DDoS Mitigation at CloudFlare

Martin J. Levy

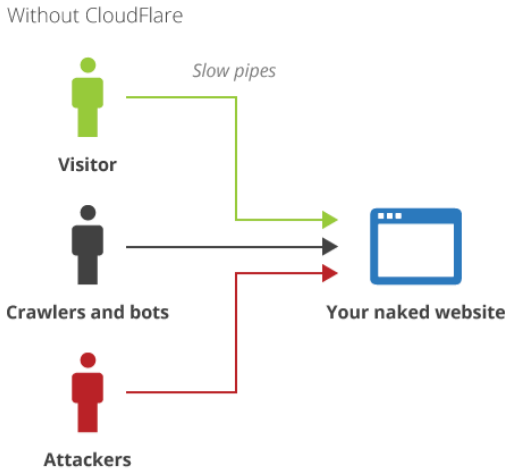
CEE Peering Day 2015 – Bratislava Slovakia
19 March 2015

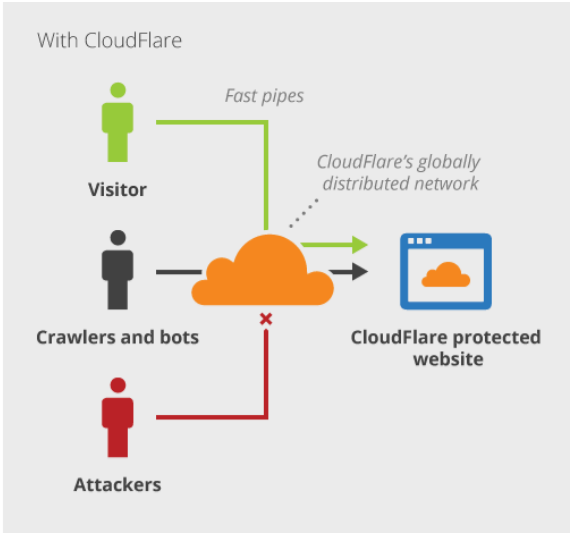
CloudFlare

What is CloudFlare?

CloudFlare makes websites faster and safer using our globally distributed network to deliver essential services to any website

- Performance
- Content
- Optimization
- Security
- 3rd party services
- Analytics





How does CloudFlare work?

CloudFlare works at the network level

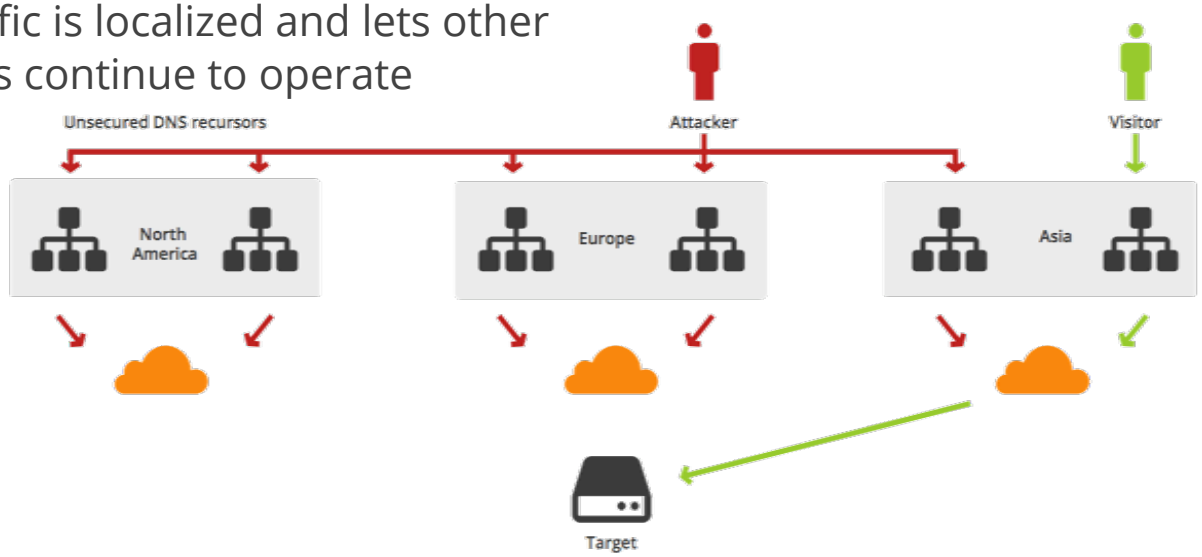
- Once a website is part of the CloudFlare community, its web traffic is routed through CloudFlare's global network of 30+ (and growing) data centers.

- At each edge node, CloudFlare manages DNS, caching, bot filtering, web content optimization and third party app installations.

CloudFlare works globally

CloudFlare protects globally

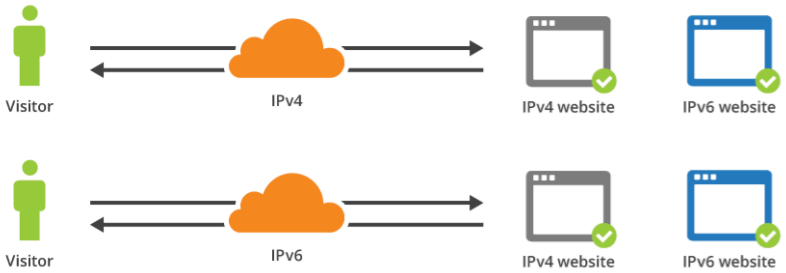
- DDoS attack traffic is localized and lets other geographic areas continue to operate



IPv4/IPv6 - automagically enabled

CloudFlare offers an automatic IPv6 gateway seamlessly bridging the IPv4 and IPv6 networks

- For most businesses, enabling IPv6 is costly and time consuming
- CloudFlare's solution requires NO hardware, software, or other infrastructure changes by the site owner or hosting provider
- Enabled via the flip of a switch on the site owner's CloudFlare dashboard
- Users can choose two options: (FULL) which will enable IPv6 on all subdomains that are CloudFlare Enabled, or (SAFE) which will automatically create specific IPv6-only subdomains (e.g. www.ipv6.yoursite.com)



IPv6 - automagically enabled

enabled by default!

