SHADOWSERVER
Lighting the way to a more secure Internet
Internet Spelunking
IPv6 Scanning and Device Fingerprinting

Dave De Coster // Piotr Kijewski
decoster@shadowserver.org // piotr@shadowserver.org

30th June, 2022
2022 FIRST Annual Conference, Dublin
149,281,685 Reported IPs
103,553,198,124  UDP Probes
214,618,534,185  TCP SYN
351,695,957     Full Handshakes
Ground Rules

Do no harm
Never exploit
Test, test, test, 1/250th test
Test some more
First, do no harm

- Scans will not compromise, harm, or degrade system performance
  - Use the smallest and most minimal packet possible to get the results
  - Test repeatedly before a full Internet scan occurs
  - 1/250th test

- Only scan what is necessary for remediation
  - Vulnerable or misconfigured systems
  - Specific ports used by criminal infrastructures

- Scans will not break any US laws
How Did We Get Here?

No (good?) deed goes unpunished.
You can all thank Christian Rossow for publishing:

“Amplification Hell: Revisiting Network Protocols for DDoS Abuse”

The Origin

• Laid out 14 UDP protocols that could be used for a DDoS, including populations and actual amplification of each protocol
  • 11 were the most worrisome
• We focused on seven
<table>
<thead>
<tr>
<th>Protocol</th>
<th>Port</th>
</tr>
</thead>
<tbody>
<tr>
<td>SNMPv2</td>
<td>UDP/161</td>
</tr>
<tr>
<td>NTP</td>
<td>UDP/123</td>
</tr>
<tr>
<td>DNS</td>
<td>UDP/53</td>
</tr>
<tr>
<td>NetBIOS</td>
<td>UDP/137</td>
</tr>
<tr>
<td>SSDP</td>
<td>UDP/1900</td>
</tr>
<tr>
<td>CharGen</td>
<td>UDP/19</td>
</tr>
<tr>
<td>QOTD</td>
<td>UDP/17</td>
</tr>
</tbody>
</table>
The Origin

- Started with DNS
  - It was easy
  - Miscreants were already abusing it
  - There were already two open DNS scanners available for us to confirm results against
    - Other data sets were deemed too polluted to be used easily for reporting purposes
    - Cleaning other data sets was difficult and the actual methodology of scanning was flawed by both other scanning entities
    - Better to build something new to meet our more narrow scope and mission
The Origin Story

- First scan took 91 hours to complete
- 16.9 million responses (53/udp only)
- 12.25 million openly recursive
*sigh*
Fast Forward to curdate()

• The DNS scan now runs in 4 hours
  • 6 million total responses (53/udp only)
  • 1.8 million recursive resolvers
\~10.4 million IPs that are no longer abusable
After discovering that the scanning worked, we:

- Acquired more hardware
- Acquired more bandwidth
- Wrote new scanning tools
- Proceeded to implement scans on the rest of the named UDP targets

Hey, It worked!
Smooth sailing until October 2014

- POODLE (SSLv3 Downgrade)
  - Padding Oracle On Downgraded Legacy Encryption
Discovered that scanning /0 for UDP is *much* easier than TCP

- UDP is just Spray’n’Pray (with some limits)
  - Self DDoS’s can hurt if not controlled and rate limited
- TCP you have to track state and scan twice
  - And you have to talk x509!
First reported POODLE data:

- November 2014
- 15,573,251 IPs vulnerable to a downgrade attack
POODLE (SSLv3) now:

- 2,157,293
- Still a big number, but better
Expansion of the beast

We couldn’t let all the lessons we learned sit idle, so we added in a *few* more scans..
<table>
<thead>
<tr>
<th>Protocol</th>
<th>Port</th>
<th>Protocol</th>
<th>Port</th>
<th>Protocol</th>
<th>Port</th>
<th>Protocol</th>
<th>Port</th>
<th>Protocol</th>
<th>Port</th>
</tr>
</thead>
<tbody>
<tr>
<td>AMQP</td>
<td>5672</td>
<td>tcp</td>
<td>DVR</td>
<td>5001/tcp</td>
<td>tcp</td>
<td>HTTPS</td>
<td>5007/tcp</td>
<td>OMRON FINS</td>
<td>9600/udp</td>
</tr>
<tr>
<td>Android Debug Bridge</td>
<td>5555</td>
<td>tcp</td>
<td>DHCPDiscover</td>
<td>37810/udp</td>
<td>tcp</td>
<td>HTTPS</td>
<td>4433/tcp</td>
<td>OPC-UA</td>
<td>4840/udp</td>
</tr>
<tr>
<td>Apple File Protocol</td>
<td>548</td>
<td>tcp</td>
<td>EPMD</td>
<td>4936/tcp</td>
<td>tcp</td>
<td>HTTPS</td>
<td>6643/tcp</td>
<td>PCWORX</td>
<td>1962/udp</td>
</tr>
<tr>
<td>Apple Remote Management</td>
<td>3283</td>
<td>udp</td>
<td>EtherCAT</td>
<td>34980/udp</td>
<td>tcp</td>
<td>HTTPS</td>
<td>447/tcp</td>
<td>Microsoft Exchange</td>
<td>443/tcp</td>
</tr>
<tr>
<td>BACnet</td>
<td>47808</td>
<td>tcp</td>
<td>EtherCAT/IP</td>
<td>44818/udp</td>
<td>tcp</td>
<td>HTTPS</td>
<td>4117/tcp</td>
<td>Middleware</td>
<td>80/tcp</td>
</tr>
<tr>
<td>CharGEN</td>
<td>19</td>
<td>udp</td>
<td>FTP</td>
<td>21/tcp</td>
<td>tcp</td>
<td>HTTPS</td>
<td>8080/tcp</td>
<td>ProConOS</td>
<td>20547/tcp</td>
</tr>
<tr>
<td>cLDP</td>
<td>389</td>
<td>udp</td>
<td>GE-SRTP</td>
<td>18245/tcp</td>
<td>tcp</td>
<td>HTTPS</td>
<td>5443/tcp</td>
<td>Mitel</td>
<td>10074/udp</td>
</tr>
<tr>
<td>CoAP (v1)</td>
<td>5683</td>
<td>udp</td>
<td>Hadoop</td>
<td>50075/tcp</td>
<td>tcp</td>
<td>HTTPS</td>
<td>7443/tcp</td>
<td>MODBUS</td>
<td>502/tcp</td>
</tr>
<tr>
<td>CoAP (v2)</td>
<td>5683</td>
<td>udp</td>
<td>Hadoop</td>
<td>50070/tcp</td>
<td>tcp</td>
<td>HTTPS</td>
<td>443/tcp</td>
<td>MongoDB</td>
<td>27017/tcp</td>
</tr>
<tr>
<td>CODESYS IEC 61131-3</td>
<td>2455</td>
<td>tcp</td>
<td>HART</td>
<td>5094/tcp</td>
<td>tcp</td>
<td>HTTPS</td>
<td>443/tcp</td>
<td>MQTT</td>
<td>1883/tcp</td>
</tr>
<tr>
<td>CODESYS IEC 61131-3</td>
<td>1200</td>
<td>tcp</td>
<td>HTTP</td>
<td>80/tcp</td>
<td>tcp</td>
<td>ICCP</td>
<td>102/tcp</td>
<td>MQITT SSL</td>
<td>8883/tcp</td>
</tr>
<tr>
<td>CouchDB</td>
<td>5984</td>
<td>tcp</td>
<td>HTTP</td>
<td>8080/tcp</td>
<td>tcp</td>
<td>IEC 60870-5-104</td>
<td>2404/tcp</td>
<td>MS-SQL</td>
<td>1434/udp</td>
</tr>
<tr>
<td>Crimson (Red Lion)</td>
<td>789</td>
<td>tcp</td>
<td>HTTP</td>
<td>8000/tcp</td>
<td>tcp</td>
<td>IPMI</td>
<td>623/tcp</td>
<td>MySQL</td>
<td>3306/tcp</td>
</tr>
<tr>
<td>CWMP</td>
<td>7547</td>
<td>tcp</td>
<td>HTTP</td>
<td>80/tcp</td>
<td>tcp</td>
<td>IPP</td>
<td>631/tcp</td>
<td>MySQL (IPv6)</td>
<td>3306/tcp</td>
</tr>
<tr>
<td>CWMP</td>
<td>30005</td>
<td>tcp</td>
<td>HTTPS</td>
<td>8443/tcp</td>
<td>tcp</td>
<td>ISAMKP</td>
<td>500/udp</td>
<td>NAT-PMP</td>
<td>5351/udp</td>
</tr>
<tr>
<td>DB2</td>
<td>523</td>
<td>tcp</td>
<td>HTTPS</td>
<td>9000/tcp</td>
<td>tcp</td>
<td>Kubernetes</td>
<td>6443/tcp</td>
<td>NetBIOS</td>
<td>137/udp</td>
</tr>
<tr>
<td>DNP3</td>
<td>20000</td>
<td>tcp</td>
<td>HTTPS</td>
<td>449/tcp</td>
<td>tcp</td>
<td>Kubernetes</td>
<td>443/tcp</td>
<td>Nettis</td>
<td>53413/udp</td>
</tr>
<tr>
<td>DNS</td>
<td>53</td>
<td>udp</td>
<td>HTTPS</td>
<td>10443/tcp</td>
<td>tcp</td>
<td>LDAP</td>
<td>389/tcp</td>
<td>NTP (Monitor)</td>
<td>123/udp</td>
</tr>
<tr>
<td>Docker</td>
<td>2375</td>
<td>tcp</td>
<td>HTTPS</td>
<td>8010/tcp</td>
<td>tcp</td>
<td>mDNS</td>
<td>5353/udp</td>
<td>NTP (Version)</td>
<td>123/udp</td>
</tr>
</tbody>
</table>

