# TsuNAME: exploiting misconfiguration and vulnerability to DDoS DNS

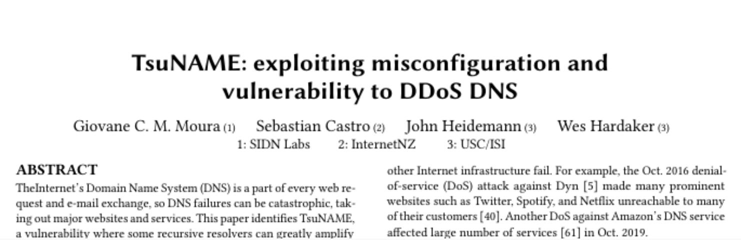
**Giovane C. M. Moura**[1], Sebastian Castro[2],
John Heidemann[3], Wes Hardaker[3]

1: SIDN Labs, 2: InternetNZ, 3: USC/ISI

**IETF 112**
**MAPRG**
*Virtual Meeting*
2021-11-09

# Prelude

1. Our paper appeared at ACM IMC 2021:
   - PDF: https://www.isi.edu/~johnh/PAPERS/Moura21b.pdf



## TsuNAME: exploiting misconfiguration and vulnerability to DDoS DNS

Giovane C. M. Moura (1) Sebastian Castro (2) John Heidemann (3) Wes Hardaker (3)
1: SIDN Labs 2: InternetNZ 3: USC/ISI

### ABSTRACT

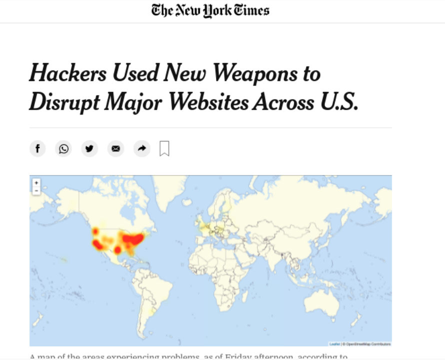
TheInternet's Domain Name System (DNS) is a part of every web request and e-mail exchange, so DNS failures can be catastrophic, taking out major websites and services. This paper identifies TsuNAME, a vulnerability where some recursive resolvers can greatly amplify

other Internet infrastructure fail. For example, the Oct. 2016 denial-of-service (DoS) attack against Dyn [5] made many prominent websites such as Twitter, Spotify, and Netflix unreachable to many of their customers [40]. Another DoS against Amazon's DNS service affected large number of services [61] in Oct. 2019.

2. We identify problems and propose solutions for current RFCs:
   - New draft: draft-moura-dnsop-negative-cache-loop

# Introduction

- The DNS is one of the **core** services on the Internet
- People notice it when it **breaks**:
  - 2016 DDoS against Dyn DNS 2016 [1, 6]
    - affected Netflix, Spotify, Airbnb, Reddit, and others.
  - 2019 DDoS against Amazon AWS [7]

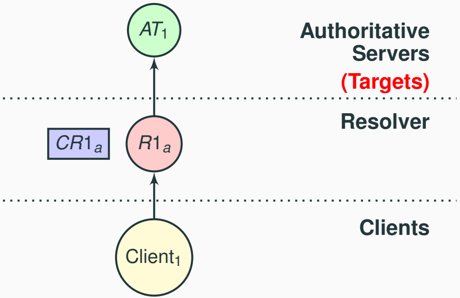


The New York Times

*Hackers Used New Weapons to Disrupt Major Websites Across U.S.*

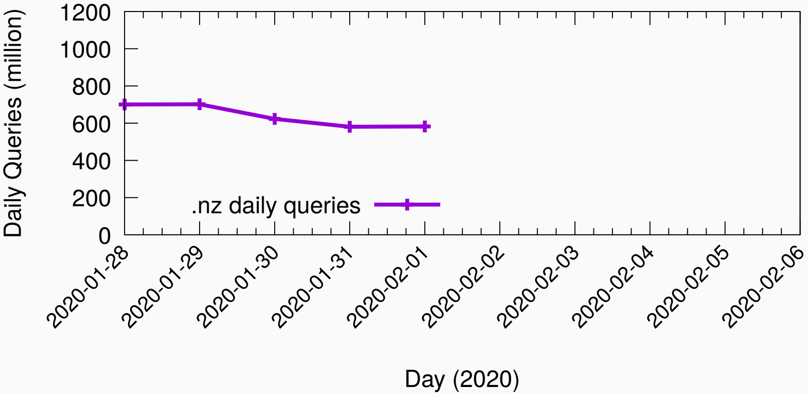
A map of the areas experiencing problems, as of Friday afternoon, according to

