

AI Training Bot Swarms and Catalog Disruptions at University Libraries: Technical Patterns, Impacts, and Mitigation Strategies

Executive Summary

Since 2024, university libraries worldwide have faced an unprecedented wave of automated bot traffic driven by generative AI companies seeking training data. The University of North Carolina at Chapel Hill (UNC) Libraries was among the first institutions to experience severe service disruptions, documenting a layered defense evolution from basic IP blocking to Cloudflare Turnstile in-browser verification ([Code4Lib Journal](#)). Duke University Libraries deployed Anubis proof-of-work challenges beginning June 2025, reporting faster website response afterward ([Duke University Libraries](#)). Sector-wide surveys confirm that over 90% of institutional repositories encounter aggressive bots, with 80% reporting service disruptions ([COAR](#)). The most effective mitigation strategies combine multiple layers: facet-based bot detection (81% request reduction at UNC), commercial WAF filtering, and in-browser client verification via Cloudflare Turnstile or Anubis proof-of-work ([Code4Lib Journal](#)). However, bot operators rapidly adapt to each countermeasure, shifting IP sources, user agents, and geographic origins, making this an ongoing "cat-and-mouse" challenge ([COAR](#)).

Scope and Evidence Base

This report synthesizes evidence from primary institutional sources, library-technology publications, sector-wide surveys, academic literature, and technical documentation. The strongest quantitative evidence comes from a detailed [Code4Lib Journal](#) article by UNC Libraries staff documenting their year-long mitigation journey. Duke University Libraries provides a complementary case study on Anubis deployment. Two major sector surveys—the GLAM-E Lab report (43 institutions) and the COAR survey (66 repositories)—provide breadth. Academic literature from IEEE and Taylor & Francis adds peer-reviewed context on traffic

composition and institutional responses. Technical documentation from Cloudflare and the Anubis project (TecharoHQ) supplies implementation details.

Evidence Quality Assessment

Source type	Evidence quality	Quantitative rigor
UNC Code4Lib article	High—first-hand technical account with specific metrics	Strong traffic, ban, and effectiveness data
Duke University Libraries page	Medium—official institutional documentation	Qualitative only; no quantitative metrics
GLAM-E Lab survey report	High—structured survey with 43 respondents	Mixed; some quantitative examples, mostly qualitative
COAR survey report	High—structured survey with 66 respondents	Strong prevalence and impact statistics
Anubis GitHub/TecharoHQ docs	Medium—technical project documentation	Limited; deployment examples but few metrics
Cloudflare Turnstile docs	Medium—vendor documentation	Technical specifications; no independent benchmarks
IEEE paper (Aguilar-Bueno et al.)	Medium—peer-reviewed conference paper	Strong: 89% non-human traffic finding
Taylor & Francis paper (Green)	Medium—peer-reviewed journal article	Survey-based: 116 respondents, 42% workload increase

The current evidence base is strongest in library-technology journals, official institutional pages, and repository-sector reports. Peer-reviewed incident studies remain scarce; the phenomenon is too recent for extensive academic publication.

Incident Landscape

UNC Chapel Hill University Libraries

UNC Libraries experienced two major waves of bot traffic. The first wave began in Spring 2024, producing sluggish catalog responses and targeting full-text resources, images, and audiovisual content. Traffic initially originated from commercial cloud providers, primarily Amazon AWS and Microsoft Azure. UNC engineers mitigated this wave relatively quickly ([Code4Lib Journal](#)).

A second, "seismic" wave began in late Fall 2024, substantially more intense than the first. This wave used masked traffic sources and evolved behavior to evade each blocking effort. The new traffic pattern appeared first on the catalog discovery layer rather than on full-text resource sites. Bots used a generic browser user agent and residential ISP addresses, defeating earlier user-agent-based controls. The cross-departmental response involved libraries IT staff, campus security, and campus networking staff. Over approximately one year, the team escalated through fourteen distinct mitigation layers ([Code4Lib Journal](#)).