CloudFlare has customers globally

Nearly two million websites

Anycast CDN

Anycast CDN – equally IPv4 and IPv6

Anycast prefixes

- Same IP prefixes (IPv4 & IPv6) advertised in each of the 30+ sites around the world (and growing)
- Unicast (from separate site-specific prefixes) used to pull traffic from “origin” web source

Traffic Control

- Eyeball ISPs (should) route to closest node, resulting in a very low latency to our services from everywhere in the world
- If ISP A routes to CloudFlare in Germany then traffic will be served from Frankfurt or Düsseldorf
- If ISP B routes to CloudFlare in Texas then traffic will be served from Dallas

This results in a reasonable distribution of attack traffic between our sites

- Easier to mitigate 10 sites receiving a ~50Gbit DDoS than 1 site receiving 1,500Gbit DDoS

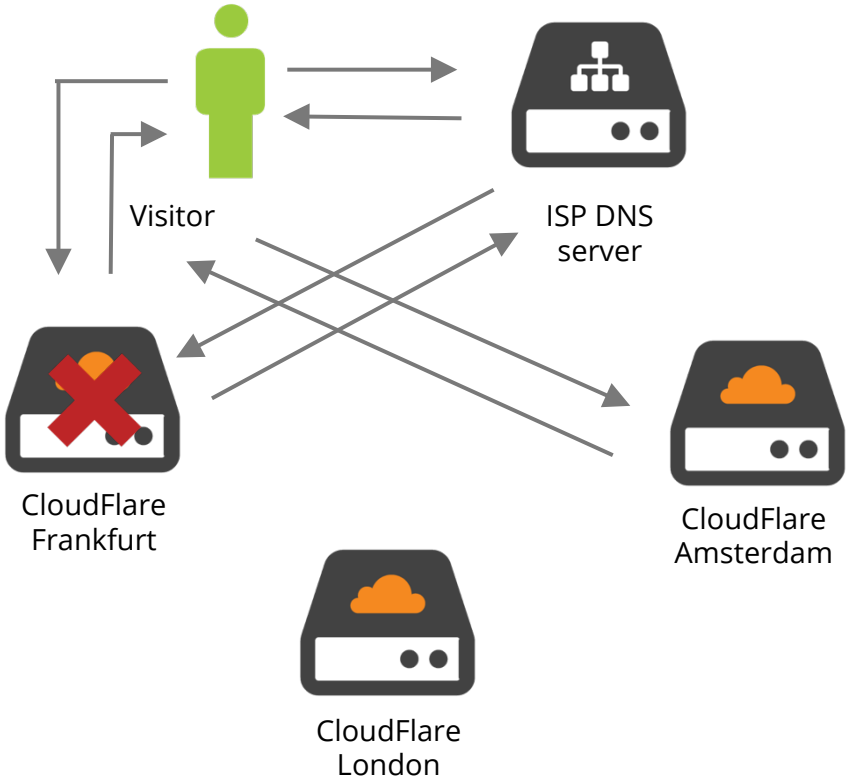
Anycast CDN

How does it work?

- DNS Query - to anycast DNS address
- DNS result returned with “Anycast” IP
- Client makes connection to closest server
- CloudFlare replies - session established

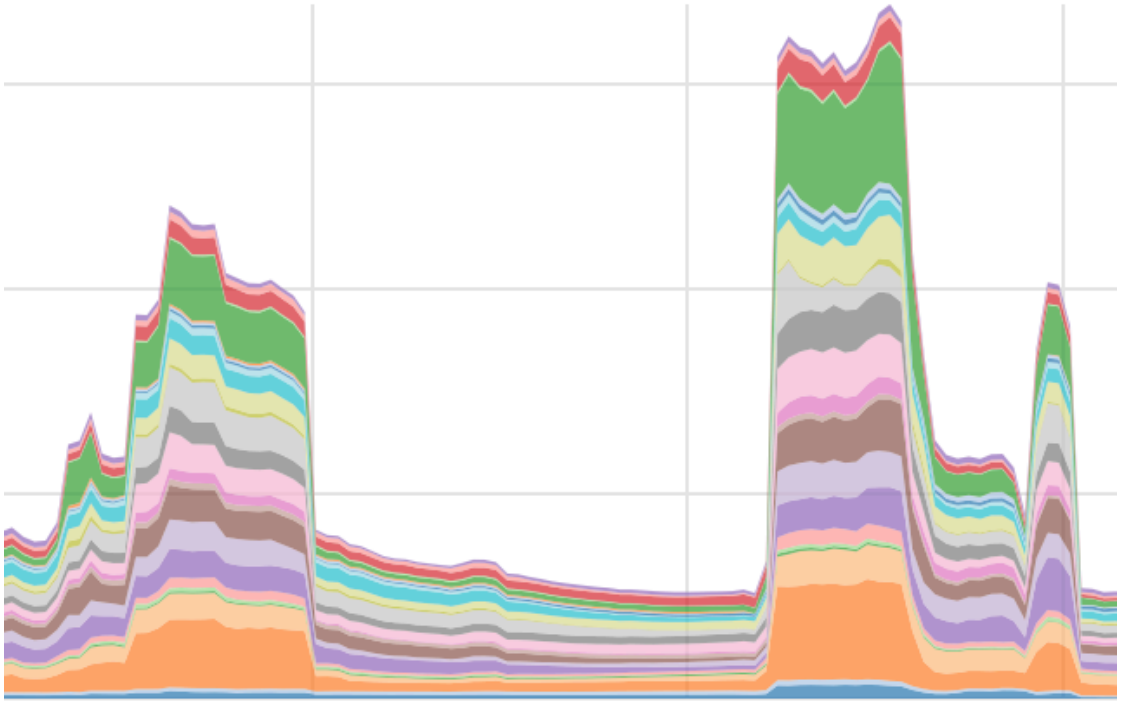
What happens in the event of an outage?

