DNS Security: Beyond DNSSEC

A "He Must Be Nearing Retirement" Manifesto

Edward Lewis
ed.lewis@neustar.biz
OARC
May 12/13, 2013

Relax - "DNSSEC" only appears on this slide! I promise.

neustar

The Game Changer

- Reflection Attacks are profoundly troublesome
- Reflection Attacks represent a fundamental change
- Reflection Attacks make a victim an unwitting accomplice

Why This Bothers Me So

- Everything* done to protect the DNS is completely** wrong
- Why?
- Because everything* done to the improve DNS protocol is now being turned around to harm someone else
- The DNS is now the world's most capable and reliable attack traffic utility, like gas, electric, water, and uh, POTS

*, **

- Well, *almost* everything and *almost* completely...I'm using the word for dramatic affect

Prompting This

- Why I am giving this talk here and now
  - "Refused" replies to ANY queries over UDP
  - Response Rate Limiting

- A tipping point
  - They called to mind other factors, trends, turning points

How We Got Here

- The role and significance of the global public internet is fundamentally changing
  - How it is used
  - How it is valued
- This is true for "both sides" in the security theater
- The landscape of operations is changing
- The threatscape of malicious activity is changing

For Example

- When rolling out anycast, the theory was that when a node goes down, you pull the route advertisement
  - For a random error, you can lose a node for a while (diminished capacity) and bounce it back up later
  - For a randomly triggered systemic error, you do the same with a time budget to fix it
- But in an environment with active persistent threats (APT)
  - Systemic errors mean that taking out one failed instance will shift the attack traffic to the next "target"
  - You help the opposition mow you down!
- Anycast is for containment now, not robustness

Running The DNS

- While shifting focus from protocol engineering to operations
  - A schism has become clear between what operators see as important and what protocol designers see as important
  - Growing scale of operations spreads the challenges
  - No time to anticipate, no "threat" document, real time diagnosis
  - Operators don’t write code on the fly, they use available tools
  - Operators push for greater automation

Attacking (With) The DNS

- As the Internet grows in (economic) significance
  - First simple kill packets and host break-ins
  - Then botnets - stolen, uncapped attack capability
  - Then crowd sourced flood attacks - social activity
  - Then internet assisted flood attacks - amplification
  - Then unmonitored assists to floods - open recursive servers
- Attackers measure and monitor their successes, failures and modifying attacks in flight (sophistication)
  - While debugging, the cause shifts

Back To Reflection

- What happens when a reflection attack hits an operator
  - Malicious traffic comes, but not enough to take down a server
  - An operator probably has other things to deal with, maybe buy more capacity (via cloud) to be able to ignore it
  - The result is more capacity for the attackers to use to hit someone else (the true victim)
  - If an operator moves to stop the attack, the attackers use a different technique or another operator

How Did We Get Here

- We built the DNS for robustness by
  - Building in looseness (agility)
  - Building excess capacity
  - Building global reach
  - Building more into messages
- Our reward
  - More ways an attacker can abuse the system
  - More capacity that an attacker can make use of
  - A high-availability global utility for malicious use
  - More payload to deliver

A Manifesto Is Born

- First we have the moment of time in which we realize the old defenses are not the right ones
- Second we begin to realize that we no longer have the same stockpile of capabilities

- We need to reset our thinking on security
  - Ideas that have been considered to be crazy
  - Ideas that will take a long horizon to appear
  - Ideas that may use old pests as assets

Crazy Ideas And Taboos

- These are techniques that we usually dismiss out of hand, but now we should reconsider them
  - Changing the protocol
  - Increase the reliance on TCP
  - Threatening the "open nature" of the Internet
  - Using firewalls to choke bad behavior
  - Negotiation

Stop Blaming Someone Else

- BCP 38
  - Sorry, that's not helping
  - It's the right thing but who is going to enforce it?
- Going after middle-box operators to close poorly thought-out designs
  - This does work, but it takes a lot of effort by those willing to slog through the studies to isolate the source
- We still need cooperation...the Internet runs only with that
  - But not all agree on the direction

Changing The Protocol

- Yes, sigh, this will take a while
- But anything short of some changes means a continued burden on operators and a munitions for the attackers
- For example
  - Shift more types to be "TCP only" as AXFR is today
  - Aggressively coerce smaller responses
  - I'm purposefully being vague here
- The changes need not be drastic but we should reconsider
  - Does anyone really need to ask for each of the RR types?

Eliminating ANY Over UDP

- Ahh, the topic that launched the talk
- Rationale for doing so
  - No one has ever demonstrated a good reason for ANY
  - It has it's uses, but predominately malicious now
    - But prior to attacks, only used in debugging and for an application that is roundly disrespected
    - After the attacks, it is a large component of the traffic involved in an attack
  - By restricting ANY to TCP, reflection attacks become less interesting
  - There's alternatives that are less open to abuse

Moving The Fight?

