Welcome to

# DDoS Workshop

## The Camp 2013

Henrik Lund Kramshøj, internet samurai
hlk@solido.net

http://www.solidonetworks.com

# Agenda

Intro

Graphs and Dashboards!

Taxonomy of DDoS Attacks

Netflow NFSen

Defense in depth - multiple layers of security

Routing RTBH

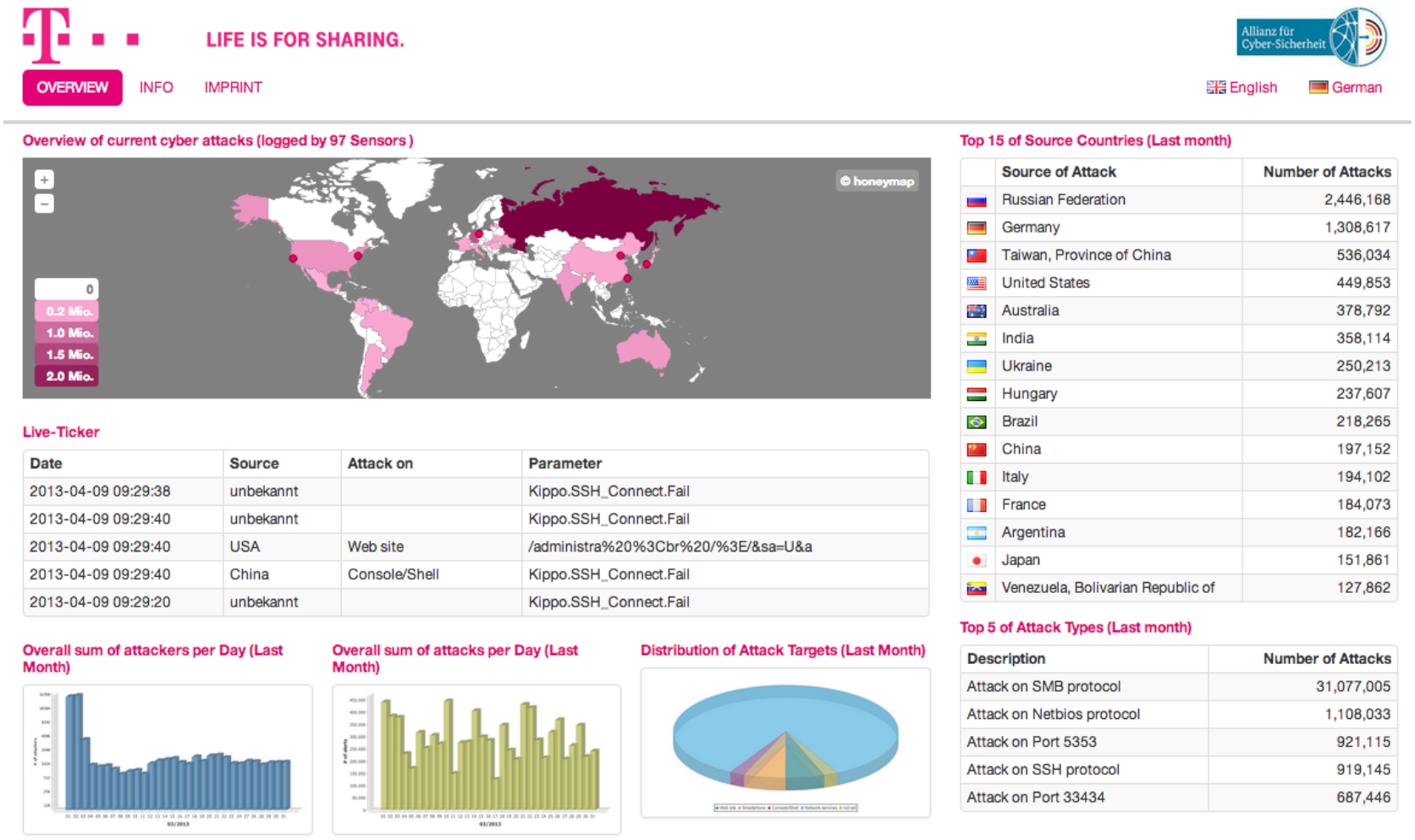
Troubleshooting

# Intro

DDoS is very much in the media

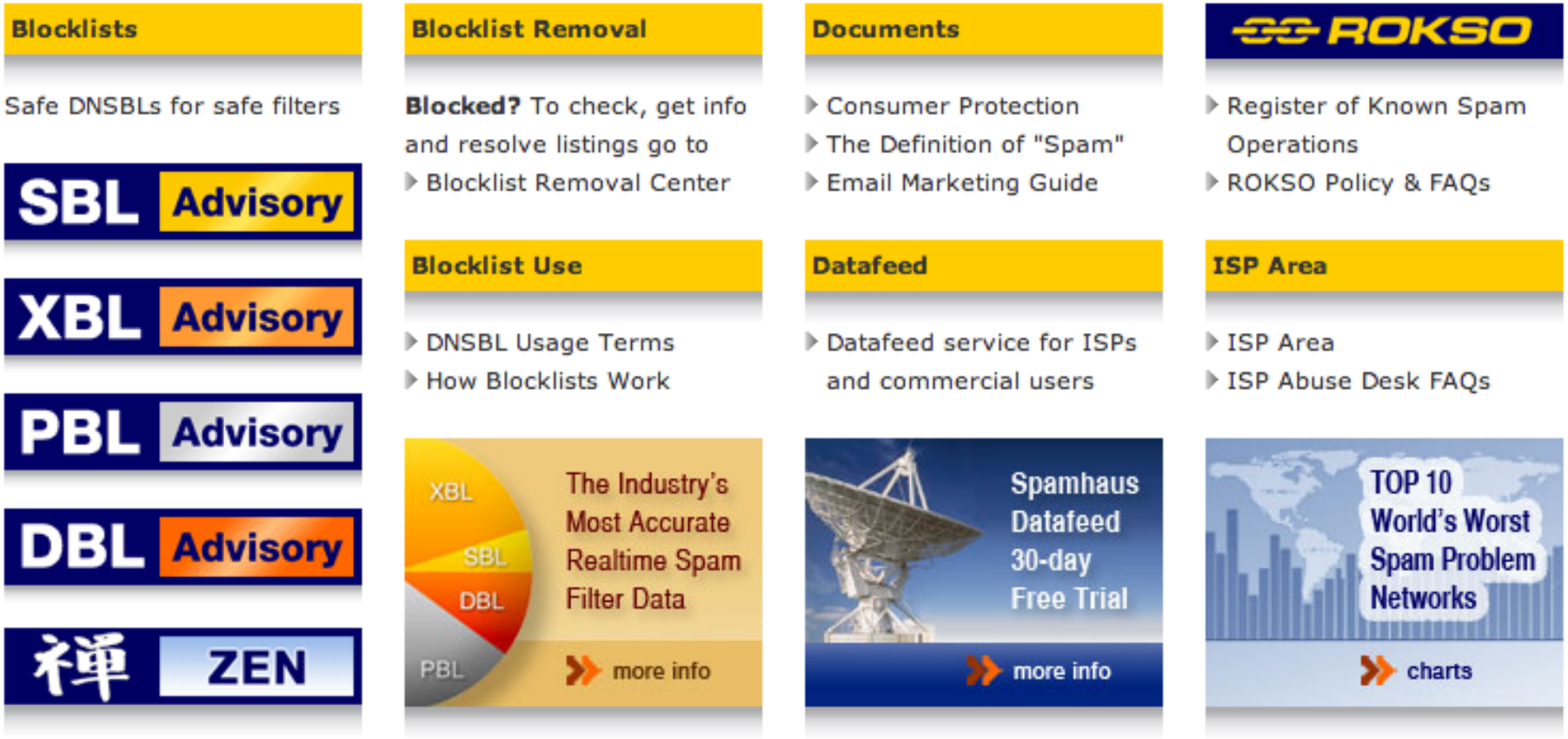
Vendors say:
Prolexic did mitigate a 130 Gbps attack in March and more than 10 percent of attacks directed at Prolexic's global client base exceeded 60 Gigabits per second (Gbps). Source: Prolexic Quarterly Global DDoS Attack Report Q1 2013

