Evolving DDoS Landscape

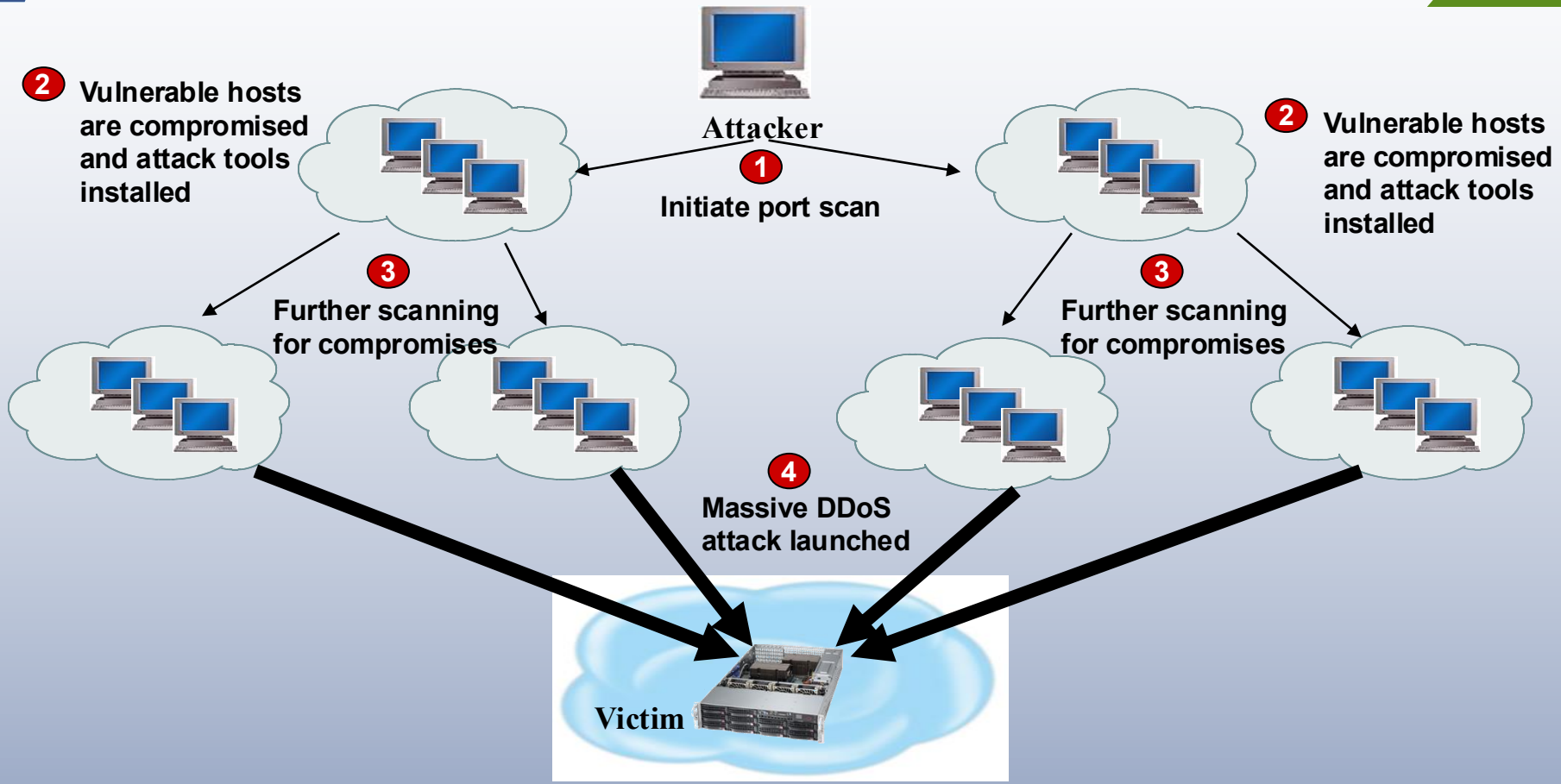
ABQNOG – Albuquerque, NM
September 17, 2026