Duke University Libraries

Duke University Libraries reported that AI bot harvesting agents targeted the Duke Digital Repository and Archives & Manuscripts Collection Guides, pulling content for use by large language models. Bot traffic rendered sites inaccessible to human patrons. The library noted that bots sent very large numbers of requests from thousands of individual IP addresses, disguised as real browsers, and followed links across pages including potentially endless links in the Duke Digital Repository. Previous mitigation approaches—monitoring IP addresses, adding temporary blocks, and increasing application resources—became less effective as bots used wide IP ranges and consumed any additional capacity. In June 2025, Duke began rolling out Anubis proof-of-work challenges, reporting that websites responded "much faster" afterward ([Duke University Libraries](#)).

Broader Sector Incidents

The GLAM-E Lab surveyed 43 GLAM institutions across Europe, North America, and Oceania. Of 39 respondents experiencing increased traffic, 27 attributed it wholly or partly to AI training bots, with 7 more believing bots might be contributing. One respondent's traffic grew from 600,000 visitors in February 2025 to 1 million in March 2025, attributed to automated extraction.

Another institution's reported visits fell from 1 million to 125,000 after suspected bot traffic was filtered. The Perseus Digital Library reported continuing denial-of-service activity from Alibaba subnets. Wikimedia reported that bots accounted for 65% of its most expensive traffic ([GLAM-E Lab](#)).

The COAR survey gathered 66 responses from repositories worldwide. Over 90% of respondents indicated their repository encounters aggressive bots, usually more than once a week. 80% reported service disruptions including slowdowns, short downtimes, and major outages. One respondent documented more than 196,000 accesses in 60 hours. Another reported performance degradation once or twice daily, with complete system crashes at least weekly requiring service restarts. According to the cited "2024 Bot Usage Activity Report," at least 11 AI assistants, data harvesters, and search agents accounted for at least 25% of all bot downloads (503,447 repository downloads) and 15% of all bot pageviews (21,048 pageviews) ([COAR](#)).

A peer-reviewed IEEE study found that approximately 89% of recorded traffic on an institutional repository was non-human, significantly inflating raw key performance indicators such as downloads and views. The study also found massive non-human traffic originating from only 0.01% of IP addresses, while 99% of human users performed up to 8 downloads per day ([IEEE](#)).

A Taylor & Francis study of 116 U.S. institutional repository professionals found that 42% had experienced workload increases due to AI, and many expected AI to distort engagement metrics over the next two years ([Taylor & Francis](#)).

Technical Patterns of Bot Harvesting

Request Volume and Distribution

- At UNC, hundreds of thousands of clients made approximately 10 requests per minute in bursts lasting 3–5 minutes, effectively functioning as a distributed denial-of-service attack ([Code4Lib Journal](#)).

- On December 4, 2024, UNC recorded 648,861 requests from 22,861 distinct IP addresses in a single day, excluding the millions of IP addresses already blocked ([Code4Lib Journal](#)).
- The week before facet-based blocking (November 28–December 4, 2024), UNC received 3,604,924 requests from 207,351 distinct IP addresses ([Code4Lib Journal](#)).
- One GLAM-E Lab respondent reported over 200 IP addresses used simultaneously by a single harvester; swarms involved hundreds of thousands or millions of IP addresses ([GLAM-E Lab](#)).
- The COAR survey documented more than 196,000 accesses in 60 hours at one repository ([COAR](#)).

IP Source Evolution

Bot traffic sources evolved systematically to evade blocking:

1. **Cloud provider IPs:** Initial high-volume requests originated from Amazon AWS and Microsoft Azure ([Code4Lib Journal](#)).
2. **Residential ISP IPs:** Traffic shifted to mainstream consumer ISPs including AT&T, Comcast, Spectrum, and Verizon-Home, using residential proxy networks that can distribute traffic over tens of millions of unrelated internet addresses. Proxy nodes were concealed within mobile apps, browser extensions, and other software ([Code4Lib Journal](#)).
3. **Geographic shifting:** After cloud and ISP blocking, traffic moved to East Asia (notably China), then Brazil, then other South American and European locations. Blocking one region caused near-immediate traffic migration to another ([Code4Lib Journal](#)).
4. **Starlink and regional ISPs:** Some GLAM-E Lab respondents reported increasing IP addresses associated with Starlink and smaller regional ISPs ([GLAM-E Lab](#)).

