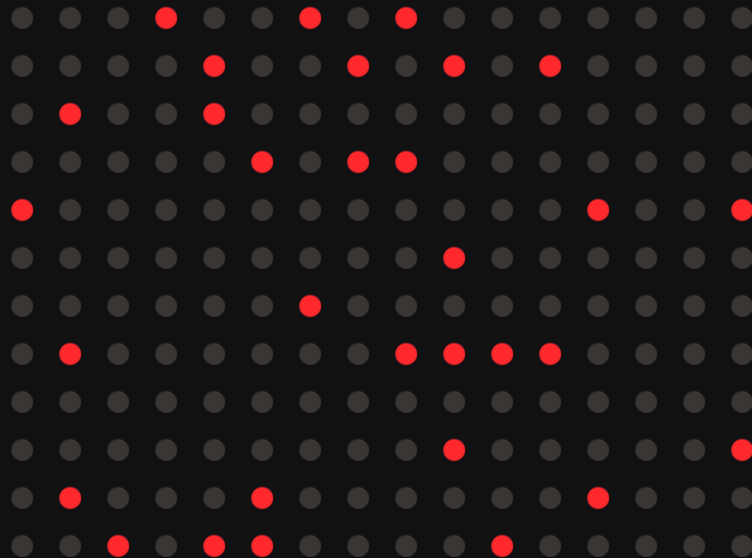


The Shadow Traffic Problem

Combating AI-driven automated threats to websites



Tin Zaw Project Leader, OWASP Automated Threats · Staff Technical PM, Fastly

ABOUT THE SPEAKER

Defending cyberspace since ...



Role

Staff Technical Product Manager,
security products, at an edge cloud
provider



Focus

Denial of Service protection, bot
management, application security, and
safe and ethical AI



OWASP

Co-lead of the Automated Threats to
Web Applications project since 2016 —
handbook, ontology (OAT-001...021),
examples, countermeasures



Find me

linkedin.com/in/tinzaw
[owasp.org/www-project-automated-thr
eats-to-web-applications](https://owasp.org/www-project-automated-threats-to-web-applications)

Agenda

01

Automated Threats, A Brief History

OWASP Automated Threats projects, threat ontology (21 OATs), examples and counter measures

02

Today's Bot Landscape

Scraping as a Service, residential IP addresses

03

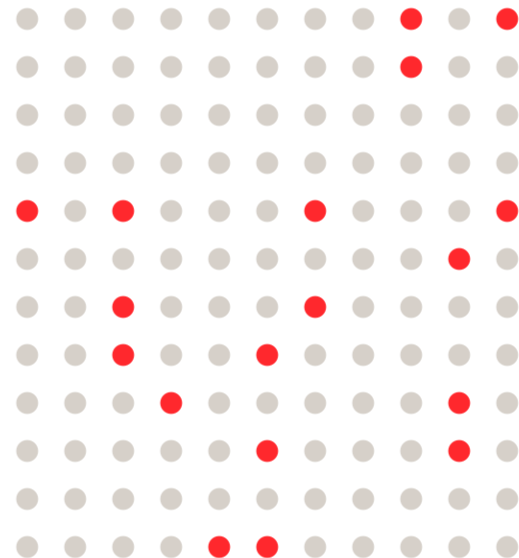
How AI is making it worse

Rise of Agentic Coding

04

How We Can Address the Problem

Use all tools available and possible



01

Automated Threats, A Brief History

Automation that is shaped like a person defeats signals designed to detect and defend machines.



THE CORE PROBLEM

You cannot patch a **feature that is working as intended.**

The problem with automated threats is that the automation exploits features designed for humans.

THE CORE PROBLEM

No bugs, but no **problem**?

“

All high and medium vulnerabilities eliminated, **OWASP Top 10s** covered, and the S-SDLC ticking along nicely.



Product Security

“

Wait! My Ops team is battling against attacks all the time.