**Over 100 Full Scans a Day**
How and Why are the next targets chosen

- Topical – new blog comes out with a vulnerability that can be remotely tested
  - Netis, Synfulknock, ISAKMP, etc
- Looking at legacy protocols that really should not be exposed
  - Telnet, rsh, etc
- Current protocols that really should not be exposed
  - MongoDB, Kubernetes, etc
- Someone asked us to look for it
Some scans are easier than others

- **“Banner” services**
  - Things that respond to a single packet are easy
  - Telnet, TFTP, et cetera

- **Negotiated services**
  - Services where you need a HELO or client/server agreement
  - SSL, SSH

- **Multi-Step services**
  - Services that require a stepwise response to get an answer
  - IEC 60870-5-104
We have sent (with daily repeats):

- 209,724,213,326,259 UDP Probes
  - 209.7 Trillion UDP Probes
- 221,639,352,853,200 TCP SYNs
  - 221.6 Trillion TCP Syns
- 508,013,815,018 Full Protocol Connections
  - 508 Billion Connections
- 287,916,573,658 Services for remediation
  - 287.9 Billion Reported
Sorry for the noise...
The Gear
How the work gets done – Grab the hearing protection
Stack o’ Boxes in a Colo

Just a pile of leftover gear

• 37 x Cisco C220 M3’s
  • 256 GB Memory
  • 5 TB Disk (8 x 1tb RAID 6)
• 2 x 10 Gb/s lines
• 5 x /26 IP blocks (and 1 /24)
Dirtiest CIDRs on the net?

