Bug-Bounty-Boosters –
Scanning for exploitable vulnerabilities at scale

Karsten Nohl <nohl@srlabs.de>
Our regular vulnerability scans find many easy-to-exploit security issues on the Internet.

IP addresses of companies running bug bounties have fewer issues on average, but even these companies often have issues on assets that are hard to discover.

We want to make you more successful in finding some of these issues, in two steps:

1. We show you how to **find hard-to-discover assets**
2. We then go into **automated vulnerability scanning** that efficiently discover issues

This talk shows 10 easy tricks to discover assets and automatically scan for a large range of vulnerabilities.

By using these tricks, we discovered thousands of critical issues on corporate IP addresses globally.
10 easy tricks can multiply your vulnerability search results

<table>
<thead>
<tr>
<th>Company name</th>
<th>Domains</th>
<th>IPs</th>
<th>Services</th>
<th>Vulnerabilities</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 Subsidiary search</td>
<td>2 Certificate hunting</td>
<td>5 BGP search</td>
<td>6 masscan + nmap scan split</td>
<td>9 subjack</td>
</tr>
<tr>
<td>3 Subscribe to cert updates</td>
<td>4 Mine subject alt names</td>
<td>7 nmap resolve-all</td>
<td></td>
<td>10 gobuster + SRLabs_words</td>
</tr>
<tr>
<td>10 easy tricks can multiply your vulnerability search results</td>
<td></td>
<td>8 Deep dive: Telco protocol scan</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Subsidiary search: Scan whole families of companies

The Coca-Cola Company / Subsidiaries

Minute Maid  
Costa Coffee  
Energy Brands  
Honest Tea  
Coca-Cola (Japan) Company, Ltd.  
Barq's  
Coca-Cola Enterprises  
Odwalla  
Jugos del Valle  
Hindustan Coca-Cola Beverages P...  
Simply Orange Juice Company  
Coca-Cola India  
SA Coca-Cola Services NV  
European Refreshments
TLS certificates contain useful information

Motivation to find certificates

- Finding interesting assets of a bug bounty company involves finding the company’s domains
- TLS certificates help find domains and verify what company they belong to
- Let’s look at a few techniques to mine certificates
Collection of certificates are available to download for free

Certificate repository published by Rapid7

- Includes the Organization, a free-text field that often includes the company name
- Generated by scanning all IP addresses on ports known to provide TLS certificates
- Can also be used for finding potentially unmaintained 3rd party boxes in IP ranges that are known to belong to the target

https://opendata.rapid7.com/sonar.ssl/

Download Now (No Account Required)

See below for datasets updated on the first of every month—no sign up required.

<table>
<thead>
<tr>
<th>File Name</th>
<th>SHA1-Fingerprint</th>
<th>Size</th>
<th>Updated At</th>
</tr>
</thead>
<tbody>
<tr>
<td>20190925/2019-09-25-1569439494-https_get_8443_names.gz</td>
<td>7645879b6959310e3f5226dedc2ea86adf127bda</td>
<td>24.6 MB</td>
<td>Sept. 25, 2019</td>
</tr>
<tr>
<td>20190925/2019-09-25-1569439494-https_get_8443_hosts.gz</td>
<td>8e5ab07fad2b3fa747bdd3bd97a9114e87bae9f6</td>
<td>99.7 MB</td>
<td>Sept. 25, 2019</td>
</tr>
</tbody>
</table>
CAs publish all issued certificates into a public log

Certificate transparency

- Started in 2013
- If no wildcard certificate is used, this leaks all subdomains that a TLS cert was issued for
- They are hosted on a searchable website, https://crt.sh
- The process of extracting subdomains can be automated

```bash
$ domain=srlabs.de; curl https://crt.sh/?q=%25.$domain 2>/dev/null |
grep -oE "<TD>(.*\.$domain)</TD>" | sed -e 's/<[^>]*>//g' | sort | uniq

autobahn.srlabs.de
brest.srlabs.de
datatop.srlabs.de
etherpad.srlabs.de
exchange.srlabs.de
gitlab.srlabs.de
heating.srlabs.de
```
It is possible to directly request live cert updates using CertStream

Real-time certificate transparency

- A real-time certificate transparency log is being streamed at https://certstream.calidog.io/ by Cali Dog
- This allows being immediately notified, e.g. a new subdomain of a company with a bug bounty (with a wildcard domain scope) was created
- Subscribe to the stream to be the first to look for bounty bugs
Certs often include related domains as *alt names*.

