A **subdomain** is a **child domain** that exists under a main (root) domain.
It helps organize and separate different sections, services, or environments of a website.

Ex:blog.raju.com

While organizations invest heavily in securing their primary web properties (www.raju.com and raju.com), subdomains often remain the silent entry point for attackers.

Forgotten, misconfigured, or poorly maintained subdomains dramatically expand the attack surface and frequently become the real weakest link during breaches, penetration tests, and bug bounty discoveries.

The Illusion of Strong Perimeter Security

Most companies focus protection efforts on:

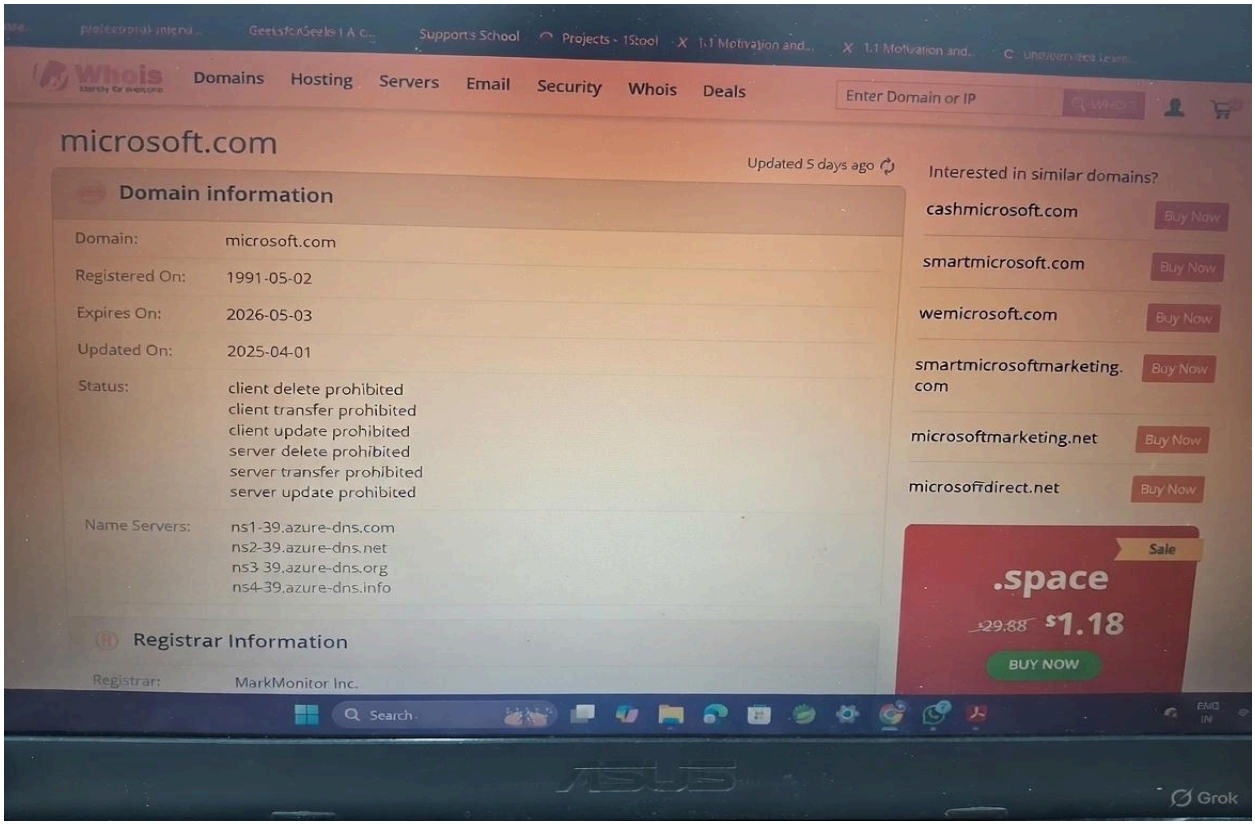
- www.raju.com
- raju.com (root domain)

These flagship assets typically receive:

- Web Application Firewall (WAF) rules
- Strict Content Security Policy (CSP) & HTTP headers
- Regular patching & vulnerability scanning
- **Continuous monitoring & logging**
- Multi-factor authentication (MFA) enforcement

However, the reality is different for subdomains.