- Merike Kaeo
- vCISO, Double Shot Security
- merike@doubleshotsecurity.com

DDoS - Conventional

- Traffic Sources
  - Compromised Internet-facing devices
  - Amplification attacks using well known services
  - Spoofed addresses
- DDoS for Hire
  - Also known as DDoS ‘booters’ and ‘stressors’
  - Inexpensive and started with gamers
- Mitigation
  - Rate limiting
  - Traffic scrubbing
  - Harden services against amplification attacks
  - Ingress filtering (BCP38) to protect against spoofing
  - RTBH [remotely triggered black-hole routing]

Automated Distributed Denial of Service Attack



Amplification Attack – DNS Open Resolvers

Attacker

Use forged IP address of intended victim to send legitimate queries to open recursive DNS servers.

1

Open Resolvers

Open recursive DNS server legitimate responses create massive DDoS attack to victim's IP address.

4

Open recursive DNS servers send legitimate queries to authoritative servers

2

3

Authoritative DNS Servers

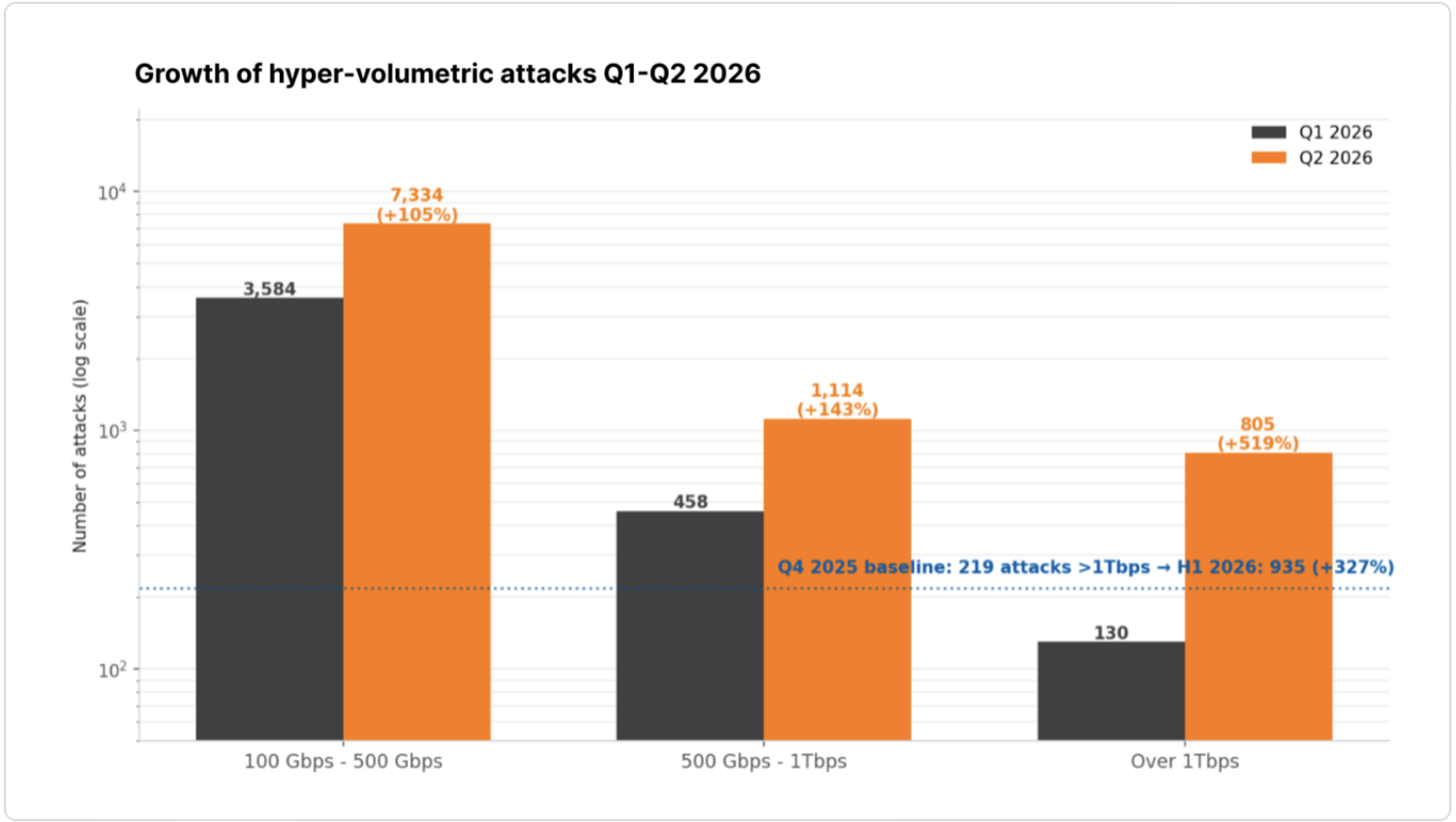
Authoritative servers send back legitimate replies to recursive DNS servers