# Two main type of DNS servers



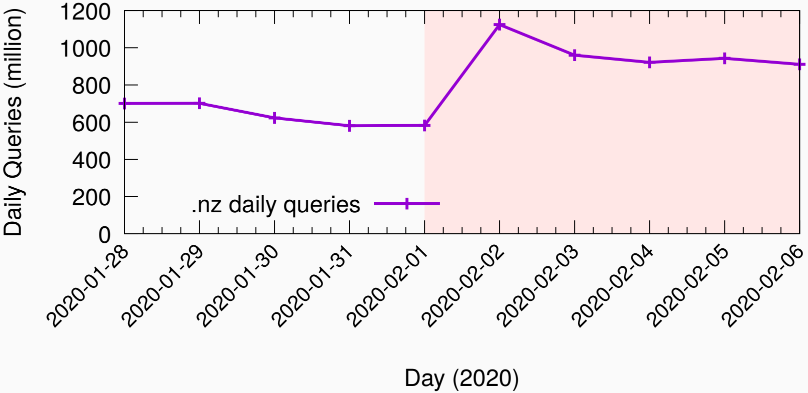
TsuNAME affects **traffic to authoritative servers**

# New Zealand's `.nz` event



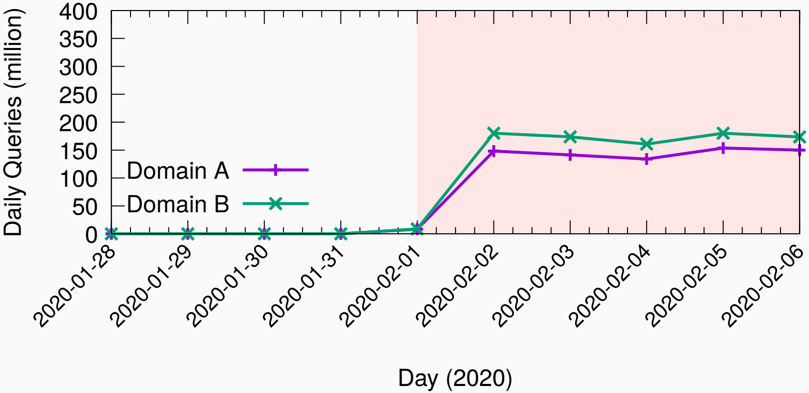
- Normal traffic on `.nz` authoritative servers

# Big traffic increase



- Operators see something strange:
  - 50 % traffic **surge** on `.nz` authoritative servers

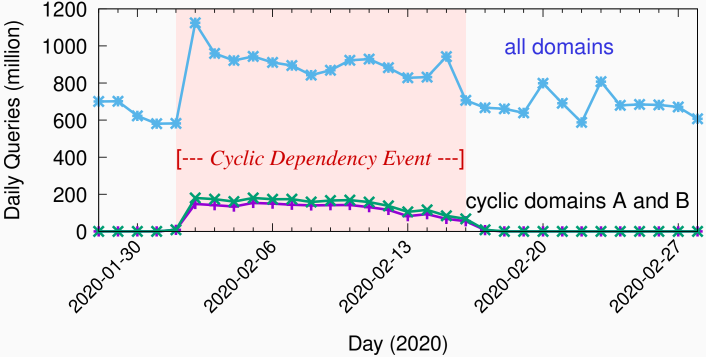
# New Zealand's `.nz` event: an accident?