- Traffic re-routes to next closest DC
  - TCP session resets at this point

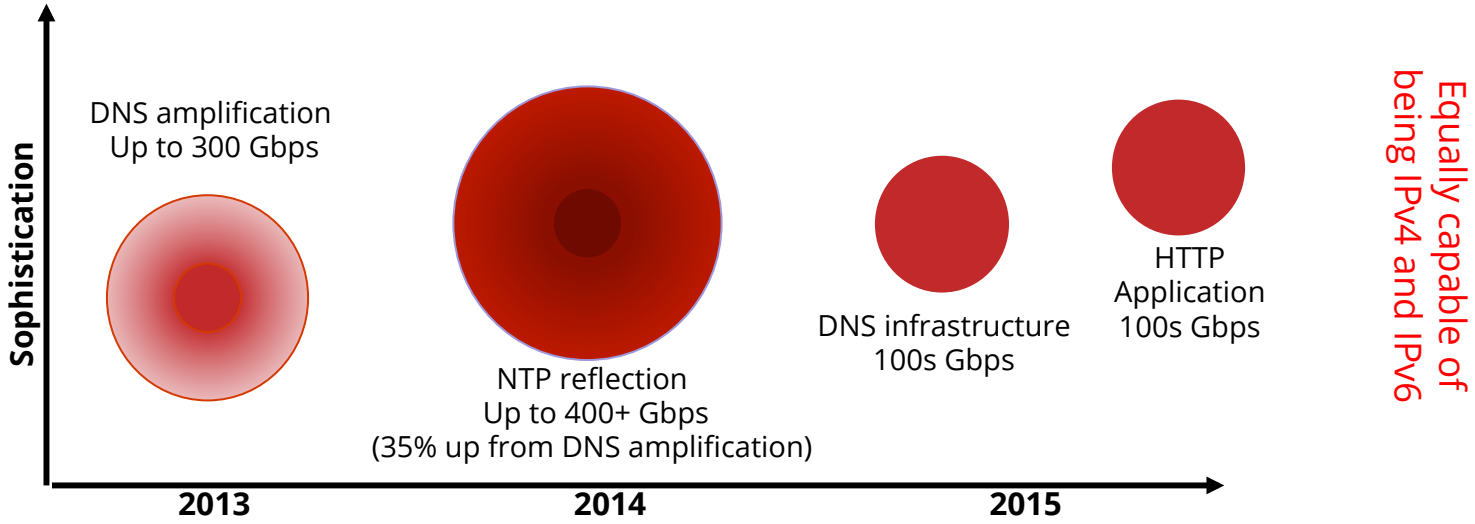


DDoS Mitigation

Hundreds of millions of packets per second



The Evolving Landscape of DDoS Attacks



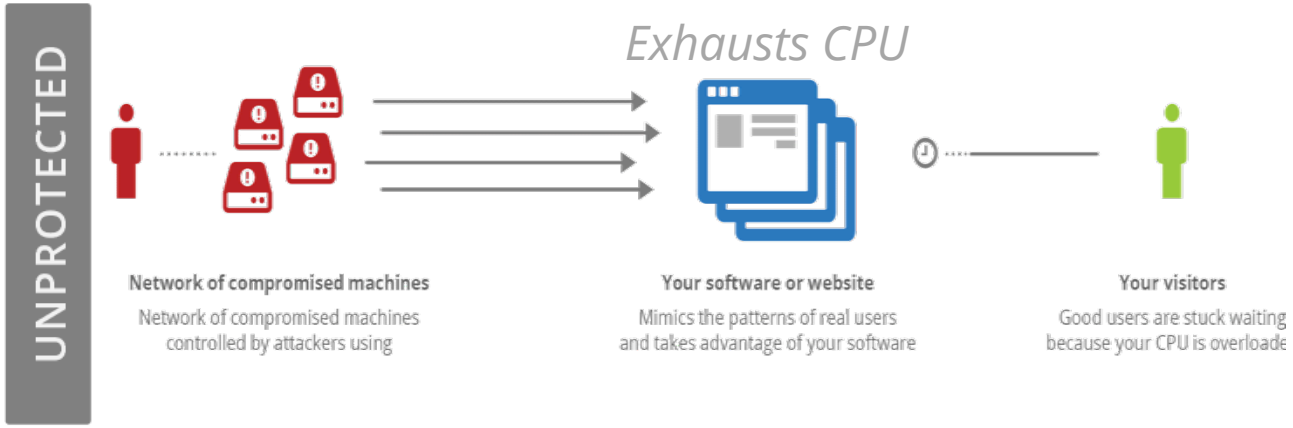
ATTACK TYPE	TREND
• Volumetric Layer 3 / 4	⬇
• DNS Infrastructure	⬆
• HTTPS application	⬆
• Origin: 100s of countries	⬆

More sophisticated DDoS mitigation and larger surface area to block volumetric attacks has forced hackers to change tactics.

New DNS infrastructure and HTTP layer 7 attack signatures that mimic human-like behavior are increasing in frequency.