# Attack overview




http://www.sicherheitstacho.eu/?lang=en

# Spamhaus




Source: `http://www.spamhaus.org/`

# Massive DDoS

Title: Massive DDoS against Spamhaus reaches 300Gbps Description: Following a dispute between Dutch hosting provider Cyberbunker and anti-spam group Spamhous, the latter suffered what initially began as a relatively small - 10 Gbps - DDoS, which escalated over the course of last week to a 300Gbps flood.

Source: `http://blog.cloudflare.com/the-ddos-that-almost-broke-the-internet`

# Massive DDoS a lie

SOLIDO NETWORKS

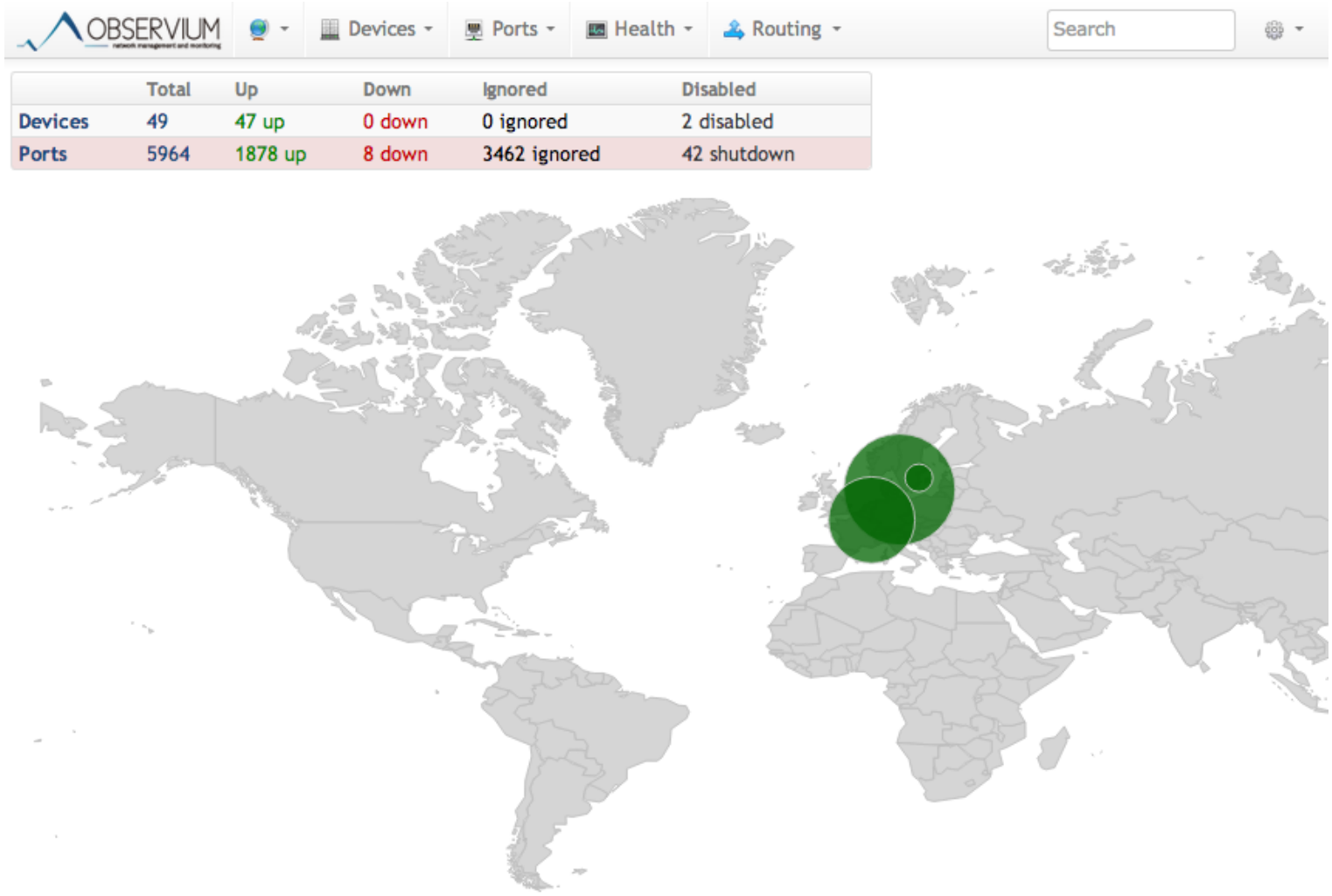
> CloudFlare CEO Matthew Prince said he was sure of the 300Gbps figure, pointing to an online comment from Richard Steenbergen, CTO of nLayer, one of the upstream network providers of CloudFlare. Although Steenbergen said the company saw a 300Gbps hit going after CloudFlare, which targeted ”pieces” of the core network, it was nothing ”record smashing” or ”game changing”
>
> Actual data proving a 300Gbps hit remains thin on the ground. Hammack said his firm had not seen anything above 160Gbps in a single DDoS, with 144 million packets sent per second, and he doesn’t believe there has been one higher. He won’t be convinced otherwise unless someone shows him proof one organisation’s network took more traffic in an attack.

Source: Prolexic CEO: Biggest Cyber Attack Ever Was Built On Lies

`http://www.techweekeurope.co.uk/news/prolexic-ceo-scott-hammack-biggest-cyber-attack-lies-spam`

Ohhh only 160Gbps ☺

# Graphs and Dashboards!




https://observium.solido.net/

# Riorey Taxonomy of DDoS Attacks

SOLIDO NETWORKS

What are DDoS? and DoS?