Operations

The 21 vectors of OWASP Automated Threats (OATs)

The OWASP ontology classifies 21 distinct OAT events grouped by attacker intent.

Not a Top 10. A complete vocabulary.

Identity & Credentials	Payment & Inventory	Competitive & Market	Reconnaissance & Testing	Content & System Abuse
OAT-007 Credential Cracking	OAT-001 Carding	OAT-005 Scalping	OAT-004 Fingerprinting	OAT-017 Spamming
OAT-008 Credential Stuffing	OAT-010 Card Cracking	OAT-013 Sniping	OAT-018 Footprinting	OAT-003 Cost-Inflation Fraud
OAT-019 Account Creation	OAT-012 Cashing Out	OAT-011 Scraping	OAT-014 Vulnerability Scan	OAT-015 Denial of Service
OAT-020 Account Aggregation	OAT-021 Denial of Inventory	OAT-016 Skewing		OAT-009 CAPTCHA Defeat
				OAT-002 Token Cracking
				OAT-006 Expediting

OATs – Account Take Over (ATO)

Cracking vs. Stuffing — same goal, different means

● OAT-007 | Credential Cracking

Goal: To take over an account

Method: Brute force, dictionary (word list) and guessing attacks

Detection Signal: High number of failed login attempts, containing variations on account name and/or password

Countermeasures: Authenticating (preventing users from weak passwords), Rate (limiting attempts), Monitoring, Contract (service limits)

● OAT-008 | Credential Stuffing

Goal: To take over an account

Method: authentication credentials stolen from elsewhere, tested against authentication mechanisms

Detection Signal: Sequential login attempts with different credentials from the same HTTP client

Countermeasures: Rate (limiting attempts, success or failure per session/client/fingerprint), Monitoring, Reputation, Fingerprinting

Denial of Inventory and Cost-inflation Fraud

● OAT-021 | Denial of Inventory

Goal: Holding limited inventory items for resale or to deny them to legitimate buyers

Method: Selection and holding of items from a limited inventory, but never actually buying

Detection Signal: Increased stock held in baskets or reservations, elevated basket abandonment

Countermeasures: Authentication (only for registered users), Monitoring (their activities), Rate, Contract (explicitly ban such behavior)

● OAT-003 | Cost-inflation Fraud (formerly Ad Fraud)

Goal: Inflation of charges to the service provider (website operator or ad provider)