• We scan from 558 IPs:
  184.105.139.64/26
  184.105.247.192/26
  216.218.206.64/26
  74.82.47.0/26
  65.49.20.64/26
  64.62.197.0/24

• Nodes are each assigned 15 IPs
• Evenly split across 2x 10 gb lines
Scanning Methodology

- TCP and UDP scans are handled differently

  - TCP Scans are:
    - Broken into shards
      - Shard is 1/250\textsuperscript{th} of the IP space to be scanned
      - IPs in a shard are algorithmically determined by a random seed that is supplied to every shard.
    - Will use the entire cluster to scan
    - Performed using commodity software

  - UDP Scans are:
    - Monolithic
    - Run from a single node
    - Performed using custom software
UDP Scans

• Meet “railgun”
  • Designed to send a single UDP packet as randomly as possible and as fast as possible to all 3.4B IPs
  • Tuned for sending small packets
  • Will send packets using all available IPs
  • Has very few safety measures
Railgun can usually scan the internet for one service in around four hours.

- Highly dependent on the number of responding devices.
TCP Scans

- Commodity tools
  - Assignment of jobs:
    - HTCondor
  - Actual scanning:
    - Zmap performs the initial sweep
    - Zgrab (mostly) performs the connection
    - Other tools for doing custom things
TCP Scans

Each service takes between ten minutes and three hours

- Dependent on the complexity of the scan
  - Things with no crypto (Telnet) are fast
    - 8 minutes in human time
    - 3 hours and 57 minutes in machine time
  - Things with crypto (HTTPS) are much slower
    - 2 hours and 29 minutes in human time
    - 82 hours in machine time
The raw data is:

- Parsed (protocol specific)
- Sanity checked (bad data?)
- Standardized
- Shipped off to the Datacenter to get turned into reports
IPv6
You want to scan what?
Surprisingly Familiar

• Like IPv4, just a LOT more of it
• Not feasible to scan it all, so curated lists
  • IPv6 addresses sourced from SSL certificates, IPv6 Hitlist, other.
• Currently scanning 814,675,045 IPv6 addresses
IPv6 space is $3.48 \times 10^{38}$ unique addresses

Time to scan $\sim 6.33 \times 10^{32}$ seconds

Roughly $2 \times 10^{25}$ years
Blindly Scanning is Infeasible

- Use curated lists from:
  - DNS AAAA records (passive DNS)
  - IPv6 Hitlist: https://ipv6hitlist.github.io/
  - Certificate transparency streams
  - Sinkholes
  - Partners
Yet Different...

Fewer options for scanning tools

• **zmap6** from https://github.com/tumi8/zmap

• **zgrab/zgrab2** have native IPv6 support

• Other tools.. Not so much
And Slower...

IPv6 requires more gentle timings than IPv4

- IPv4: Potential packet loss at > 500,000 pps
- IPv6: Potential packet loss at > 100,000 pps
IPv6 requires more gentle timings than IPv4

- IPv4: Packet loss at > 3500 concurrent senders
- IPv6: Packet loss at > 1500 concurrent senders
And Slower...

Average number of IPs/second that can be processed

- IPv4: 243,116 IPs/second
- IPv6: 58,542 IPs/second
IPv4 and IPv6 scans don’t like running at the same time on the same interface.
IPv6 Scans

- SSL (443/tcp, 8443/tcp)
- SMTP (25/tcp)
- TELNET (23/tcp)
- SSH (22/tcp)
- HTTP (80/tcp, 8080/tcp)
- MySQL (3301/tcp)
- FTP (21/tcp)
## IPv6 Scan Stats

<table>
<thead>
<tr>
<th>Scan</th>
<th>Port</th>
<th>Responses</th>
</tr>
</thead>
<tbody>
<tr>
<td>SSL</td>
<td>443/tcp</td>
<td>8,192,360</td>
</tr>
<tr>
<td>SSL</td>
<td>8443/tcp</td>
<td>75,432</td>
</tr>
<tr>
<td>SMTP</td>
<td>25/tcp</td>
<td>407,521</td>
</tr>
<tr>
<td>Telnet</td>
<td>23/tcp</td>
<td>25,267</td>
</tr>
<tr>
<td>SSH</td>
<td>22/tcp</td>
<td>839,575</td>
</tr>
<tr>
<td>HTTP</td>
<td>80/tcp</td>
<td>109,845,303</td>
</tr>
<tr>
<td>HTTP</td>
<td>8080/tcp</td>
<td>415,989</td>
</tr>
<tr>
<td>MySQL</td>
<td>3306/tcp</td>
<td>1,424,136</td>
</tr>
<tr>
<td>FTP</td>
<td>21/tcp</td>
<td>2,622,208</td>
</tr>
</tbody>
</table>
IPv6 Scans (Observations)

