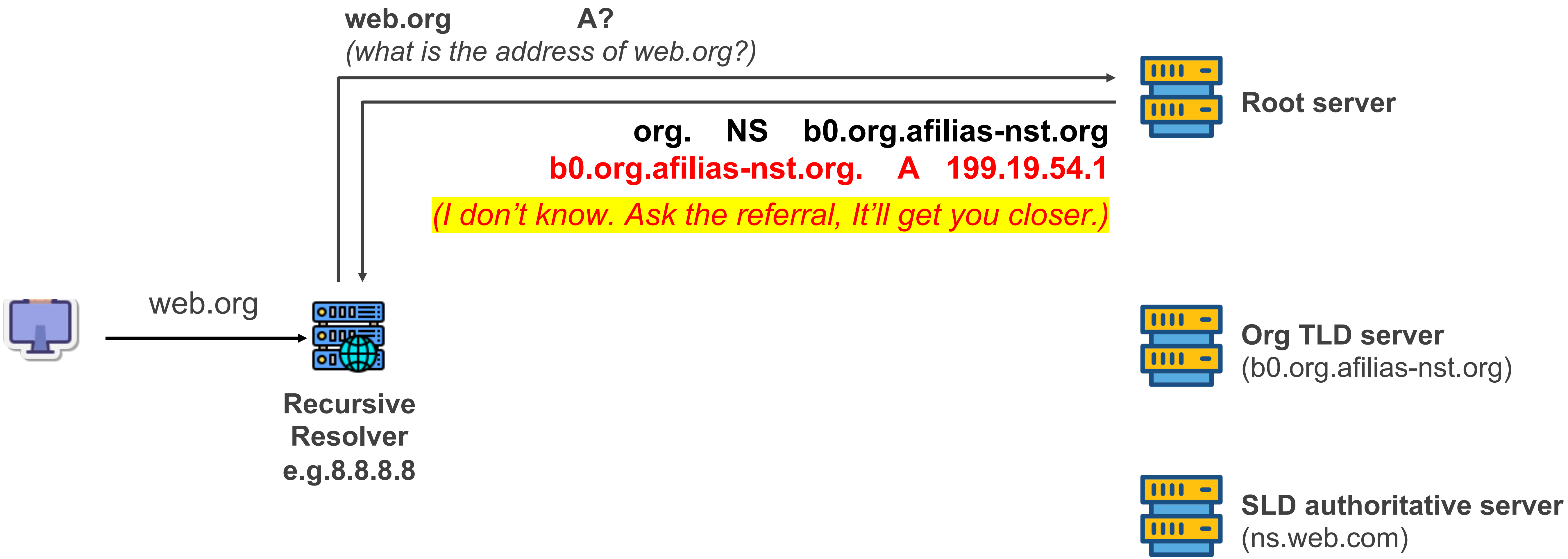
Authoritative Name Server Owned by attacker

Victim

[1] King: estimating latency between arbitrary internet end hosts, ACM CCR 2002

# DNS resolution guided by *referrals*

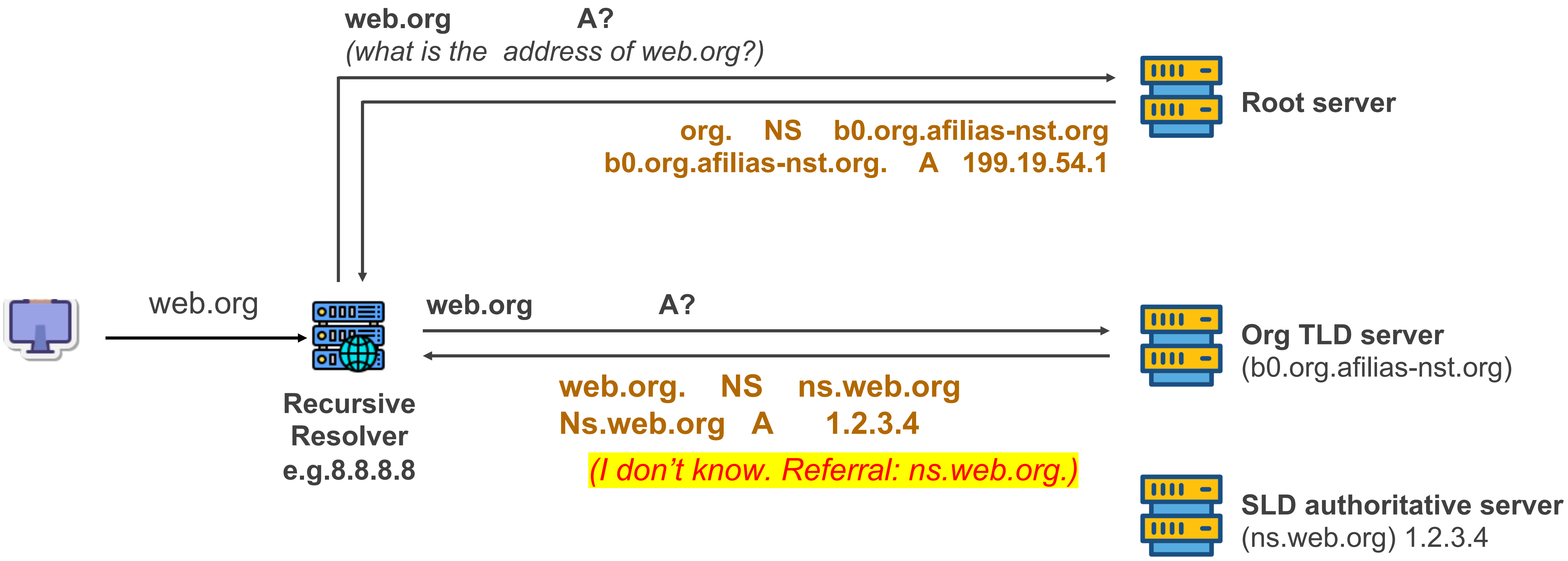
❖ Referrals *tell recursive resolvers who to ask next*



**web.org A?**
*(what is the address of web.org?)*
**org. NS b0.org.afilias-nst.org**
**b0.org.afilias-nst.org. A 199.19.54.1**
*(I don't know. Ask the referral, It'll get you closer.)*
web.org
**Recursive Resolver e.g.8.8.8.8**
**Root server**
**Org TLD server**
(b0.org.afilias-nst.org)
**SLD authoritative server**
(ns.web.com)

# DNS resolution guided by *referrals*

❖ Referrals *tell recursive resolvers who to ask next*



**web.org A?**

*(what is the address of web.org?)*

**org. NS b0.org.afilias-nst.org**

**b0.org.afilias-nst.org. A 199.19.54.1**

**Root server**

web.org

**web.org A?**

**Org TLD server**

(b0.org.afilias-nst.org)