User-Agent Spoofing and Evasion

- Early traffic included identifiable AI bot user agents, including Anthropic's "ClaudeBot" ([Code4Lib Journal](#)).

- Later bots used a generic browser user agent (Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/136.0.0.0 Safari/537.36 Edg/136.0.0.0), defeating user-agent-based controls ([Code4Lib Journal](#)).
- User-agent strings were routinely rotated or randomized, making request-header pattern analysis largely unsuccessful ([Code4Lib Journal](#)).
- GLAM-E Lab respondents reported that user-agent strings could be changed for every request, or left empty ([GLAM-E Lab](#)).

Facet-Query Abuse

A distinctive pattern at UNC involved exhaustive facet combinations in the catalog discovery layer:

- In November 2024, the catalog received 15 searches for "Finnish" and "Music." On December 4, 2024, it received 11,329 such searches ([Code4Lib Journal](#)).
- November 2024 saw 294,398 requests with 15 or more facets applied across the month. On December 4 alone, 140,104 requests used 15+ facets, 17,527 used 20+ facets, and 203 used 25+ facets ([Code4Lib Journal](#)).
- Facet queries were resource-intensive and slow to return, contributing to large request queues and repeated query attempts ([Code4Lib Journal](#)).
- Many queries involved subject headings outside UNC's collecting strengths, with facet combinations unusually deep compared with expected human behavior ([Code4Lib Journal](#)).

Protocol Noncompliance

- Bots did not consistently follow robots.txt, sitemaps, or reasonable rate limits ([Code4Lib Journal](#)).
- AI-training bots were reported to ignore robots.txt across the GLAM-E Lab survey, which characterized the protocol as voluntary and unable to independently prevent bot access ([GLAM-E Lab](#)).

- The COAR survey confirmed that less scrupulous systems ignore robots.txt rules, and even well-intentioned services may not reliably comply ([COAR](#)).

Adaptive Behavior

Bots demonstrated systematic adaptation to mitigation measures:

- Switched IP addresses after individual IP blocks ([Code4Lib Journal](#))
- Changed user agents after user-agent-based blocking ([Code4Lib Journal](#))
- Moved from cloud-provider IPs to residential ISP IPs ([Code4Lib Journal](#))
- Limited the number of facets after facet-based bans were implemented ([Code4Lib Journal](#))
- Drew from larger sets of IP addresses after subnet blocking ([Code4Lib Journal](#))
- Shifted traffic geographically after regional blocks ([Code4Lib Journal](#))
- Bots appeared to probe different access vectors, analyze responses, and modify their approach. Patterns that appeared consistent shifted within days or sometimes hours after blocking measures were implemented ([Code4Lib Journal](#)).

Operational Impacts

Server Load and Performance

Metric	Source	Value
UNC peak daily requests (Dec 4, 2024)	Code4Lib	648,861 from 22,861 IPs
UNC weekly requests before facet blocking	Code4Lib	3,604,924 from 207,351 IPs
UNC weekly requests after facet blocking	Code4Lib	673,031 from 220,110 IPs (81% drop)
GLAM-E Lab: server utilization during bot spike	GLAM-E Lab	From below 25% to 100%

GLAM-E Lab: DDoS-style incident frequency	GLAM-E Lab	Once per day, ~3 minutes duration
COAR: accesses in 60 hours	COAR	196,000+
GLAM-E Lab: traffic spike example	GLAM-E Lab	500 normal visits/day to 10,000 bot visits (20x increase)
IEEE: non-human traffic percentage	IEEE	~89% of recorded traffic
Wikimedia: bot share of expensive traffic	GLAM-E Lab	65%

Service Disruptions

UNC experienced intermittent service disruptions for multiple consecutive days, with sluggish catalog responses, overwhelmed server resources, large request queues, and renewed outages when facet-based blocking was removed. The traffic first affected the public catalog interface and later expanded to digital collections services. Scaling and performance optimization eventually failed because new server resources were consumed by bot traffic. UNC's on-premises infrastructure had physical hardware limitations that could not be addressed quickly ([Code4Lib Journal](#)).