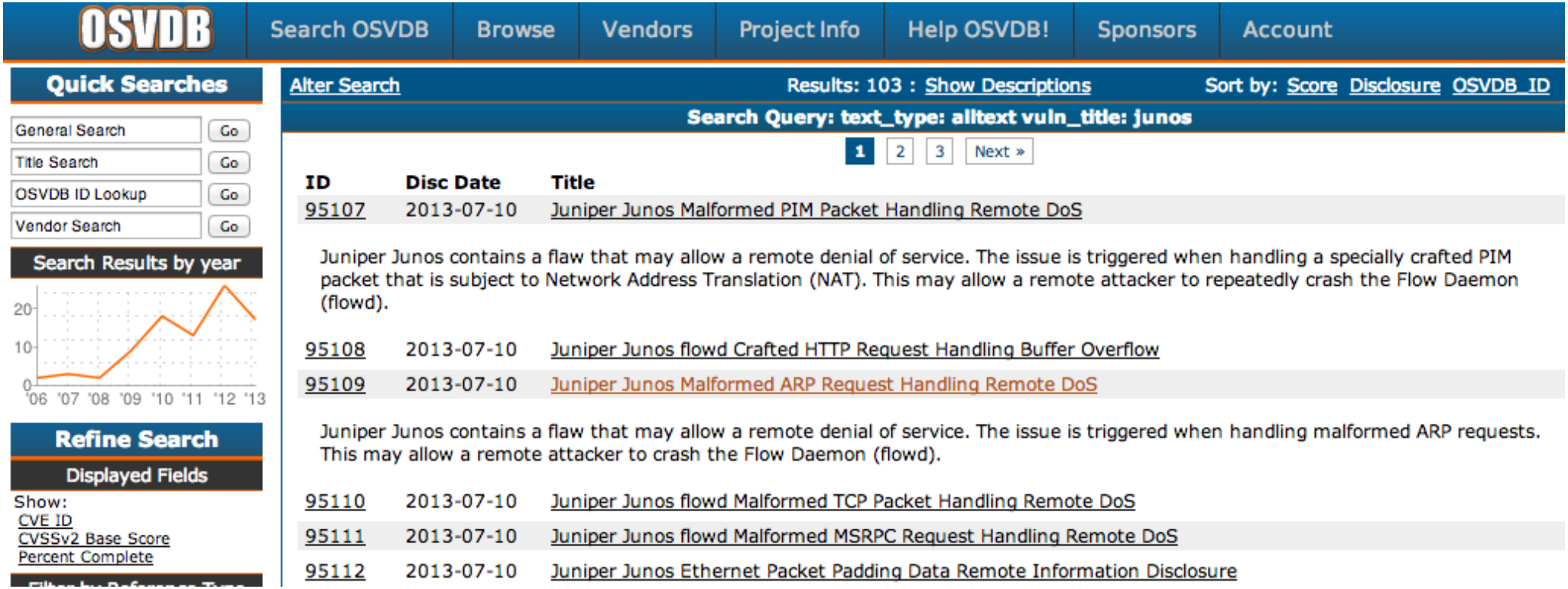
Denial of Service attack - prevents authorized users access to resources

Can be a single request to HTTP service, sequence of network packets

Distributed Denial of Service attack - many (spoofed) sources

`https://en.wikipedia.org/wiki/Denial-of-service_attack`

# Denial of Service description




Source: `http://osvdb.org/`

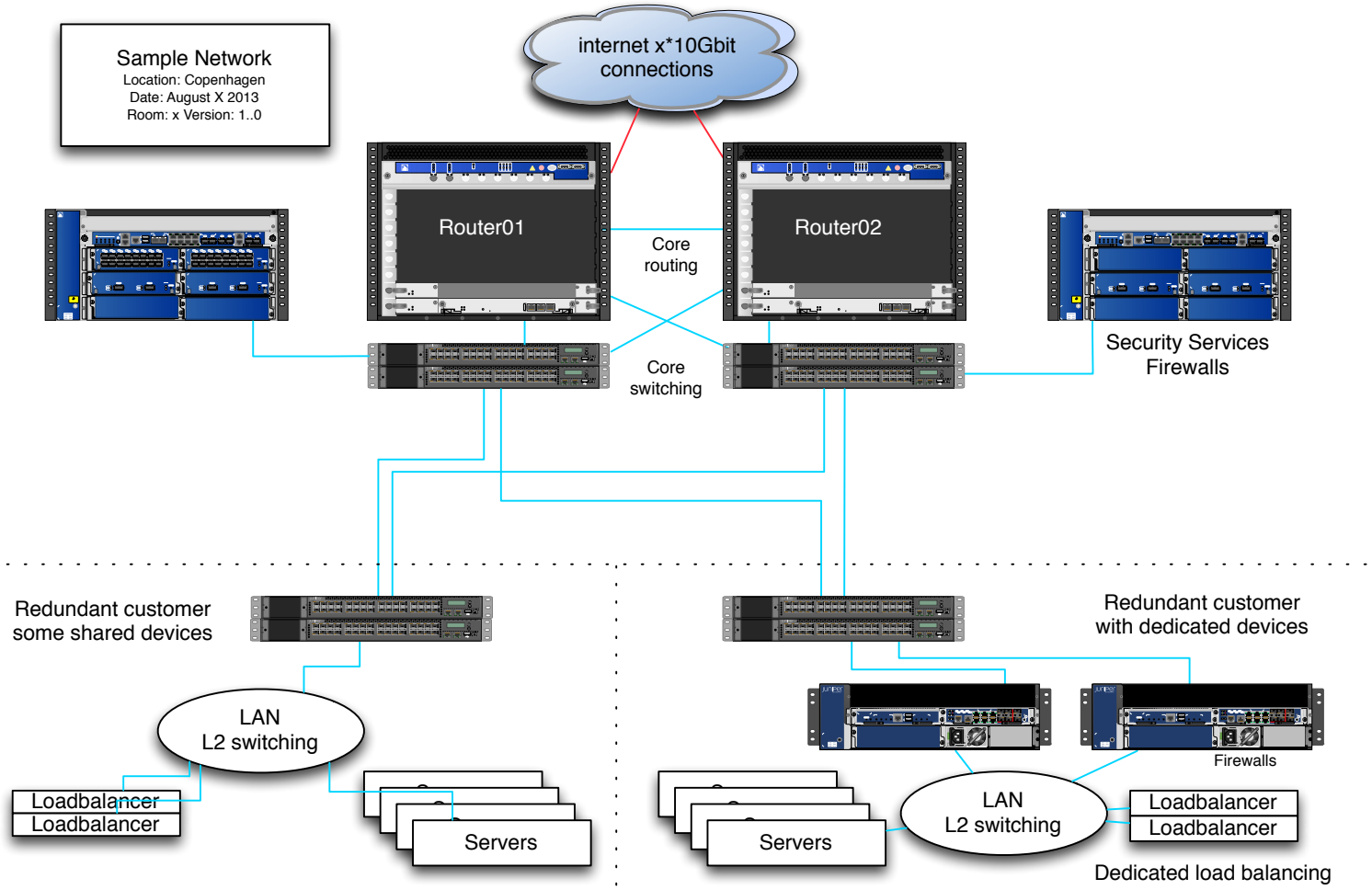
# Cisco DoS *exploit* script

SOLIDO NETWORKS

```
#!/bin/sh
# 2003-07-21 pdonahue
# cisco-44020.sh
# -- this shell script is just a wrapper for hping (http://www.hping.org)
#    with the parameters necessary to fill the input queue on
# exploitable IOS device
# -- refer to "Cisco Security Advisory: Cisco IOS Interface Blocked by
# IPv4 Packets"
# (http://www.cisco.com/warp/public/707/cisco-sa-20030717-blocked.shtml)
#for more information
...
for protocol in $PROT
    do
       $HPING $HOST --rawip $ADDR --ttl $TTL --ipproto $protocol

       --count $NUMB --interval u250 --data $SIZE --file /dev/urandom
    done
```