CLOUDFLARE

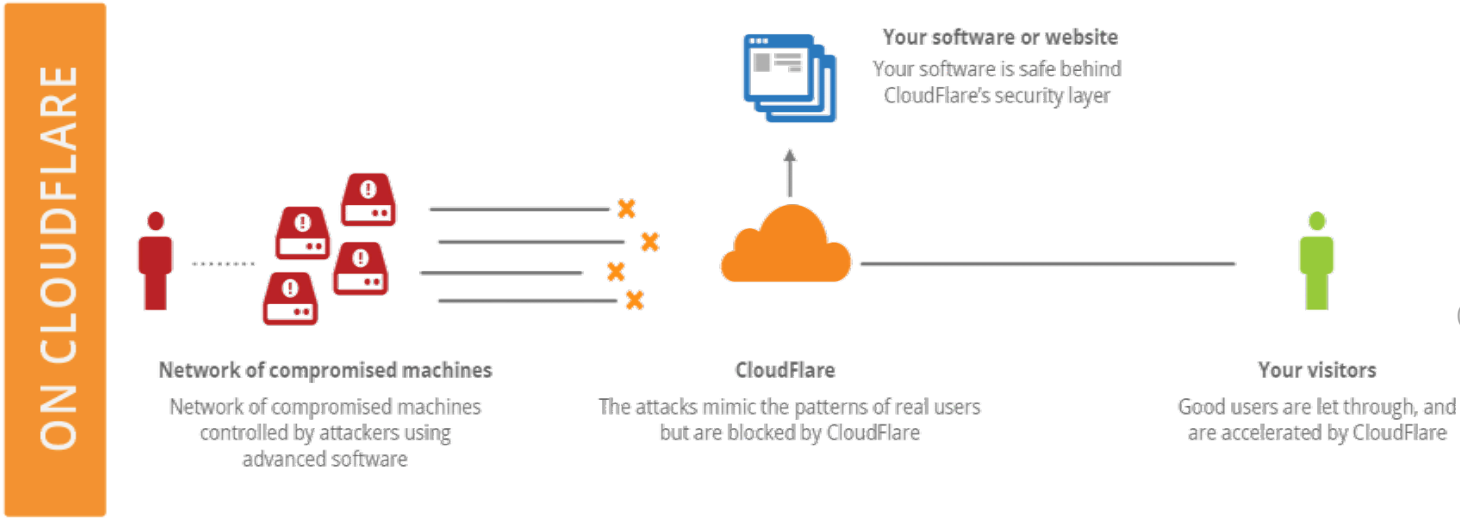
Layer 7 Attack methodology



Attackers use millions of compromised machines to launch a sophisticated attack that mimics real users and overloads the slow points in your web property.

As infected machines (both end user boxes and datacenter/hosting boxes) become IPv6 enabled, these Layer 7 attacks are showing up as IPv6 attacks

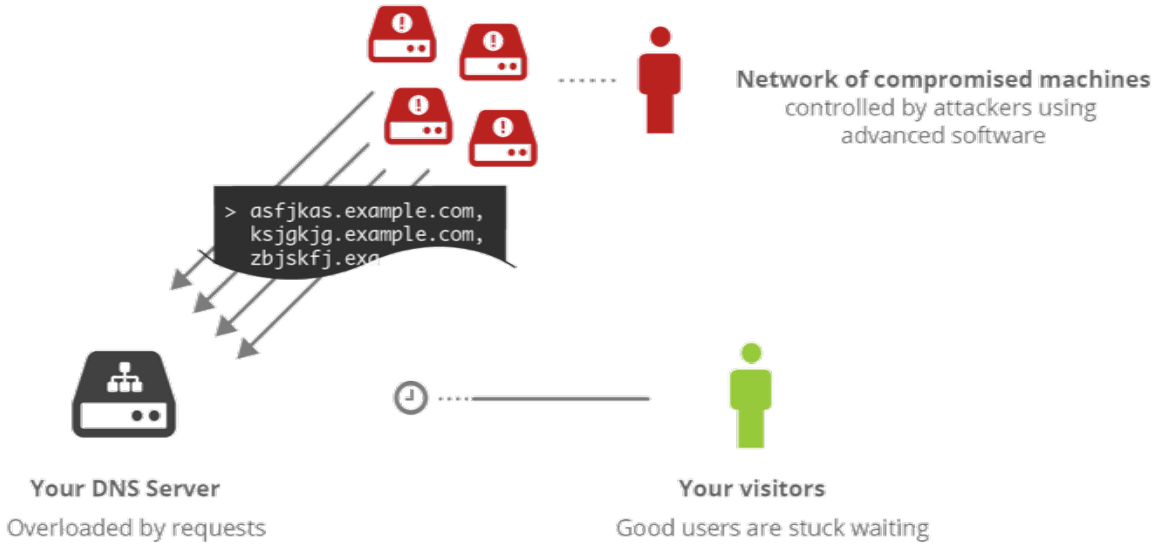
Layer 7 Attack methodology (protected)



A highly advanced attack that mimics real users is detected and blocked by CloudFlare before it can overload the slow parts of web server software.

Protecting at Layer 7 can be independent of IPv4 or IPv6 usage

DNS Infrastructure Attacks



Protecting DNS can be independent of IPv4 or IPv6 usage

DNSSEC and DNS attacks and open resolvers

```
$ dig ANY isc.org @63.21*.**.** +edns=0 +notcp +bufsize=4096
```