SSL
- Fewer hosts with really old ciphers (SSLv3, TLSv1.0, TLSv1.1)
- 3.86% IPv4 vs 0.04% IPv6

FTP
- Far higher ratio of FTP+SSL
- 55% IPv4 vs 91% IPv6

MySQL
- Far fewer hosts with deny rules
- 42% IPv4 vs 4% IPv6
IPv6 Scans

• Always Looking for More Sources of IPv6 Targets
Device Identification
Fingerprinting all things!
Device Identification

• Take all data we collect in all our daily scans
  • match fields, banners and responses to identify device make-and-model
• Classify all IPs by:
  • device_type
  • device_vendor
  • device_model
  • device_version
  • device_sector
Device Identification

- Scan rule engine implemented
- Classifies scan data as it is submitted to the API
- Currently ~1200 scan rules implemented
- Support for detection of devices from 173 vendors
- Daily successfully classifies over 28M devices (excluding desktops/servers, web servers etc)
- More to come!
Device Identification

Scan rules

<table>
<thead>
<tr>
<th>Contact</th>
<th>Name</th>
<th>Device model</th>
<th>Device type</th>
<th>Device vendor</th>
<th>Group</th>
<th>Order</th>
<th>Test count</th>
<th>Usage</th>
<th>Enabled</th>
<th>State</th>
<th>Created</th>
<th>Actions</th>
</tr>
</thead>
<tbody>
<tr>
<td>PlotKit</td>
<td>Allegro_Software_RomPaga</td>
<td>RomPaga</td>
<td>embedded-system</td>
<td>Allegro Software</td>
<td>Allegro Software</td>
<td>100</td>
<td></td>
<td></td>
<td>✔</td>
<td>✔</td>
<td>2021-11-14</td>
<td>View</td>
</tr>
<tr>
<td>PlotKit</td>
<td>Allegro_Software_RomPaga</td>
<td>RomPaga</td>
<td>embedded-system</td>
<td>Allegro Software</td>
<td>Allegro Software</td>
<td>200</td>
<td></td>
<td></td>
<td>✔</td>
<td>✔</td>
<td>2021-11-14</td>
<td>View</td>
</tr>
<tr>
<td>PlotKit</td>
<td>Realtron_Embedded_Syst</td>
<td>Realtron</td>
<td>embedded-system</td>
<td>Realtron</td>
<td>Realtron</td>
<td>100</td>
<td></td>
<td></td>
<td>✔</td>
<td>✔</td>
<td>2022-04-24</td>
<td>View</td>
</tr>
<tr>
<td>PlotKit</td>
<td>ASUS_httpd_server_http</td>
<td>router</td>
<td>ASUS</td>
<td>ASUS</td>
<td>ASUS</td>
<td>90</td>
<td></td>
<td></td>
<td>✔</td>
<td>✔</td>
<td>2021-01-29</td>
<td>View</td>
</tr>
<tr>
<td>PlotKit</td>
<td>ASUS_by_ACloud_html_title</td>
<td>router</td>
<td>ASUS</td>
<td>ASUS</td>
<td>ASUS</td>
<td>90</td>
<td></td>
<td></td>
<td>✔</td>
<td>✔</td>
<td>2022-04-13</td>
<td>View</td>
</tr>
<tr>
<td>PlotKit</td>
<td>ASUS_capture_FTP_Banner</td>
<td>router</td>
<td>ASUS</td>
<td>ASUS</td>
<td>ASUS</td>
<td>95</td>
<td></td>
<td></td>
<td>✔</td>
<td>✔</td>
<td>2021-02-05</td>
<td>View</td>