# Networks today



# Taxonomy of DDoS Attacks

SOLIDO NETWORKS

## RioRey Taxonomy of DDoS Attacks

| Attack Types | | | Attack Matrix Dimensions | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | Nature of IP | Handshake | Source IP Range | Packet Rate | Packet Size | Packet Content | Fragmenting | Session Rate | Session Duration | VERB Rate |
| TCP BASED | 1 | SYN Flood | Spoofed | None | Large | High | Small | --- | --- | --- | --- | --- |
| | 2 | SYN-ACK Flood | Spoofed | None | Large | High | --- | --- | --- | --- | --- | --- |
| | 3 | ACK & PUSH ACK Flood | Spoofed | None | Large | High | --- | --- | --- | --- | --- | --- |
| | 4 | Fragmented ACK | Spoofed | None | Large | Moderate | Large | --- | High | --- | --- | --- |
| | 5 | RST or FIN Flood | Spoofed | None | Large | High | --- | --- | --- | --- | --- | --- |
| | 6 | Synonymous IP | Spoofed | None | Single IP | High | --- | --- | --- | --- | --- | --- |
| | 7 | Fake Session | Spoofed | None | Large | Low | --- | --- | --- | --- | --- | --- |
| | 8 | Session Attack | Non-Spoofed | Yes | Small | Low | --- | --- | --- | Low | Long | --- |
| | 9 | Misused Application | Non-Spoofed | Yes | Small | Variable | --- | --- | --- | High | Short | --- |

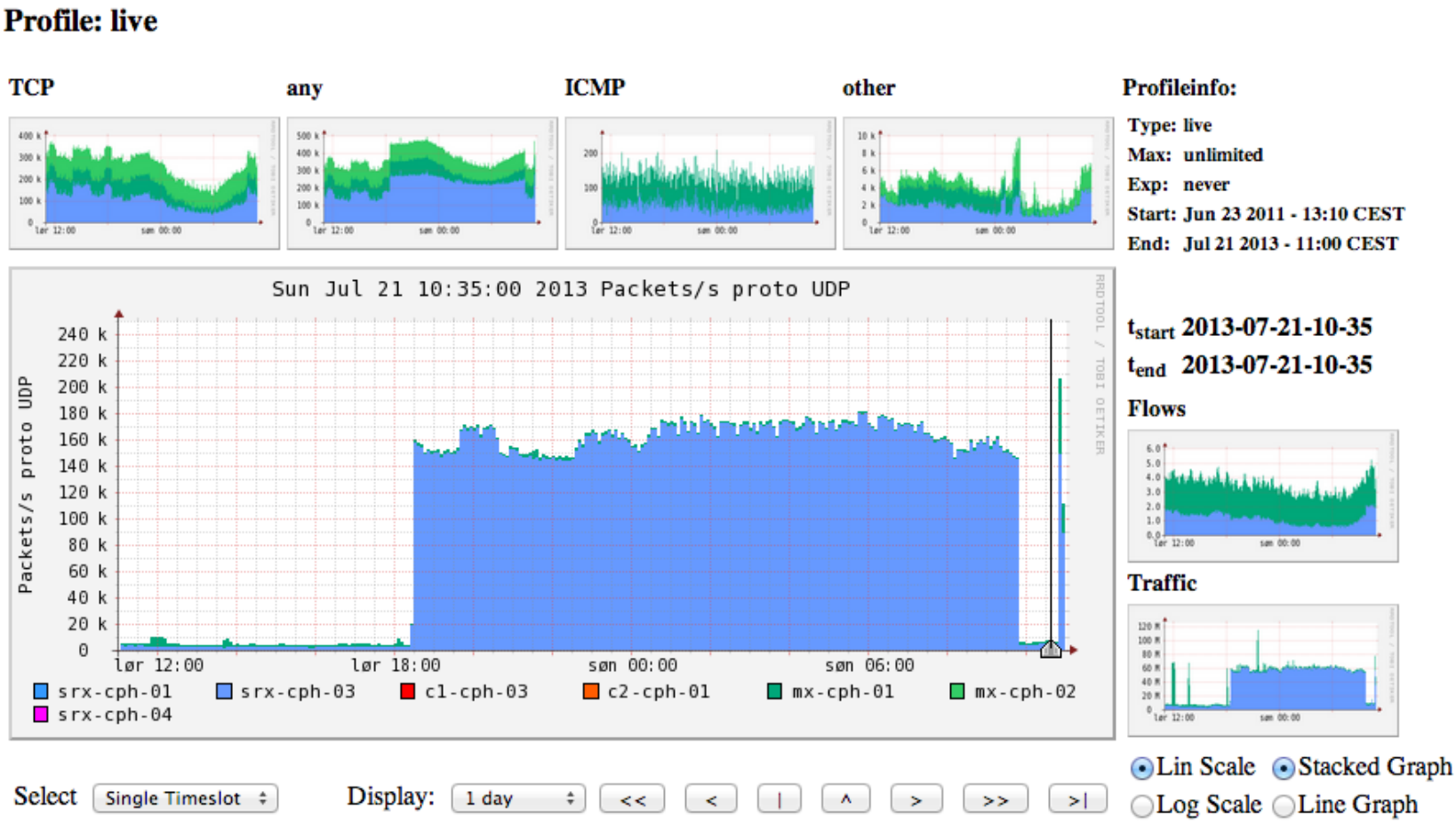
| | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| TCP HTTP BASED | 10 | **HTTP Fragmentation** | Non-Spoofed | Yes | Small | Very Low | Small | Valid | High | Very Low | Very Long | Very Low |
| | 11 | **Excessive VERB** | Non-Spoofed | Yes | Small | High | --- | Valid | --- | High | Short | High |
| | 12 | **Excessive VERB Single Session** | Non-Spoofed | Yes | Small | Low | --- | Valid | --- | Low | Moderate | High |
| | 13 | **Multiple VERB Single Request** | Non-Spoofed | Yes | Small | Very Low | Large | Valid | --- | Low | Long | High |
| | 14 | **Recursive GET** | Non-Spoofed | Yes | Small | Low | --- | Valid | --- | Low | Short | Low |
| | 15 | **Random Recursive GET** | Non-Spoofed | Yes | Small | Low | --- | Valid | --- | Low | Short | Low |
| | 16 | **Faulty Application** | Non-Spoofed | Yes | Small | Low | --- | Valid | --- | Low | Short | Low |

# UDP and ICMP

| | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| UDP BASED | 17 | UDP Flood | Spoofed | --- | Very Large | Very High | Small | Not Valid | --- | --- | --- | --- |
| | 18 | Fragmentation | Spoofed | --- | Moderate | Very High | Large | Not Valid | High | --- | --- | --- |
| | 19 | DNS Flood | Spoofed | --- | Very Large | Very High | Small | Valid | --- | --- | --- | --- |
| | 20 | VoIP Flood | Spoofed | --- | Very Large | Very High | Small | Valid | --- | --- | --- | --- |
| | 21 | Media Data Flood | Spoofed | --- | Very Large | Very High | Moderate | Valid | --- | --- | --- | --- |
| | 22 | Non-Spoofed UDP Flood | Non-Spoofed | --- | Small | Very High | --- | Valid | --- | --- | --- | --- |