64 byte query, 3,363 byte response

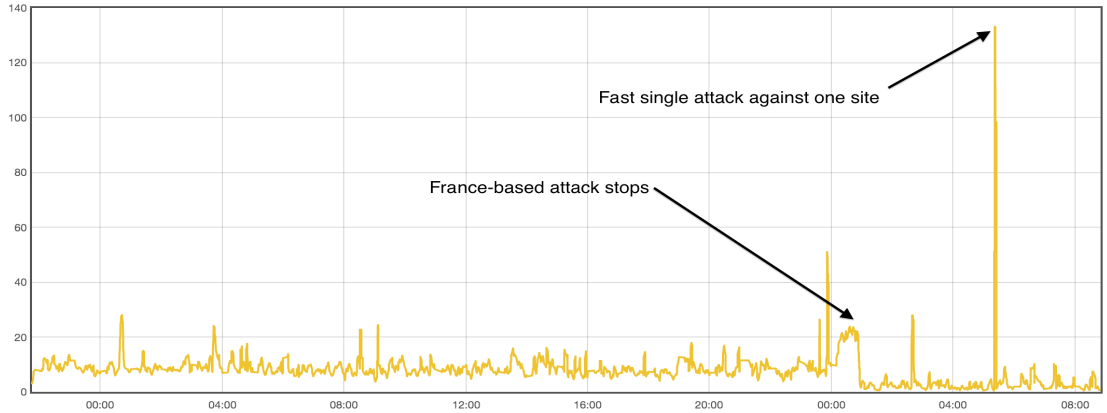
```
;; ANSWER SECTION:
isc.org.                7147    IN      SOA     ns-int.isc.org. hostmaster.isc.org. 2013073000 7200 3600 24796800 3600
isc.org.                7147    IN      NS      ns.isc.afilias-nst.info.
isc.org.                7147    IN      NS      ord.sns-pb.isc.org.
isc.org.                7147    IN      NS      ams.sns-pb.isc.org.
isc.org.                7147    IN      NS      sfba.sns-pb.isc.org.
isc.org.                7       IN      A       149.20.64.69
isc.org.                7147    IN      MX      10 mx.pao1.isc.org.
isc.org.                7147    IN      TXT     "v=spf1 a mx ip4:204.152.184.0/21 ip4:149.20.0.0/16 ip6:2001:04F8::0/32 ip6:2001:500:60::65/128 ~all"
isc.org.                7147    IN      TXT     "$Id: isc.org,v 1.1835 2013-07-24 00:15:22 dmahoney Exp $"
isc.org.                7       IN      AAAA    2001:4f8:0:2::69
isc.org.                7147    IN      NAPTR   20 0 "S" "SIP+D2U" "" _sip._udp.isc.org.
isc.org.                3547    IN      NSEC    _adsp._domainkey.isc.org. A NS SOA MX TXT AAAA NAPTR RRSIG NSEC DNSKEY SPF
isc.org.                7147    IN      DNSKEY  Z56 3 5 BQEAAAABwuHz9Cem0BJ0JQTO7C/a3McR6hMaufljs1dfG/inaJpYv7vH XTrAOm/MeKp+/x6eT4QLru0KoZkvZJnqTI8JyaFTw2OM/ItBfh/hL2lm Cft2O7n3MfeqYtvjPnY7dWghYW4sVfH7vVEGm958o9nfi79532Qeklxh x8pXWdeAaRU=
isc.org.                7147    IN      DNSKEY  257 3 5 BEAAAAOhHQDBrhQbtphqq2wQUpEQ5t4DtUHxoMVFu2hWLDMvoOMRXjGr hhCeFvAZih7yJHf8ZGfW6hd38hXG/xylYCO6Krpbdojwx8YMXLA5/kA+ u50WIL8ZR1R6KTbsYVMf/Qx5RiNbPClw+vT+U8eXEJmO20jIS1ULgqy3 47cBB1zMnnz/4LJpAOda9CbKj3A254T515sNIMcwsB8/2+2E63/zZrQz
                        Bkj0BrN/9Bexjpiks3jRhZatEsXn3dTy47R09Uix5WcJt+xzqZ7+ysyL KOOedS39z7SDmsn2eA0FKtQpwA6LXeG2w+jxmw3oA8lVUgEf/rzeC/bB yBNsO70aEFTd
isc.org.                7147    IN      SPF     "v=spf1 a mx ip4:204.152.184.0/21 ip4:149.20.0.0/16 ip6:2001:04F8::0/32 ip6:2001:500:60::65/128 ~all"
isc.org.                7147    IN      RRSIG   SPF 5 2 7200 20130828233259 20130729233259 50012 isc.org. XDoOYzkTHEVlWlV4TT50SsqXn4cxNhPvEuz3iFjq/oskLY9UOaK4GYDO GqHAjwNT0B6pUakKTQ3GvBjUBufPcEauCO17L7kb8/cC6zYifUCoW0pS moiQxmyqfrPDTzyVA894myUONGgMm6QW8HGPVvc6HzGWx9bOmjvFyX uOs=
isc.org.                7147    IN      RRSIG   DNSKEY 5 2 7200 20130828230128 20130729230128 12892 isc.org. COfF8fU6a8TBUG97SI/X+u2eKv7/mw+ixD3IWBnr3d3cWZmzF1sV8bWT YbuJebwnJMgN5OfB9PLsN+4QT617OjBe1dUF2M9jZeBiWsSsvvrdHnHM P8KwX6eayByDUoFsYe9VAH6C94bTQ7h7Cr0ytaVSXUytqFZV+DGn
                        v3kqSi50V3YkFNPAJDqqs0trenjwv/sPlqWVqEAoU/KMZMtYpCEMbHVP 8nbRP3jj/10WLccPtjHhiw4Ka9Sk4o+b7BMYCgXGXlhaap2lSUqkytHt 8RJdVxxd5Cj7Bi+O4LXODlS4bAZEDG7UoHR27MzXtvMZogsNyyNUKHSs FtUvBg==
isc.org.                7147    IN      RRSIG   DNSKEY 5 2 7200 20130828230128 20130729230128 50012 isc.org. Mbr/QqJPoIuf3K5jCuABUIG0/zSHQ8iWZpqvHx7olVBEmTxhi3/vW+IE DW6VCE1EpZc1S1lMMRZUnbBnVSpe0rZl3BxoLlRQqvsbC3jD15Se41WB DscDOS63C0GLqI9IhSyVuqtlpqIAOluSqtABHbAktPP05Rm0tST2Y A5Q=
isc.org.                3547    IN      RRSIG   NSEC 5 2 3600 20130828233259 20130729233259 50012 isc.org. v7G42xYTY9WFlvsBlRFuJ2rox/QjftLoRrDCMfqPW6kb5ZswjKt5zho 4o2sIrylTqad68O+lMxrDcg+7c2D8Hdh848CODEkjunBXkGBtLtaJvO5 zMn+d/OgUY5O7wtkeryb3wZdiHcFwIcvsPKJYZWKCdaaCWibne7c wls=
isc.org.                7147    IN      RRSIG   NAPTR 5 2 7200 20130828233259 20130729233259 50012 isc.org. gWUV6KACaYgsCgtRS4iKkHBBidfJfqS4drUf4kuPX2Et19fjlYrgOQK QFB5kBrzJLKhlIF4YpV+KYVUF8213AtpsohpUH5Uyc3yD3rlCUDVyVvc T9qUrIuRpZLInD2kBLmDaG76MRzQ/ZAkdXmwxZJhgTrfMLy+Uw/Ktk H7w=
isc.org.                7       IN      RRSIG   AAAA 5 2 60 20130828233259 20130729233259 50012 isc.org. dfzIo0VGT0MptTaPOus4FxDxSpeuOg127QedlqLGTxKGN1ppV/Bd6R0 WktMagZY9rSqmjfXNPlF3Q+7YeTpMssQhHqjE/tDoj9q9r8RXuBLJ1+a VRq3+xMbxb5EXAyQVZw24ILuldqNpXPeGCXNINSWd7VZEIDNqhu9C g7U=
isc.org.                7147    IN      RRSIG   TXT 5 2 7200 20130828233259 20130729233259 50012 isc.org. WtB3SYzcOKpNbOtBlnmtsI0DCbDB4k1v/HBY24PTZyWF/3tI8l+wZ+/p MfJ/SblbAzT67DO5Rfxl0hr8UlRKVa7OoqinQp5+rqiS67lv1hGO6ArO k+J0jLTis9Uz32653dgAxlgjEdWDRqSQ2TaHirAXxyI8fos5WNl/h4 GLo=
isc.org.                7147    IN      RRSIG   MX 5 2 7200 20130828233259 20130729233259 50012 isc.org. BSXC42oV6MCF0dX2icyxnvyijhy569BJCoanm5VrIIUimeEqo26lFQJx 7ofFCWa4fKOoa+EZ0qloNPfDiczStr8MmK8Lznu6+8IRfdmcG/kURuSi JdvDa0swxjmCm9aYu2nhoyHs+jqbJ+ZHHI0iDUX1fiM+9G2K9BjLru NxU=
isc.org.                7       IN      RRSIG   A 5 2 60 20130828233259 20130729233259 50012 isc.org. Gmb8tt8d7kxx4HsA8L6IdFYGGSJCA8PTWexUP3CBLna39e4a6gVz[illegible]HuiZBEXNx3dSagpjiTJYfMML8AY0uO0tgyjqaTyzwPPV5lW xQKVC092BPJx9IeKw+DC57f3m9LOaHJlMIh7WYFn8jxqeg1lSwJN0e35 Qvc=
isc.org.                7147    IN      RRSIG   NS 5 2 7200 20130828233259 20130729233259 50012 isc.org. RBvXLeTH0726iKvElmBZYUE+AWG3s2YRxKxuCnrhg7o9[illegible]RRz4f0QxODnPm+frpWIPbCpRf0SUFDQ82opQDwAb2CM 0D9N95y1t9hYfSeHEsEEk2yWgIymd9[illegible]S2XXCmwuVVZ7ZeYQmVEVkF7Jt V3A=
isc.org.                7147    IN      RRSIG   SOA 5 2 7200 20130828233259 20130729233259 50012 isc.org. iiDnH6tvmap0h2cdULI8Ihme+zbtQ2+D3ycKRqBc9[illegible]P2DNLm5nkpNsgA+Ur+YQ6pr0hZKzbDkBllBIW4C0LV DsjzPX3qLPH4G3x/20M+TeGe4uzPB[illegible]nw0VxB8g8ZP5znvdiZG6qen jas=

;; AUTHORITY SECTION:
isc.org.                7147    IN      NS      ns.isc.afilias-nst.info.
isc.org.                7147    IN      NS      ord.sns-pb.isc.org.
isc.org.                7147    IN      NS      ams.sns-pb.isc.org.
isc.org.                7147    IN      NS      sfba.sns-pb.isc.org.

;; ADDITIONAL SECTION:
ns.isc.afilias-nst.info. 56648  IN      A       199.254.63.254
ns.isc.afilias-nst.info. 56652  IN      AAAA    2001:500:2c::254
ord.sns-pb.isc.org.     31018   IN      AAAA    2001:500:71::30
ord.sns-pb.isc.org.     31018   IN      A       199.6.0.30
ams.sns-pb.isc.org.     31018   IN      AAAA    2001:500:60::30
ams.sns-pb.isc.org.     31018   IN      A       199.6.1.30
sfba.sns-pb.isc.org.    31018   IN      AAAA    2001:4f8:0:2::19
sfba.sns-pb.isc.org.    31018   IN      A       149.20.64.3
mx.pao1.isc.org.        3547    IN      AAAA    2001:4f8:0:2::2b
mx.pao1.isc.org.        3547    IN      A       149.20.64.53
_sip._udp.isc.org.      7147    IN      SRV     0 1 5060 asterisk.isc.org.
```

