

Foot printing (Information Gathering)

Foot printing :

- Foot printing is the systematic collection of publicly available (and optionally directly queried) information about a target (domain, network, web app, person, organization) to build an asset/profile map.
- It's the first phase of any security assessment or attack lifecycle — and the best place for defenders to find and reduce exposure.

Type of foot printing :

1. Passive :
 - Collecting information about a target **without directly contacting the target's systems**.
 - You use publicly available sources (search engines, archives, public APIs, third-party indexes) so the target's infrastructure isn't queried and is less likely to detect you.
 - Build an initial picture of the target safely and legally; good for discovery and reconnaissance when you must avoid detection.
2. Active :
 - Gathering information by **directly interacting with the target's infrastructure or services**.
 - Examples: DNS queries to authoritative servers, HTTP requests, crawling, traceroute, or controlled port checks. These actions touch the target and are likely to be logged.
 - To confirm reachability, enumerate live hosts/services, discover hidden endpoints, or collect banners/version info that passive sources don't show.

Various step of information gathering :

1) WHOIS (domain registrar info)

- **Goal:** Learn what WHOIS reveals and how to interpret registrar/expiry data.
- **Setup:** Create a fake test domain in a local hosts file (e.g., lab-example.local) and also use a public test domain you own OR use example.com for demonstration (passive read-only).

```

kali@kali:~$ whois example.com
Domain Name: EXAMPLE.COM
Registry Domain ID: 2336799_DOMAIN_COM-VRSN
Registrar WHOIS Server: whois.iana.org
Registrar URL: http://res-dom.iana.org
Updated Date: 2025-08-14T07:01:39Z
Creation Date: 1995-08-14T04:00:00Z
Registry Expiry Date: 2026-08-13T04:00:00Z
Registrar: RESERVED-Internet Assigned Numbers Authority
Registrar IANA ID: 376
Registrar Abuse Contact Email:
Registrar Abuse Contact Phone:
Domain Status: clientDeleteProhibited https://icann.org/epp#clientDeleteProhibited
Domain Status: clientTransferProhibited https://icann.org/epp#clientTransferProhibited
Domain Status: clientUpdateProhibited https://icann.org/epp#clientUpdateProhibited
Name Server: A.IANA-SERVERS.NET
Name Server: B.IANA-SERVERS.NET
DNSSEC: signedDelegation
DNSSEC DS Data: 370 13 2 BE74359954660069D5C63D200C39F5603827D7DD02856F120EE9F3A86764247C
URL of the ICANN Whois Inaccuracy Complaint Form: https://www.icann.org/wicf/
>>> Last update of whois database: 2025-11-07T15:50:43Z <<<

For more information on Whois status codes, please visit https://icann.org/epp

NOTICE: The expiration date displayed in this record is the date the
registrar's sponsorship of the domain name registration in the registry is
currently set to expire. This date does not necessarily reflect the expiration
date of the domain name registrant's agreement with the sponsoring
registrar. Users may consult the sponsoring registrar's Whois database to
view the registrar's reported date of expiration for this registration.

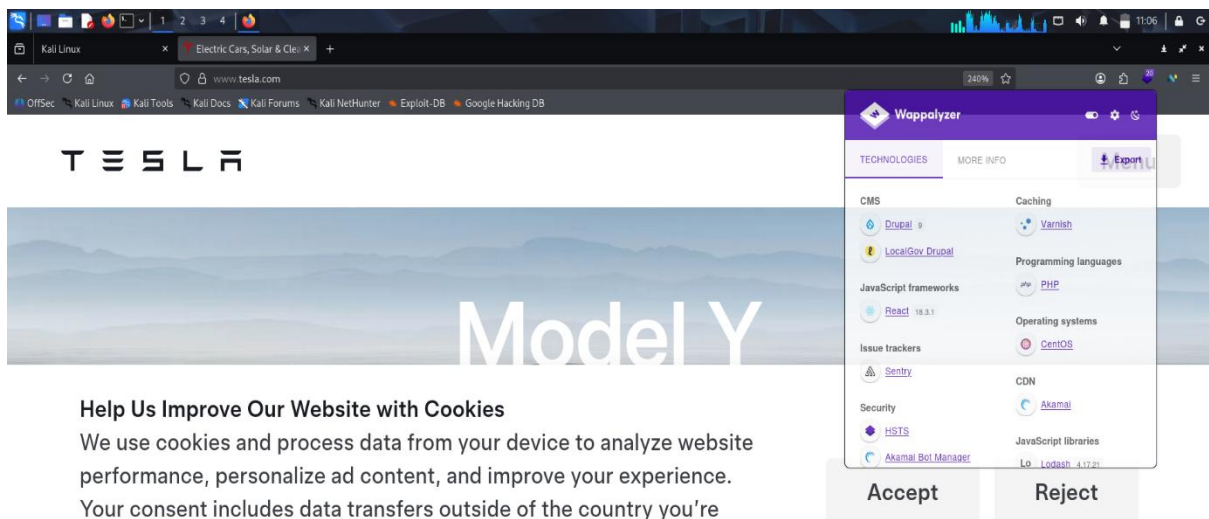
TERMS OF USE: You are not authorized to access or query our Whois
database through the use of electronic processes that are high-volume and
automated except as reasonably necessary to register domain names or
modify existing registrations; the Data in VeriSign Global Registry
Services' ("VeriSign") Whois database is provided by VeriSign for
information purposes only, and to assist persons in obtaining information