**web.org. NS ns.web.org**

**Ns.web.org A 1.2.3.4**

*(I don't know. Referral: ns.web.org.)*

**Recursive Resolver**

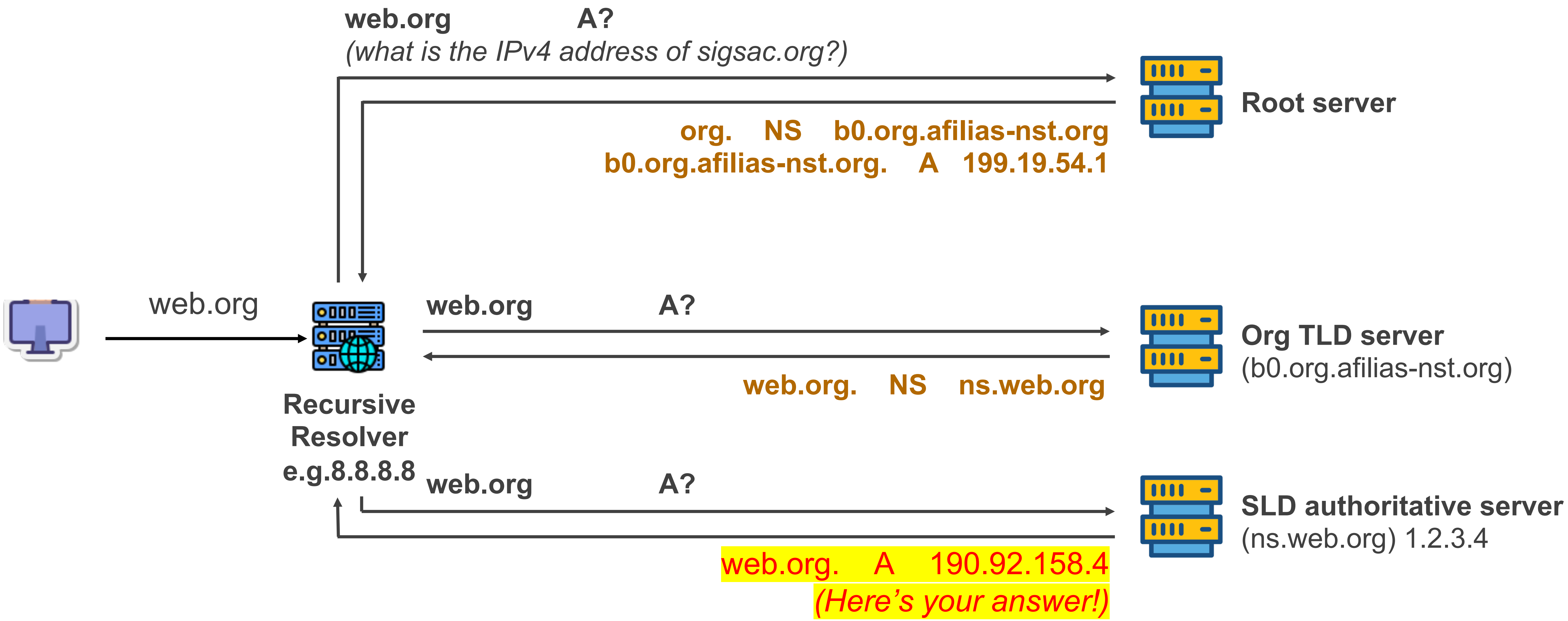
**e.g.8.8.8.8**

**SLD authoritative server**

(ns.web.org) 1.2.3.4

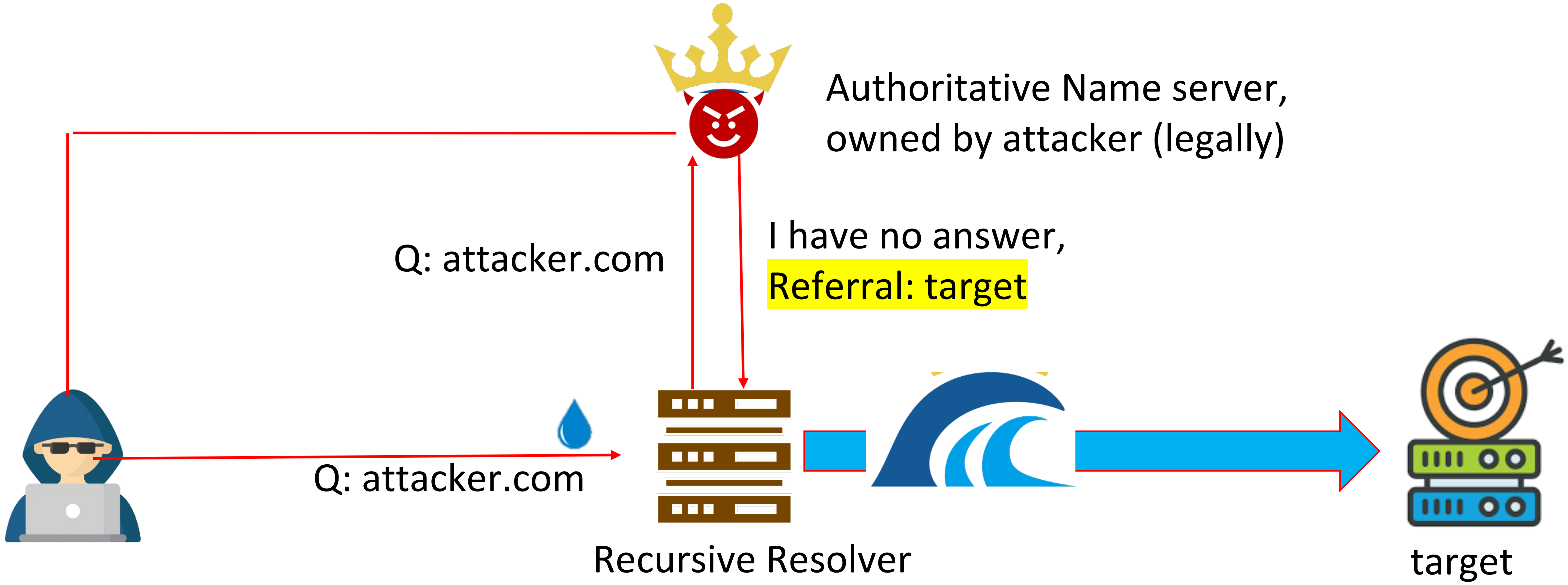
# DNS resolution guided by *referrals*

- **Recursive DNS resolution guided by *referrals***
  - Referrals *tell recursive resolvers who to ask next*



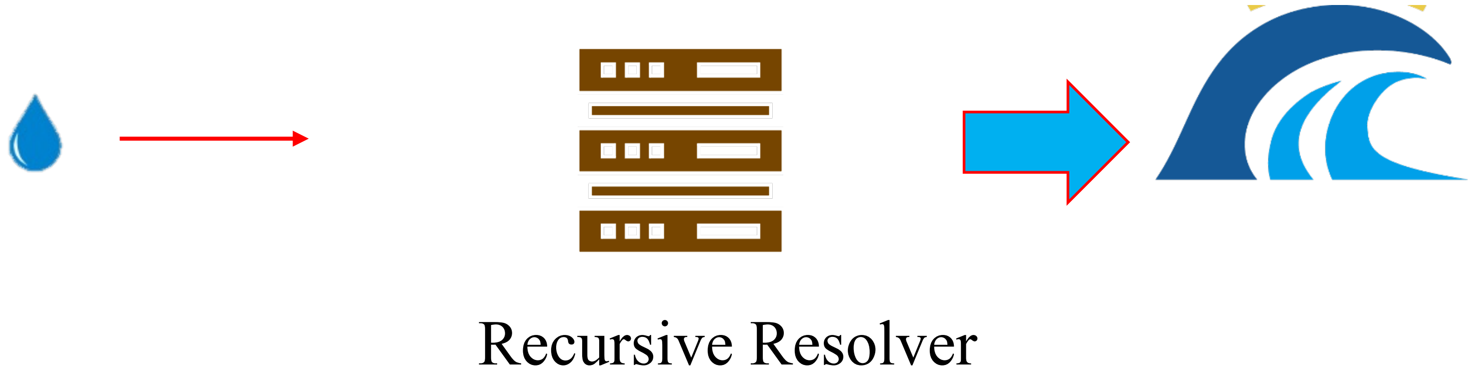
# Threat model of TsuKing

❖ Attacker sends DNS query for his own domain name



# Sounds simple, but

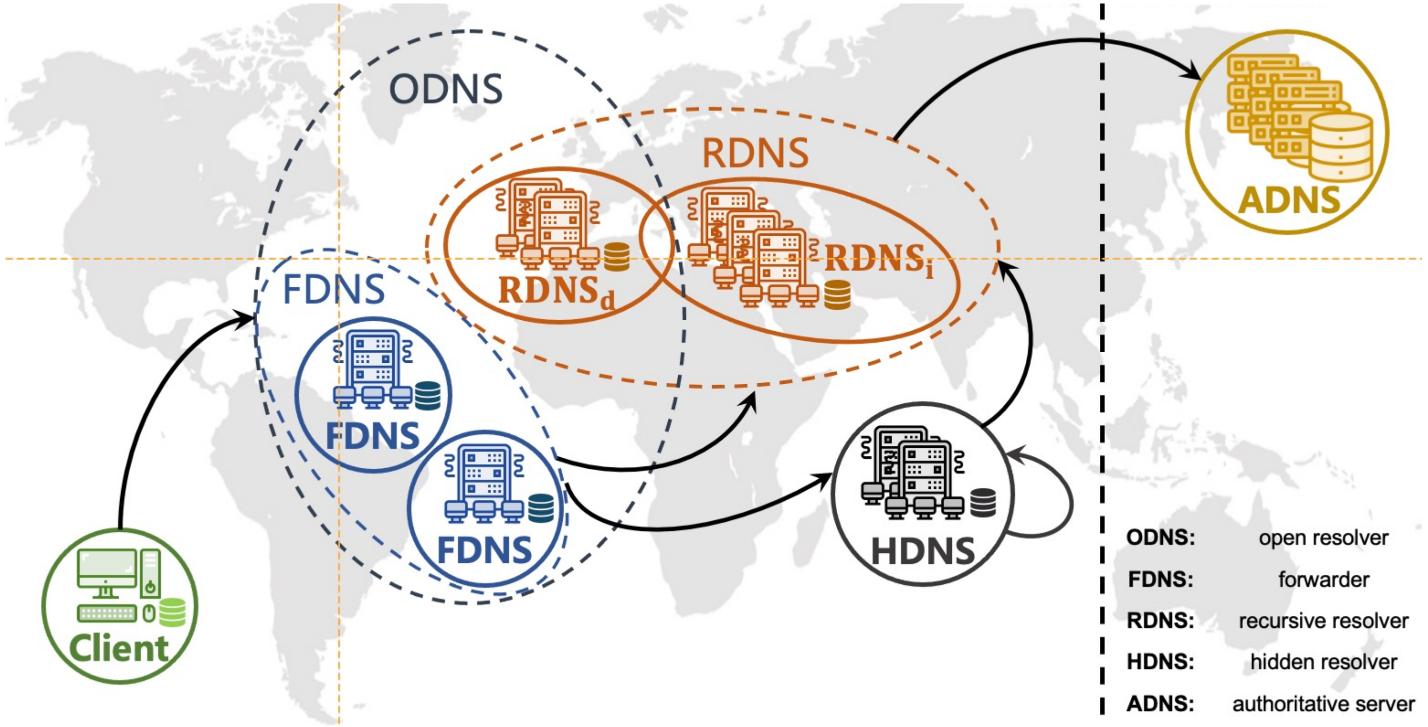
## Why does a resolver amplify query traffic?
## Is it that powerful?



# DNS as a complex infrastructure

- Multiple *types* and *layers* of DNS servers
  - DNS forwarders ➔ pass queries to upstream *(e.g., another forwarder)*
  - Large public DNS services ➔ complexes of load balancers, caches, egress servers, etc.