Victim

DDoS - Today

- DDoS BotNets have become more sophisticated
  - AI tools allow operators to be *less* sophisticate
  - Objectives can be defined in plain language
    - Define timing and size parameters without knowing technical details
- Size is exponentially increasing
  - Multi-Tbps / multi-Gpps
    - Cloudflare has reported 7.3 Tbps / 37.4 TB in ~45 seconds
    - Aisuru/Kimwolf Botnet – 1-4million infectd hosts 31.4Tbps /14.1Bpps
  - Increase in “hyper volumetric” bursts (large / short duration)
    - Even short bursts cause routing instability and server downtime
- Traffic Sources
  - Same as Conventional DDoS (more automation/bw/devices)
  - ***Residential Proxies***

Some Statistics from 2026



Source: https://blog.cloudflare.com/ddos-threat-report-2026-h1/

Residential Proxies

- What are Residential Proxies?
  - Software Development Kits (SDKs) are embedded into applications or consumer devices to create proxies
  - Since these proxies exist on consumer endpoints (tv and streaming devices, doorbells, media players) as well as mobile apps used on smartphones they are classified as ‘residential proxies’
- How are they Exploited?
  - Many developers allow 3rd parties to access SDKs for a fee
  - A threat actor can have millions of devices at their disposal
  - Residential proxies can be exploited to source attack traffic that bypass firewalls
- Known DDoS BotNets that are using 100-200M devices
- Issue: Cannot just block legitimate IP addresses

DDoS BotNets – Nokia Deepfield ERT (public research)

Directory	Description
aisuru	Mirai-derivative DDoS botnet, active since August 2024
cecbot	CECbot: Android TV botnet with HDMI-CEC abuse, successor to Katana
cecilio	CatDDoS derivative with modified RC4 cipher, OpenNIC C2
datasurge	Mirai-lineage bot with no self-propagation; competitor-killing scanner larger than its DDoS engine, plus operator RAT features
ddosia	DDoS client of the pro-Russian hacktivist group NoName057(16); crowdsourced, gamified crypto-reward leaderboard whose self-reported impact is trivially fabricated
drifter	Independent DDoS botnet on ADB attack surface, CCTV-themed C2 domains
ipmoyu	MoYu / BadBox 2.0 residential proxy delivered at runtime by clean grey-market Android-TV IPTV apps (the tigertv family)
iranbot	Self-branded IRAN-BOTNET DDoS family whose three builds shed encryption and branding while gaining a backdoor and then self-propagation, never reusing infrastructure — disposable by design
jackskid	Mirai variant sharing code lineage with Aisuru, DoH C2 via mbedTLS
katana	Mirai variant with on-device compiled rootkit, targeting Android TV set-top boxes
kbotne	Mirai-lineage DDoS botnet with WebSocket C2 on port 80, hex-encoded config strings, and a broken Android APK
kimwolf	Dual-purpose residential proxy and DDoS botnet, 3M+ devices observed
maskify	Dual-purpose proxy/DDoS botnet with ENS, IPFS, and custom P2P mesh
mossadproxy	Android TV/IoT DDoS botnet via ADB, operationally linked to ecosystem