Layer 7 – Malicious Payload

- Request sent to exploit vulnerability on server
  - WAF on CloudFlare blocks 1.2 billion request per day
- Shellshock
  - 10 to 15 attacks per second during the first week
  - Top countries: France (80%), US (7%), Netherlands (7%)



DDoS mitigation

- How can we solve this?
- Detection
- Mitigation
  - Server Farms
  - Network

- Scaling to respond
- IPv6 challenges

How can we solve this?

How can we solve this?

- At this scale no one solution is going to be a “silver bullet”
- The cost per Gbit of scrubbing hardware is insane and not really an option.
- However…. there is a lot you can do with commodity servers and vendor routers.
- Can filter any criterion that can be matched up to layer4 in hardware on any major router vendor’s hardware.
- Can do a bunch of tricks to reduce the burden of filtering traffic on your server metal.
- Plus a bunch more.

Detection

How do we even work out what to mitigate

Detection - how do we do it?

If I asked you to tell me what was DDoSing you, without expensive vendor hardware how would you do it?

- tcpdump(1)?
- Some other packet sniffer

Servers under load during attacks (CPU, RAM, etc), despite great scale.

tcpdump attempts to find a large block of contiguous free RAM, then times out if this is not possible, leaving it often useless until the attack is over.

It is also very resource intensive to start sniffing all traffic on a server.

Detection - how do we do it?

So how do we do this?

- Taking the burden of detection away from the device being attacked can be very helpful
  - Export NetFlow records from the edge routers
  - Export sFlow from the switches in our datacenters
  - Automating this process has helped considerable

- Reading data from the application
  - NGINX logs tell you a lot that is useful.

