RRL -- Strategies for a Successful Deployment

November 2013

ISC & DNSco

Welcome!

- Presentation – 45 minutes
  - Interactive Question & Answer Format
- All attendees are on **mute**
- Q&A during the session
  - Use WebEx chat window to submit questions
  - In the interest of time, please email unanswered questions to **info@isc.org**
- A recording of this event will be sent to all registered attendees

Agenda

- ISC and DNSco
- RRL Question & Answer
  - Configuration
  - Utilizing Log Files
  - Gotchas
  - Additional Classifier Options
- Summary

Presenters

- Eddy Winstead, Senior Systems Engineer
- Peter Losher, Senior Operations Architect

ISC at a Glance

Sponsored R&D	Open Home Gateway Open Source Routing	ISC
Public Benefit	Hosted@ F-Root Open Source Software	ISC
Commercial Services	Subscription Services DNS Hosting Training	DNS co

RRL OVERVIEW

Response Rate Limiting

- **An Enhancement to the DNS**
  - A mechanism for limiting the number of unique responses returned by a DNS server
  - A mitigation tool for the problem of DNS Amplification Attacks
  - The only practical defense available for filtering in the name server
    - **BIND 9.9.4** includes RRL as a key feature
      - Available for download at https://www.isc.org/downloads/

Normal Traffic

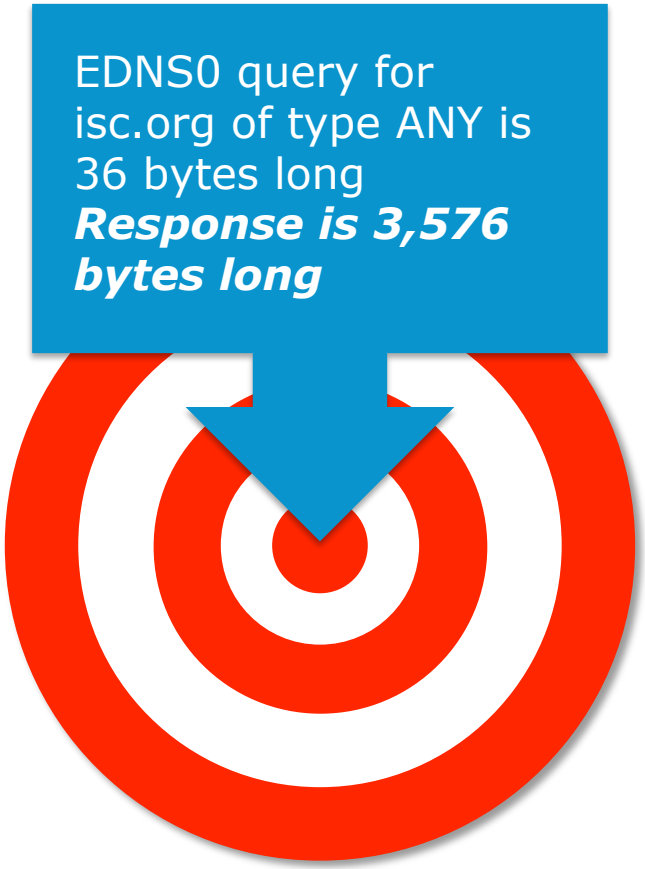
rDoS Attack


5.6.7.8
Source IP Address:
5.6.7.8
1.2.3.4
Internet
Responds to 1.2.3.4
1.2.3.4

ISC'S EXPERIENCE

How did RRL come about?

- ISC signed our zones in 2006
- Observed queries that were occurring too frequently from the same IP
- Defensive strategy sessions with Paul led to RRL