```

- **Observe:** registrar, creation/expiry, name servers, registrant contact (may be redacted).
- **Deliverable:** short note: registrar | created | expires | nameservers | privacy on/off.
- **Precautions:** WHOIS is passive; avoid querying personal/private domains without consent.

2) Wappalyzer / BuiltWith (tech fingerprinting)

- **Goal:** Identify web technologies used by a target site.
- **Setup:** Run OWASP Juice Shop on target VM and open its web UI.
- Install Wappalyzer browser extension on your Kali desktop browser OR run:



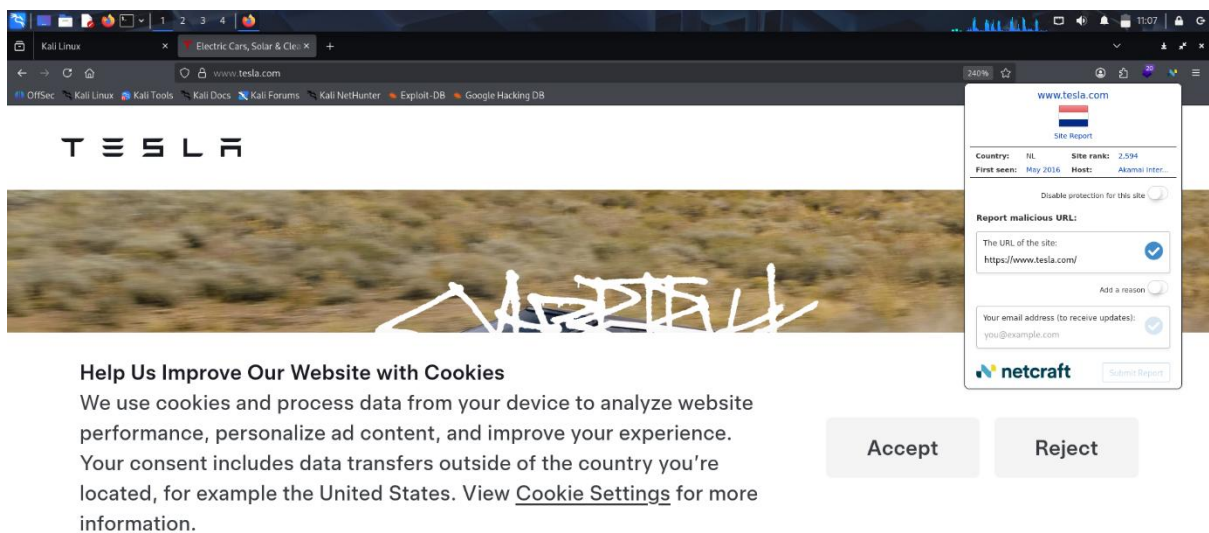
Help Us Improve Our Website with Cookies

We use cookies and process data from your device to analyze website performance, personalize ad content, and improve your experience. Your consent includes data transfers outside of the country you're located, for example the United States. View [Cookie Settings](#) for more information.

- **Observe:** frameworks, server, JS libraries, analytics, CDN.
- **Deliverable:** table: component | evidence (header, script, cookie).
- **Precautions:** Passive detection — run only against lab targets.