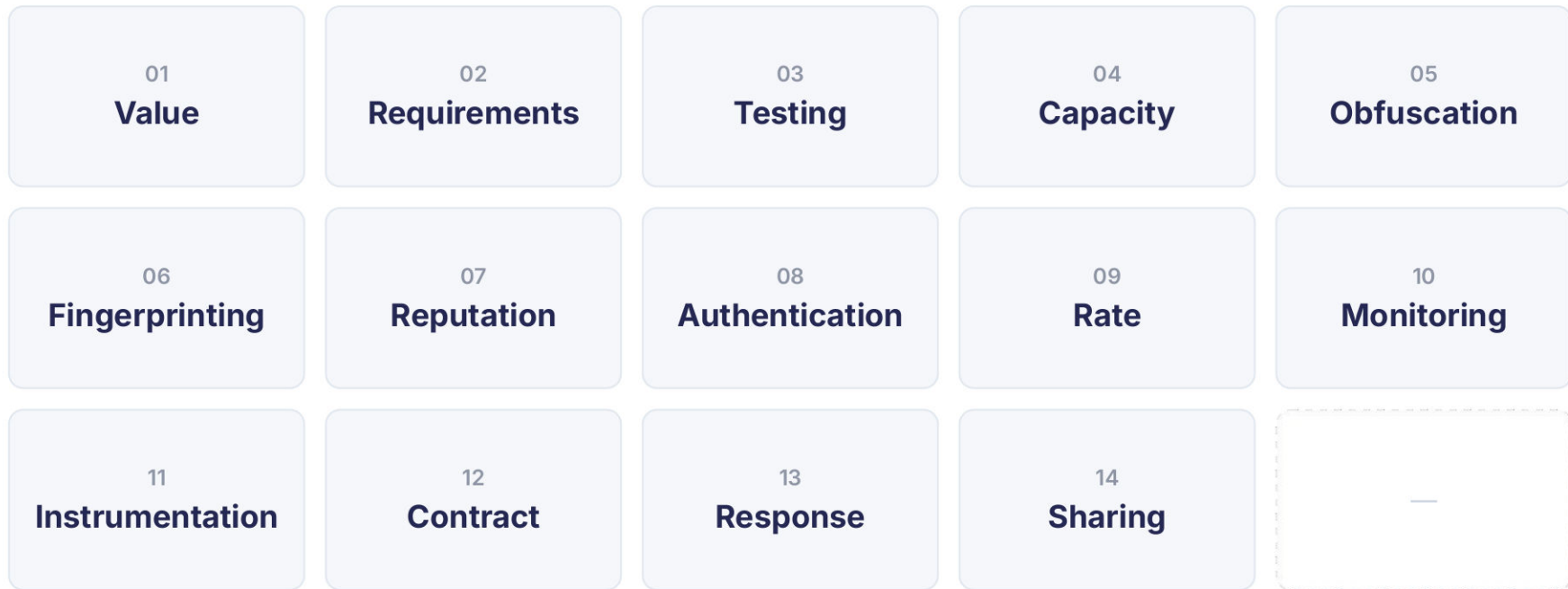
Method: Automated and repeated invocation of services (ad impressions, SMS messages) for illegitimate uses

Detection Signal: Disproportionate use of functionality requiring third-party APIs, resulting in escalated charges

Countermeasures: Rate (limiting per user, session, IP), Monitoring, Contract (excluding automated bot usage)

Countermeasures

Fourteen Countermeasures



Countermeasures

Countermeasure Classifications (Examples)

Countermeasure name	SDLC Stage		Countermeasure Type		
	Builder	Defender	Prevent	Detect	Recover
Value	✓ Y		✓ Y		
Requirements	✓ Y	✓ Y	✓ Y	✓ Y	✓ Y
Rate	✓ Y	✓ Y	✓ Y	✓ Y	
Contract		✓ Y	✓ Y		

02

Today's Bot Landscape

Automation with commercial motivations



Today's Bot Landscape

The signals that separated humans from bots have inverted

● Yesterday's assumptions

Bots come from data centers

Bots don't run JavaScript

Bots fail CAPTCHAs

Bots are fast and repetitive

● Today's shadow traffic

Residential and mobile egress, same ASNs as customers

Full browser engines with real fingerprints

Agents solve challenges, or hand them to humans

Paced, randomized, and session-aware

New SaaS: OAT-11 Scraping as-a Service

- Apify
- Zyte
- SerpApi
- Acurast, decentralized powered by *your* phones

Bot net powered by you:

The Hidden Backdoors Inside Millions of Smart Devices | WSJ

<https://www.youtube.com/watch?v=apEPPKYgLL0>

Characteristics

- Realtime
- Geo distributed
- Residential IPs
- Structured data from unstructured sources
- API-driven, usage based pricing



In the greed-driven world, CAPTCHA is a losing cat-and-mouse game!

Today's Bot Landscape

Automation got cheaper, smarter, and better disguised



Cheaper

Hosted headless browsers and agent frameworks turn a weekend script into a one-line API call. The cost floor for a convincing client is now near zero.



Smarter

LLM-driven agents read the page, recover from errors, and solve the puzzles that used to stop them. Behavior signatures drift toward human.



Disguised

Residential proxies, real device fingerprints, and rotating identities make the IP reputation era feel quaint.

03

How AI Is Making It Worse

Automation that is shaped like a person defeats signals built to find machines.