Accidental? Enemies

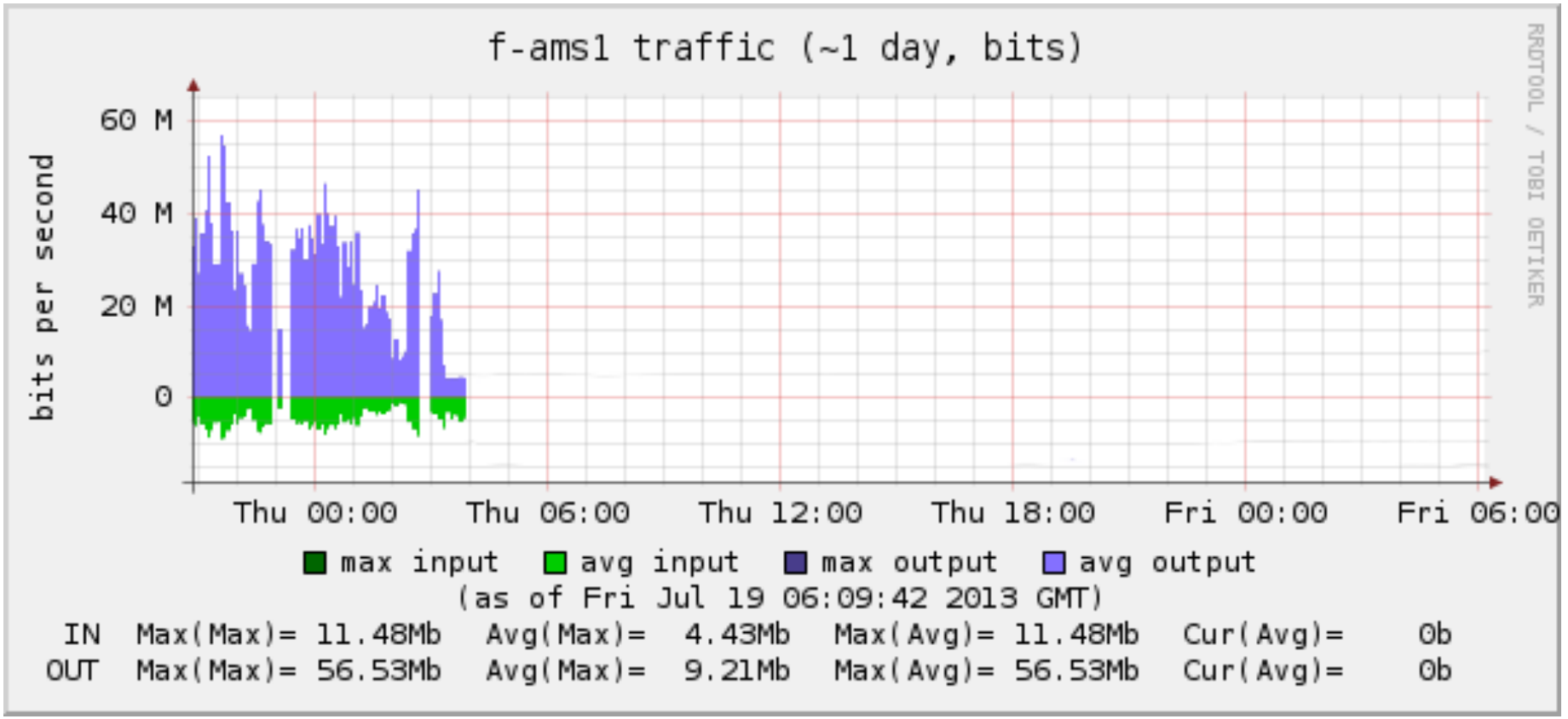
Poor Network Hygiene

- Non-caching name servers
- Too frequent flushing
- Open recursive servers