3) Netcraft (historical/hosting info)

- **Goal:** See hosting/provider metadata and historical hosting if available.
- **Setup:** Use a public demo site you own OR use results from Netcraft for a public benign domain (only lookup).
- **Steps:** Visit Netcraft's site and enter the domain; note historical hosting, server header info.



- **Observe:** hosting provider, server/version, historical hosting changes.
- **Deliverable:** note of hosting provider + any interesting historical change.
- **Precautions:** Passive lookup only.

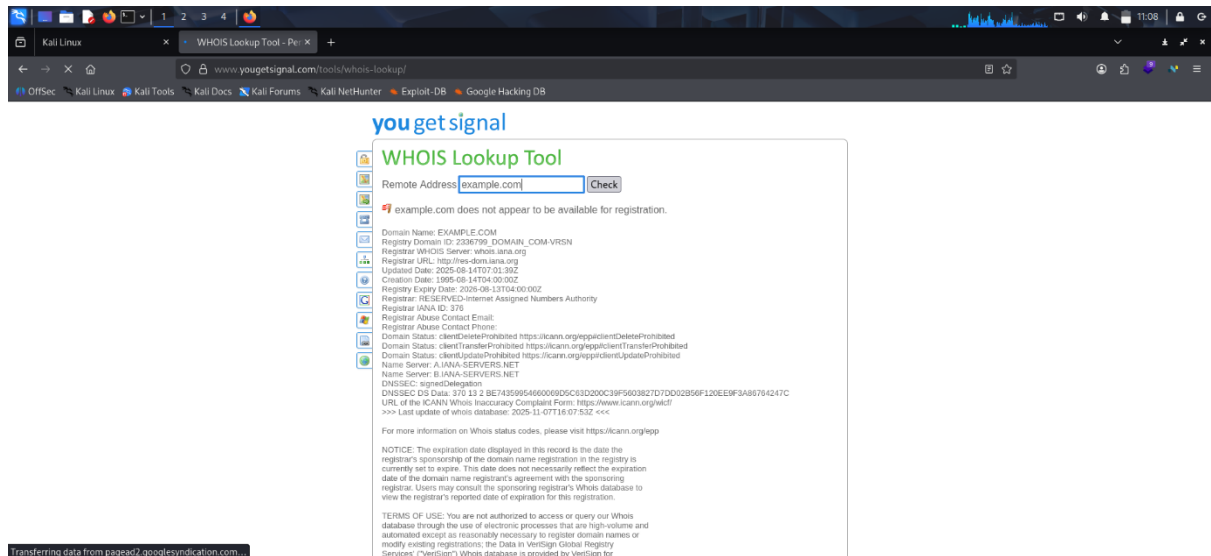
4) MX/DNS lookups (dig, nslookup, mxlookup)

- **Goal:** Extract DNS records and mail configuration (MX, TXT/SPF, NS).
- **Setup:** Use a domain you control for full visibility. For lab practice use dig against public resolver for example.com.

- **Precautions:** Passive only.

6) YouGetSignal-style reachability (simulate with nc / local checks)

- **Goal:** Understand remote port reachability; **do not** probe external hosts without permission.
- **Setup:** On target VM open a service (e.g., `python -m http.server 8000`) and from Kali test reachability.



- **Observe:** connection success/failure, open port presence.
- **Deliverable:** table: target IP | port | reachable | service.
- **Precautions:** Use local lab only. If using third-party web tools (YouGetSignal), be aware they perform remote checks.

7) GHDB (Google Dorks) & GitHub search

- **Goal:** Learn how Google dorks and GitHub searches reveal exposed files and credentials.
- **Setup:** Use Google with safe dorks on your own test repo or on public benign sites you manage.
- **Example dorks:**
 - `site:github.com "your-organization" "API_KEY"`
 - `site:example.com filetype:pdf "confidential"`
- **Observe:** exposed files, leaked config snippets, references to hosts/APIs.
- **Deliverable:** list of dorks used + results + remediation suggestions.

- **Precautions:** Do not use aggressive dorks on domains you don't control. Respect robots and privacy.

8) Shodan (internet-exposed service discovery)

- **Goal:** See what Shodan can reveal about exposed services (banners, ports).
- **Setup:** Use Shodan web UI or CLI against your lab public IP only if you expose it to internet (not recommended). Safer: use Shodan to search for your own public IPs that you control.
- **Example CLI:**
- shodan host <your-public-ip>
- **Observe:** service banners, ports, device type (if indexed).
- **Deliverable:** screenshot/export of Shodan host output + mitigation plan.
- **Precautions:** Don't index or expose lab VMs to the public network unless intentional and secure.