Threat research from the Nokia Deepfield Emergency Response Team (ERT), focused on DDoS botnets and related infrastructure. Each directory covers a botnet family with a brief summary and machine-readable indicators of compromise (IoCs).

Talk at RIPE52: https://pretalx.ripe.net/media/ripe-92/submissions/7AETZG/resources/RIPE_92_-_Nokia_Deep_T2jCpjR.pdf

Detecting DDoS

- Use flow telemetry (NetFlow/IPFIX/sFlow) and application/service logs to determine ‘normal’ behavior
- Utilizing DDoS specific TI sources / feeds
- Characterizing ‘normal’ traffic should include:
  - Protocols, Ports
  - Packet sizes
  - Packets-per-second (pps)
  - Bits-per-second (bps)
  - Connections-per-second (cps)
  - Requests-per-second (rps)
  - TLS negotiation parameters

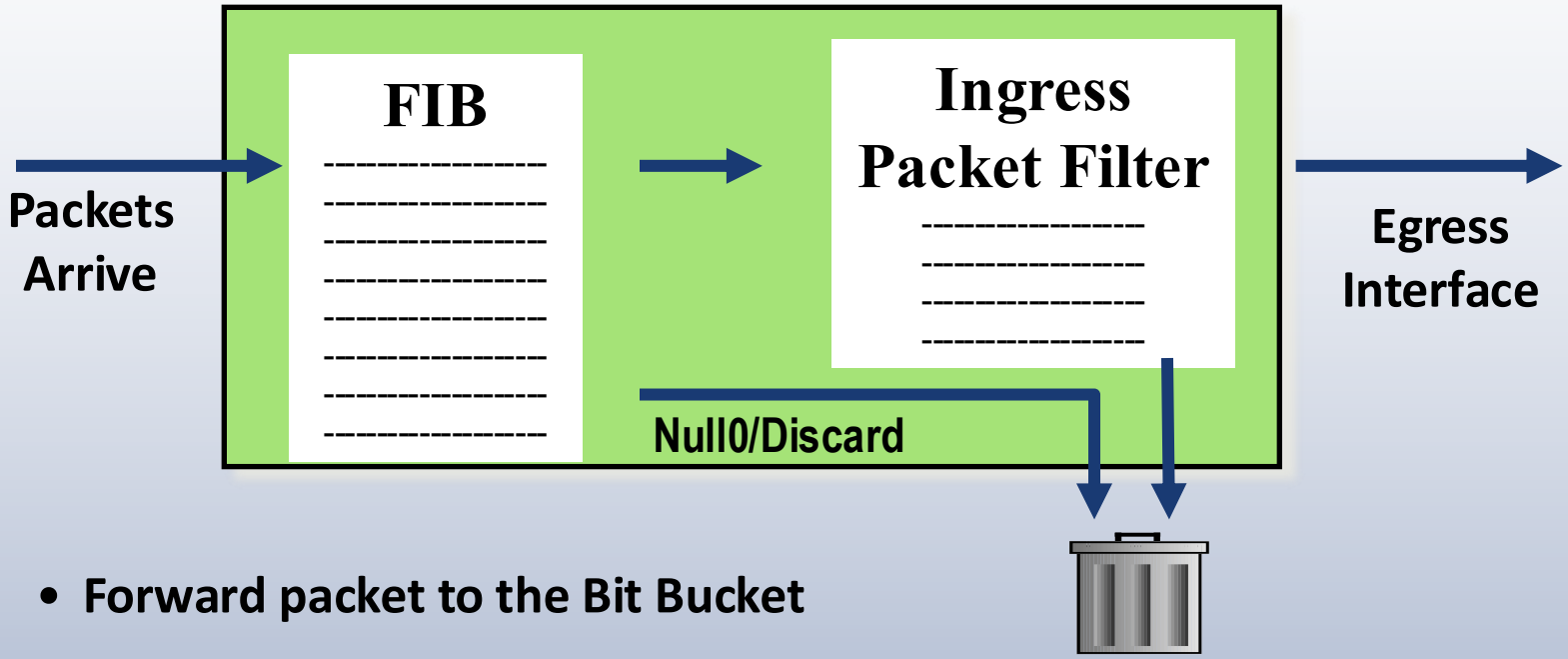
Defending Against DDoS

- Ingress Filtering
  - BCP38 protects against successful IP address spoofing
  - https://www.rfc-editor.org/rfc/rfc2267.html
- Use commercial always-on scrubbing services