Bot landscape – Early 2026

Bots vs. Humans in 2026

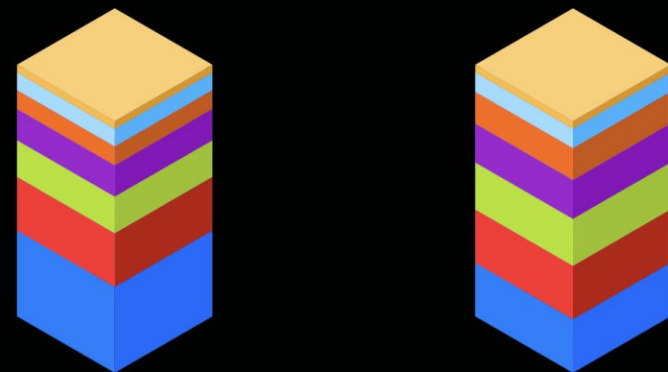
The Global Baseline Received

51% human traffic

48% unwanted bot traffic

1% wanted bot traffic

Source: Fastly Threat Insights Report



Cache

Origin



Fastly's Vantage Point

166

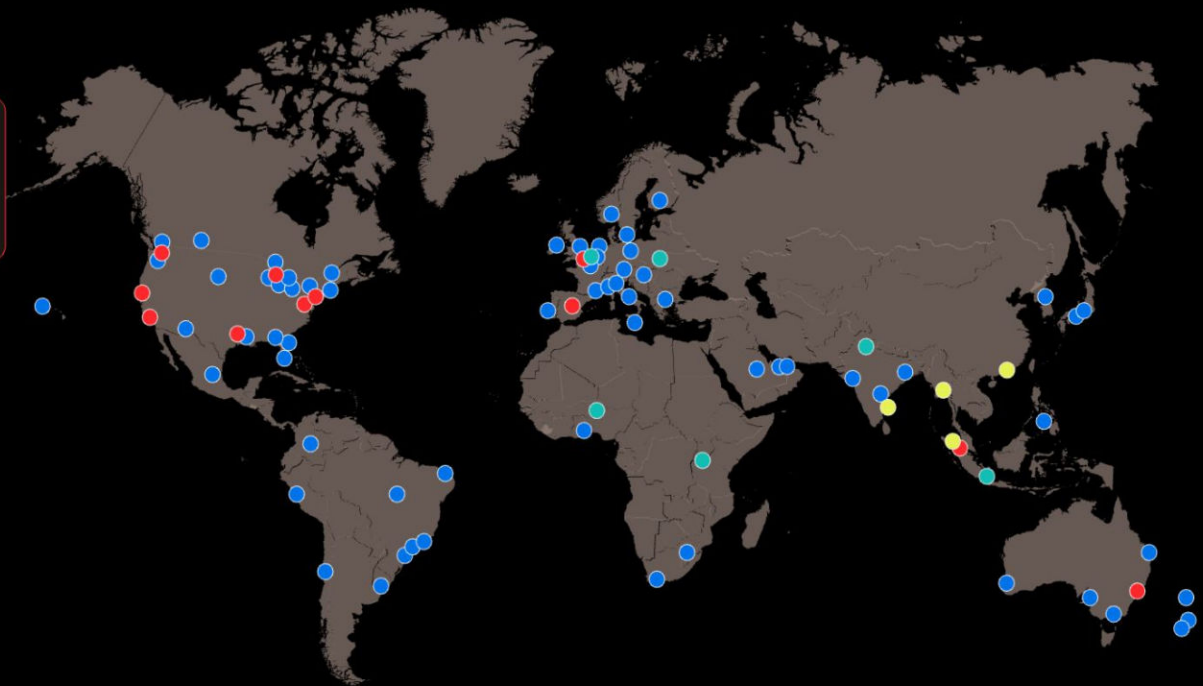
POPs

622 Tbps

Edge network capacity

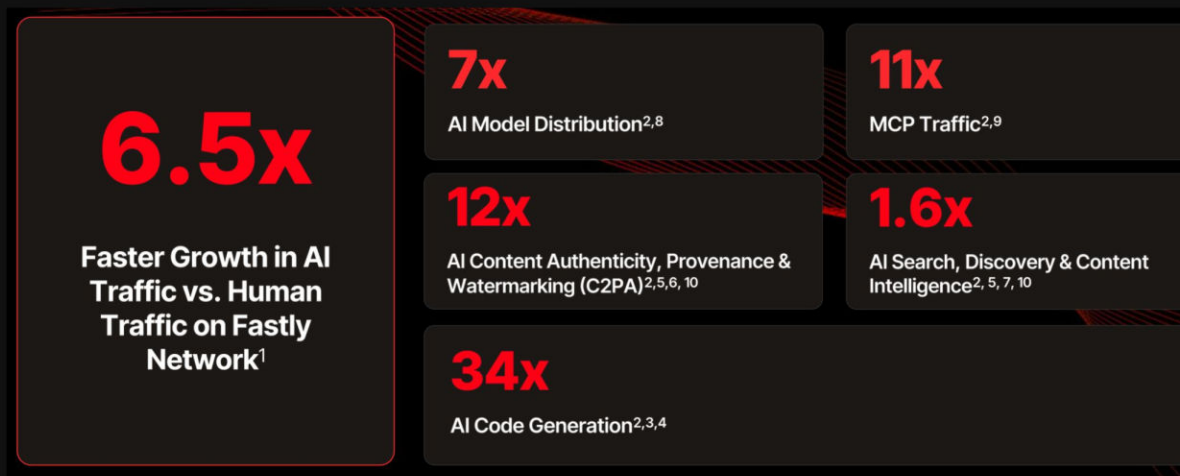