## The complex DNS infrastructure

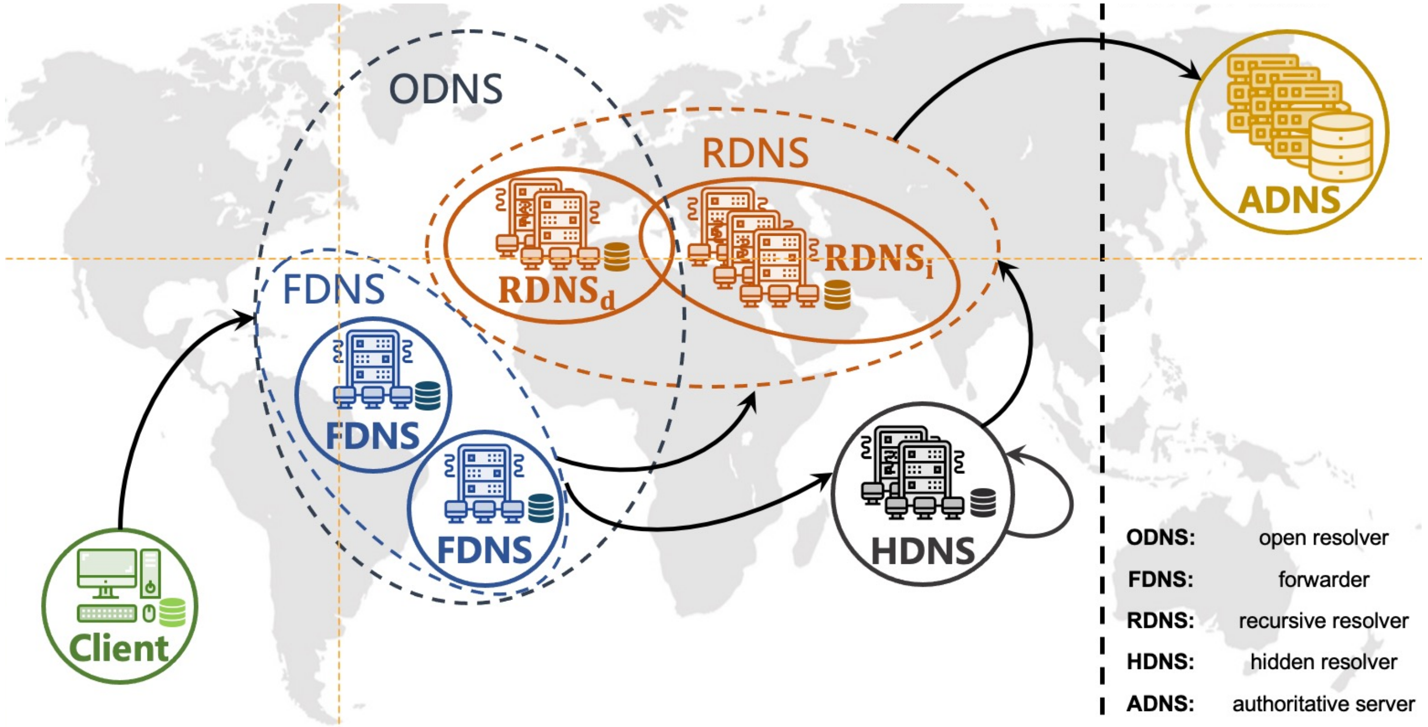


Schomp, et al. On Measuring the Client-side DNS Infrastructure, IMC 2013

# DNS as a complex infrastructure

- Multiple *types* and *layers* of DNS servers
  - DNS forwarders ➔ pass queries to upstream *(e.g., another forwarder)*
  - Large public DNS services ➔ complexes of load balancers, caches, egress servers, etc.

## The complex DNS infrastructure

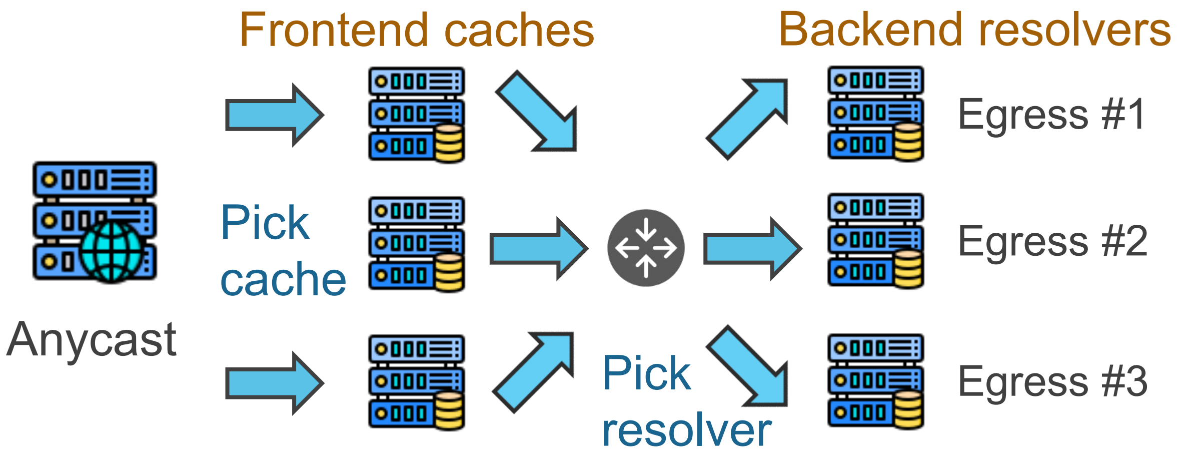


| | |
|---|---|
| **ODNS:** | open resolver |
| **FDNS:** | forwarder |
| **RDNS:** | recursive resolver |
| **HDNS:** | hidden resolver |
| **ADNS:** | authoritative server |

Schomp, et al. On Measuring the Client-side DNS Infrastructure, IMC 2013

## Large public DNS service

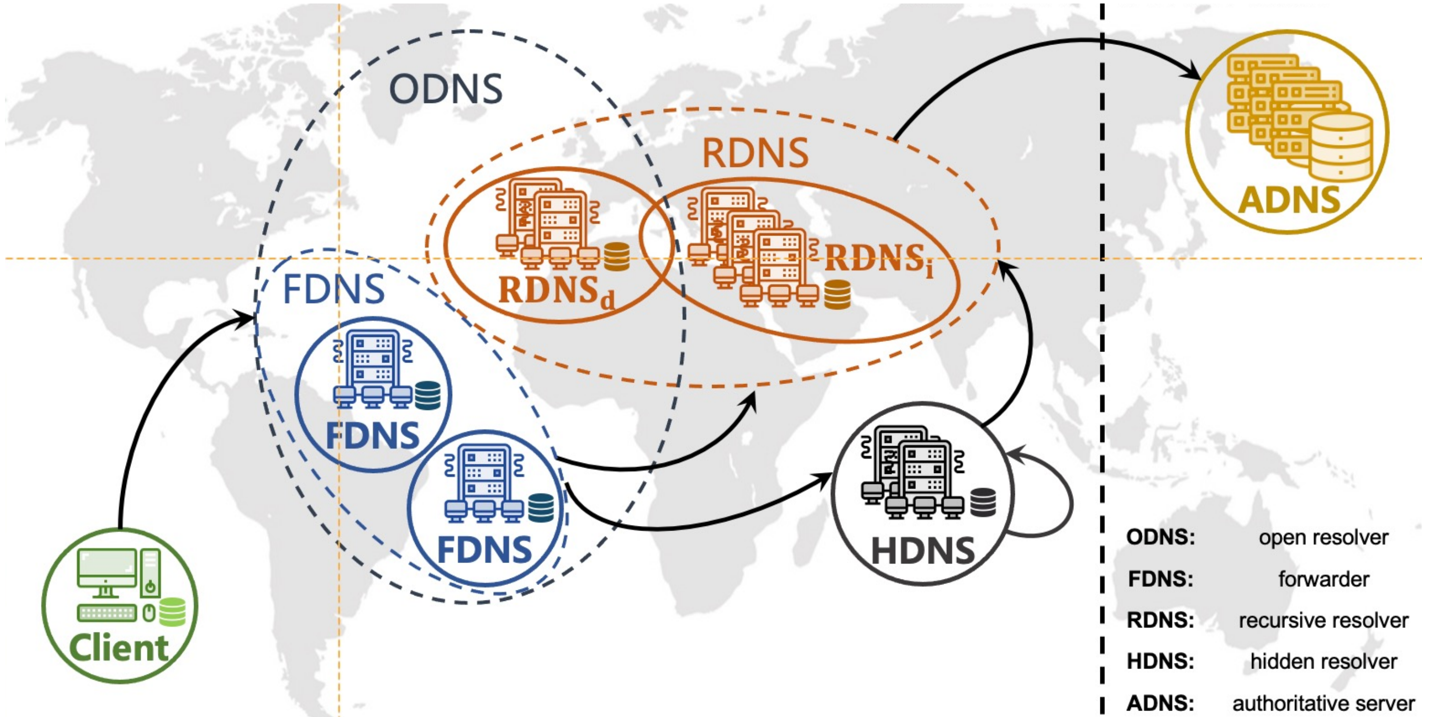
(e.g., Google Public DNS)



# DNS as a complex infrastructure

- **Multiple *types* and *layers* of DNS servers**
  - DNS forwarders ➔ pass queries to upstream *(e.g., another forwarder)*
  - Large public DNS services ➔ complexes of load balancers, caches, egress servers, etc.

## The complex DNS infrastructure

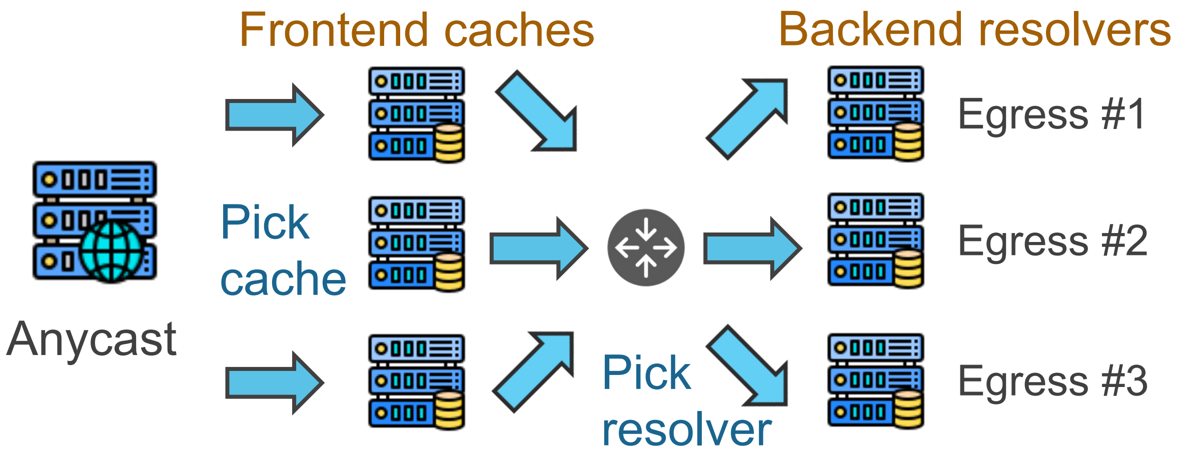


| | |
|---|---|
| **ODNS:** | open resolver |
| **FDNS:** | forwarder |
| **RDNS:** | recursive resolver |
| **HDNS:** | hidden resolver |
| **ADNS:** | authoritative server |

Schomp, et al. On Measuring the Client-side DNS Infrastructure, IMC 2013

## Large public DNS service

(e.g., Google Public DNS)