Duke reported that websites became very slow, completely stalled, and inaccessible to human patrons. Adding resources did not reliably resolve the problem because bots consumed the additional capacity ([Duke University Libraries](#)).

The COAR survey found that 80% of respondents experienced service disruptions, with about one-third reporting at least one downtime lasting 2–24 hours and about one-fifth reporting outages lasting several days. One case reported long-lasting damage to the repository system. Reported resource effects included excessive CPU use, high server temperature, and energy costs ([COAR](#)).

The GLAM-E Lab report documented servers becoming unresponsive, entire collections going offline, administrative login pages becoming inaccessible, and unexpected user emails

reporting collections were offline. A hosted solution increased customer hosting fees by 10% due to bot traffic ([GLAM-E Lab](#)).

Staffing and Labor Costs

The COAR survey found that the average repository effort was about three days per month, approximately 10% of a full-time equivalent. One respondent reported 0.5 FTE per week spent on education, experimentation, and responding to outages. Another reported 20% of a full-time engineer's effort investigating and blocking AI bots. Measures were described as highly labor-intensive and a "blunt instrument" ([COAR](#)).

The Taylor & Francis study found that 42% of 116 U.S. institutional repository professionals had experienced workload increases due to AI ([Taylor & Francis](#)).

Analytics Distortion

Bot traffic massively distorted analytics reporting. One GLAM-E Lab respondent's March 2025 report showed 1 million visits; after filtering suspected bot traffic, the figure was revised to 125,000. Bot traffic caused analytics to show a reversal from mobile-oriented to desktop-oriented traffic because bots often identified as desktop browsers. The IEEE study confirmed that approximately 89% of recorded traffic on an institutional repository was non-human, significantly inflating raw KPIs such as downloads and views ([GLAM-E Lab](#); [IEEE](#)).

Mitigation Strategies

Layer 1: Server Scaling and Application Optimization

UNC initially increased server resources, tuned web-server configurations, profiled the catalog application, and implemented targeted performance improvements. New resources were eventually consumed by bot traffic. Other institutions scaled more aggressively behind load balancers; some later reported similar problems and concerns about cloud-scaling fees. UNC's on-premises infrastructure had physical hardware limitations ([Code4Lib Journal](#)).

The GLAM-E Lab report confirmed that increasing capacity helped some collections handle bot spikes, but respondents noted it costs money and were reluctant to spend limited resources on improving the bot experience. Some collections that had recently upgraded infrastructure experienced minimal disruption ([GLAM-E Lab](#)).

Layer 2: User-Agent and IP Blocking

UNC traditionally blocked malicious IP addresses in local host-based firewalls. The library used shared IP and user-agent lists from NCSU Libraries, Princeton University Libraries, and other sources. Blocking addresses associated with AWS, Azure, and other cloud providers provided only temporary relief as traffic shifted to adjacent IPs. Targeting AI bot user agents including "ClaudeBot" was effective only until bots switched to generic browser user agents ([Code4Lib Journal](#)).

The GLAM-E Lab report found that IP-address blocking could overblock welcome users. One respondent blocked entire IP ranges associated with Alibaba cloud servers. Geographic blocking was used when regions were not historically significant sources of human traffic—respondents blocked Singapore, Hong Kong, and Brazil—but this also blocked welcome human users ([GLAM-E Lab](#)).