One network designed and
sized for the AI Era

- Single POP
- Multiple POPs
- Planned new POP
- Planned upgrade



Bot landscape – Late 2026

- Bot traffic has surpassed human-generated traffic as of the second half 2026
- Agents are much chattier than humans



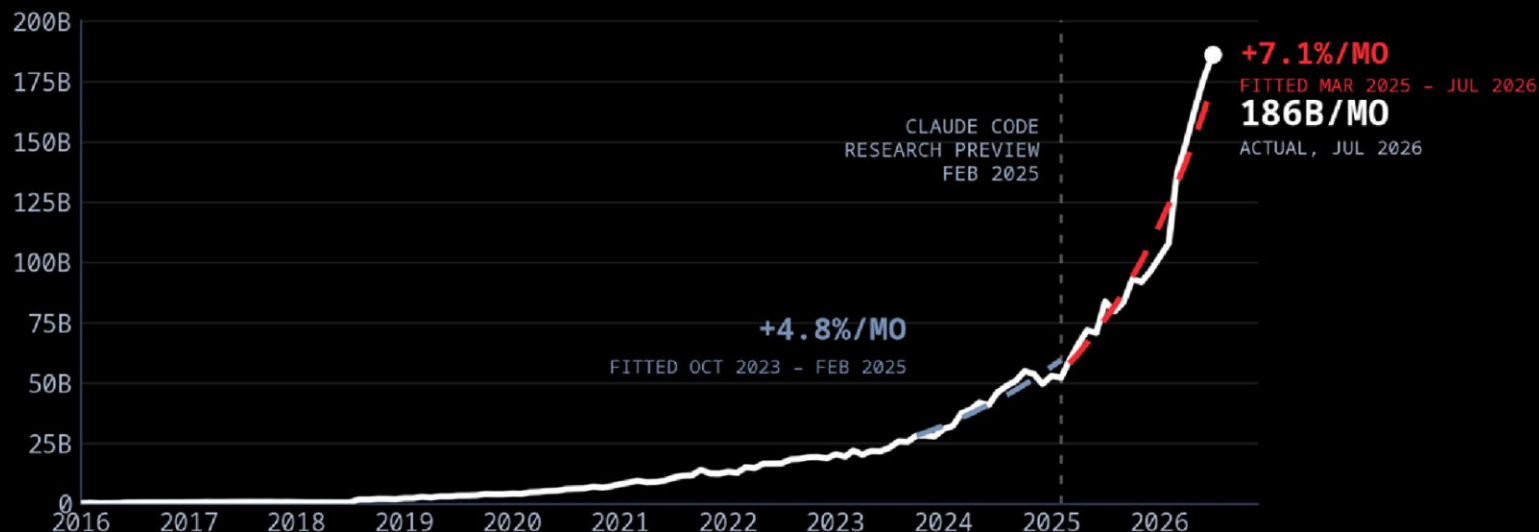
Check out next Fastly Threat Insights Report (available soon)