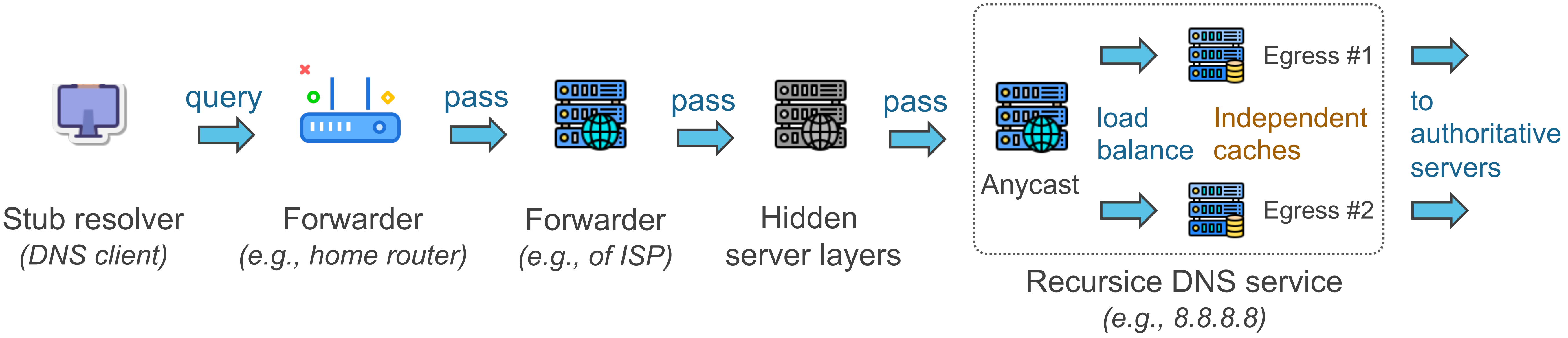
**2.27 Million**

Open DNS servers

* Data from Censys, Oct 2023

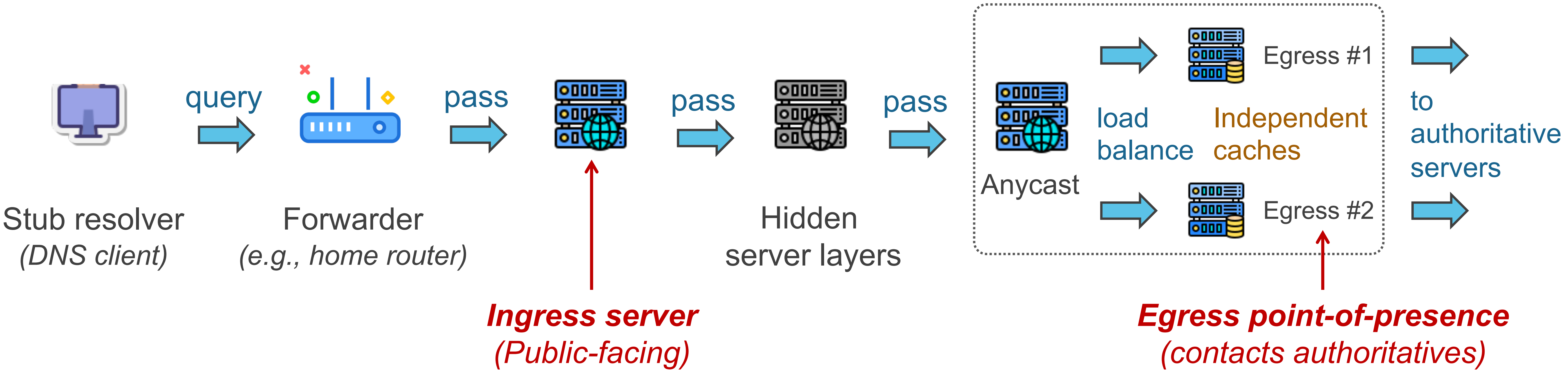
# A typical domain name resolution path

- **Multiple *types* and *layers* of DNS servers**
  - DNS forwarders ➔ pass queries to upstream *(e.g., another forwarder)*
  - Large public DNS services ➔ complexes of load balancers, caches, egress servers, etc.
- **A *typical* DNS resolution path now looks like this**



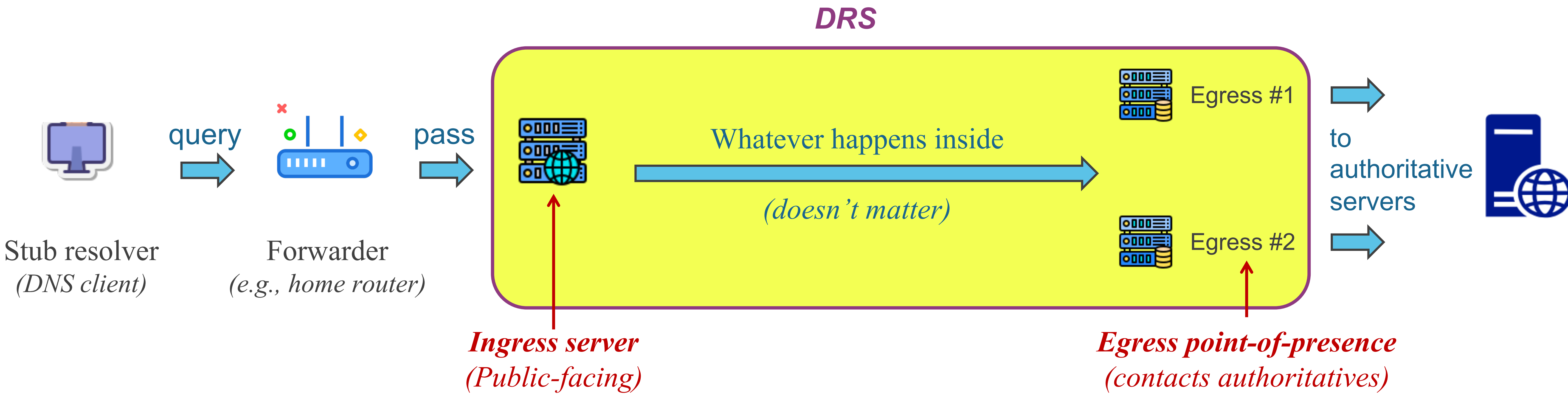
# DNS as a complex infrastructure

- **Multiple *types* and *layers* of DNS servers**
  - DNS forwarders ➔ pass queries to upstream *(e.g., another forwarder)*
  - Large public DNS services ➔ complexes of load balancers, caches, egress servers, etc.
- **A *typical* DNS resolution path now looks like this**



# Definition of DNS Resolver System(DRS)

- **DNS resolver system (DRS)**
  - A public-facing DNS server, together with everything between it and authoritative servers
- **Black box inside**



# DNS as a complex infrastructure

## OK, I get it.
## DNS resolver is a complex system.

But how is this relevant to traffic amplifcation?

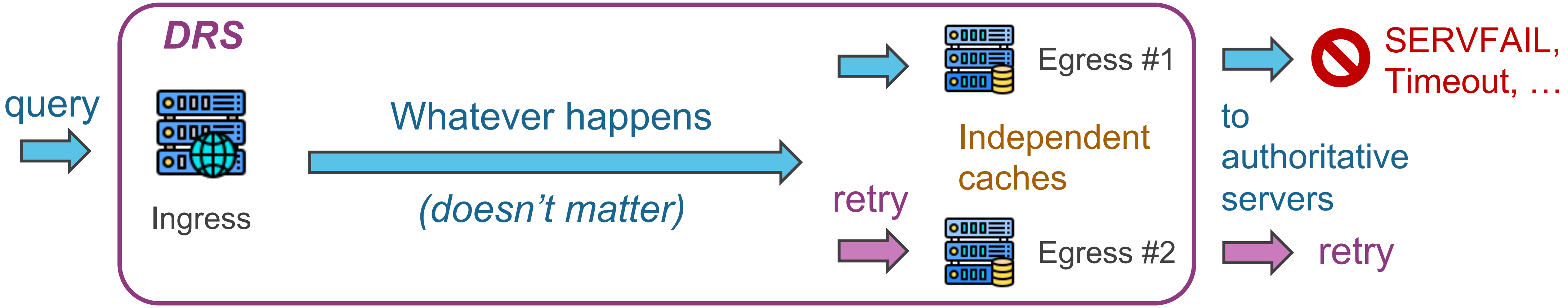
# Amplification ability: DNS retries

- **DNS query could fail for variety of reasons**
  - Packet lost, server fail, routing problems
- **So upon failure, please *retry* for a few more times**
  - Adopted by mainstream DNS software
  - ***THE amplification potential exploited by our attack***

| DNS software | # of retries |
|---|---|
| BIND9 | 13 |
| Unbound | 9 |
| Knot | 3 |

# Amplification ability: DNS retries

- **For a DRS, retries may exit from *different egresses***
  - Egress servers don't share cache
  - Prevents *query aggregation* and *cache hits*



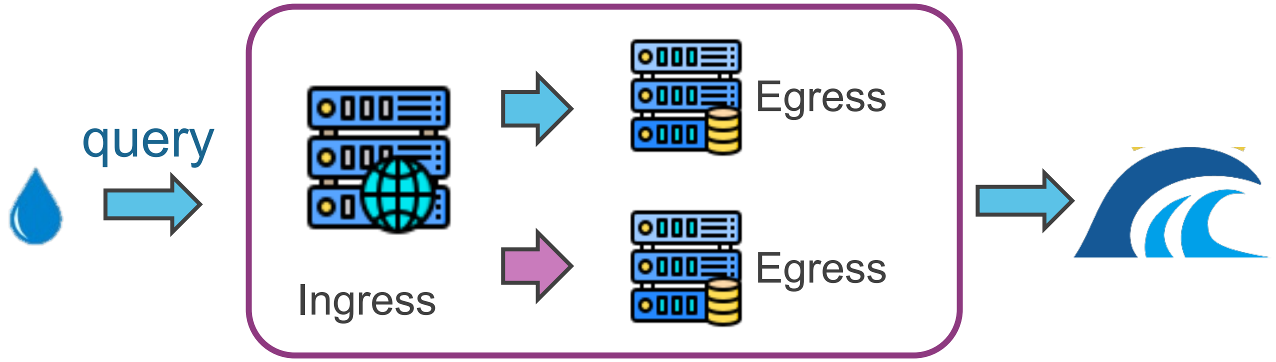
# Amplification ability: DNS retries

## Wait… You exploit retries?

That’s not even enough
to cause ripples!