- It is possible to discover domains and subdomains belonging to a company by mining alt names.
- Recursive mining can be done on each alt name to find more assets.
- Filtering needed, e.g. Cloudflare used to put many unrelated companies into the same TLS cert.
Large companies usually have multiple Autonomous Systems
Use BGP search to also find subnets that are not part of these AS, but hosted by third-parties
Combine masscan and nmap to have the best of both worlds

### Fast scanning:
- Asynchronous SYN scanner
- Only basic functionality above that, hard to add anything new
- Can scan one port on the Internet in 5 minutes with a fast connection (Still 224 days for 65k TCP ports!)

### Deep scanning and vulnerability tests:
- Fully-fledged port scanner
- Extensive service detection library, easy to extend through lua – search on Github for latest scripts
- Relatively slow and resource-hungry, especially some advanced script tests

1. Scan top ports on all potential IPs
2. Scan all ports on alive hosts
3. Run service detection on discovered ports
4. Run scripts fitting the detected service
Spot configuration mismatches by scanning all IPs

- A domain can resolve to many IPv4/IPv6 addresses
- Scan all IP addresses for open ports
- HTTP(s) services can behave differently based on which virtual host is specified
  - Target each IP directly
  - Target each IP with the virtual host set to the domain name:

```bash
$ nmap -p443 [...] --resolve-all costa.co.uk
$ nmap -p443 [...] -6 --resolve-all costa.co.uk
```
Deep dive: Telco protocol scanning

Scanning for telco protocols reveals hidden hosts which are often not detected by the common sets of nmap ports.

<table>
<thead>
<tr>
<th>Telco Protocols</th>
<th>Generation</th>
<th>4G</th>
<th>2G/3G</th>
</tr>
</thead>
<tbody>
<tr>
<td>UDP*: 2123</td>
<td>MME</td>
<td>GTP</td>
<td></td>
</tr>
<tr>
<td>2152</td>
<td>PGW</td>
<td>S1AP</td>
<td></td>
</tr>
<tr>
<td>3386</td>
<td>SGW</td>
<td>X2AP</td>
<td></td>
</tr>
<tr>
<td>SCTP: 36422</td>
<td>eNodeB</td>
<td>Diameter</td>
<td></td>
</tr>
<tr>
<td>SCTP: 36412</td>
<td>SGSN</td>
<td>MGCP</td>
<td></td>
</tr>
<tr>
<td>SCTP: 3868</td>
<td>GGSN</td>
<td>H.323</td>
<td>SIGTRAN</td>
</tr>
</tbody>
</table>

$ nmap -sS -sY -sU -pT:2427,...,U:2427,...,S:36412$

<table>
<thead>
<tr>
<th>Protocol</th>
<th>Message type</th>
<th>Generation</th>
<th>4G</th>
<th>2G/3G</th>
</tr>
</thead>
<tbody>
<tr>
<td>GTPv2</td>
<td>Downlink Data Notification</td>
<td>MME</td>
<td>GTPv2</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Modify Bearer Request</td>
<td>PGW</td>
<td>X2AP</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Release Bearer Request</td>
<td>SGW</td>
<td>GTPv1</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Handover Request</td>
<td>eNodeB</td>
<td></td>
<td>SGSN Context Request</td>
</tr>
<tr>
<td></td>
<td>SGSN Context Request</td>
<td>SGSN</td>
<td></td>
<td>CreatePDP</td>
</tr>
</tbody>
</table>

* Scanning UDP ports often requires sending a valid packet for the targeted protocol as the other side might otherwise not respond.
Unmaintained subdomains allow hijacking

1. Get address of sub.domain.com
2. Get address of sub.domain.com
3. Found CNAME olddomain.com
4. Get address of olddomain.com
5. Not found
6. Not found