<https://www.fastly.com/threat-insights>

Arrival of Agentic Coding

PyPI Changed Gear

Monthly file downloads, 2016–2026 · linear scale · dashed = log-linear fit per window

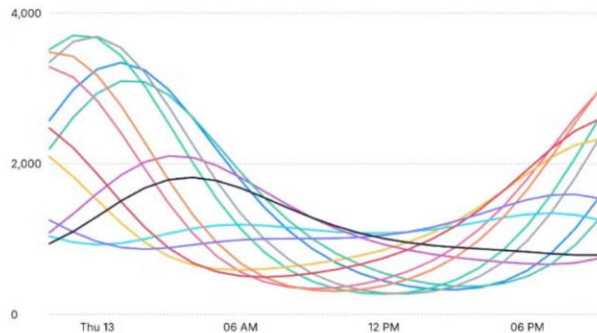


Today's Bot Landscape

AI Crawlers (One Day in August, 2026)

Bot traffic by category

Request volume and traffic distribution across all automated categories



AI Crawler	35.96K
AI Fetcher	37.36K
Accessibility	34.19K
Content Fetcher	29.36K
Monitoring	27.32K
Online Marketing	29.93K
Page Preview	34.81K
Platform Integrations	37.46K
Research	35.45K
Search Engine Crawler	30.62K
Search Engine Optimization	27.42K
Security Tools	28.79K

Top bots by requests

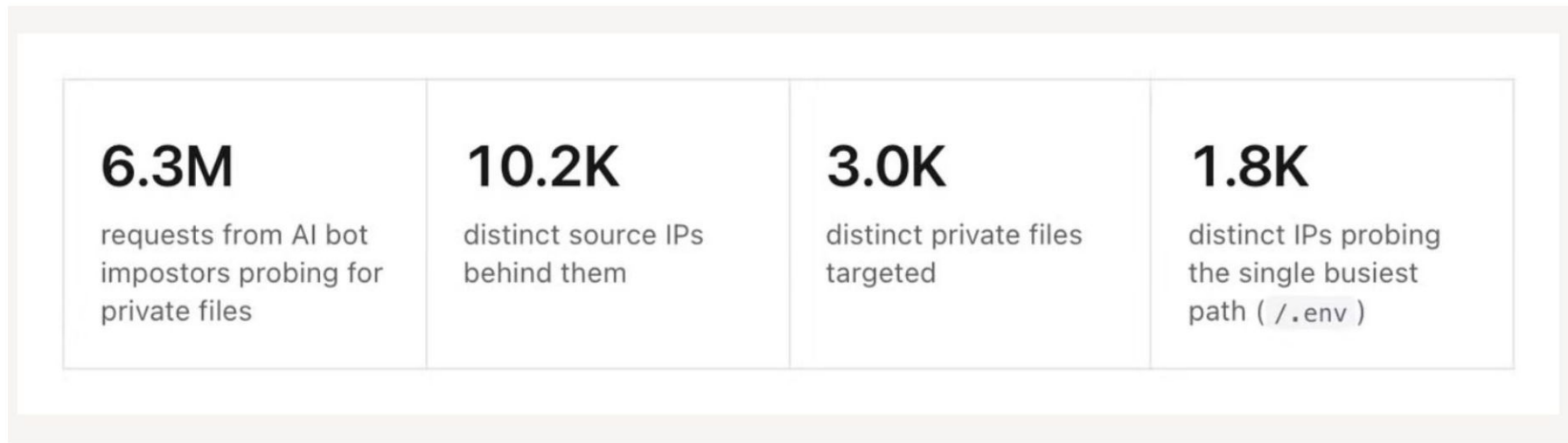
Most active automated actors by request volume