# Attack variant I: DNS-Retry

- Some bogus DRS implementations that retry aggressively
- In 1.3M DRS, 2.4% (>30,000) retry more than 100 times
- 529 DRSes retry more than 1,000 times
- Max # of retries by one DRS: ***117,541***



Amplification by one DRS only is big enough

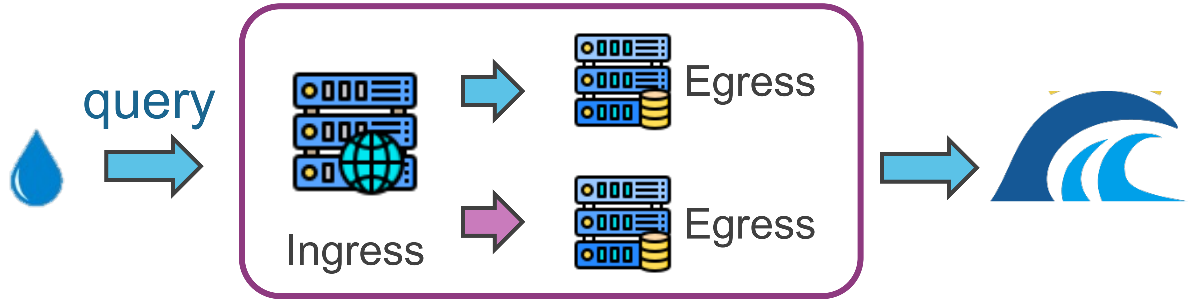
In 1.3M open DNS Resolver System(DRS)

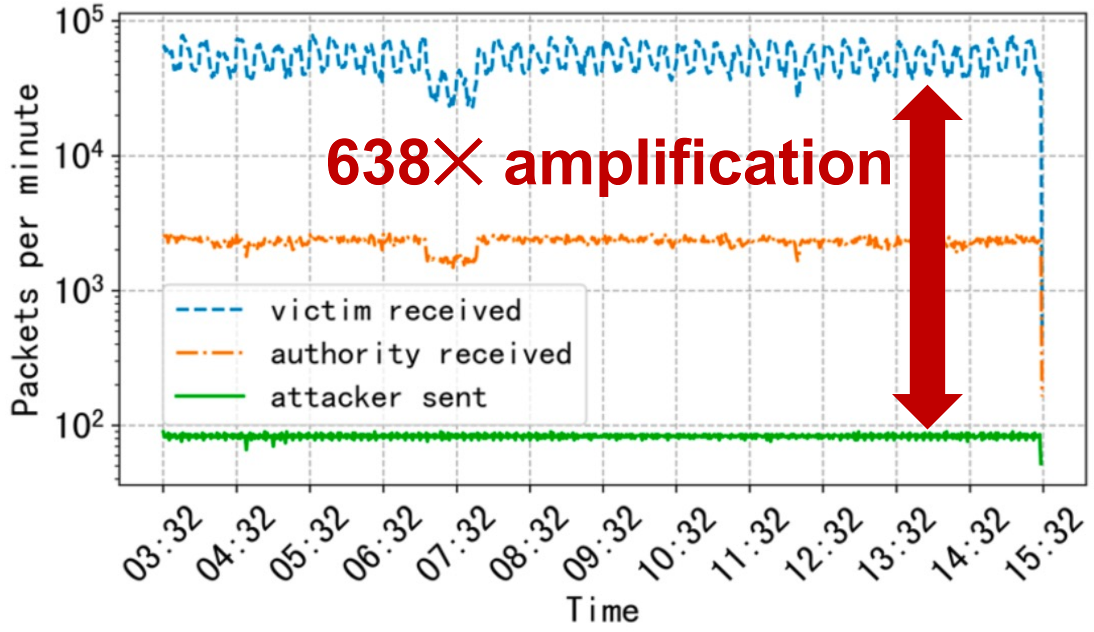
| # of retries | # of open DRSes | % of tested |
|---|---|---|
| > 2 | 925,500 | 69.8% |
| > 10 | 407,581 | 30.7% |
| > 100 | 31,660 | 2.4% |
| **> 1,000** | **529** | **0.04%** |

# DNS-Retry Evaluation

## ❖ Evaluation in controlled environment

- ❖ Select 10 DRSes that retry aggresively
- ❖ Attacker sends 1.3 pkt/s ➔ **Victim receives 882 pkt/s**