9) traceroute / tracert (network path mapping)

- **Goal:** Map routing path from attacker to target and identify hops.
- **Setup:** Use local lab network.

```

kali@kali:~$ traceroute google.com
traceroute to google.com (2404:6800:4009:81c::200e), 30 hops max, 80 byte packets
 1  2402:3a80:45d3:b01a:8d65:a57d:5332:4a29 (2402:3a80:45d3:b01a:8d65:a57d:5332:4a29)  3.670 ms  3.470 ms  3.391 ms
 2  * * *
 3  fd00:0:0:91::31 (fd00:0:0:91::31)  190.044 ms  189.978 ms  189.917 ms
 4  fd00:0:16:16::6a (fd00:0:16:16::6a)  189.859 ms  189.798 ms  189.739 ms
 5  fd00:0:17:17::69 (fd00:0:17:17::69)  189.731 ms  189.660 ms  189.550 ms
 6  2400:5200:1400:9f::32 (2400:5200:1400:9f::32)  189.490 ms  116.623 ms  116.518 ms
 7  2402:8100:4000::7ba (2402:8100:4000::7ba)  116.341 ms  116.188 ms  116.097 ms
 8  2001:4860:111::ea8 (2001:4860:111::ea8)  116.000 ms  115.936 ms  115.864 ms
 9  2404:6800:8118::1 (2404:6800:8118::1)  115.735 ms  2404:6800:8201::1 (2404:6800:8201::1)  115.733 ms  2404:6800:80b3::1 (2404:6800:80b3::1)  115.669 ms
10  2001:4860:0:1::5cd4 (2001:4860:0:1::5cd4)  115.600 ms  2001:4860:0:1::7ba0 (2001:4860:0:1::7ba0)  115.549 ms  2001:4860:0:1::3132 (2001:4860:0:1::3132)  60.828 ms
11  2001:4860:0:1::8766 (2001:4860:0:1::8766)  60.553 ms  2001:4860:0:1::4b57 (2001:4860:0:1::4b57)  62.061 ms  2001:4860:0:1::870c (2001:4860:0:1::870c)  61.993 ms
12  bom07s26-in-x0e.1e100.net (2404:6800:4009:81c::200e)  61.922 ms  61.858 ms  2001:4860:0:1::1baf (2001:4860:0:1::1baf)  66.501 ms

```

- **Observe:** intermediate hops, latency spikes, possible NATs/routers.
- **Deliverable:** traceroute output + notes on network topology.

- **Precautions:** traceroute is low-risk; avoid using it for external reconnaissance without permission.

10) SpiderFoot (automated OSINT aggregator)

- **Goal:** Use SpiderFoot to aggregate passive OSINT into one report.
- **Setup:** Install SpiderFoot on attacker VM and configure with no active modules enabled unless authorized. Point at example.com or your own domain.
- **Example:**
 - spiderfoot -s example.com -o spider_report.html
- **Observe:** subdomains, certs, public leaks, email addresses, third-party links.
- **Deliverable:** export HTML report + CSV of discovered assets.
- **Precautions:** Disable active modules (port scans, zone transfers) unless in-scope.

11) Dmitry & theHarvester (quick harvest tools)

- **Goal:** Compare outputs from small footprinting tools.
- **Setup:** Use lab domain.

```

(kali@kali)~$ dmitry -i nixsecura.com
Deepmagic Information Gathering Tool
"There be some deep magic going on"

HostIP:62.72.28.150
HostName:nixsecura.com

Gathered Inet-whois information for 62.72.28.150

inetnum:        62.72.28.0 - 62.72.31.255
netname:        HOSTINGER-HOSTING
country:        IN
admin-c:        HN1858-RIPE
tech-c:         HN1858-RIPE
status:         SUB-ALLOCATED PA
geofeed:        https://raw.githubusercontent.com/hostinger/geofeed/main/geofeed.csv
geoloc:         19.076090 72.877426
mnt-by:         MNT-HOSTINGER
created:        2023-07-31T08:00:43Z
last-modified:  2023-07-31T08:00:43Z
source:         RIPE

person:         Hostinger NOC
address:         Hostinger International Ltd.
address:         61 Lordou Vryonas
address:         Lumiel Building, 4th floor
address:         6023
address:         Larnaca
address:         CYPRUS
phone:          +37064503378
nic-hdl:        HN1858-RIPE
mnt-by:         HN19812-MNT
created:        2013-12-02T20:17:12Z
last-modified:  2024-07-09T12:29:29Z