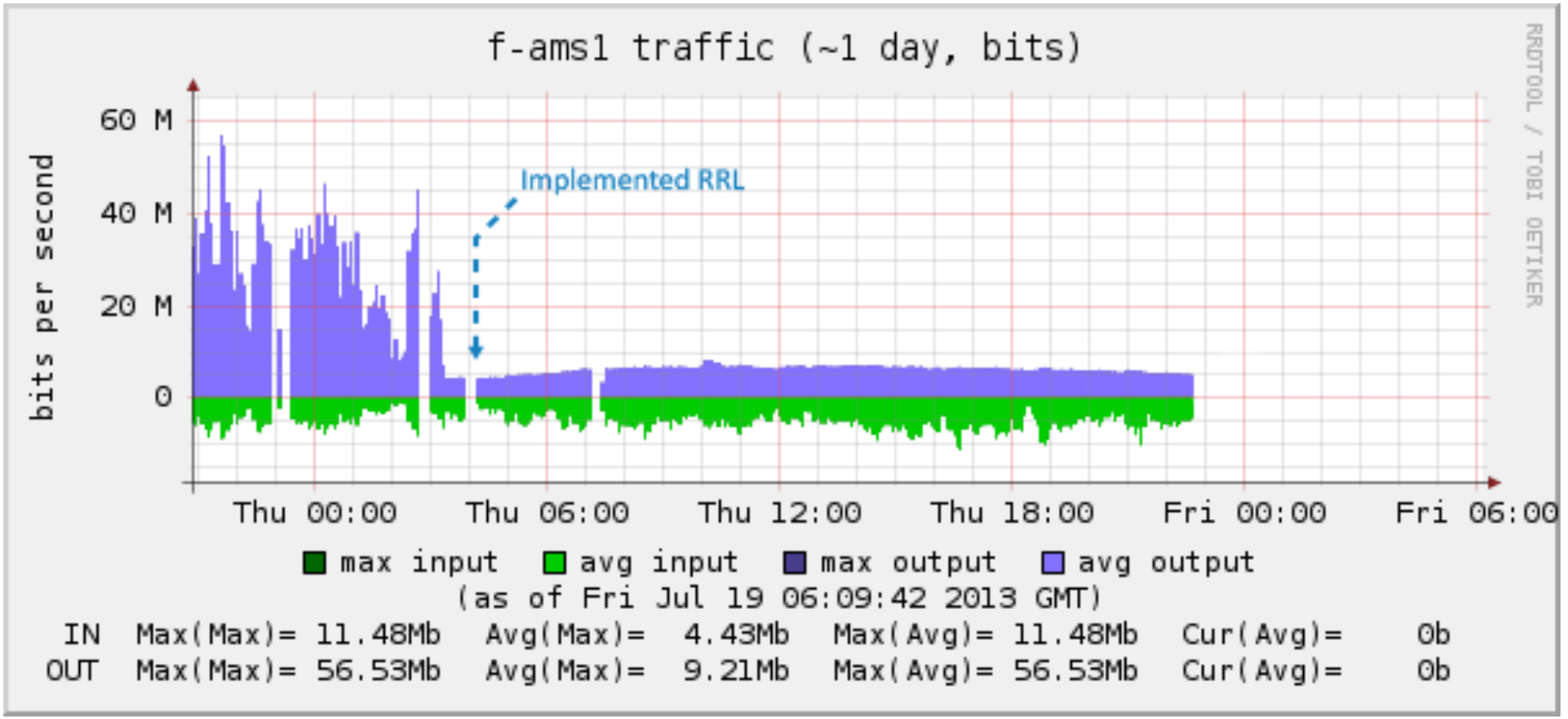
RRL on ISC network

- Deployed on isc.org and SNS in Spring of 2012
- Deployed on F-root in Summer of 2013