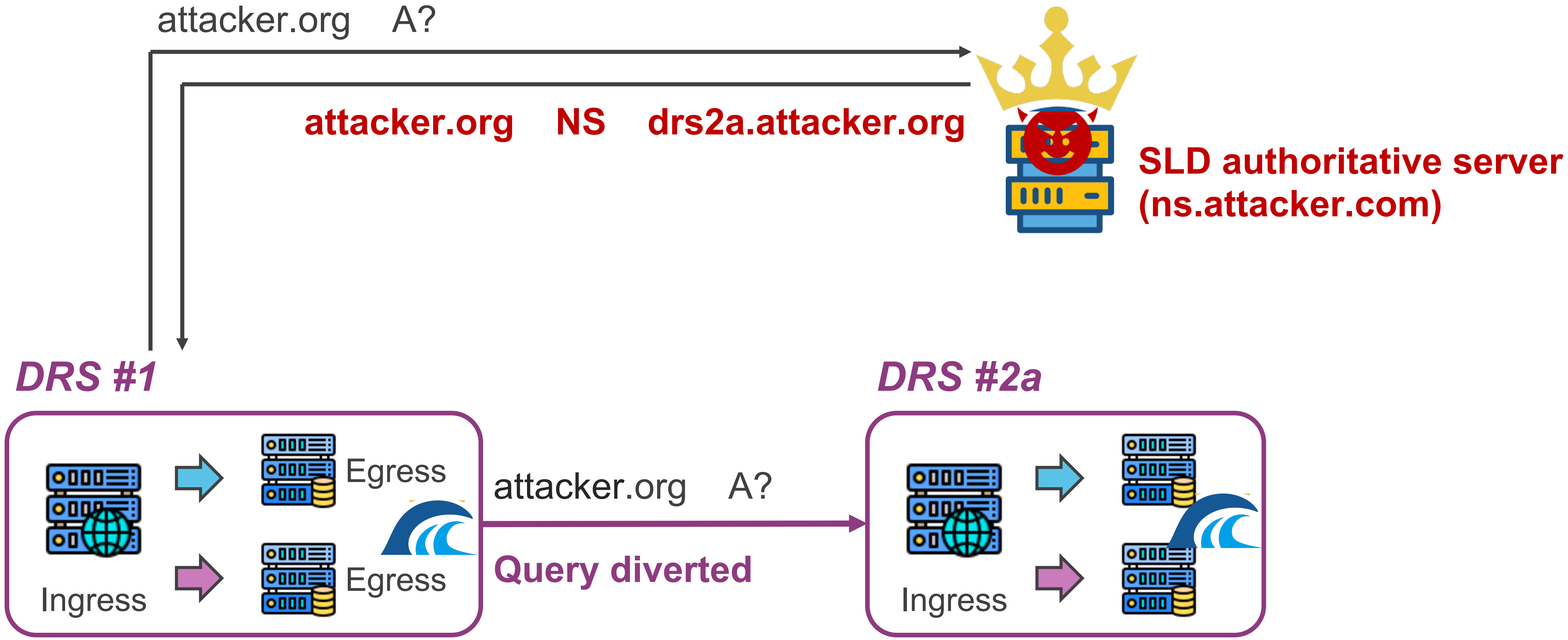
638✕ amplification

# Alright, but lots of them are not aggressive at all.
# Only modest retries…

# Let’s *chain* these ripples into bigger waves!

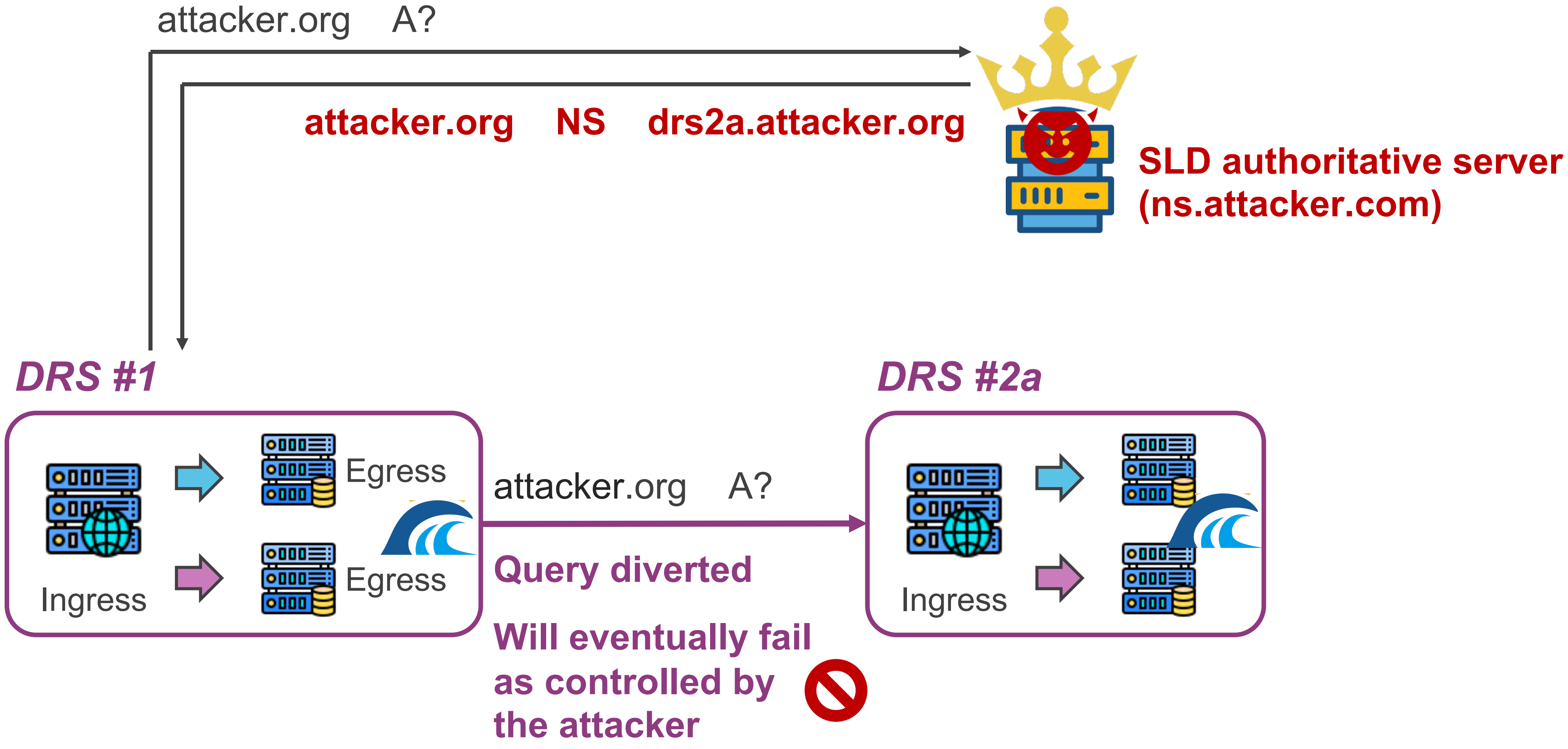
# Attack variant II: DNS-Chain

❖ Recursive DNS resolution guided by ***evil referrals***



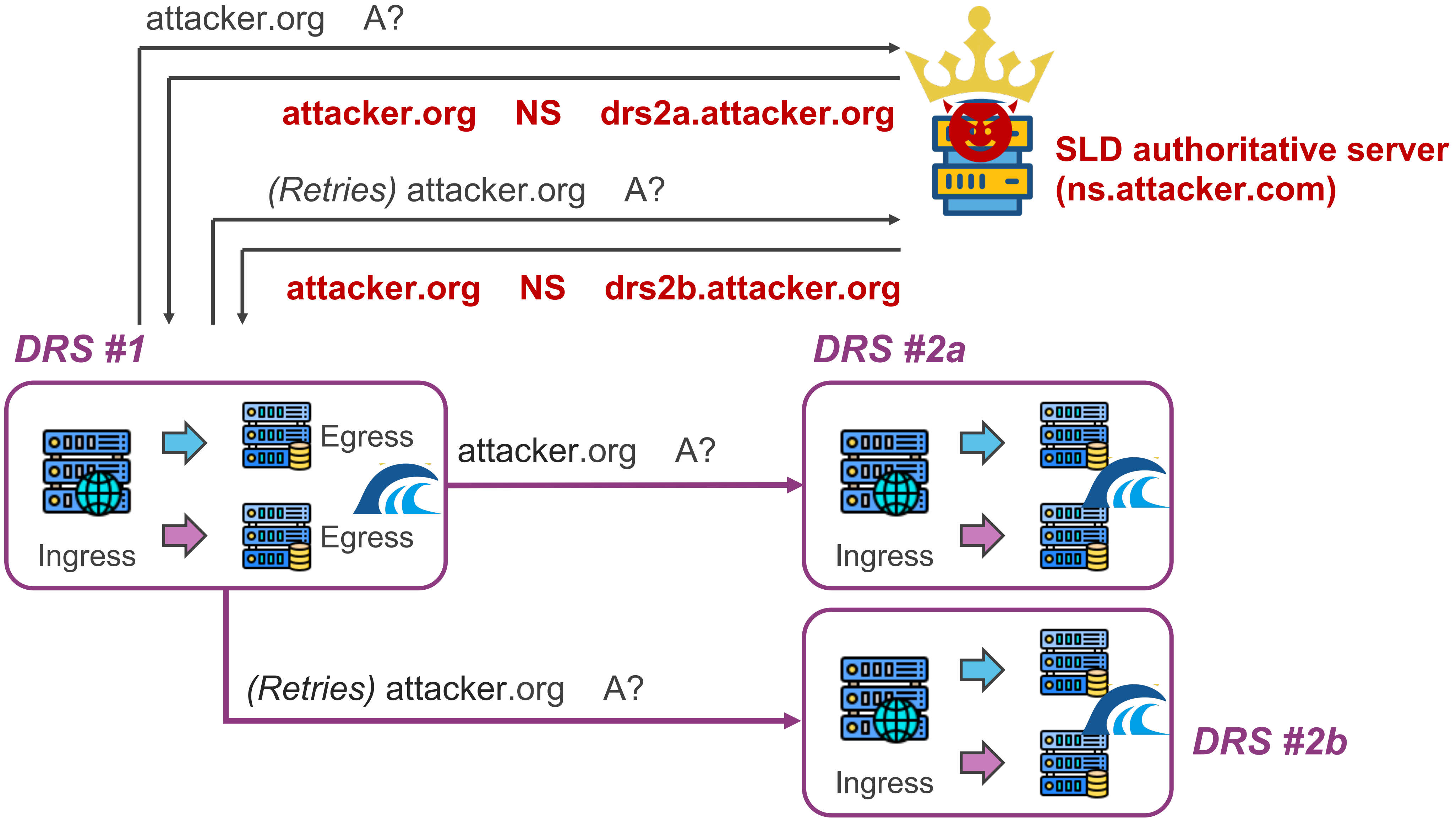
# Attack variant II: DNS-Chain

❖ Recursive DNS resolution guided by *evil referrals*



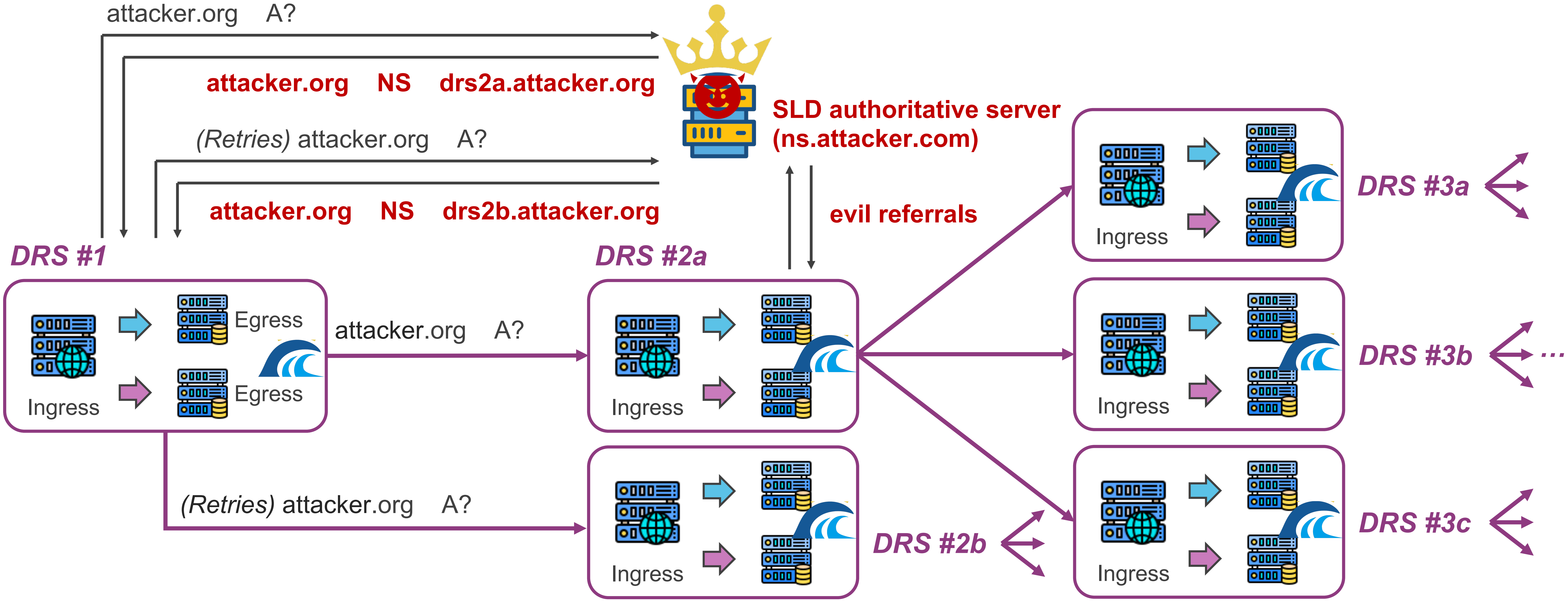
# Attack variant II: DNS-Chain

❖ Recursive DNS resolution guided by ***evil referrals***


attacker.org A?
**attacker.org NS drs2a.attacker.org**
*(Retries)* attacker.org A?
**attacker.org NS drs2b.attacker.org**
**SLD authoritative server (ns.attacker.com)**
***DRS #1***
Egress
Egress
Ingress
attacker.org A?
***DRS #2a***
Ingress
*(Retries)* attacker.org A?
Ingress
***DRS #2b***

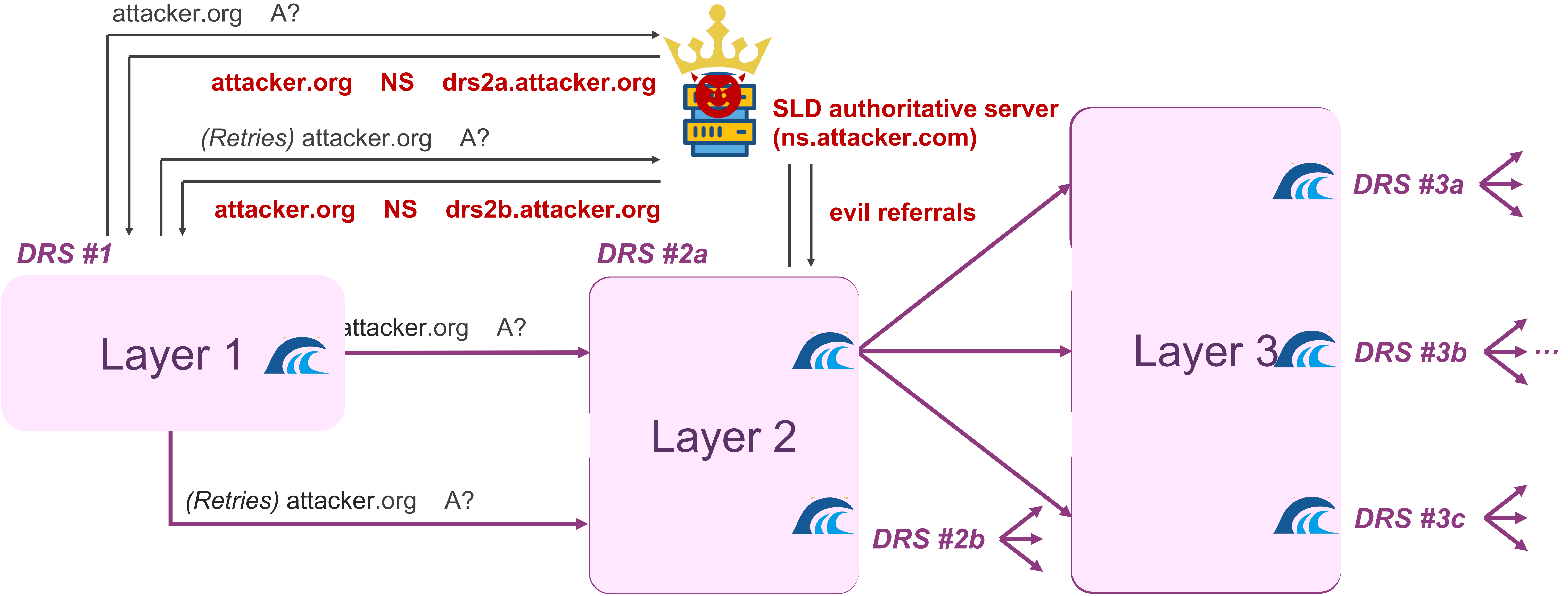
# Attack variant II: DNS-Chain

❖ Recursive DNS resolution guided by *evil referrals*


attacker.org A?
attacker.org NS drs2a.attacker.org
(Retries) attacker.org A?
attacker.org NS drs2b.attacker.org
SLD authoritative server (ns.attacker.com)
evil referrals
DRS #1
Ingress
Egress
Egress
attacker.org A?
DRS #2a
Ingress
(Retries) attacker.org A?
Ingress
DRS #2b
Ingress
DRS #3a
Ingress
DRS #3b
…
Ingress
DRS #3c