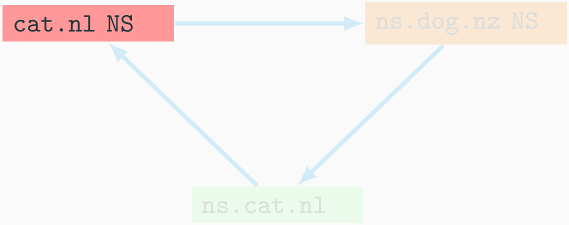
- Two domain names suddenly start to receive millions of queries
- **a DDoS attack?**

# Cause: DNS Loops (cyclic dependency)

## Loop: domainA → domainB → domainA

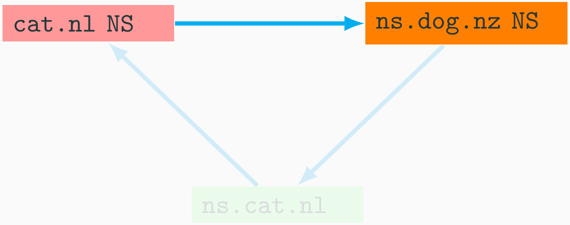


# Cyclic Dependency is a loop; an error



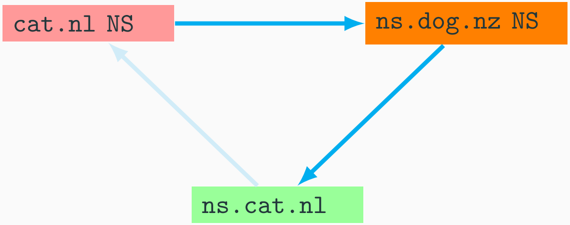
- Described in RFC1536, and later in Pappas2004 [5]
- Such names can never be resolved

# Cyclic Dependency is a loop; an error



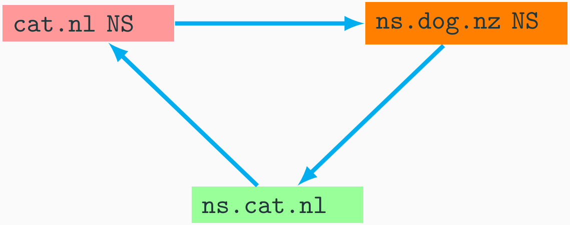
- Described in RFC1536, and later in Pappas2004 [5]
- Such names can never be resolved

# Cyclic Dependency is a loop; an error



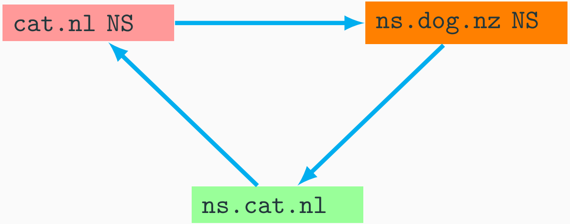
- Described in RFC1536, and later in Pappas2004 [5]
- Such names can never be resolved

# Cyclic Dependency is a loop; an error



- Described in RFC1536, and later in Pappas2004 [5]
- Such names can never be resolved

# Cyclic Dependency is a loop; an error



- Described in RFC1536, and later in Pappas2004 [5]
- Such names can never be resolved

# Contributions

1. **Understanding**: show how TsuNAME can be weaponized (§3 and §4)
2. **Prevention**: provide tool for DNS ops (§5)
   - `CycleHunter`: so they can detect loops in their zones
   - identifying what's missing in RFCs
3. **Fixing Bugs** (§6):
   - Responsible disclosure
   - **Google** fixed their Public DNS 🙂
   - **Cisco** fixed OpenDNS 🙂

# Contributions

1. **Understanding**: show how TsuNAME can be weaponized (§3 and §4)
2. **Prevention**: provide tool for DNS ops (§5)
    - `CycleHunter`: so they can detect loops in their zones
    - identifying what's missing in RFCs
3. **Fixing Bugs** (§6):
    - Responsible disclosure
    - **Google** fixed their Public DNS 🙂
    - **Cisco** fixed OpenDNS 🙂

# Contributions

1. **Understanding**: show how TsuNAME can be weaponized (§3 and §4)
2. **Prevention**: provide tool for DNS ops (§5)
   - `CycleHunter`: so they can detect loops in their zones
   - identifying what's missing in RFCs
3. **Fixing Bugs** (§6):
   - Responsible disclosure
   - **Google** fixed their Public DNS 🙂
   - **Cisco** fixed OpenDNS 🙂

# The Real Threat: weaponization

- 2 domains in .nz → 50% total traffic surge
- **The threat:**
  - Adversary holds many domains
  - Reconfigure to create loops of NS records
  - Trigger recursive resolvers from a botnet

This got us very **concerned**.