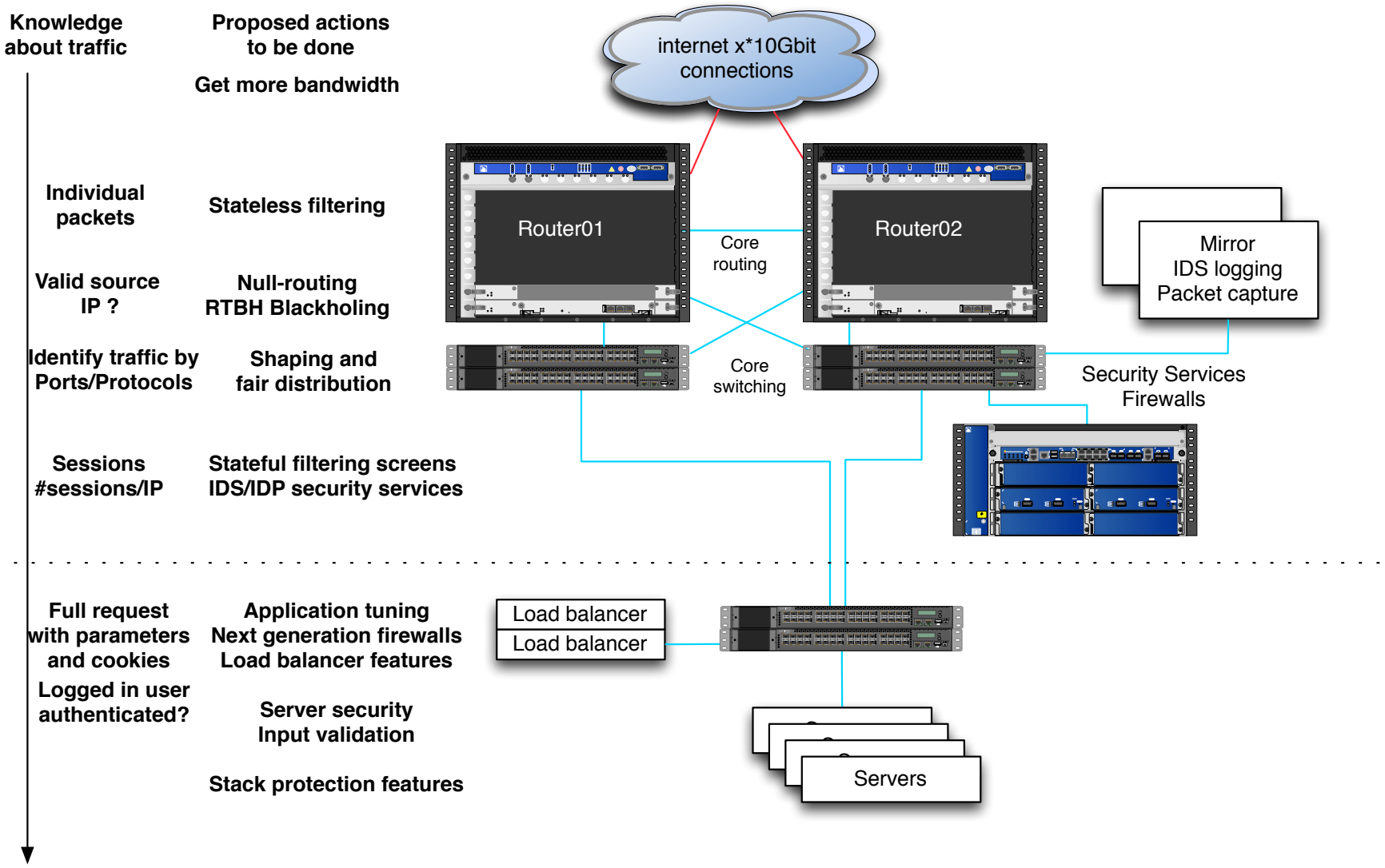
| | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| ICMP BASED | 23 | ICMP Flood | Spoofed | --- | Very Large | Very High | Variable | Not Valid | --- | --- | --- | --- |
| | 24 | Fragmentation | Spoofed | --- | Moderate | Very High | Large | Not Valid | High | --- | --- | --- |
| | 25 | Ping Flood | Spoofed | --- | Very Large | Very High | Small | Valid | --- | --- | --- | --- |

# Netflow NFSen

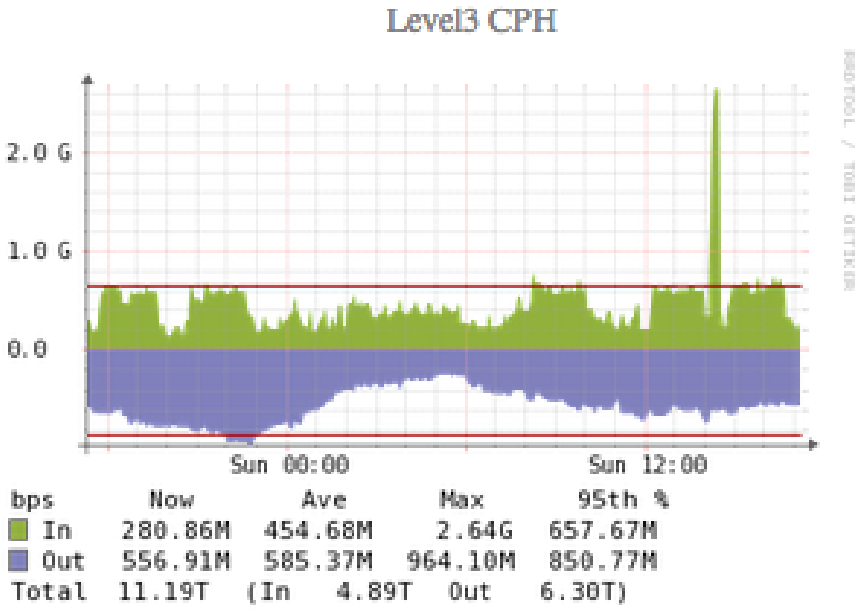


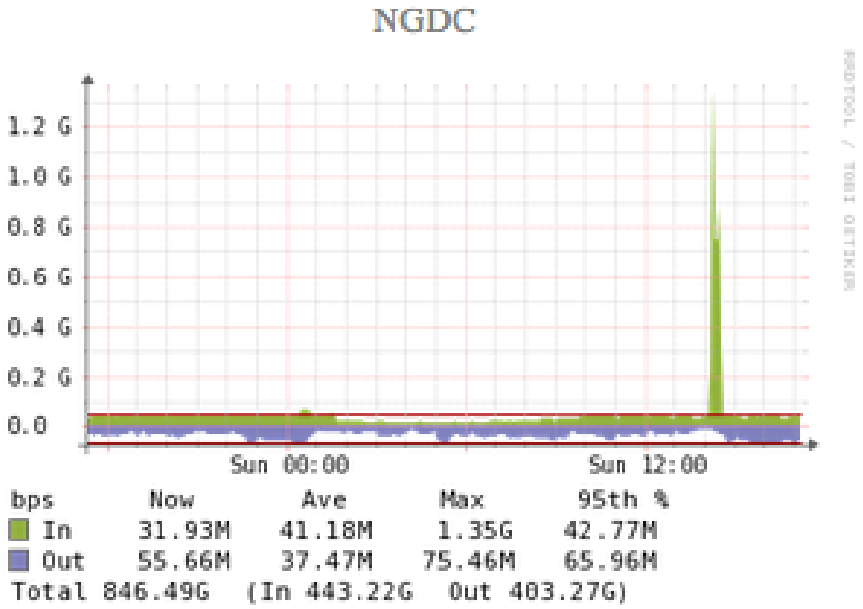

An extra 100k packets per second from this netflow source (source is a router)