- An attack could then shift from ANY to some other type, and if attackers are sane, they would
  - Let's be more aggressive and move other types to TCP too

- What about RRL? (*That other topic*)
  - That's moving the fight too, from authoritative servers to recursive servers
    - The latter being, overall, less managed, numerous, and hidden from control and enforcement
- The fight will move - unless we just let the attacks happen

Avoiding Blasphemy

- Response Rate Limiting is a solid tactic
  - But it is just a tactic
  - It is a drag on processing (not a lot but still)
  - Potential for false positives (but it has good heuristics)
- The heuristics it relies upon are not generally applicable to recursive servers
  - How do we defend recursive servers?
- If it comes to manual mitigations, there's a human error rate to consider

Is TCP So Bad?

- True it induces a round trip to set up the connection
  - But this can be amortized over long-lived connections
  - But with closer servers, a round trip isn't what it used to be
  - But web servers seem to deal with it
  - But we have had a lot of improvements with TCP state consumption attacks

- TCP is not a panacea but it is not as scary as it was in the 1990's
  - Just thinking about reflection attacks here...

Is EDNS0 So Good?

- Come to think of it, EDNS0's expansion of the buffer size is a key ingredient in amplification
  - In as much as a failure to deploy BCP 38, this is an enabler
  - But this was supposed to a good improvement!
  - We had to fight off firewall rules to get this working
- With TCP, EDNS0's buffer size is not needed (just sayin')

- Sigh

By The Way

- Lame delegations
  - Signaled by "referrals to the root" or "upward referrals" has already been flagged as a bad thing
- Negative answers (RFC 2308) added more size to the response, contributing to the problem today
- Each of these were benign, helpful steps

Negotiation

- There's been a campaign to ban negotiation from DNS, for the same reasons as
  - We want to avoid the TCP handshakes
  - Looking at other protocols (IPSEC for one) it's a lot of work
- "RD" is about as far as we've gone with negotiation
- Now, what about "cookies" and other ways to allow a server to place trust in the earnestness of a client?
  - This idea keeps rising up, why does the idea linger?
  - Can a server judge and make use of the reputation of a client?

Open Nature

- The internet is supposed to be a mesh of interconnected networks
  - Works best when there are no "bumps" no "borders"
  - Protocol works smoothly if there are no stateful middleboxes
  - End to end addressing uniqueness
  - Unfettered reachability for robustness
- Remember those days?
  - I suppose so - we complain about NAT
  - And about SPAM, how it is harder to route mail
- I've seen networks that were very "tied down," not pretty

Threatening the Open Nature

- In my dream world there would be no exchange of UDP port 53 between ISPs
  - Instead each ISP would peer directly with a DNS cluster
  - Remove the "D" in DDoS
  - Eliminates the need for BCP38
- But a dream it is, not all ISPs and not all DNS services will play
  - For those that do, charges of "preferential treatment" have been levied

Firewalls

- Those things that filtered DNS packets over 512 bytes, those that voided EDNS0's benefits
  - A real and royal pain
  - To this day I uncover them
- Can we turn this around?
  - To speed up deployment of radical changes, curtail abusive behavior of older software
  - This is playing with fire, I know, that's why it is a taboo
- But can we harness this asset?
  - I don't know, just throwing this out there

Enforcing Better Behavior

- Often it is said that it's too late to fix code once it is released
- But we can filter out traffic we decide it is malicious
- It's our network(s), we can do what we want, we don't have to "take it"
- There are products that enforce security rules, they can be put to use to protect enclaves

What Do I Want?

- I want a simpler-to-operate protocol
  - In general, fewer knobs and dials for basic features
  - Restrict large sized responses to TCP
  - Retain as much of the benign behavior we have now
  - Essentially, a refactoring of the protocol
- I want controls that make it easier for an operator to have a predictable traffic pattern and to return to that when needed
  - Controlled via routing perhaps
- I want us to do what we can about the old code that is open to abuse

Where Do I See A Start

- Layer 7
  - Restricting types are available to anonymous UDP queries
  - Refactoring our current security meta-data to be smaller
  - TCP-only for more types than AXFR
- Layer 3
  - Increasing the use of no-export peering over transit
  - Trending downward of UDP port 53 crossing ISP boundaries

The Manifesto

- It's time the DNS protocol refactor itself to be a better neighbor in the Internet
  - Instead of looking for causes in other areas, accept responsibility
  - Realize that the past is gone, the network is changing
- It's time to consider drastic changes that will take a long time to complete
  - The protocol is entrenched but it is not indelible
  - It has a caveman mentality, optimizing for fixed width fields as an example
  - Let's face it, it's a badly designed protocol

Discussion

- You can ask questions but I don't think I have the answers

  - It's a manifesto...
  - If you disagree with me, I'll just stare at you for a while