Layer 3: fail2ban with Progressive Bans

UNC deployed fail2ban as its first automated response. The tool continuously monitors log files, detects user-defined patterns, inserts firewall blocks, and removes blocks after configurable durations. UNC created a fail2ban jail for known AI-bot user agents and adopted a progressive ban cycle: 1 day, 7 days, 1 month, 6 months, 1 year. Initial short bans lasting a few hours were insufficient. Later, facet-based rules were also implemented through fail2ban using regular-expression filters for requests containing more than a specified number of facets ([Code4Lib Journal](#)).

The COAR survey also found respondents using fail2ban and the mod_evasive module for Apache. One respondent used Splunk to provide rate limiting for a DSpace repository ([COAR](#)).

Layer 4: Request Throttling

UNC initially blocked clients after 80 requests within 5 minutes, then reduced to 60 requests within 5 minutes. These thresholds produced only brief respites before new bots returned with slightly slower request streams. The deep residential-proxy IP pool allowed new clients to continue after individual clients were throttled. Request throttling alone failed to stabilize the service ([Code4Lib Journal](#)).

Layer 5: Facet-Based Bot Detection

UNC's most innovative library-specific defense exploited the observation that bots requested unusually large combinations of facets. The library developed a regular-expression filter for fail2ban that isolated requests containing more than a specified number of facets. The threshold was set to 8–12 facets depending on traffic severity. All queries from on-campus and VPN networks were allowed. The method treated faceted interfaces as security honeypots and bot traps.

Results:

Interval after deployment (Dec 5, 2024)	IP addresses banned		
First 30 minutes	~700		
First 24 hours	>10,000		
First 48 hours	~16,000		
Period	Requests	Distinct IPs	Outcome
---	---	---	---
Nov 28–Dec 4 (before facet blocks)	3,604,924	207,351	Baseline week
Dec 6–12 (after facet blocks)	673,031	220,110	Over 81% request reduction

The facet-based method remained sufficient for nearly one month before bots evaded it by limiting facet counts and using larger IP pools. In the five months after introducing the facet limit, UNC received only one user report of a problem with the limit ([Code4Lib Journal](#)).

Layer 6: Subnet-Based Blocking

After several weeks of stability, UNC faced huge numbers of neighboring IP addresses. The library changed facet-based rules from blocking only the offending IP to blocking the entire /24 subnet (256 IP addresses). UNC also lowered the facet limit and began banning clients after one flagged query rather than multiple. This broadened protection while minimizing collateral impact on human users ([Code4Lib Journal](#)).

Layer 7: Regional Traffic Blocking and Prioritization

UNC briefly banned all addresses from China, which caused traffic to move immediately to Brazil. Blocking Brazil caused traffic to move to South America and Europe. UNC described the process as "whack-a-mole." The institution sought a non-local solution positioned between incoming traffic and the server to block bots and prioritize on-campus and VPN users. Future concepts include tiered access channels prioritizing an institution's primary service areas during high load, allowlisting known international research partners, and transparent communication about resource allocation. The article notes that regional tiering conflicts with the professional commitment to open access ([Code4Lib Journal](#)).

Layer 8: Commercial Web Application Firewall

UNC rapidly implemented a trial of a cloud-based, GDPR-compliant WAF for the catalog server. The WAF provided bot filtering based on client "trustworthiness" ratings, easier monitoring, and easier blocking. The WAF initially provided relief. After nearly one month of stability, large numbers of bots began passing through the WAF. UNC licensed the WAF for one year. The WAF's cloud location prevented direct inclusion of the campus VPN in regional rules ([Code4Lib Journal](#)).

The GLAM-E Lab report identified Cloudflare as the most popular and widely used third-party bot-countermeasure service among respondents. One respondent reported that after implementing Cloudflare protections, bot spikes remained visible in the dashboard but no longer negatively affected the system. However, some respondents worried about reliance on Cloudflare indefinitely, questioned whether they could independently verify its effectiveness,

and expressed concern that widespread reliance could create a monoculture vulnerable to industrial-scale countermeasures. False positives were reported—one school was incorrectly identified as a bot and blocked ([GLAM-E Lab](#)).