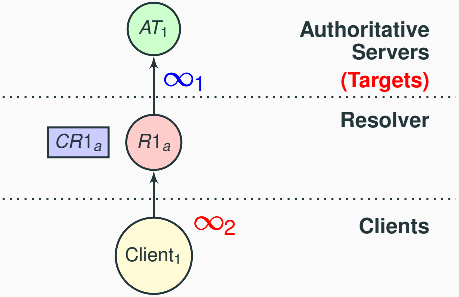
- How many anycast providers/TLDs can withstand that?

# The Real Threat: weaponization

- 2 domains in .nz → 50% total traffic surge
- **The threat:**
  - Adversary holds many domains
  - Reconfigure to create loops of NS records
  - Trigger recursive resolvers from a botnet

  This got us very **concerned**.
- How many anycast providers/TLDs can withstand that?

# TsuNAME in practice: Root Causes

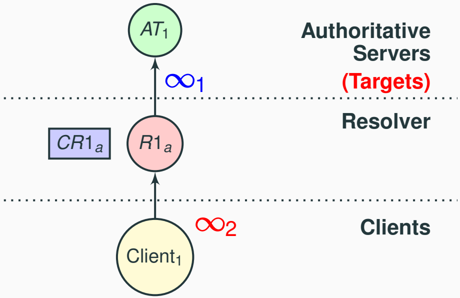


A client sends a query to the recursive. We found three cases:

1. Resolvers that loop indefinitely ($\infty_1$)
2. Clients that loop indefinitely ($\infty_2$)
3. Both

**We will see solutions later**

# TsuNAME in practice: Root Causes

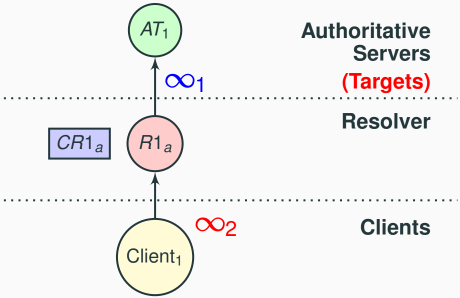


A client sends a query to the recursive. We found three cases:

1. Resolvers that loop indefinitely ($\infty_1$)
2. Clients that loop indefinitely ($\infty_2$)
3. Both

We will see solutions later

# TsuNAME in practice: Root Causes

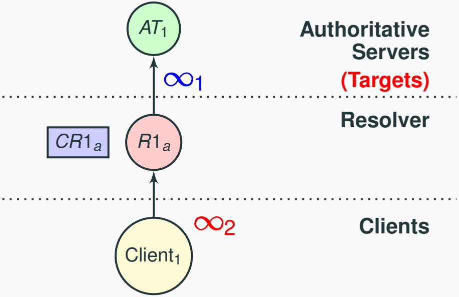


A client sends a query to the recursive. We found three cases:

1. Resolvers that loop indefinitely ($\infty_1$)
2. Clients that loop indefinitely ($\infty_2$)
3. Both

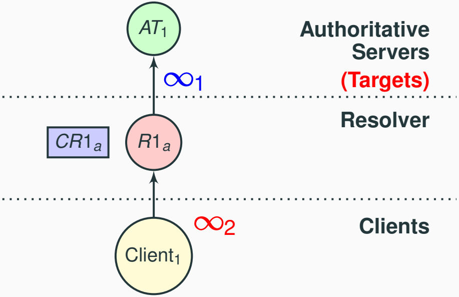
**We will see solutions later**

# Isn’t this a known and solved problem?



1. RFC1034 [3] is very **vague**
    - “resolvers should bound the amount of work” to avoid infinite loops
2. Offers no protection from looping clients ($\infty_2$)
    - amplification is proportional to client query rate

# Isn't this a known and solved problem?

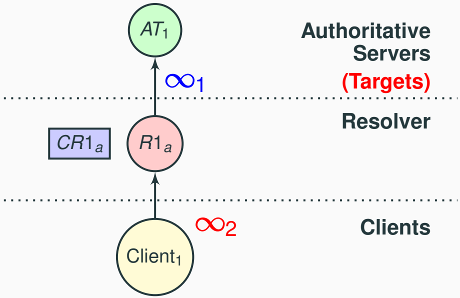


1. RFC1034 [3] is very **vague**
   - "resolvers should bound the amount of work" to avoid infinite loops

Offers no protection from looping clients ($\infty_2$)

- amplification is proportional to client query rate

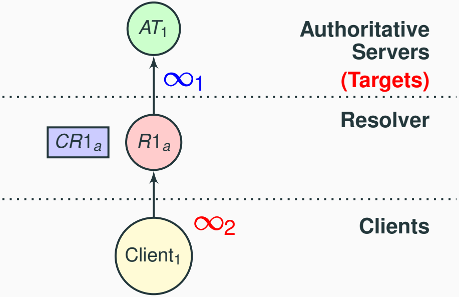
# Isn’t this a known and solved problem?