- Sometimes calming the attack down to a manageable level with blunt rules (rate-limit all traffic from these 5x /16s to this single /32) can help to be able to then do deeper inspection and fine-tune the rules we implement to mitigate

Mitigation - on our server farms

DNS - BPF tools + lots and lots of DNS IPs

DNS attacks have a number of unique solutions;

- CloudFlare have many many thousands of DNS servers

```
$ host bob.ns.cloudflare.com
bob.ns.cloudflare.com has address 173.245.59.104
bob.ns.cloudflare.com has IPv6 address 2400:cb00:2049:1::adf5:3b68
$
```

  - Allows us to distribute the attack more effectively
  - Can null route specific DNS server IPs with minimal impact
- BPF (Berkeley Packet Filter) tools
  - High performance pattern matching driven filtering
  - Allows us to filter out DNS attack traffic using far less CPU resource
  - *http://blog.cloudflare.com/introducing-the-bpf-tools/*
  - *https://github.com/cloudflare/bpftools*

Hashlimits

Enforce “no more than X connection attempts per minute for this hash”, otherwise blacklist

Hash is made up from whatever criterion you want, but for our purposes combo of src + dest IPs

Fairly effective method of easily detecting “ddos-like” traffic.

Trick is preventing false detections.

- Customer with many millions of users released an application update causing the application to regularly perform JSON queries against their application.
- Users behind a CG-NAT appeared as if they were coming from a single IP.
- Triggered enforcement on non-malicious traffic.

“I’m under attack” mode

Customer enabled mode that forces users to a challenge page.

Uses an early code path in NGINX.

Significantly less CPU required to process requests than going through the full process of serving their request.

Connection tracking to validate sessions

Another way to perform non-cpu-intensive processing of malicious traffic.

Using conntrack to inspect all sessions for certain source/destination IP combinations can be incredibly useful, but it has it's risks as well.

While maintaining a session table can be incredibly helpful and allow you to mitigate malicious traffic early on, it also presents another attack surface (session table flooding).

Used incredibly selectively, particularly for large ACK attacks.

ECMP to distribute traffic between servers

Allows us to ensure no one server bears the entire brunt (for traffic coming into a given site) of the attack load aimed at a single IP. 16 servers can more easily mitigate an attack than 1.

All our servers speak BGP to our routing infrastructure, so this is not particularly difficult to implement.

By default, ECMP hashes will be re-calculated every time there is a next-hop change.

- Causes flows to shift between servers
  - TCP sessions reset
- Can solve this with consistent ECMP hashing
  - Available in Junos from 13.3R3 for any trio based chipset
  - Only works for up to 1k unicast prefixes, so struggles to scale

Solarflare cards and OpenOnload

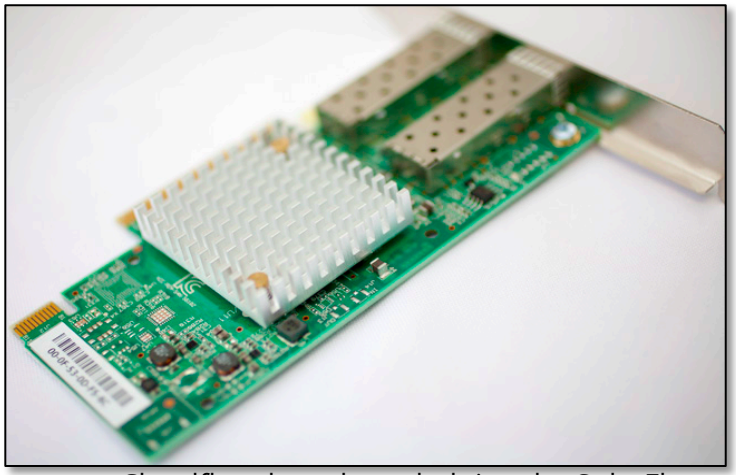
In our latest generation of server hardware we;

- Made the move to 2x10Gbit per server (from 6x1Gbit LAGs)
- Did this with NICs from Solarflare.

SolarFlare NICs have very cool abilities to pre-process traffic on-board before handing to the CPU (OpenOnload).

Can identify certain types of traffic and assign it to cores based on rules pushed in the cards.

Can handle certain requests in userspace without creating CPU interrupts



Cloudflare have been helping the SolarFlare develop this functionality for their cards.

http://blog.cloudflare.com/a-tour-inside-cloudflares-latest-generation-servers/

Mitigation - in the network

Null route and move on

When an attacker targets a website or a service, while they may want to take this website/service down, they target the IP address in order to do this.

First order of business can be to update the DNS A/AAAA record and move on.

If the attacker follows, keep doing this.

Easy to automate, requires an attacker to continually change the attack to follow.

Depends on rDNS service operators honouring our TTLs

FlowSpec (RFC 5575)

Important to understand from the outset that ALL flowspec does is automate the provisioning of a backplane-wide firewall filter on multiple devices. Having said that, **it does this really well**.

Can use most “from” and “then” actions available in Juniper firewall filters in FlowSpec. While Juniper have been an early adopter, other vendors have struggled to get this into their code. Even Juniper has only recently implemented IPv6 support for FlowSpec.

Being able to match “TCP packets from this /24, to this /32, with SYN but no ACK and a packet length of 63 bytes” and “rate-limit to 5Mbit” per edge router is incredibly useful.

Being able to configure this in one place and have it push to the entire network is awesome!

Regional enforcement

Under certain circumstances, it makes sense to enforce regionally

- Seeing 300Gbit of traffic targeted at AMS, LHR, FRA, CDG for a website with 99% of legitimate traffic being served into HKG and SIN
  - Can implement strict flowspec enforcement in sites targeted, while no enforcement needed in sites traffic is legitimately needed in.
  - Take advantage of any opportunity presented

Regional null routing can also be worthwhile at times

- Want to move site to new IPs and move on.
  - Null route in only the regions that are being targeted.

Have your transit provider configure firewall filters in their network to filter certain packet types / lengths / src-IPs / dst-IPs / etc upstream in one region only to help filter malicious traffic.

Dealing with attacks on infrastructure IPs

Relatively easy to mitigate attacks on Anycast IP space.