# Defense in depth - multiple layers of security



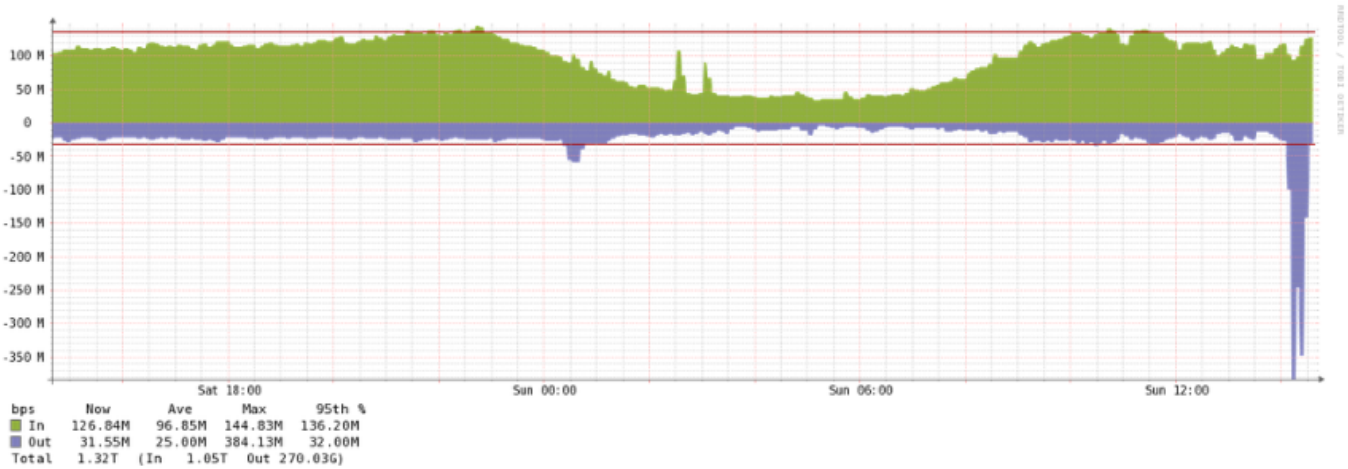
# DDoS traffic before filtering




Only two links shown, at least 3Gbit incoming for this single IP

# DDoS traffic after filtering




Link toward server (next level firewall actually) about 350Mbit outgoing

# Stateless firewall filter throw stuff away

```
hlk@MX-CPH-02> show configuration firewall filter all | no-more
/* This is a sample, better to use BGP flowspec and RTBH */
inactive: term edgeblocker {
    from {
        source-address {
            84.180.xxx.173/32;
...
            87.245.xxx.171/32;
        }
        destination-address {
            91.102.91.16/28;
        }
        protocol [ tcp udp icmp ];
    }
    then {
        count edge-block;
        discard;
    }
}
```

Hint: can also leave out protocol and then it will match all protocols

# Stateless firewall filter limit protocols

```
term limit-icmp {
    from {
        protocol icmp;
    }
    then {
        policer ICMP-100M;
        accept;
    }
}
term limit-udp {
    from {
        protocol udp;
    }
    then {
        policer UDP-1000M;
        accept;
    }
}
```

Routers have extensive Class-of-Service (CoS) tools today

# Strict filtering for some servers, still stateless!

```
term some-server-allow {
    from {
        destination-address {
            109.238.xx.0/xx;
        }
        protocol tcp;
        destination-port [ 80 443 ];
    }
    then accept;
}
term some-server-block-unneeded {
    from {
        destination-address {
            109.238.xx.0/xx;
        }
        protocol-except icmp;
    }
    then {
        count some-server-block;
        discard;
    }
}
```

Wut - no UDP, yes UDP service is not used on these servers

# Firewalls - screens, IDS like features

When you know regular traffic you can decide:

```
hlk@srx-kas-05# show security screen ids-option untrust-screen
icmp {
    ping-death;
}
ip {
    source-route-option;
    tear-drop;
}
tcp {    Note: UDP flood setting also exist
    syn-flood {
        alarm-threshold 1024;
        attack-threshold 200;
        source-threshold 1024;
        destination-threshold 2048;
        timeout 20;
    }
    land;
} Always select your own settings YMMV
```

# Routing RTBH

What about a really big DDoS?

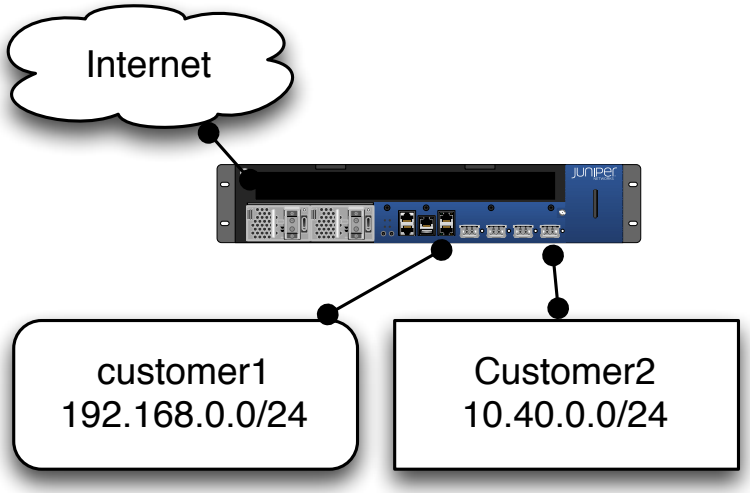
and routers can do more

# uRPF unicast Reverse Path Forwarding

Reverse path forwarding (RPF) is a technique used in modern routers for the purposes of ensuring loop-free forwarding of multicast packets in multicast routing and to help prevent IP address spoofing in unicast routing.

Source: `http://en.wikipedia.org/wiki/Reverse_path_forwarding`

# Strict vs loose mode RPF




```
user@router# show interfaces
ge-0/0/0 {
        unit 2 {
                family inet {
                        rpf-check fail-filter rpf-special-case-dhcp;
                        address 192.168.0.254/24;
                }
        }
}
ge-0/0/1 {
        unit 2 {
                family inet {
                        rpf-check fail-filter rpf-special-case-dhcp;
                        address 10.40.0.254/24;
                }
        }
}
```