Looking them in whois lookup tool



How Organizations Create Hundreds of Subdomains Over Time

Business needs lead to rapid creation of subdomains for legitimate purposes:

- Testing & Quality Assurance —• test.raju.com
- Development environments •dev.raju.com, developer.raju.com
- Staging / Pre-production —• staging.raju.com, uat.raju.com

- Admin & management panels
  admin.raju.com, portal.raju.com,
  manage.raju.com

• Legacy & old projects —• old.raju.com, legacy.raju.com,
v1.raju.com

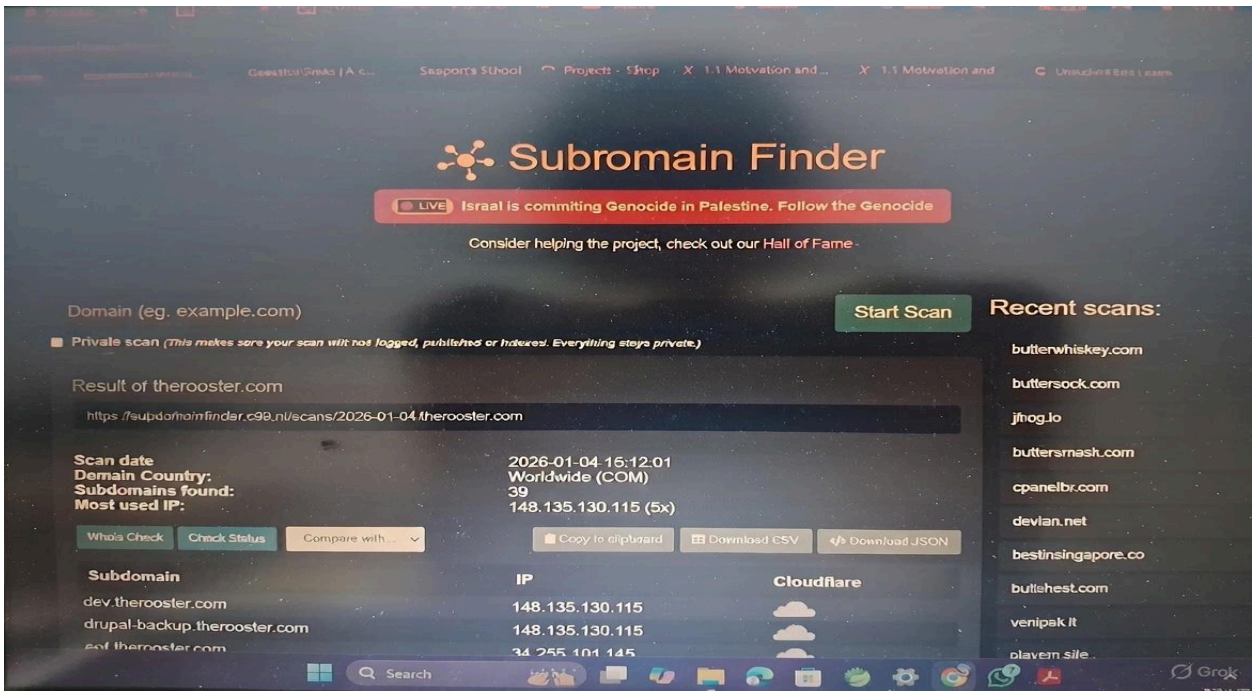
- Marketing campaigns & temporary
  microsites —• summer2025.company.com,
  blackfriday.raju.com

- VPN & internal tools —• vpn.raju.com, remote.raju.com

- Third-party integrations —•
  api.partner.raju.com, webhook.raju.com

Many of these are created quickly, used briefly, then forgotten — but never properly decommissioned.

We can find them using subfinder tool



Why Forgotten Subdomains Are Prime Targets for Attackers

Dangerous conditions:

• Outdated & unpatched software

—• Running old versions of Apache, WordPress, PHP, Node.js, etc. with known CVEs (e.g., Log4Shell remnants, outdated Laravel, etc.)

• Lack of WAF / monitoring

—• Bypass the main domain's protection entirely; no alerts on suspicious activity

- Weak or default credentials

—• admin:admin, root:toor, no MFA, or exposed .htaccess bypasses

- No regular patching or vulnerability management

—• Years-old vulnerabilities remain exploitable

• Exposed sensitive data

—• Debug modes enabled, .env files, source code backups, database dumps, API keys in javascript

• Dangling DNS records (subdomain takeover risk)

—• CNAME points to deleted cloud resources (AWS S3, Azure Blob, GitHub Pages, Heroku, etc.) —• attackers claim & control the subdomain

Result: While the main website appears "locked down," attackers quietly gain foothold through these forgotten doors.