2. RFC1035 [4] (§7.2) **set counters**:
   - “the resolver should have a global per-request counter to limit work on a single request.”

   Still no protection from looping clients ($\infty_2$)
   - amplification is proportional to client query rate

# Isn't this a known and solved problem?

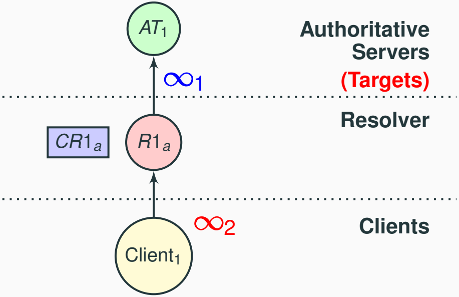


2. RFC1035 [4] (§7.2) **set counters**:
   - "the resolver should have a global per-request counter to limit work on a single request."

   Still no protection from looping clients ($\infty_2$)

   - amplification is proportional to client query rate

# Isn’t this a known and solved problem?

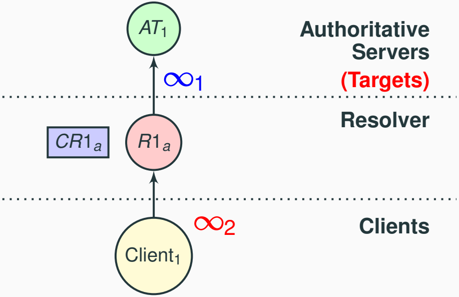


3. RFC1536 [2](§2) warns that **loops can occur** :
   - “a set of servers might form a loop wherein A refers to B and B refers to A”
   - Offers no new solution

Still no protection from looping clients ($\infty_2$)

- amplification is proportional to client query rate

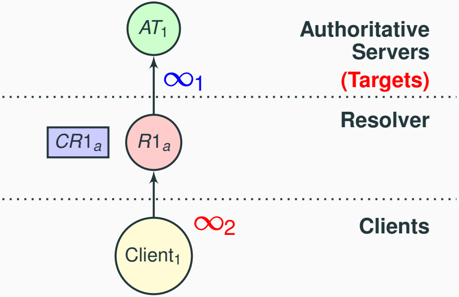
# Isn’t this a known and solved problem?



3. RFC1536 [2](§2) warns that **loops can occur** :
    - “a set of servers might form a loop wherein A refers to B and B refers to A”
    - Offers no new solution

   Still no protection from looping clients ($\infty_2$)
    - amplification is proportional to client query rate

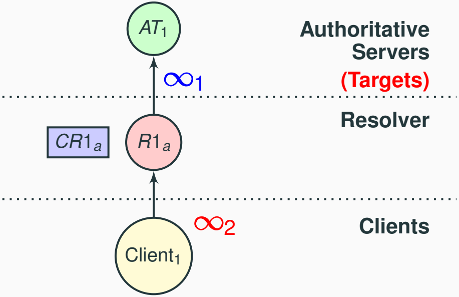
# Solution: detect & cache



**Solution:** detect loops and don't repeat them (**negative caching**)

- Not in any RFC at the moment.
- Resolvers **MUST** cache these looping records
- That minimizes $\infty$1 and prevents $\infty_2$
- draft-moura-dnsop-negative-cache-loop

# Solution: detect & cache



**Solution:** detect loops and don't repeat them (**negative caching**)

- Not in any RFC at the moment.
- Resolvers **MUST** cache these looping records
- That minimizes $\infty 1$ and prevents $\infty_2$
- draft-moura-dnsop-negative-cache-loop

# Reproducing TsuNAME: a controlled experiment

- We run our authoritative servers
- Each Atlas probe sends 1 query
  - to each local resolver
- **Goal**: determine if we can trigger loops with 1 query only
- We collect traffic and analyze it