Layer 9: Commercial AI-Driven Bot Detection

UNC evaluated an AI-driven bot-detection system offered as a WAF component. The system appeared to solve the problem, but its cost would have more than doubled the WAF price, making it prohibitively expensive. UNC therefore investigated regional traffic prioritization and other options including Cloudflare Turnstile ([Code4Lib Journal](#)).

Layer 10: Cloudflare Turnstile (In-Browser Client Verification)

UNC adopted Cloudflare Turnstile despite concerns about accessibility, user annoyance, and the prospect of requiring a CAPTCHA-like mechanism. Turnstile was free to use and generally operated automatically. According to the UNC Code4Lib article, Turnstile uses browser fingerprinting, behavioral analysis, and lightweight challenges that typically complete without user intervention, and Cloudflare asserts that Turnstile is WCAG 2.1 Level AA compliant ([Code4Lib Journal](#)).

UNC initially applied Turnstile selectively to the most resource-intensive catalog endpoints—complex faceted searches and endpoints vulnerable to bot abuse—and exempted on-campus and VPN users entirely. No challenges were issued for record views or searches without facets. A client was challenged no more than once per day.

Technical mechanism: Turnstile works in two steps: a JavaScript widget runs challenges in the visitor's browser and produces a token (up to 2,048 characters), then the server sends that token to Cloudflare's Siteverify API to confirm validity. Tokens expire after 300 seconds and can only be validated once. Three widget types are available: Managed (recommended; selects challenge based on browser signals, presents interaction only if automated traffic is detected), Non-Interactive (widget displayed but no user interaction required), and Invisible (completely invisible, challenge runs in background) ([Cloudflare](#); [Cloudflare Challenges](#)).

Results at UNC:

- Server load returned to normal within hours of deployment
- Very few further outages occurred on protected endpoints
- Remaining outages were eliminated by restoring facet-based bans
- Only two incidents involving legitimate-user service disruption occurred in the first four months
- The article identifies Turnstile as one of the most effective current layers of the overall strategy

([Code4Lib Journal](#))

Layer 11: Anubis (Proof-of-Work Challenges)

Anubis is an open-source project (TecharoHQ) that "weighs the soul of your connection" using a SHA-256 proof-of-work challenge to protect upstream resources from automated bots ([Anubis GitHub](#)). The browser completes a small computational task that is trivial for individual sessions but potentially prohibitively expensive at scale. According to the UNC Code4Lib article, computational requirements can be dynamically adjusted based on system load—the barrier increases during suspected bot activity and remains negligible during normal operations ([Code4Lib Journal](#)). Installing Anubis will likely result in a website not being indexed by some search engines, which the project considers a feature ([Anubis GitHub](#)).

The TecharoHQ solutions page cites proof-of-work protection at [git.kernel.org](#) and [lore.kernel.org](#), and states that SourceHut deployed Anubis to [git.sr.ht](#), describing the solution as "robust and reliable," sufficient to mitigate LLM traffic, with minimal user impact. The Anubis challenge drawing appears on average for less than 5 seconds ([TecharoHQ](#)).

Duke University Libraries began rolling out Anubis in June 2025. Users not on a Duke network may see a "Making sure you're not a bot!" message, indicating the application is asking the user's computer to execute a small amount of JavaScript. After implementation, Duke observed that websites were responding "much faster." No quantitative before-and-after measurements

were provided. Users on older browsers, with JavaScript disabled, or with cookies disabled face higher risk of being blocked ([Duke University Libraries](#)).

Several academic libraries and archives implemented Anubis, including:

- Duke University Libraries' Archives & Manuscripts
- Connecticut's Archives Online
- UCLA Library Digital Collections
- Columbia University Libraries Finding Aids

([Code4Lib Journal](#))

Layer 12: robots.txt and AI Bot Lists

Almost all GLAM-E Lab respondents updated their robots.txt files. This was generally unsuccessful—AI-training bots ignored the protocol. No respondent reported that updating robots.txt significantly reduced traffic ([GLAM-E Lab](#)).