Through domain registration

Through registration at third-party

1. Get content for company.3rdpartyservice.com
2. Content

Web server on 1.2.3.4 (3rd party)

DNS server for domain.com
DNS server for olddomain.com
<table>
<thead>
<tr>
<th>Engine</th>
<th>Fingerprint</th>
</tr>
</thead>
<tbody>
<tr>
<td>AWS/S3</td>
<td>“The specified bucket does not exist”</td>
</tr>
<tr>
<td>Bitbucket</td>
<td>“Repository not found”</td>
</tr>
<tr>
<td>Ghost</td>
<td>“The thing you were looking for is no longer here”</td>
</tr>
<tr>
<td>Github</td>
<td>“There isn’t a Github Pages site here.”</td>
</tr>
<tr>
<td>Microsoft Azure</td>
<td>Responds with NXDOMAIN</td>
</tr>
<tr>
<td>Pantheon</td>
<td>“404 error unknown site!”</td>
</tr>
<tr>
<td>Readme.io</td>
<td>“Project doesn’t exist... yet!”</td>
</tr>
<tr>
<td>Statuspage</td>
<td>Redirection to <a href="https://www.statuspage.io">https://www.statuspage.io</a></td>
</tr>
<tr>
<td>Wordpress</td>
<td>“Do you want to register *.wordpress.com?”</td>
</tr>
</tbody>
</table>
Tool release: Enhanced gobuster with SRLabs_wordlist

- **Gobuster** is a popular **directory** and **files brute-force** tool
- SRLabs uses a custom version of Gobuster to:
  a. Reduce **false positives**
  b. Increase the **expressiveness** of the tests by leveraging the Nikto database

### Web-scanners
- **Common Problems**
  - Identify custom 404 responses
  - 200 "Not found" responses
  - Identify **static pages** responses
  - "Want to buy this domain?" responses
  - Identify **custom** and **WAF error** pages
  - All requests to /admin* blocked

### Our solution
- Use **404 wildcards** based on the **fuzzy hash** of the response bodies
- Filter out long lists of **similar responses** from the results
- **Fuzz the URI** of the positive results and identify similar pages
- Use the expressiveness and existence of **nikto tests** with **gobuster** performance

### Test expressiveness vs. tool performance
- Gobuster "dumb" bruteforcing

---

**Exclusive AVTokyo 2019 release:** https://opensource.srlabs.de/projects/srlabs-gobuster
The presented scanning approach discovers a wide range of security issues

<table>
<thead>
<tr>
<th>Security issue example</th>
<th>Tool</th>
<th>Frequency (per 1mio IPs)</th>
<th>Chance to exploit</th>
</tr>
</thead>
<tbody>
<tr>
<td>Exposed git repository</td>
<td></td>
<td>1.950</td>
<td>Medium – In many cases, the exposed data is not sensitive</td>
</tr>
<tr>
<td>Exposed svn repository</td>
<td>gobuster + SRLabs_wordlist</td>
<td>1.050</td>
<td>High</td>
</tr>
<tr>
<td>Database file leak</td>
<td></td>
<td>830</td>
<td></td>
</tr>
<tr>
<td>Pulse VPN</td>
<td></td>
<td>210</td>
<td></td>
</tr>
<tr>
<td>Heartbleed</td>
<td></td>
<td>630</td>
<td></td>
</tr>
<tr>
<td>Cisco Smart Install</td>
<td>nmap</td>
<td>450</td>
<td></td>
</tr>
<tr>
<td>Bluekeep</td>
<td></td>
<td>380</td>
<td>High</td>
</tr>
<tr>
<td>Unauthenticated MQTT</td>
<td></td>
<td>90</td>
<td></td>
</tr>
<tr>
<td>Hijackable domain</td>
<td>subjack</td>
<td>1.700 (per 1mio domains)</td>
<td></td>
</tr>
</tbody>
</table>

gobuster and nmap find hundreds of other types of security issues
We scanned thousands of large companies globally

Start with 40,000 companies
We selected a representative sample:
- Diverse in industry and location
- Large enough to have their own technology assets
- Reach an Internet exposure threshold

Use global databases
- IP WHOIS
- Domain WHOIS
- TLS certificates
- Some AI to extrapolate
- Open datasets
- Google search
- Manual search

Aggregate information by company
- 270 million IP addresses
- 1.3 million base domains
- Industry
- Financial data
- Year of founding
- Headquarter location
- Bug bounties

These preparation steps provide context for each IP address and domain in our scan
Our vulnerability scanner, Autobahn, computes a Hackability Score from a large range of issues.