- Apply caching and rate limits
- Harden services against amplification attacks
- Use own routing infrastructure to help
  - Manually null route all traffic to IP address under attack
  - *Automated* solution via *Remotely Triggered Blackhole* Filtering (RTBH)

DDoS Mitigation - RTBH Basics

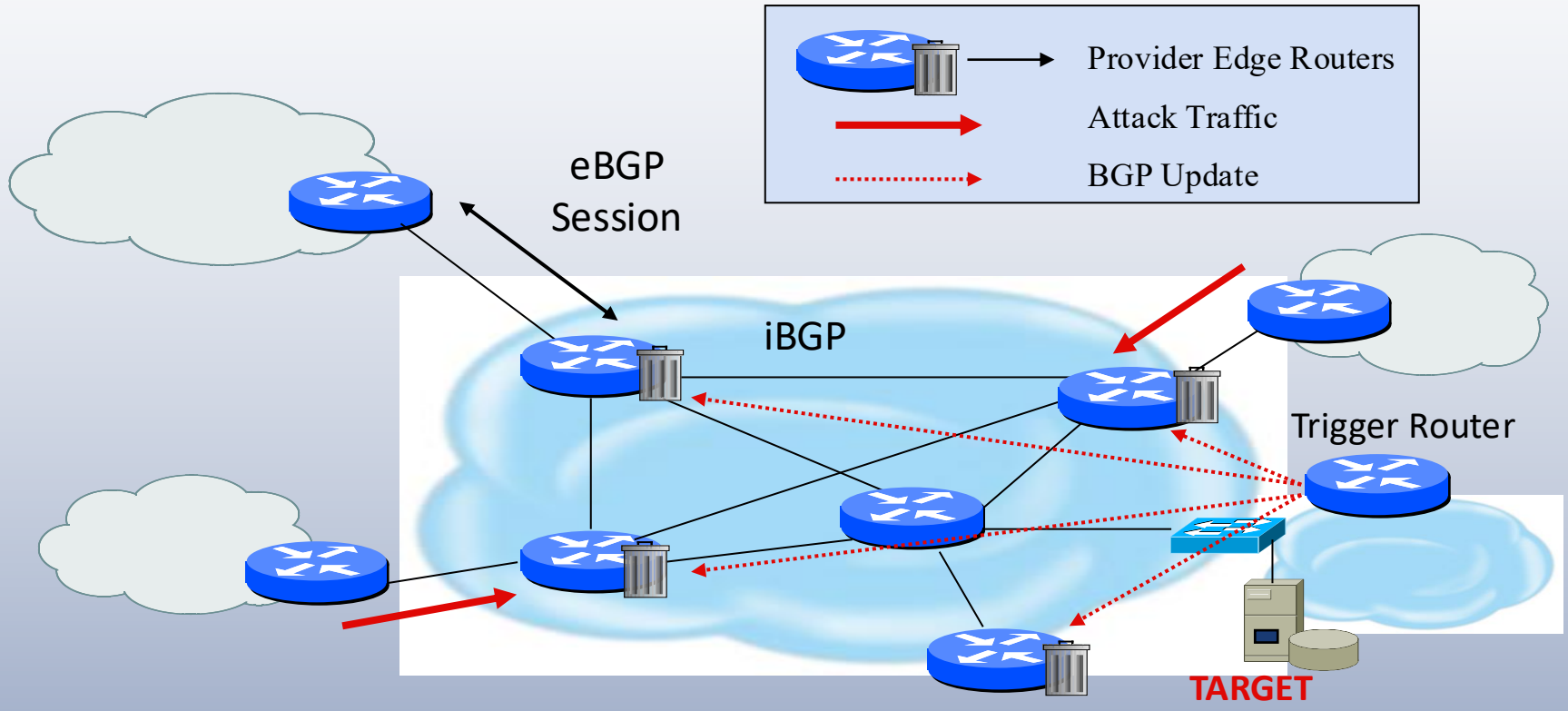
- BGP used to trigger network wide response
  - Exploits router's forwarding logic to drop packets
  - Packets are forwarded to a Null interface (aka 'Discard Interface')
- Effective against spoofed and valid source IP addresses
- Fast response times
  - Static route and BGP allows ISP to trigger network wide black holes as fast as iBGP can update the network
- Operational Deployments / Standardization
  - Operationally used since the early 2000s
  - RFC3882 "Configuring BGP to Block Denial-of-Service Attacks"(2004)
  - RFC5635 "Remotely Triggered Black Hole Filtering with uRPF" (2009)
  - RFC7999 "Blackhole Community" (2016)