</tr>
<tr>
<td>PlotKit</td>
<td>ASUS_router.asus.com</td>
<td>router</td>
<td>ASUS</td>
<td>ASUS</td>
<td>ASUS</td>
<td>100</td>
<td></td>
<td></td>
<td>✔</td>
<td>✔</td>
<td>2020-11-13</td>
<td>View</td>
</tr>
<tr>
<td>PlotKit</td>
<td>ASUS_by_AUSSTek_cert</td>
<td>router</td>
<td>ASUS</td>
<td>ASUS</td>
<td>ASUS</td>
<td>100</td>
<td></td>
<td></td>
<td>✔</td>
<td>✔</td>
<td>2022-04-14</td>
<td>View</td>
</tr>
<tr>
<td>PlotKit</td>
<td>ASUS_aususcomm issuer</td>
<td>router</td>
<td>ASUS</td>
<td>ASUS</td>
<td>ASUS</td>
<td>101</td>
<td></td>
<td></td>
<td>✔</td>
<td>✔</td>
<td>2020-11-23</td>
<td>View</td>
</tr>
<tr>
<td>PlotKit</td>
<td>ASUS_aususcomm_lets_en</td>
<td>router</td>
<td>ASUS</td>
<td>ASUS</td>
<td>ASUS</td>
<td>102</td>
<td></td>
<td></td>
<td>✔</td>
<td>✔</td>
<td>2020-11-23</td>
<td>View</td>
</tr>
<tr>
<td>PlotKit</td>
<td>ASUS_aususcomm_lets_en</td>
<td>router</td>
<td>ASUS</td>
<td>ASUS</td>
<td>ASUS</td>
<td>105</td>
<td></td>
<td></td>
<td>✔</td>
<td>✔</td>
<td>2021-02-01</td>
<td>View</td>
</tr>
<tr>
<td>PlotKit</td>
<td>ASUS_AUSUSWRTחשוב</td>
<td>router</td>
<td>ASUS</td>
<td>ASUS</td>
<td>ASUS</td>
<td>120</td>
<td></td>
<td></td>
<td>✔</td>
<td>✔</td>
<td>2020-11-23</td>
<td>View</td>
</tr>
<tr>
<td>PlotKit</td>
<td>ASUS_AUSUSWRT_HGG_is</td>
<td>router</td>
<td>ASUS</td>
<td>ASUS</td>
<td>ASUS</td>
<td>200</td>
<td></td>
<td></td>
<td>✔</td>
<td>✔</td>
<td>2020-11-23</td>
<td>View</td>
</tr>
<tr>
<td>PlotKit</td>
<td>ASUS_Merlin_Koolshare_i</td>
<td>router</td>
<td>ASUS</td>
<td>ASUS</td>
<td>ASUS</td>
<td>202</td>
<td></td>
<td></td>
<td>✔</td>
<td>✔</td>
<td>2020-11-23</td>
<td>View</td>
</tr>
<tr>
<td>PlotKit</td>
<td>ASUS_Merlin_Koolshare_i</td>
<td>router</td>
<td>ASUS</td>
<td>ASUS</td>
<td>ASUS</td>
<td>203</td>
<td></td>
<td></td>
<td>✔</td>
<td>✔</td>
<td>2020-11-23</td>
<td>View</td>
</tr>
<tr>
<td>PlotKit</td>
<td>ASUS_Merlin_Koolshare_i</td>
<td>router</td>
<td>ASUS</td>
<td>ASUS</td>
<td>ASUS</td>
<td>204</td>
<td></td>
<td></td>
<td>✔</td>
<td>✔</td>
<td>2020-11-23</td>
<td>View</td>
</tr>
<tr>
<td>PlotKit</td>
<td>ASUS_Merlin_Koolshare_i</td>
<td>router</td>
<td>ASUS</td>
<td>ASUS</td>
<td>ASUS</td>
<td>205</td>
<td></td>
<td></td>
<td>✔</td>
<td>✔</td>
<td>2020-11-23</td>
<td>View</td>
</tr>
<tr>
<td>PlotKit</td>
<td>ASUS_Merlin_Koolshare_i</td>
<td>router</td>
<td>ASUS</td>
<td>ASUS</td>
<td>ASUS</td>
<td>206</td>
<td></td>
<td></td>
<td>✔</td>
<td>✔</td>
<td>2020-11-23</td>
<td>View</td>
</tr>
</tbody>
</table>
Device Identification - Scan rules