```

- **Observe:** email lists, subdomains, netcraft/whois results aggregated.
- **Deliverable:** combined list of emails/subdomains found and confidence level.
- **Precautions:** theHarvester uses public sources; be mindful of API limits and terms.

```

kali@kali: ~
Session Actions Edit View Help
status: SUB-ALLOCATED PA
geofeed: https://raw.githubusercontent.com/hostinger/geofeed/main/geofeed.csv
geoloc: 19.076090 72.877426
mnt-by: MNT-HOSTINGER
created: 2023-07-31T08:00:43Z
last-modified: 2023-07-31T08:00:43Z
source: RIPE

person:
  Hostinger NOC
  address: Hostinger International Ltd.
  address: 61 Lordou Vryonou
  address: Lumiel Building, 4th floor
  address: 6023
  address: Larnaca
  address: CYPRUS
  phone: +37064503378
  nic-hdl: HN1858-RIPE
  mnt-by: HN19812-MNT
  created: 2013-12-02T20:17:12Z
  last-modified: 2024-07-09T12:29:29Z
  source: RIPE # Filtered

% Information related to '62.72.28.0/22AS47583'
route: 62.72.28.0/22
origin: AS47583
mnt-by: MNT-HOSTINGER
created: 2013-12-02T20:17:12Z
last-modified: 2023-07-31T07:59:45Z
source: RIPE
descr: HOSTINGER IN

% This query was served by the RIPE Database Query Service version 1.119 (DEXTER)

All scans completed, exiting

```

- **Goal:** Correlate all findings into a single asset inventory and prioritize.
- **Steps:** Merge outputs (WHOIS, DNS, subdomains, certs, services) into a spreadsheet with columns: asset | type | evidence | source | confidence | remediation.
- **Deliverable:** final spreadsheet + short slide with top 5 risky findings and mitigation.

12) Whatweb

- **Fast tech fingerprinting:** finds server types, CMS, frameworks, common plugins and sometimes version hints.
- **Prioritization:** helps you decide which follow-up checks to run (e.g., look for WordPress plugins, Node endpoints, or outdated server versions).
- **Defensive check:** defenders run it against their own sites to see what version/metadata they leak and what to harden.
- **Mostly active (low-noise):** WhatWeb makes HTTP requests to the target and inspects responses.
- It's less noisy than a port scan but still *touches* the target, so it's active and will be logged by the target's web servers.
- Use it only on sites you own or have explicit permission to test.

```

kali@kali: ~
Session Actions Edit View Help

kali@kali:~$ whatweb tesla.com
https://tesla.com [403 Forbidden] Akamai-Global-Host, Country[EUROPEAN UNION][EU], HTTPServer[AkamaiGHost], IP[2.18.53.207], Title[Access Denied], UncommonHeaders[x-reference-error,x-ak-cache,permissions-policy]
https://tesla.com [403 Forbidden] Akamai-Global-Host, Country[EUROPEAN UNION][EU], HTTPServer[AkamaiGHost], IP[2.18.53.207], Strict-Transport-Security[max-age=15768000], Title[Access Denied], UncommonHeaders[x-reference-error,x-ak-cache,permissions-policy]

```

13) wafw00f

- wafw00f is a small Python tool that detects and fingerprints Web Application Firewalls (WAFs) protecting a website.

- It starts with a normal HTTP request and, if needed, sends additional crafted requests to match signatures in its database.
- Use it to learn whether a site is behind a WAF and (often) which vendor/product is present.
- **Role in footprinting:** it's an *active* web-fingerprinting step used after passive OSINT to confirm whether a target is protected by a WAF and to inform further (authorized) testing.

```
(kali@kali):~$ wafw00f tesla.com

  ( Woof! )
  / \
 /   \
(   )
 \   /
  \ /

- WAFW00F : v2.2.1 -
The Web Application Firewall Fingerprinting Toolkit

[*] Checking https://tesla.com
[*] The site https://tesla.com is behind CacheWall (Varnish) WAF.
[*] Number of requests: 2

(kali@kali):~$
```