Blackhole Filtering CPU Advantages



- Forward packet to the Bit Bucket
- Blackhole Filtering Saves on CPU and ACL processing

Components of RTBH in the Network



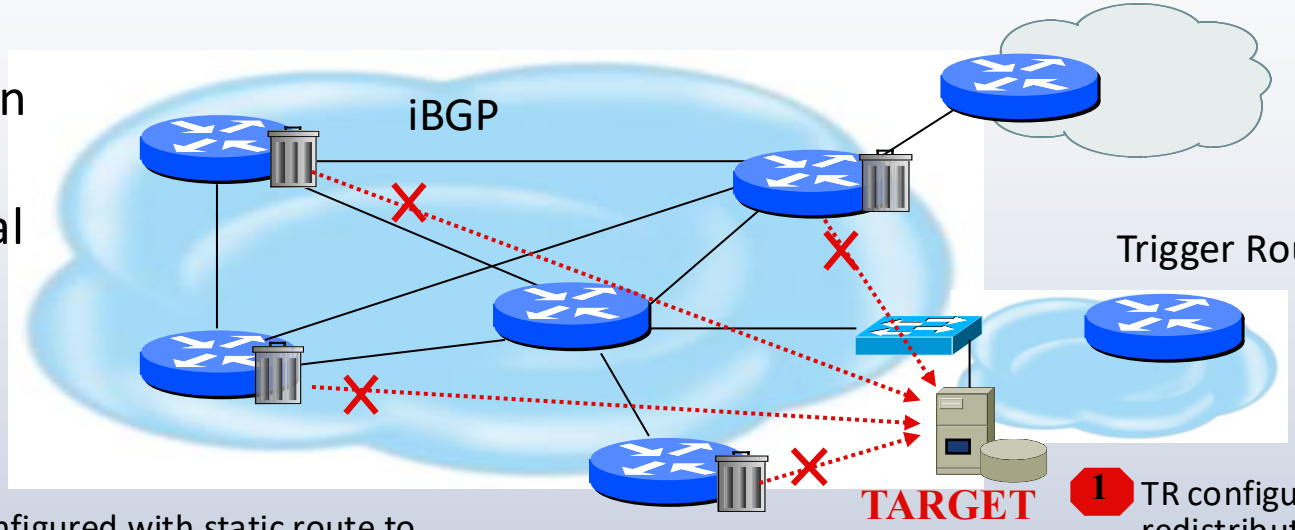
RTBH Configuration Steps

- Configure all edge routers with static route to Null0
  - Must use ‘reserved’ network space
  - IPv4: Typical to use ‘test network 192.0.2.0/24 [e.g. 192.0.2.6]
  - IPv6: Typical to use IANA assigned RTBH prefix from RFC6666 [100::/64]
- Configure trigger router (can be simple BGP device)
  - Configured to redistribute static route to all iBGP peers
  - Recommend using a dedicated router
- Activate black hole
  - Redistribute static route for victim into BGP with next hop set to 192.0.2.6
  - Route is propagated using BGP to all BGP speakers
  - All BGP speakers install victim network reachability via 192.0.2.6 route
  - All traffic to victim is now sent to Null0 (i.e. dropped)

Destination-Based RTBH

Steps

1. Preparation
2. Trigger
3. Withdrawal



1. PE configured with static route to unused space set to Null0 (192.0.2.6/32 set to Null0)
2. Receives iBGP update which states next hop for target is 192.0.2.6/32
3. Installs new (valid) route to target

NOTE: All traffic to the target is dropped, even legitimate traffic

1. TR configured to redistribute static into every iBGP peer
2. Add static route for Victim IP which sets next hop to target destination (192.0.2.6)
3. Manually remove static route which causes BGP route withdrawl

Unicast Reverse Path Forwarding (uRPF)

- Originally created to scale BCP 38 ingress filtering at ISP
  - Customer edge of an ISP's network
- Check router's FIB for matching source IP address entry
- Original mode was strict
  - Is there a matching entry for SRC IP in the router's forwarding table?
  - Is same interface used to reach SRC IP as where packet was received?
- Added loose mode
  - Is there a matching entry for the SRC IP in the router's forwarding table?
  - Does the FIB table point that SRC IP to Null0?
- Loose mode uRPF provided ISPs with the means to trigger a network wide, source-based black hole filter