### Example issues

<table>
<thead>
<tr>
<th>Authentication and credential issues</th>
<th>Unnecessary exposure</th>
<th>Hardening gaps</th>
<th>Missing patches</th>
</tr>
</thead>
<tbody>
<tr>
<td>Weak password</td>
<td>Exposed VMWare ESXi</td>
<td>Accessible .git</td>
<td>Heartbleed</td>
</tr>
<tr>
<td>HTTP default credentials</td>
<td>Exposed Cisco SmartInstall</td>
<td>Accessible Linux home folder</td>
<td>RDP vulnerability</td>
</tr>
<tr>
<td>Unauthenticated Redis</td>
<td>Exposed HP Remote Console</td>
<td>Writable anon FTP</td>
<td>Vulnerable Coldfusion</td>
</tr>
<tr>
<td>Unauthenticated MQTT</td>
<td>Exposed Lantronix config</td>
<td>HTTP path traversal</td>
<td>Vulnerable Struts2</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Issues per million active IPs</th>
<th>297</th>
<th>2.154</th>
<th>3.369</th>
<th>1.080</th>
</tr>
</thead>
<tbody>
<tr>
<td>Research scope: 827k active IPs – of 270 million IPs belonging to companies that we scanned</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

- Researchers focus on novel bug classes, while most issues found on the Internet are well-known issues.
- The vast majority of Internet-exposed security issues would be addressed by basic security practices: Change default passwords, use a firewall well, harden your servers, and patch them regularly.
- Most of the surveyed companies conduct regular vulnerability scans, but cannot act on the data.
- They need more actionable scan results, which we deliver with Autobahn.
For easy comparison of security between different companies, we summarize issues from four best practice areas in a Hackability Score.

<table>
<thead>
<tr>
<th>Issue examples</th>
<th>Hackability score</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Severity 4</strong></td>
<td>3 x 8</td>
</tr>
<tr>
<td>- Exploit</td>
<td></td>
</tr>
<tr>
<td>Tomcat with default or weak credentials</td>
<td>x 8</td>
</tr>
<tr>
<td>NFS share mountable</td>
<td></td>
</tr>
<tr>
<td>Cisco Smart Install exposed</td>
<td></td>
</tr>
<tr>
<td>Java Debug Wire protocol exposed</td>
<td></td>
</tr>
<tr>
<td>CMS backup files can be downloaded</td>
<td></td>
</tr>
<tr>
<td>Directory traversal</td>
<td></td>
</tr>
<tr>
<td>Apache Struts vulnerability</td>
<td></td>
</tr>
<tr>
<td>HP iLO 4 vulnerability</td>
<td></td>
</tr>
</tbody>
</table>

| **Severity 3**  | 2 x 4              |
| - Exploit fragment       |                  |
| Printer with default credentials |           |
| Weak SNMP pass w/ write access |           |
| Java RMI exposed |           |
| Industrial control system protocol exposed |           |
| .git accessible |           |
| Home directory exposed in web root |           |
| Oracle TNS poison attack |           |
| Cisco iOS older than 3 years |           |
| .git accessible |           |
| Home directory exposed in web root |           |
| Oracle TNS poison attack |           |
| Cisco iOS older than 3 years |           |

| **Severity 2**  | 1 x 1              |
| - Best practice deviation       |                  |
| Known leaked TLS private key used |           |
| Weak SNMP pass w/ read access |           |
| Database exposed |           |
| Server management interface exposed |           |
| Open SMTP relay |           |
| DNS server allows zone transfers |           |
| EOL IIS |           |
| EOL OpenSSH |           |

**Definition:** The Hackability Score is the sum over Internet-exposed issues, multiplied by their severity class.

- If one issue type is present multiple times, each additional occurrence is weighted less to account for the diminishing return to the hacker.