**Configuring Unicast RPF Strict Mode**

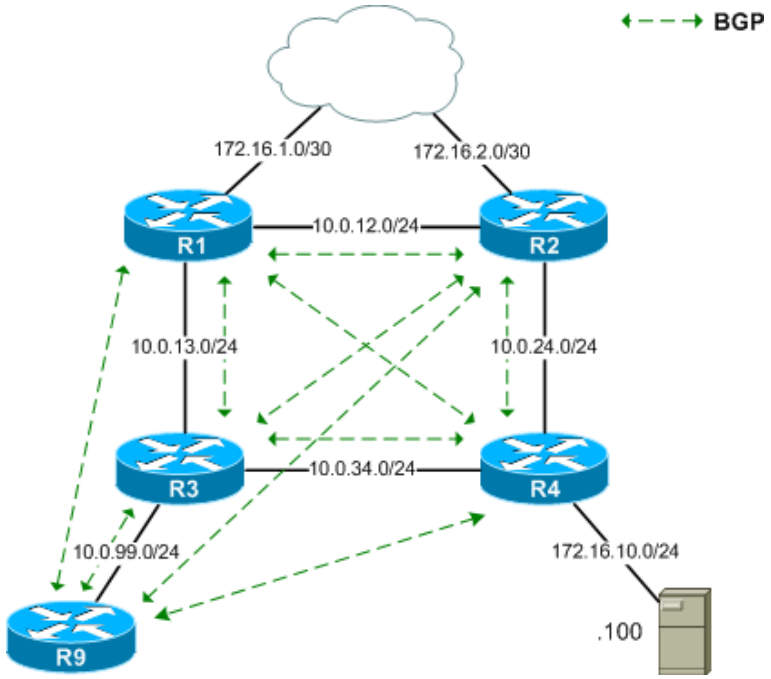
In strict mode, unicast RPF checks whether the incoming packet has a source address that matches a prefix in the routing table, **and whether the interface expects to receive a packet with this source address prefix.**

# uRPF Junos config with loose mode

```
xe-5/1/1 {
    description "Transit: Blah (AS65512)";
    unit 0 {
        family inet {
            rpf-check {
                mode loose;
            }
            filter {
                input all;
                output all;
            }
            address xx.yy.xx.yy/30;
        }
        family inet6 {
            rpf-check {
                mode loose;
            }
            address 2001:xx:yy/126;
        }
    }
}
```

See also: `http://www.version2.dk/blog/den-danske-internettrafik-og-bgp-49401`

# Remotely Triggered Black Hole Configurations




Picture from packetlife.net showing R9 as a standalone ”management” router for route injection.

`http://packetlife.net/blog/2009/jul/6/remotely-triggered-black-hole-rtbh-routing/`

`https://ripe65.ripe.net/presentations/285-inex-ripe-routingwg-amsterdam-2012-09-27.pdf`

`https://www.inex.ie/rtbh`

# Remotely Triggered Black Hole at upstreams

```
6.  Black Hole Server (Optional)

   ##################################################################
   #                            NOTE                                #
   #  The Cogent Black Hole server will allow customers to announce a /32 route      #
   #  to Cogent and have all traffic to that network blocked at Cogents backbone.    #
   #  All peers on the Cogent black hole server require a password and IP address    #
   #  from your network for Cogent to peer with.                                     #
   ##################################################################

       [   ]  Please set up a BGP peer on the Cogent Black Hole server

       Black Hole server password:
       Black Hole server peer IP:

       North American Black Hole Peer:  66.28.8.1
       European Black Hole Peer:  130.117.20.1
```

Source:

http://cogentco.com/files/docs/customer_service/guide/bgpq.sample.txt

Better drop single /32 host than whole network!

# BGP speaking software

Example BGP speakers - which can be used for RTBH

OpenBGPD from OpenBSD easy to install, also on FreeBSD, NetBSD

BIRD `http://bird.network.cz/` great BGP daemon, used as Router Server in some internet exchanges

Vyatta `http://www.vyatta.org/` complete BGP routing, firewall etc.

Exabgp `http://code.google.com/p/exabgp/` BGP engine useful for injecting routes and flowspec

Unfortunately there is not a lot of open source ”scrubbing” software

# BCP38 Network Ingress Filtering

SOLIDO NETWORKS

Network Working Group
Request for Comments: 2827
Obsoletes: 2267
BCP: 38
Category: Best Current Practice

P. Ferguson
Cisco Systems, Inc.
D. Senie
Amaranth Networks Inc.
May 2000

Network Ingress Filtering:
Defeating Denial of Service Attacks which employ
IP Source Address Spoofing

Note: you should try validating INCOMING traffic from customers, also note the date!

http://tools.ietf.org/html/bcp38

# Remember those BGP import filters, perhaps try bgpq3

```
hlk@katana:bgpq3-0.1.16$ ./bgpq3 -Jl larsen-data AS197495
policy-options {
replace:
 prefix-list larsen-data {
     91.221.196.0/23;
     185.10.8.0/22;
 }
}
```

http://snar.spb.ru/prog/bgpq3/

# The Spamhaus Don’t Route Or Peer Lists

SOLIDO NETWORKS

The Spamhaus Don’t Route Or Peer Lists

DROP (Don’t Route Or Peer) and EDROP are advisory ”drop all traffic” lists, consisting of stolen ’hijacked’ netblocks and netblocks controlled entirely by criminals and professional spammers. DROP and EDROP are a tiny subset of the SBL designed for use by firewalls and routing equipment.

`http://www.spamhaus.org/drop/`

# Flowspec Self inflicted DoS

```
+     route 173.X.X.X/32-DNS-DROP {
+         match {
+             destination 173.X.X.X/32;
+             port 53;
+             packet-length [ 99971 99985 ];
+         }
+         then discard;
+     }
```

Resulted in router crashes - ooopps

http://blog.cloudflare.com/todays-outage-post-mortem-82515

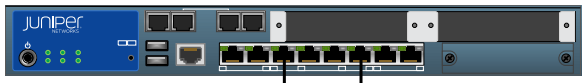
http://www.slideshare.net/sfouant/an-introduction-to-bgp-flow-spec

https://code.google.com/p/exabgp/wiki/flowspec

http://www.slideshare.net/junipernetworks/flowspec-bay-area-juniper-user-group-

# Demo network

Juniper SRX220H
Firewall performance (max) 950 Mbps
Maximum concurrent sessions 96K

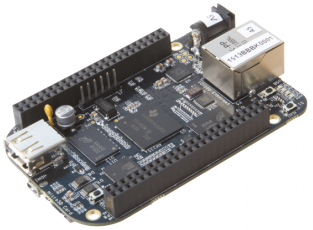


Intel Core i7 - 960 (3.2GHz)
16Gb memory Gbit + 10G NIC

**Bad girl!**

Nice Web
server Nginx

Beagleboard Black
Processor: AM335x 1GHz ARM®
Cortex-A8 512MB DDR3L



SOLIDO NETWORKS

## Getting SNMP data from Juniper SRX:

```
hlk@srx-kas-05> show snmp mib walk jnxJsSPUMonitoringObjectsTable
jnxJsSPUMonitoringFPCIndex.0 = 0
jnxJsSPUMonitoringSPUIndex.0 = 0
jnxJsSPUMonitoringCPUUsage.0 = 0
jnxJsSPUMonitoringMemoryUsage.0 = 50
jnxJsSPUMonitoringCurrentFlowSession.0 = 20
jnxJsSPUMonitoringMaxFlowSession.0 = 65536
jnxJsSPUMonitoringCurrentCPSession.0 = 0
jnxJsSPUMonitoringMaxCPSession.0 = 0
jnxJsSPUMonitoringNodeIndex.0 = 0
jnxJsSPUMonitoringNodeDescr.0 = single
```