Bot identity	Category	Action	Traffic (~24h)
PetalBot	Search Engines	Allow	17.11K 1.96% blocked
Naver-Yeti	Search Engines	Allow	16.99K 1.98% blocked
Exabot	Search Engines	Allow	16.46K 2.04% blocked
AlexaBot	Search Engines	Allow	16.03K 1.95% blocked
facebot	Search Engines	Allow	14.86K 1.94% blocked
AI2Bot	AI Crawlers	Dynamic challenge	11.23K 89.96% blocked
MojeekBot	Search Engines	Allow	11.18K 1.93% blocked
Meta-ExternalAgent	AI Crawlers	Dynamic challenge	10.66K 90.09% blocked

Today's Bot Landscape

AI Imposers

How much impostor AI traffic goes looking for private files in a single day?



Out of approximately 10,200 identified IP addresses, roughly 1,600 were responsible for 94% of that day's impostor requests. (Source: GrayNoise and Fastly)

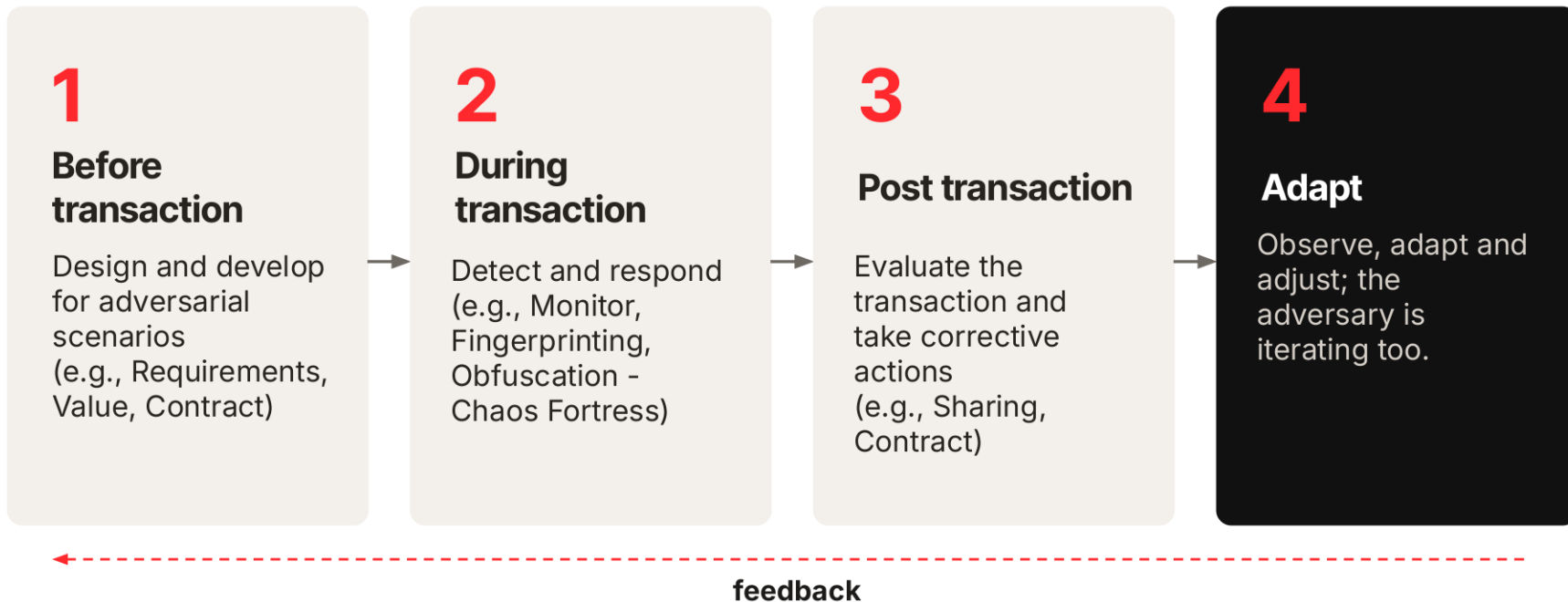
04

How We Can Address the Problem

Use all tools available and possible



Holistic Approach: A continuous process, not a rule set



What do laws say about scraping?



In the **US**, if data is publicly visible without a **login**, scraping it is generally permissible unless you break a technical barrier or agreed to a **contract**.



In **Europe**, "publicly visible" does not mean "free to process"—especially if the data touches **personal information** or machine-readable **opt-outs**.



Singapore combines one of the world's most **permissive** copyright exceptions for AI and data analytics with practical **exemptions** for public personal data.