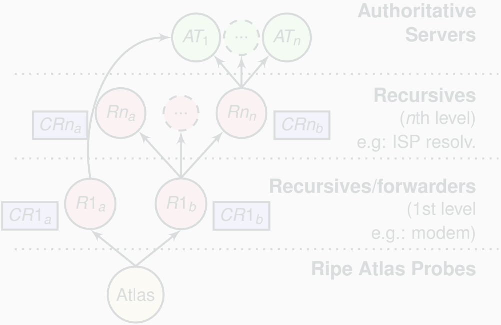


**Figure 1:** Ripe Atlas, Resolvers, and Auth. Servers

# Reproducing TsuNAME: a controlled experiment

- We run our authoritative servers
- Each Atlas probe sends 1 query
  - to each local resolver
- **Goal**: determine if we can trigger loops with 1 query only
- We collect traffic and analyze it

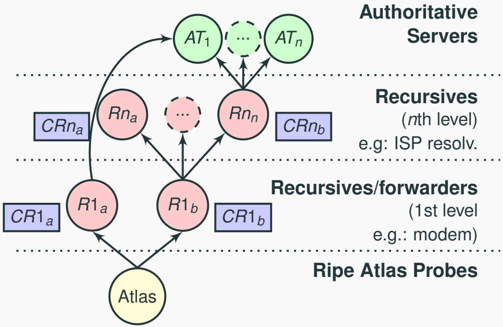
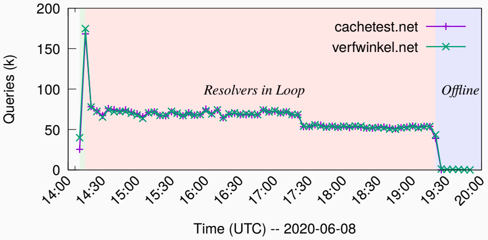
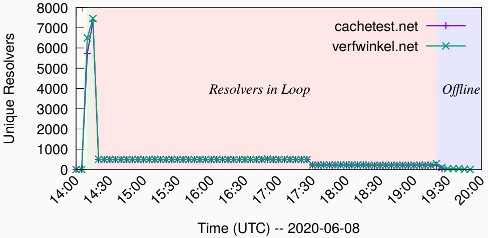


**Figure 1:** Ripe Atlas, Resolvers, and Auth. Servers

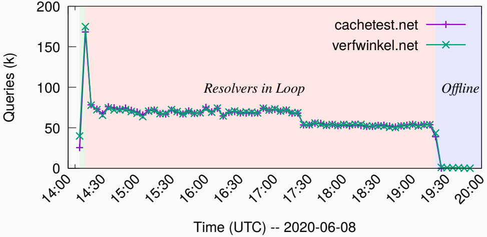
# Reproducing TsuNAME: results

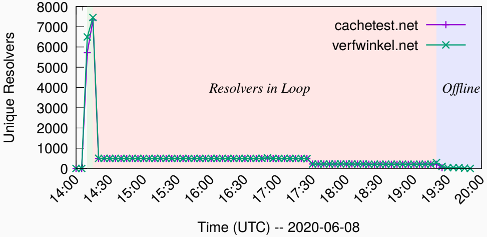




- 574 recursives looped (34 ASes)
  - Including Google Public DNS and Cisco Open DNS
- It lasted for **hours**
- (we had to stop the experiment)
- Paper: more complex scenarios
  - Using non-Atlas vantage points

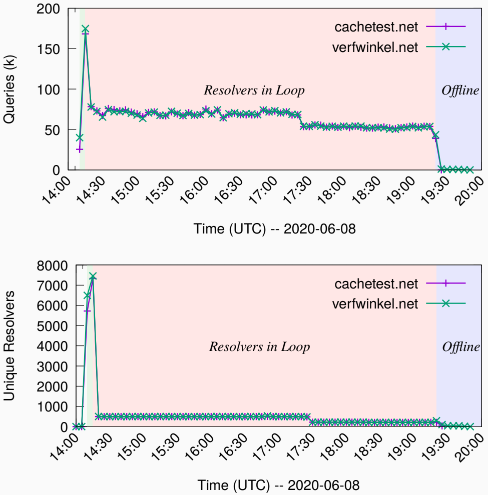
# Reproducing TsuNAME: results





- 574 recursives looped (34 ASes)
  - Including Google Public DNS and Cisco Open DNS
- It lasted for **hours**
- (we had to stop the experiment)
- Paper: more complex scenarios
  - Using non-Atlas vantage points