# SNMP OID value

SOLIDO NETWORKS

You should download and install MIBs, but quick and dirty:

```
hlk@srx-kas-05> show snmp mib walk jnxJsSPUMonitoringObjectsTable | display xml
<rpc-reply xmlns:junos="http://xml.juniper.net/junos/11.4R8/junos">
    <snmp-object-information xmlns="http://xml.juniper.net/junos/11.4R8/junos-snmp">
...
<snmp-object>
            <name>jnxJsSPUMonitoringCurrentFlowSession.0</name>
            <index>
                <index-name>jnxJsSPUMonitoringIndex</index-name>
                <index-value>0</index-value>
            </index>
            <object-value-type>gauge</object-value-type>
            <object-value>20</object-value>
            <oid> 1.3.6.1.4.1.2636.3.39.1.12.1.1.1.6.0</oid>
        </snmp-object>

hlk@katana:hlk$ snmpget -v 2c -c public 192.168.18.236  1.3.6.1.4.1.2636.3.39.1.12.1.1.1.6.0
SNMPv2-SMI::enterprises.2636.3.39.1.12.1.1.1.6.0 = Gauge32: 26
```

# With installed MIBs

SOLIDO NETWORKS

You can use better descriptive references:

```
hlk@katana:hlk$ snmpget -v 2c -c public 192.168.18.236 jnxJsSPUMonitoringCurrentFlowSession.0
JUNIPER-SRX5000-SPU-MONITORING-MIB::jnxJsSPUMonitoringCurrentFlowSession.0 = Gauge32: 45
```

# Ruby SNMP

```
#!/usr/bin/env ruby
require 'net-snmp'
router='192.168.18.236'

#snmpget -v 2c -c public 192.168.18.236 1.3.6.1.4.1.2636.3.39.1.12.1.1.1.6.0
session = Net::SNMP::Session.open(:peername => router, :community => "public" )
begin
  pdu = session.get("1.3.6.1.4.1.2636.3.39.1.12.1.1.1.6.0")
  puts pdu.varbinds.first.value
rescue Net::SNMP::Error => e
  puts e.message
end
session.close

hlk@katana:ruby-snmp-graph$ ./test2.rb
18
```

# Kali Linux the new backtrack


BackTrack `http://www.backtrack-linux.org`

Kali `http://www.kali.org/`

# it's a Unix system, I know this

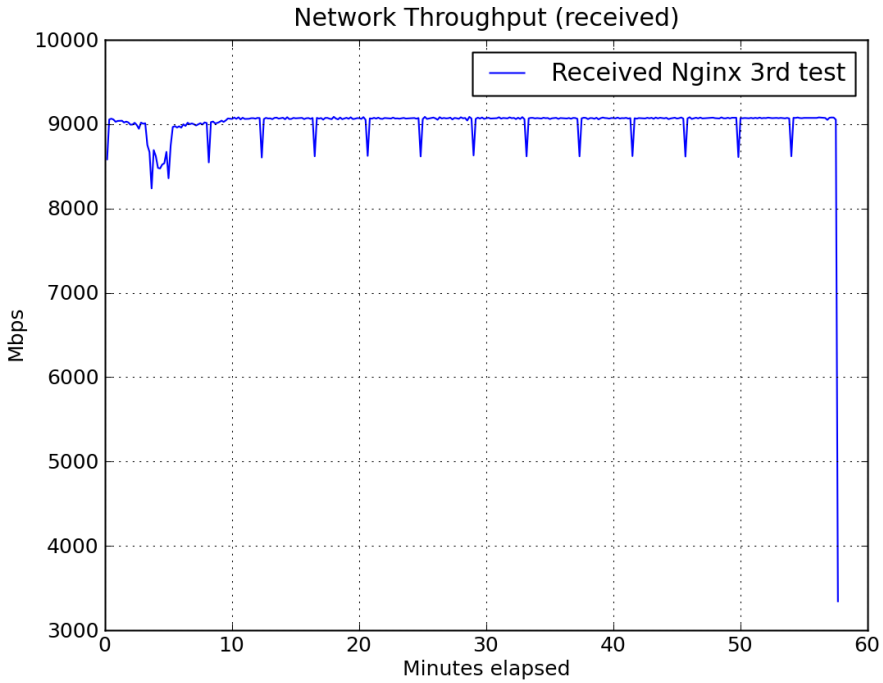

Getting into security?

Configure your own network, perhaps just a small virtualized VMware, Virtualbox, Parallels, Xen, GNS3, ...

Use Kali/BackTrack, watch youtube videos

Quote from Jurassic Park `http://www.youtube.com/watch?v=dFUlAQZB9Ng`

# More application testing




Tsung can be used to stress HTTP, WebDAV, SOAP, PostgreSQL, MySQL, LDAP and Jabber/XMPP servers `http://tsung.erlang-projects.org/`

Apache benchmark included with Apache

# Security Onion


securityonion.blogspot.dk

SOLIDO NETWORKS

**The Bro Network Security Monitor**

Bro is a powerful network analysis framework that is much different from the typical IDS you may know.

While focusing on network security monitoring, Bro provides a comprehensive platform for more general network traffic analysis as well. Well grounded in more than 15 years of research, Bro has successfully bridged the traditional gap between academia and operations since its inception.

`http://www.bro.org/`

# BRO more than an IDS

> The key point that helped me understand was the explanation that Bro is a domain-specific language for networking applications and that Bro-IDS (http://bro-ids.org/) is an application written with Bro.

Why I think you should try Bro

`https://isc.sans.edu/diary.html?storyid=15259`

# Next steps

In our network we are always improving things:

More Suricata IDS `http://www.openinfosecfoundation.org/`

More graphs, with automatic identification of IPs under attack

More identification of short sessions without data - spoofed addresses

ARP sponge, ARP watch etc.

**Conclusion:** use anything you can! Combine tools!

See also:
`http://www.version2.dk/blog/hvad-er-ddos-distributed-denial-of-ser`

# Problems and Troubleshooting

Note: some security features does not work well when DDoS hits

For instance firewall sessions can be depleted by attack traffic

# Questions?

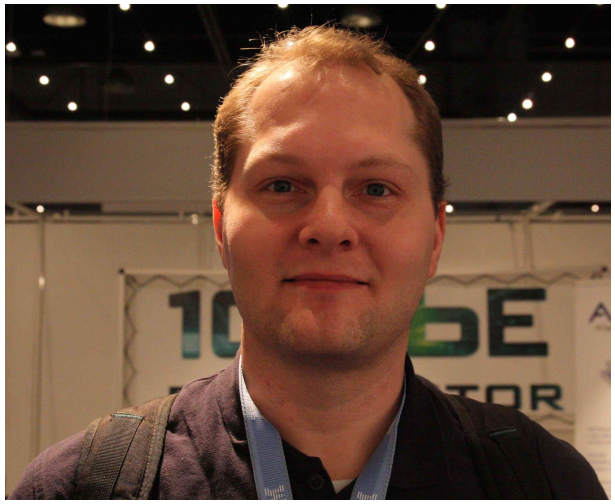
SOLIDO NETWORKS

Henrik Lund Kramshøj, internet samurai
hlk@solido.net

http://www.solidonetworks.com

You are always welcome to send me questions later via email

# Contact information



- Henrik Lund Kramshøj, IT-security and internet samurai
- Email: hlk@solido.net Mobile: +45 2026 6000
- Educated from the Computer Science Department at the University of Copenhagen, DIKU
- CISSP certified
- 2003 - 2010 Independent security consultant
- 2010 - owner and partner in Solido Networks ApS