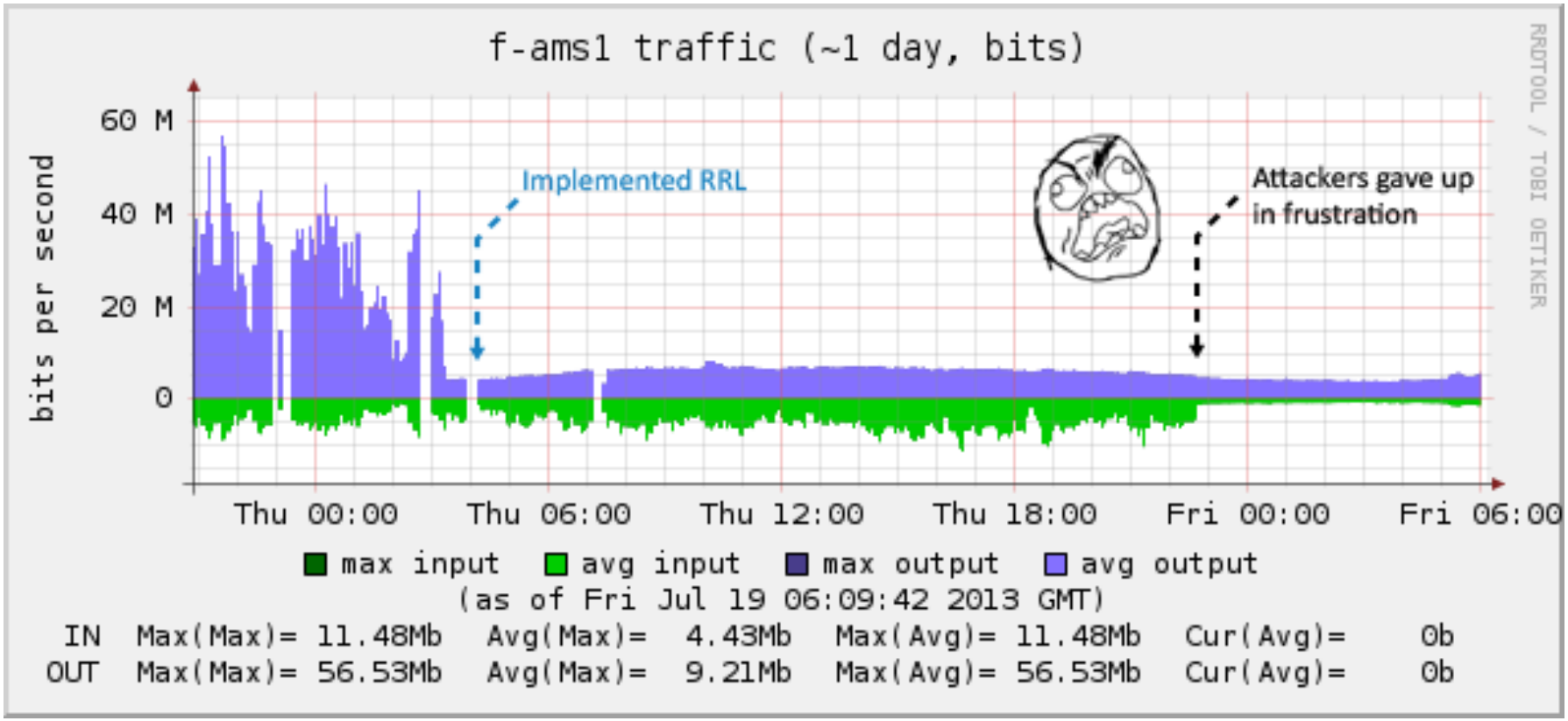
ISC F-Root



ISC F-Root



ISC F-Root



CONFIGURING RRL

K.I.S.S.

- SLIP
  - How many UDP requests can be answered with a truncated response.
  - Setting to “2” means every other query gets a short answer
- Window
  - 1 to 3600 second timeframe for defining identical response threshold
  - Highly variable based on conditions
- Responses-per-second
  - How many responses per second for identical query from a single subnet
  - Highly variable based on conditions

RRL Config

```
rate-limit {
  slip 2;                    // Every other response truncated
  window 15;                 // Seconds to bucket
  responses-per-second 5;// # of good responses per prefix-length/sec
```

RRL Config

```
rate-limit {
  slip 2;                   // Every other response truncated
  window 15;                // Seconds to bucket
  responses-per-second 5;   // # of good responses per prefix-length/sec
  referrals-per-second 5;   // referral responses
  nodata-per-second 5;      // nodata responses
  nxdomains-per-second 5;   // nxdomain responses
  errors-per-second 5;      // error responses
  all-per-second 20;        // When we drop all
```

RRL Config

```
rate-limit {
  slip 2;                  // Every other response truncated
  window 15;               // Seconds to bucket
  responses-per-second 5;// # of good responses per prefix-length/sec
  referrals-per-second 5;  //          referral responses
  nodata-per-second 5;     //          nodata responses
  nxdomains-per-second 5;  //          nxdomain responses
  errors-per-second 5;     //          error responses
  all-per-second 20;       // When we drop all

  log-only no;     // Debugging mode
```