- Multiple hundred gig attack on an anycast IP
  - Distributed over 28 sites
  - Multiple tens of gigs per site

Vs:

- Multiple hundred gig attack on an IP specific to a single router, link or DC
  - Very hard to mitigate
  - Multiple hundred gig attack traffic > 100Gbit link

How do we prevent this from happening?

What can we do about it? What gain do you get from exposing this?

Attacks on Infrastructure - obfuscation of IPs

Traceroutes that show you the full path are nice... but... at what expense?

- Reveals a lot of the IP addressing information of your infrastructure to the entire internet
  - Becomes easy to figure out what to attack.
  - Makes every linknet, loopback, and infrastructure IP a target

Worth at least considering obscuring some of your infrastructure

- Stop responding to ICMP and UDP ttl expired
- Avoid ICMP-Packet-Too-Big in IPv6
  - Killing this can cause serious problems.`

Attacks on Infrastructure - kill routability to IPs

Can take the next step and kill reachability entirely.

Make your linknet IPs non-routable;

- Take all your linknet IPs from a /24 that is not advertised on the internet
- Use RFC1918 space
- Blackhole all your linknets
  - Don't forget to blackhole the provider side also!

This can make debugging significantly harder!

A lot of work will need to be done in the pre-sales stage with transit providers to ensure that one of these options is possible.

Peering exchanges should not be reachable on the internet anyway

Scaling the network to respond

Scaling the network - capacity

Ultimately, this is all a capacity game.

If you are seeing attacks roughly equivalent to your transit capacity you'll struggle to mitigate it, however if your transit capacity is 10x or 100x the size of the attack you'll struggle a lot less

Attack traffic will often be very very small packets (<80bytes) so need to ensure that any routers, line cards, are specced for capacity based on those numbers.

- As an example, MPC4E cards perform at nowhere near line rate when challenged with hundreds of gigs of 64 byte packets
- How oversubscribed is your backplane
- Measurements should always be in PPS rather than Gbit/sec

As you scale up your routers, you may discover that PPS bottlenecks simply move to your transit providers.

Peering vs Transit

Far easier to mitigate a DDoS coming in on an IX than a DDoS coming in via a transit provider.

- Can negotiate with transit provider for features such as RTBH, NOC implementing firewall filters for you, etc
- Peering exchanges generally don't have these features.

Peering exchanges are also surprisingly expensive to scale up for DDoS. Generally will be more expensive to order more 10Gbit ports at an IX vs additional handovers to a transit provider.

Often end up de-peering a network sourcing large amounts of attack traffic to force them onto a transit provider where you have more control.

This seems broken - surely there is a better way to ingest this traffic?

The solution - lots and lots of PNIs

PNI with networks sourcing DDoS traffic in multiple locations

- Limits the scope of impact to the network sourcing the attack, or their upstream who you are peering with

Easy to scale very cheaply

- Only costs are router interfaces, DC cross connects

Can easily filter traffic from that specific network on your router interfaces

Scaling with more sites

The more we scale, the more we will distribute attack traffic, and the greater attack mitigation capacity we will have.

Scaling using caches

A step beyond scaling up the sites. Ability to place caches in hundreds, if not thousands of locations within ISP networks.

Distribute attack traffic significantly.

IPv6 is a challenge

IPv6 changes this all… a lot

For all network operators, IPv6 presents significant security challenges

- How good is your IPv6 security compared to your IPv4 security
- Obfuscation via NAT is no longer an option
- Are your IPv6 ACLs written as well as your IPv4 ACLs?
- Do all the tools you use support IPv6?
- Do your traffic measurement and DDoS mitigation tools fully support IPv6

We can reasonably assume that it is more likely that reflection attacks can be launched at routers, servers, etc using IPv6, as services won't be as consistently secured

- NTP attacks, DNS attacks, SNMP?
- Who knows what else?

IPv6 changes this all... a lot

We are currently seeing over ¼ of all DNS attacks coming in via IPv6.
Balance of malicious vs legitimate DNS traffic about the same on v4 and v6.

No CG-NAT on IPv6 means

- Not going to falsely identify legitimate traffic as malicious due to many hosts behind a CG-NAT
- One bot is not going to cause us to challenge thousands of users because their IP is in a blacklist

Scaling issues on routers

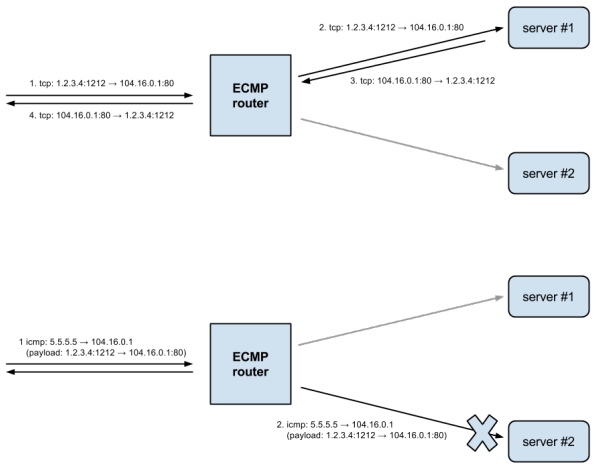
- IPv6 addresses are 400% the size of IPv4 address
  - At scale, this may long term present challenges as to the number of null routes, flowspec rules, etc that we can push to our router silicon
  - At this point, should we start doing enforcement on a per /64 basis?

ECMP & ICMP & IPv6

ECMP (Equal Cost Multi Path routing) & ICMP

TCP it hashes a tuple (src-ip, src-port, dst-ip, dst-port)

- Uses this hash to chose destination server. This guarantees that packets from one "flow" will always hit the same server
- No special knowledge of ICMP. It hashes only (src-ip, dst-ip)
- ICMP src-ip is most likely the IP of a intermediary router on the Internet
- IPv6 requires PMTU discovery to work; ECMP breaks it
- Solution: Broadcast ICMP PMTU messages to all servers
- Posted source code: https://github.com/cloudflare/pmtud



https://blog.cloudflare.com/path-mtu-discovery-in-practice/

Questions?

Thank you!

AS13335

Martin J. Levy, Network Strategy
@martin / @cloudflare
http://www.cloudflare.com/