Source: Google Gemini research with references

Framing as an economic problem

1

Every business, offline or online, has **overhead cost**. Frame it as **overhead to revenue** ratio.

2

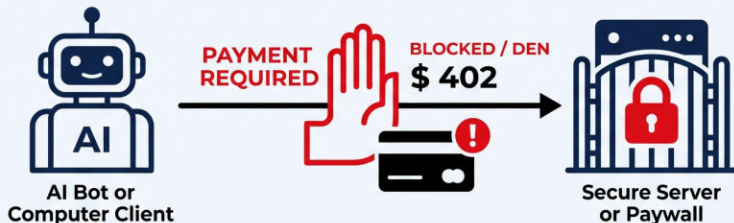
Manage unwanted and wanted **bot-generated automation as cost drivers** to the total cost of doing business.

3

Consider billing the bots.

Tools for Thought

HTTP 402 PAYMENT REQUIRED



A **non-standardized client error** reserved for paywalls, subscription thresholds, and billing issues.



PAYMENT FAILED -
The card on file was declined or transaction was unsuccessful.



QUOTA EXHAUSTED -
Monthly API credits or usage have been completely spent.

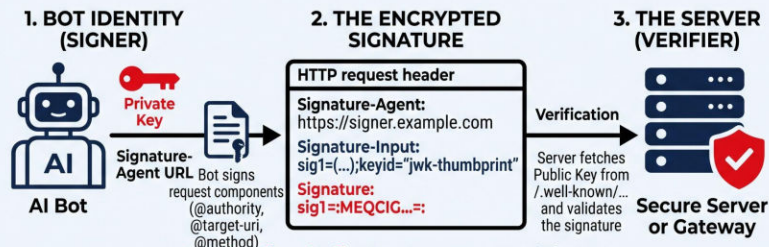


PAYWALL LOCKED -
Premium or tiered content that an upgraded subscription.

RFC 9421

web-bot-auth

HTTP MESSAGE SIGNATURES FOR BOT AUTHENTICATION



A **non-standardized client error** reserved for pay subscription thresholds, and billing issues.



ZERO-TRUST IDENTITY
Hard to spoof. Binds bot identity to the request, preventing unauthorized bots.



NETWORK INDEPENDENT
Works across dynamic IPs, cloud hosting, and serverless edge functions.



HIGH INTEGRITY
Binds host, path, and method, preventing tampering or signature replay.

Questions?

Tin Zaw

 linkedin.com/in/tinzaw

 owasp.org/www-project-automated-threats-to-web-applications

 tin.zaw@owasp.org

Slides will be published on swisscyberstorm.com