- **Rule syntax**

  \[( \text{boolean expression} ) \rightarrow \text{statement(s)} \]

- **Rule operators**

<table>
<thead>
<tr>
<th>Name</th>
<th>Operation</th>
</tr>
</thead>
<tbody>
<tr>
<td>and</td>
<td>boolean and</td>
</tr>
<tr>
<td>or</td>
<td>boolean or</td>
</tr>
<tr>
<td>=</td>
<td>case sensitive string equality</td>
</tr>
<tr>
<td>!=</td>
<td>case sensitive string inequality</td>
</tr>
<tr>
<td>=~</td>
<td>regex match</td>
</tr>
<tr>
<td>!~</td>
<td>regex difference</td>
</tr>
<tr>
<td>:=</td>
<td>assignment</td>
</tr>
</tbody>
</table>
SSL Common Names & Organization Names
HTML body content
HTTP server names
HTTP cookies
SNMP sysdesc, sysname
FTP, TELNET, SSH banners
... many more!
Example fingerprinting rule - iRobot Roomba

(issuer_common_name =~ /^Roomba/ and issuer_organization_name = "iRobot")
Device Identification - Philips HUE (2022-06-21)

~ 5300 devices
Device Identification - Siemens SIMATIC S7-300

~ 500 devices (based only on non-native ICS scans)
~ 500 devices (based only on non-native ICS scans)
Device Identification - Mikrotik (2022-06-21)

~ 3 200 000 devices
Device Identification - Fortinet (2022-06-21)

~ 1 400 000 devices
Devices identified by country (2022-06-21)

(Excluding desktop/servers & web servers)
Device Identification - Vendors (2022-06-21)

Cisco
4.6M

MikroTik
3.2M

Unknown
450.4K

ASUS
939.9K

ZTE
674.2K

Hikvision
571.2K

SonicWall
559.6K

DrayTek
515.7K

Technicolor
348.4K

WatchGuard
288K

Allegro
280.2K

QLC
244.2K

TP-Link
231.1K

F5
411.6K

Cloud Native
379.2K

NGINX
292.2K

Zyxel
155.8K

D-Link
154.3K

Vivint
150.8K

Sophos
144.8K

Transcend
138.7K

Tiggin
129.6K

Sagemcom
1.9M

Ubiquiti
369.3K

AVM
180.9K

CIG
172.1K

LANCOM
160.7K

Synology
364.1K

Cisco
4.6M

MikroTik
3.2M

Unknown
450.4K

ASUS
939.9K

ZTE
674.2K

Hikvision
571.2K

SonicWall
559.6K

DrayTek
515.7K

Technicolor
348.4K

WatchGuard
288K

Allegro
280.2K

QLC
244.2K

TP-Link
231.1K

F5
411.6K

Cloud Native
379.2K

NGINX
292.2K

Zyxel
155.8K

D-Link
154.3K

Vivint
150.8K

Sophos
144.8K

Transcend
138.7K

Tiggin
129.6K

Sagemcom
1.9M

Ubiquiti
369.3K

AVM
180.9K

CIG
172.1K

LANCOM
160.7K

Synology
364.1K

(Excluding desktop/servers & web servers)
HaDEA CEF - VARIoT Project

- July 2019 - Oct 2022
- Shadowserver role: focused on improving
  - scanning of IoT devices
  - observations of IoT attacks
  - collection & analysis of IoT malware
  - sharing of statistics as open data
- [https://variot.eu](https://variot.eu)
Subscribe to free daily threat feeds!

https://www.shadowserver.org/what-we-do/network-reporting/get-reports/
Questions?