RRL Config

```
rate-limit {
  slip 2;                   // Every other response truncated
  window 15;                // Seconds to bucket
  responses-per-second 5;// # of good responses per prefix-length/sec
  referrals-per-second 5;  //          referral responses
  nodata-per-second 5;     //          nodata responses
  nxdomains-per-second 5;  //          nxdomain responses
  errors-per-second 5;     //          error responses
  all-per-second 20;        // When we drop all

  log-only no;              // Debugging mode
  qps-scale 250;            // x / 1000 * per-second
                            //        = new drop limit
  exempt-clients  {127.0.0.1; 192.153.154.0/24;};
```


ISC & DNSco

RRL Config

```
rate-limit {
  slip 2;                   // Every other response truncated
  window 15;                // Seconds to bucket
  responses-per-second 5;// # of good responses per prefix-length/sec
  referrals-per-second 5;  //          referral responses
  nodata-per-second 5;     //          nodata responses
  nxdomains-per-second 5;  //          nxdomain responses
  errors-per-second 5;     //          error responses
  all-per-second 20;        // When we drop all

  log-only no;              // Debugging mode
  qps-scale 250;            // x / 1000 * per-second
                            //       = new drop limit
  exempt-clients  {  127.0.0.1; 192.153.154.0/24; 192.160.238.0/24  };
  ipv4-prefix-length 24;    // Define the IPv4 block size
  ipv6-prefix-length 56;    // Define the IPv6 block size
```

RRL Config

```
rate-limit {
  slip 2;                   // Every other response truncated
  window 15;                // Seconds to bucket
  responses-per-second 5;// # of good responses per prefix-length/sec
  referrals-per-second 5;  //          referral responses
  nodata-per-second 5;     //          nodata responses
  nxdomains-per-second 5;  //          nxdomain responses
  errors-per-second 5;     //          error responses
  all-per-second 20;        // When we drop all

  log-only no;              // Debugging mode
  qps-scale 250;            // x / 1000 * per-second
                            //       = new drop limit
  exempt-clients  {  127.0.0.1; 192.153.154.0/24; 192.160.238.0/24  };
  ipv4-prefix-length 24;   // Define the IPv4 block size
  ipv6-prefix-length 56;   // Define the IPv6 block size

  max-table-size 20000; // 40 bytes * this much maximum memory
  min-table-size 500;   // pre-allocate to speed startup
};
```


ISC & DNSco

Use of Logfiles

- Initially use logging
- Use a separate logging channel to segregate data from regular logs

Log only “dry run” feature to view behavior before going live with RRL

Logging Config

-=-

```
logging {

        channel query-error_log {
                file "log/query-error.log" versions 7 size 100M;
                print-category yes;
                print-severity yes;
                print-time yes;
                severity info;
        };
        category query-errors { query-error_log; };


};
```

Things to Consider

- French Connection – SLIP issue
- Window length – interrupt self-monitoring
  - Whitelist option ‘exempt clients’
- Not responding to legitimate queries

RRL Classifier

- **Expansion of RRL Basic**
  - RRL Basic filters on Destination Address of Response (source of attack traffic is assumed to be forged, but provides address of attack target)

- **2014**
  - Name Requested (QNAME)– allows for whitelisting and supports possible expansion to recursive use case
  - Size of the Response– limits amplification potential

Additional info on RRL

- Response to SLIP issue
  - https://www.isc.org/blogs/cache-poisoning-gets-a-second-wind-from-rrl-probably-not/
- Vixie Article on DNS Security
  - http://www.circleid.com/posts/20130913_on_the_time_value_of_security_features_in_dns/

FURTHER QUESTIONS?

Webinar Special Offer

- 10% discount on Advanced BIND Training Class
  - For registrations made before March15th
  - Coupon code will be in the email with the recording link

We want to hear from you!

- Look for follow-up email with links to webinar recordings and this presentation
- Take the user profile survey

https://www.surveymonkey.com/s/isc-downloads

ISC & DNSco

Webinar Survey

- Please take a moment to complete the survey now displayed.

Thank You

Internet Systems Consortium

For more information about RRL Basic, contact us at
info@isc.org

www.ISC.org

For more information about RRL Classifier, contact us at
info@dns-co.com

www.DNS-co.com