More background on how we compute and use Hackability:
[1] https://srlabs.de/bites/hackability-score/
Using Autobahn, we found Hackability to vary widely across industries

**Defense view**
Which industries can I learn from?

**Offense view**
Which industries are the easiest targets?

1. Retail 8
2. Insurance 10
3. Banking 10
4. Pharma 10
5. Real Estate 11
6. Media 12
7. Software 13
8. Hardware 13
9. Tech services 19

We scanned the Internet and computed the Hackability for thousands of companies. Download your free report of open security issues at: [https://autobahn.security](https://autobahn.security)
Banks’ hackability mostly arises from missing patches, and is worst in Europe.

**Offense view**
If your goal is to hack a bank, you would look for missing patches on unnecessarily exposed hosts, starting in Europe.

**Defense view**
If you want to secure a bank in Europe, you should focus on patching, and then learn on authentication and hardening from your peers in other regions.
How hackable is my region or industry?

Find all the statistics discussed in this talk and a lot more at srlabs.de

Play with the data at https://srlabs.de/hackability/
Many factors indicate the average hackability of a company

More hackable

- Europe
- Software, Technology Services
- High
- Pre-Internet (before 1990)
- Bad
- No bug bounty

Less hackable

- East Asia
- Banking, Retail
- Low
- After 1990
- Good
- Bug bounty
How hackable is my company?

Download details of all discovered issues

Get your company’s report at https://autobahn.security

<table>
<thead>
<tr>
<th>Scans</th>
<th>Revision</th>
<th>IPs</th>
<th>Major</th>
<th>Noteworthy</th>
<th>Minor</th>
<th>Actions</th>
</tr>
</thead>
<tbody>
<tr>
<td>AWS infrastructure</td>
<td>1</td>
<td></td>
<td>37 mins to go</td>
<td>72% complete</td>
<td></td>
<td>Cancel Scan</td>
</tr>
<tr>
<td>All company assets</td>
<td>4</td>
<td>3781</td>
<td>7</td>
<td>114</td>
<td>710</td>
<td>Download XLSX</td>
</tr>
<tr>
<td></td>
<td>3</td>
<td>3870</td>
<td>15</td>
<td>160</td>
<td>950</td>
<td>Download XLSX</td>
</tr>
</tbody>
</table>

Example scan
Revision 1

**Vulnerability Results**

- **Systems By Severity Level**
  - IPS: 208
  - Domains: 423

**Underlying Root Causes**

- **Total Hackability Score**: 61
- **Security Architecture**: Is our main weakness
  - AWS: 4
  - AWS Full: 1
- **Security Maintenance**: Is worse than average
  - IP: 4
  - Domain: 23

Next Actions

**Short-term / Technical**

1. Investigate and close the major issues: `unauth_mqtt` and `aws_full_control_bucket` affecting 2 hosts.
2. Patch and harden exposed software, starting with: `git_directory`, `writable_snmp`, `rti`
3. Block internal services at border firewall, starting with: 14 minor issues

**Long-term / Strategic**

1. Rescan regularly to measure
Take aways

- High-risk issues are commonplace on the Internet, but often reside on hard-to-discover assets
- Using available databases, such assets can be found; and using standard vulnerability scanners, many assets can be scanned automatically
- We find thousands of critical issues per million active IP addresses, using Autobahn our own vulnerability scanner

Questions?

Karsten Nohl <nohl@srlabs.de>
Old slides
10 easy tricks will multiply your vulnerability search results

1. Subsidiary search
2. Certificate hunting
3. Subscribe to cert updates
4. Mine subject alt names
5. BGP search
6. masscan+nmap scan split
7. nmap resolve-all
8. Deep dive: Telco protocol scan
9. subjack
10. gobuster + SRLabd_words
Companies often put multiple of their domains in the alt names

1. Organization name is discovered based on the given domain.
2. Discover all domains in whois data registered under that name.
3. Store discovered domains.
4. If new domains discovered: check certificate repository, and certificate transparency data to find subject names.
5. If new organization names (subject name) discovered: jump to the step #2 for each new name.

Potentially some filtering needed, e.g. Cloudflare used to put many unrelated companies into the same TLS cert