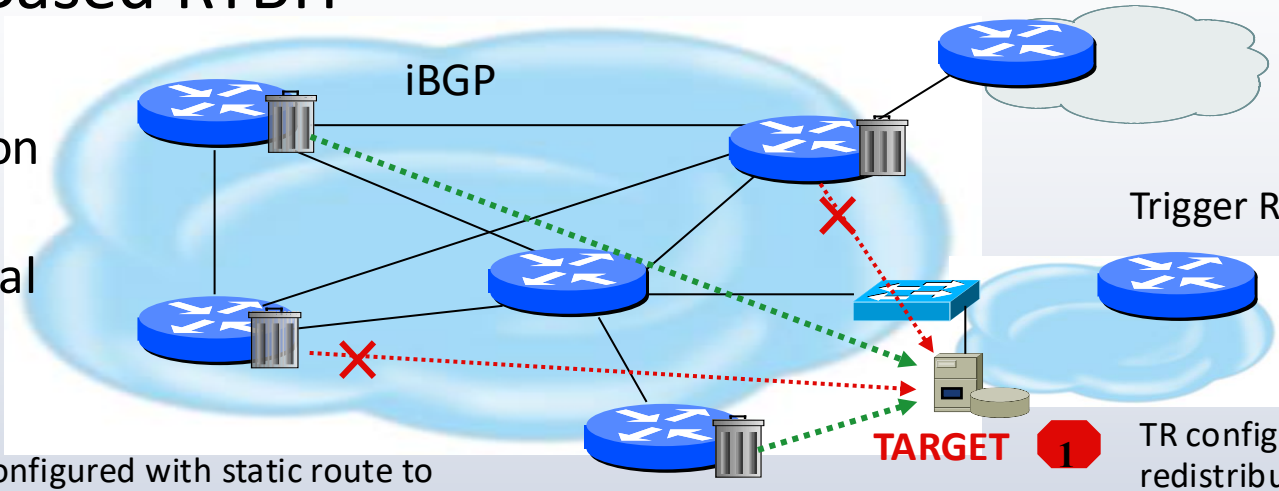
Source Based RTBH

- Edge Router Configuration
  - Loose Mode uRPF configured
  - Static routes using ‘reserved’ IP space set to Null0 [e.g. 192.0.2.6]
- Trigger Router Configuration
  - Configured to redistribute static route to all iBGP peers
- Blackhole activated
  - Adding static route for malicious SRC IP to set next hop to target destination (192.0.2.6)
  - Note that this means SRC IP is the DST IP for the FIB entry
- Traffic that is entering the edge network sourced from a host that has a route pointing to Null0 will result in a uRPF drop.

Source-Based RTBH

Steps

1. Preparation
2. Trigger
3. Withdrawal



1. PE configured with static route to unused space set to Null0 (192.0.2.6/32 set to Null0) and loose mode uRPF on external interfaces
2. Receives iBGP update which states next hop for target is 192.0.2.6/32. All traffic from source IP will fail loose uRPF check.
3. Installs new (valid) route to target

NOTE: Only traffic from the attack sources get dropped

1. TR configured to redistribute static into every iBGP peer
2. Add static route for malicious SRC IP which sets next hop to target destination (192.0.2.6)
3. Manually remove static route which causes BGP route withdrawl

BGP Community Based Trigger

- More finely tuned control over where you drop packets
- Three parts
  - Static routes to Null0 on all the routers
  - Trigger router sets the community
  - Reaction routers (on the edge) matches community and sets the next-hop to the static route to Null0
- Example communities
  - Trigger community #1 can be for all routers in the network
  - Trigger community #2 can be for all peering routers; no customer routers which allows for customers to talk to the DOSed customer within your AS

Configuring Community Based Trigger Router