# Reproducing TsuNAME: results



- 574 recursives looped (34 ASes)
  - Including Google Public DNS and Cisco Open DNS
- It lasted for **hours**
- (we had to stop the experiment)
- Paper: more complex scenarios
  - Using non-Atlas vantage points

# Contributions

1. ~~**Understanding**: show how TsuNAME can be weaponized (§3 and §4)~~
2. **Prevention**: provide tool for DNS ops (§5)
    - `CycleHunter`: so they can detect loops in their zones
    - identifying what's missing in RFCs
3. **Fixing Bugs** (§6):
    - Responsible disclosure
    - **Google** fixed their Public DNS 🙂
    - **Cisco** fixed OpenDNS 🙂

# Prevention: DNS Ops can use `CycleHunter`

To protect Authoritative Servers OPs

- `https://github.com/SIDN/CycleHunter`

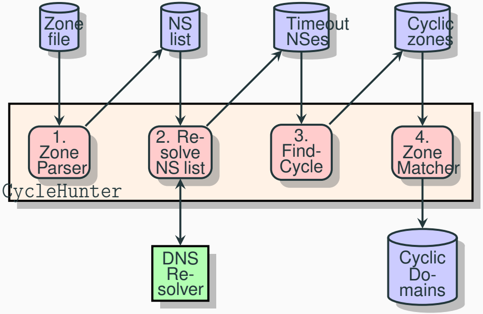


**Figure 2:** `CycleHunter` workflow

# CycleHunter in the wild: not many cyclic domains

| zone | Size | NSSet | Cyclic | Affec. | Date |
|---|---:|---:|---:|---:|---:|
| .com | 151445463 | 2199652 | 21 | 1233 | 2020-12-05 |
| .net | 13444518 | 708837 | 6 | 17 | 2020-12-10 |
| .org | 10797217 | 540819 | 13 | 121 | 2020-12-10 |
| .nl | 6072961 | 79619 | 4 | 64 | 2020-12-03 |
| .se | 1655434 | 27540 | 0 | 0 | 2020-12-10 |
| .nz | 718254 | 35738 | 0 | 0 | 2021-01-11 |
| .nu | 274018 | 10519 | 0 | 0 | 2020-12-10 |
| Root | 1506 | 115 | 0 | 0 | 2020-12-04 |
| **Total** | 184409371 | 3602839 | 44 | 1435 | |

**Table 1:** CycleHunter: evaluated DNS Zones

- Human error plays a role

# Contributions

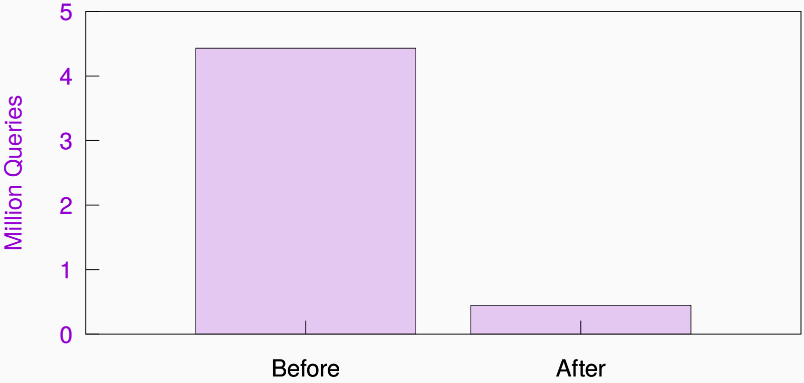
1. ~~**Understanding**: show how TsuNAME can be weaponized (§3 and §4)~~
2. ~~**Prevention**: provide tool for DNS ops (§5)~~
   - ~~`CycleHunter`: so they can detect loops in their zones~~
   - ~~identifying what’s missing in RFCs~~
3. **Fixing Bugs** (§6):
   - Responsible disclosure
   - **Google** fixed their Public DNS 🙂
   - **Cisco** fixed OpenDNS 🙂