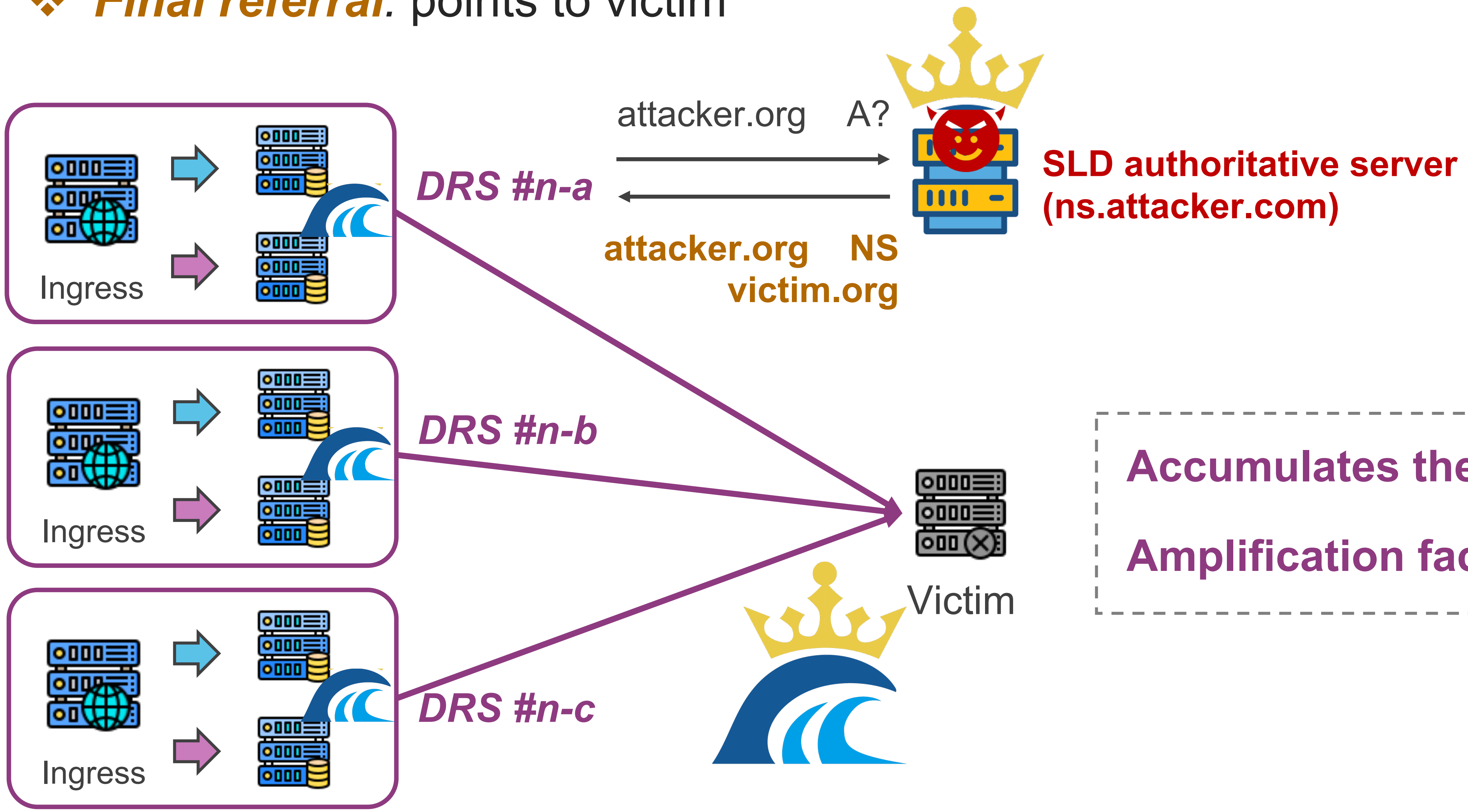
# Attack variant II: DNS-Chain

❖ Recursive DNS resolution guided by ***evil referrals***



# Attack variant II: DNS-Chain

- Recursive DNS resolution guided by *evil referrals*
  - *Final referral:* points to victim



Accumulates the power of layers of DRSes

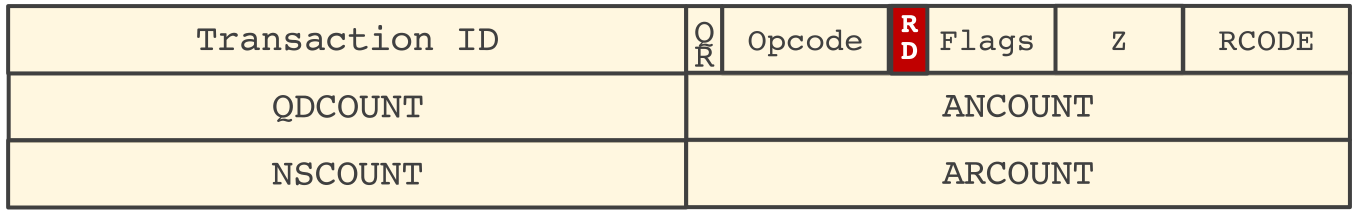
Amplification factor *multiplies*

# Seems plausible,
# but can many DRSes be used?

What are the conditions of successful attacks?

# Conditions of successful attacks

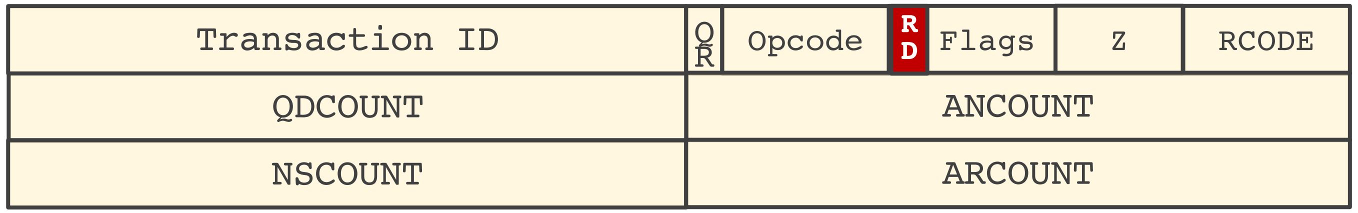
- **DRS *not honoring cleared RD bit* in DNS header**
  - RD (recursion desired) =0: *do not perform recursion, find answers locally in cache*
  - Usually *cleared by egress*, as authoritative servers cannot perform recursion
  - DRS honors RD ➔ *chain cannot continue*
  - ***27.2% of tested DRSes do not honor***



| | | | | | | |
|---|---|---|---|---|---|---|
| Transaction ID | QR | Opcode | RD | Flags | Z | RCODE |
| QDCOUNT | ANCOUNT | | | | | |
| NSCOUNT | ARCOUNT | | | | | |

# Conditions of successful attacks

- **DRS *not honoring cleared RD bit* in DNS header**
  - RD (recursion desired) =0: *do not perform recursion, find answers locally in cache*
  - Usually *cleared by egress*, as authoritative servers cannot perform recursion
  - DRS honors RD ➔ *chain cannot continue*
  - ***27.2% of tested DRSes do not honor***

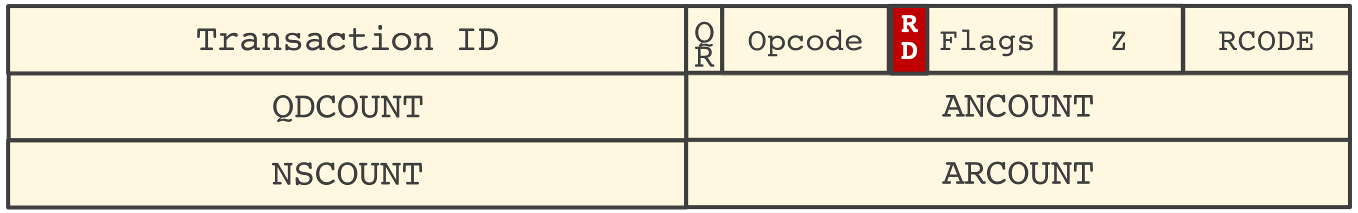


| | | | | | | |
|---|---|---|---|---|---|---|
| Transaction ID | QR | Opcode | RD | Flags | Z | RCODE |
| QDCOUNT | ANCOUNT | | | | | |
| NSCOUNT | ARCOUNT | | | | | |

- **DRS not deployed with negative caching [RFC 2308]**
  - Negative caching records DNS failures ➔ *effectively eliminates retries*
  - ***43% of tested DRSes do not deploy***

# Conditions of successful attacks

- **DRS *not honoring cleared RD bit* in DNS header**
  - RD (recursion desired) =0: *do not perform recursion, find answers locally in cache*
  - Usually *cleared by egress*, as authoritative servers cannot perform recursion
  - DRS honors RD ➔ *chain cannot continue*
  - ***27.2% of tested DRSes do not honor***



<table>
<tr><td>Transaction ID</td><td>QR</td><td>Opcode</td><td>RD</td><td>Flags</td><td>Z</td><td>RCODE</td></tr>
<tr><td>QDCOUNT</td><td colspan="6">ANCOUNT</td></tr>
<tr><td>NSCOUNT</td><td colspan="6">ARCOUNT</td></tr>
</table>