- Create a route map to catch routes which need to be blackholed
  - Static routes can be tagged in Cisco IOS
    - Tag routes to be blackholed with the value of 66
  - Set origin to be iBGP
  - Set local-preference to be 200
    - Higher than any other local-preference set in the backbone
  - Set community to be no-export and RTBH community (65535:666)
    - Don’t want prefix to leak outside the AS
  - Set next-hop to 192.0.2.6 (IPv4) or 100::1 (IPv6)

RTBH Configuration Example


```
interface Null0
no ip unreachables
!
ip route 192.0.2.6 255.255.255.255 null0
```

```
interface Null0
! avoid backscatter traffic
no ip unreachables
!
router bgp 6665
redestribute static route-map bh-trigger
!
route map bh-trigger permit 10
  match tag 77
  set local-preference 200
  set origin igp
! ensure edge router does not leak prefix outside AS
  set community no-export 65535:666
  set ip next-hop 192.0.2.6
! make sure no other static routes affected by the
! bh-trigger route map
route-map bh-trigger deny 20
!
! the manually configured trigger which ensures that
!192.168.33.0/24 is announced with next hop 192.0.2.6
ip route 192.168.33.0 255.255.255.0 null0 tag77
```

Added Resources

- Cloudflare Blog: **Cloudflare DDoS Threat Report H1 2026**
  - https://blog.cloudflare.com/ddos-threat-report-2026-h1/
- Kentik Blog on DDoS Protection and Mitigation
  - https://www.kentik.com/kentipedia/ddos-protection/
- Netscout DDoS Intelligence Report
  - https://www.netscout.com/threatreport/
- The Proxy-for-Profit Economy: Consumer Devices as Attack Infrastructure [good resource from CSA]
  - https://labs.cloudsecurityalliance.org/research/csa-research-note-residential-proxy-economy-systemic-risk-20/
- Amplification Hell [Christian Rossow]
  - https://www.ndss-symposium.org/wp-content/uploads/2017/09/01_5.pdf