# Responsible Disclosure

| Date | Type | Group |
|---|---|---|
| 2020-12-10 | Private Disclosure | Google Notification |
| 2020-12-10 | Private Disclosure | SIDN DNSOPs |
| 2021-02-05 | Private Disclosure | OARC34 |
| 2021-02-22 | Private Disclosure | APTLD |
| 2021-02-22 | Private Disclosure | NCSC-NL |
| 2021-02-23 | Private Disclosure | CENTR |
| 2021-03-04 | Private Disclosure | LACTLD |
| 2021-02-18–2021-05-05 | Private Disclosure | Private |
| 2021-05-06 | Public Disclosure | OARC35 |
| 2021-05-06 | Public Disclosure | `https://tsuname.io` |

**Table 2:** TsuNAME disclosure timeline

# We confirmed Google fixed its Public DNS



# Post-disclosure: A European ccTLD saw it too

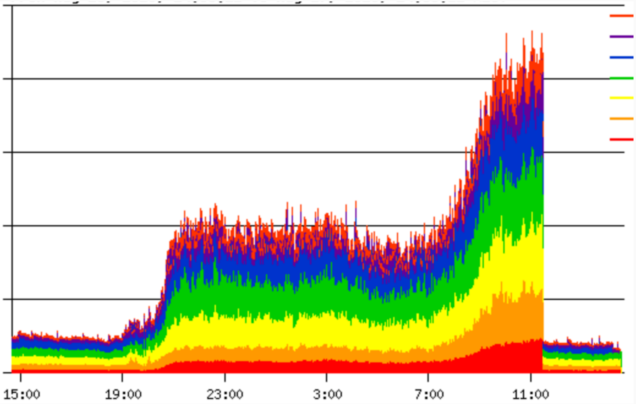


**Figure 3:** TsuNAME event at an EU-based ccTLD operator. **10x traffic growth**

# Contributions

1. ~~**Understanding**: show how TsuNAME can be weaponized (§3 and §4)~~
2. ~~**Prevention**: provide tool for DNS ops (§5)~~
    - ~~`CycleHunter`: so they can detect loops in their zones~~
    - ~~identifying what’s missing in RFCs~~
3. ~~**Fixing Bugs** (§6)~~:
    - ~~Responsible disclosure~~
    - ~~**Google** fixed their Public DNS~~ 🙂 ~~**Cisco** fixed OpenDNS~~ 🙂

# Conclusions

- NS loops are an old problem for DNS
  - we show we **MUST** address it now
- Current standards do not fully address it
  - draft-moura-dnsop-negative-cache-loop
- **What do to**?
  - DNS operators: run `CycleHunter`
  - Developers of DNS resolver: negative caching of loops

https://tsuname.io

# References i

[1] Antonakakis, M., April, T., Bailey, M., Bernhard, M., Bursztein, E., Cochran, J., Durumeric, Z., Halderman, J. A., Invernizzi, L., Kallitsis, M., Kumar, D., Lever, C., Ma, Z., Mason, J., Menscher, D., Seaman, C., Sullivan, N., Thomas, K., and Zhou, Y.

**Understanding the Mirai botnet.**

In *Proceedings of the 26th USENIX Security Symposium* (Vancouver, BC, Canada, Aug. 2017), USENIX, pp. 1093–1110.

[2] Kumar, A., Postel, J., Neuman, C., Danzig, P., and Miller, S.

**Common DNS Implementation Errors and Suggested Fixes.**

RFC 1536, IETF, Oct. 1993.

# References ii

[3] Mockapetris, P.
**Domain names - concepts and facilities.**
RFC 1034, IETF, Nov. 1987.

[4] Mockapetris, P.
**Domain names - implementation and specification.**
RFC 1035, IETF, Nov. 1987.

[5] Pappas, V., Xu, Z., Lu, S., Massey, D., Terzis, A., and Zhang, L.
**Impact of configuration errors on DNS robustness.**
*SIGCOMM Comput. Commun. Rev. 34*, 4 (Aug. 2004), 319–330.

# References iii

[6] Perlroth, N.

**Hackers used new weapons to disrupt major websites across U.S.**

*New York Times* (Oct. 22 2016), A1.

[7] Williams, C.

**Bezos DDoS'd: Amazon Web Services' DNS systems knackered by hours-long cyber-attack.**

https://www.theregister.co.uk/2019/10/22/aws_dns_ddos/, 10 2019.