- **DRS not deployed with negative caching [RFC 2308]**
  - Negative caching records DNS failures ➔ *effectively eliminates retries*
  - ***43% of tested DRSes do not deploy***
- **DRS has multiple egresses: *the more, the better***
  - ***52% of tested DRSes have over 10 egresses***

# Evaluation of DNS-Chain

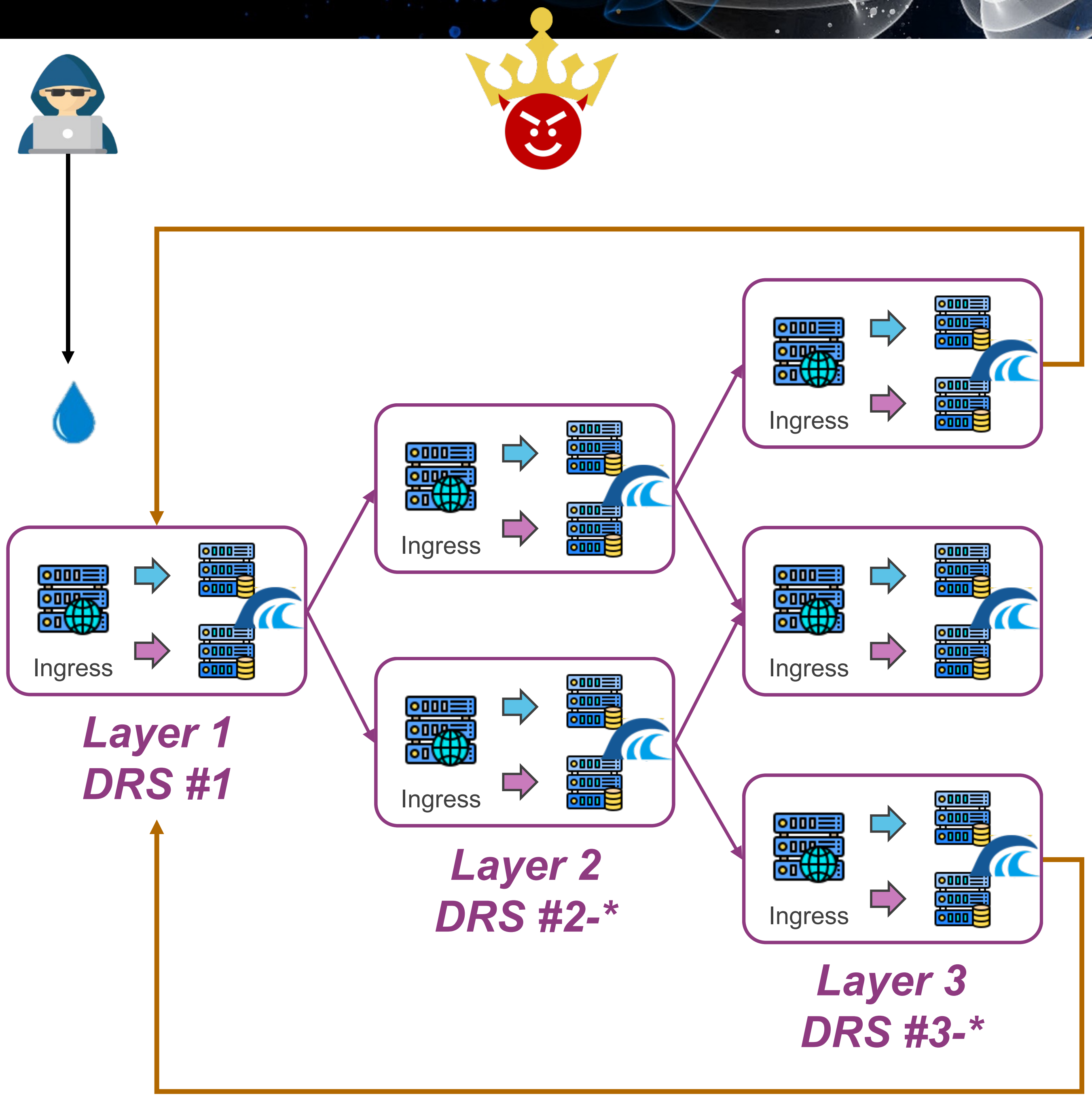
## ❖ Evaluation in controlled environment

- ❖ We select from exploitable DRSes and coordinate them into ***layers***

| Setting | # of DRSes coordinated in each layer | | | | | | | Amp. factor |
|---|---|---|---|---|---|---|---|---|
| | Layer 1 | Layer 2 | Layer 3 | Layer 4 | Layer 5 | Layer 6 | Layer 7 | |
| **# 1** | 1 | 4 | 8 | - | - | - | - | 288 |
| **# 2** | 1 | 4 | 8 | 16 | 32 | - | - | 591 |
| **# 3** | 1 | 4 | 8 | 16 | 32 | 64 | 128 | **3,702** |

# Attack variant III: DNS-Loop

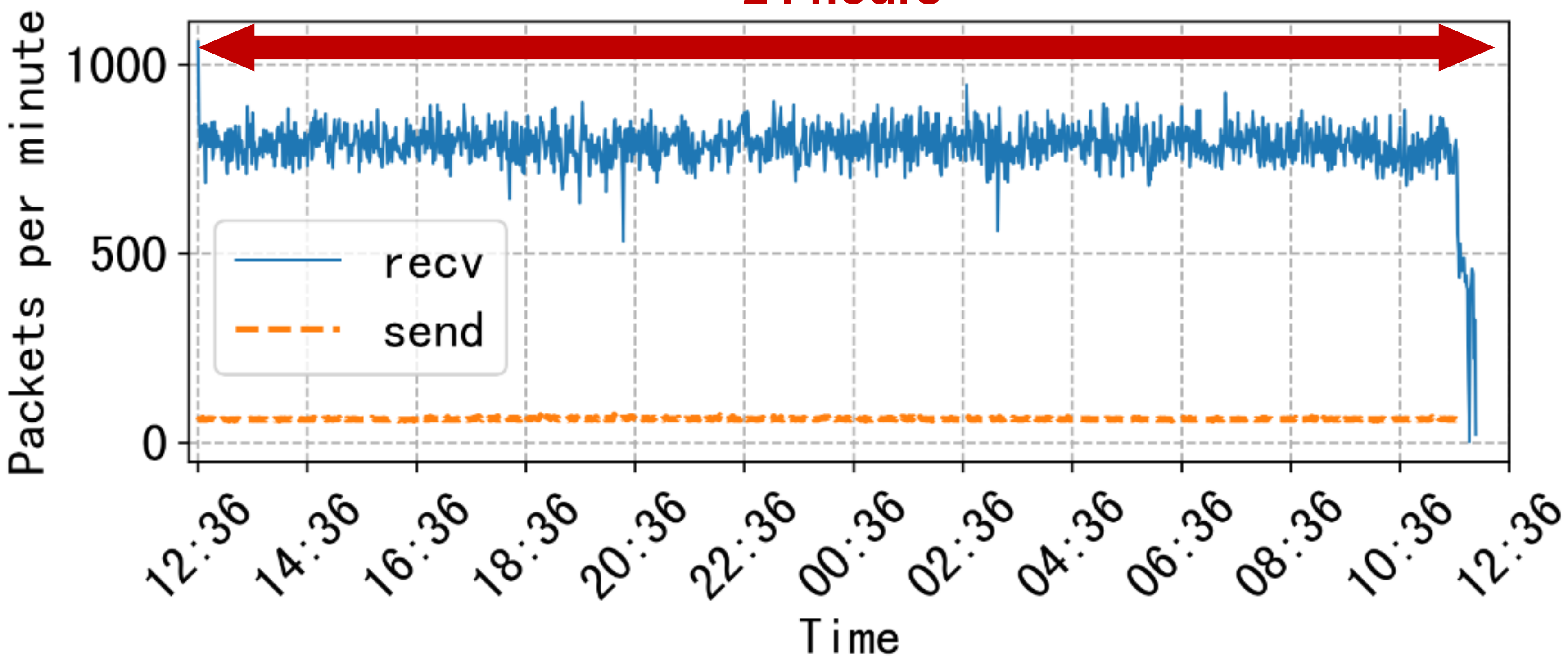
- Modified from DNSChain, creating a *loop* of retry queries
  - ***Final referral**:* points back to DRS #1
- The victim and goal change now
  - ***ALL DRSes in the loop*** become victims
  - Goal is to exhaust their resources
  - *Increasing amplification factor is a non-goal*
- Attackers may also
  - Inject new rounds of retries to the loop
  - Simply by querying DRS #1



# DNS-Loop Evaluation

- **Evaluation in controlled environment - can the loop last?**
  - Coordinates 7 layers of DRSes in the real network
  - layer #0 is our server, with *rate limit at 1 pkt/s(due to ethical considerations)*
  - Send only one DNS query Layer 0, to trigger the loop
  - ***Loop lasts for 24 hours until deliberate stop***

**24 hours**



# Mitigation

## What can we do to prevent this attack?

Correct bogus implementations such that attack conditions cannot be fulfilled.

# Causes

## Tsu-King

### Tsunami

*(Traffic amplification ability)*

- ❖ Cause 1: complex infrastructure
- ❖ Cause 2: aggressive retries

### King

*(Server coordination ability)*

- ❖ Cause 3: not following specifications

  (RD flag, negative cache)

# Mitigations

- **Avoid aggressive retries**
  - A modest number of retries should suffice, as adopted by mainstream software
- **Follow DNS specifications**
  - Honor the DNS flags: if RD tells not to perform recursion, just don’t
- **Deploy additional mechanisms that add protection**
  - Negative caching: good to reduce retries
  - Egress and cache management: reduce independence between egress servers

# Acknowledgement

## DNS Software Vendors

unbound

POWERDNS

MIKROTIK

## DNS service providers

AliDNS 公共解析服务

114DNS

腾讯云 DNSPOD

360安全DNS

# Questions?

**Paper website: https://tsuking.net**

**Contributors of the slides:**

- Wei Xu (xu-w21@mails.tsinghua.edu.cn)
- Xiang Li (x-l19@mails.tsinghua.edu.cn)
- Chaoyi Lu (luchaoyi@tsinghua.edu.cn)
- Haixin Duan (duanhx@tsinghua.edu.cn)