The COAR survey found that while most repositories implement robots.txt, there is no inherent enforcement mechanism. One respondent used Dark Visitors to obtain a publicly curated list of known AI agents and semi-automatically prepare robots.txt files. Several respondents configured robots.txt to disallow resource-intensive pages ([COAR](#)).

The ai-robots-txt GitHub project provides a community-maintained list of AI-related bots with generated configuration files for robots.txt, Apache .htaccess, Nginx, Caddy, HAProxy, and lighttpd. The repository has 4,200+ stars and 186 forks, with 756 commits as of the research date. A GitHub Action generates updated configurations from a central robots.json source. The robots.txt file implements RFC 9309, the Robots Exclusion Protocol ([ai-robots-txt GitHub](#)).

Layer 13: Cloudflare AI Labyrinth

One COAR respondent mentioned Cloudflare's AI Labyrinth, which recognizes when an AI bot has started accessing a website, uses its own AI process to generate plausible but meaningless

content, and wastes the automated system's time and bandwidth. The approach is sometimes called a honeypot—human visitors would more quickly realize the content was bogus ([COAR](#)).

Layer 14: Community-Shared Detection Patterns

UNC shared its facet-based detection method with other institutions the day after deployment. The Code4Lib Slack #bots channel served as a community coordination hub. Shared or related code repositories include:

- [samvera-labs/bot_challenge_page](#)
- [libops/captcha-protect](#)
- [NEU-Libraries/cerberus](#)
- [ai-robots-txt/ai.robots.txt](#)

UNC conducted an informal survey in March 2025. Of respondents, 65% were actively battling bot traffic or experiencing degraded services. The Engelberg Center survey found 74.4% of respondents actively engaging in bot mitigation. The COAR survey found 75.5%. Over 95% of UNC survey respondents were interested in collective knowledge sharing ([Code4Lib Journal](#)).

Comparative Mitigation Table

Incident and Impact Comparison

Institution/Source	System affected	Bot pattern	Traffic/load metric	User impact	Evidence quality
UNC Chapel Hill	Public catalog, digital collections	Facet-query abuse, residential proxy IPs, generic browser UA, adaptive evasion	648,861 req/day from 22,861 IPs; 3.6M req/week from 207K IPs	Intermittent multi-day outages; sluggish catalog	High—first-hand technical account with specific metrics

<u>Duke University Libraries</u>	Duke Digital Repository, Archives & Manuscripts	Thousands of IPs, disguised browser requests, link-following across infinite pages	"Ton of requests"; no exact figures	Sites inaccessible to patrons	Medium—official page, qualitative only
<u>GLAM-E Lab respondents</u> (43 institutions)	Various GLAM collections	Swarming bursts, 200+ IPs per harvester, millions of IPs	Up to millions of requests/day; server utilization <25% to 100%; 20x traffic increase	Servers unresponsive, collections offline, admin pages inaccessible	High—structured survey with mixed quantitative
<u>COAR respondents</u> (66 repositories)	Institutional repositories (DSpace, etc.)	Daily/near-constant bot access, multiple bots 24/7	196,000+ accesses in 60 hours; 25% of bot downloads from AI agents	80% service disruptions; weekly crashes; CPU/temperature spikes	High—structured survey with prevalence statistics
<u>IEEE study</u> (Aguilar-Bueno et al.)	Institutional repository (DSpace/Solr)	Non-human traffic from 0.01% of IPs	~89% of recorded traffic non-human	KPI inflation; analytics distortion	Medium—peer-reviewed; single-institution sample
<u>Wikimedia</u>	Wikimedia infrastructure	High-volume bot traffic on expensive endpoints	65% of most expensive traffic from bots	Not quantified	Medium—cited in GLAM-E Lab report
<u>Perseus Digital Library</u>	Digital library	Alibaba subnet denial-of-service activity	Not quantified	Continuing denial-of-service	Low—cited in GLAM-E Lab report without detail

Mitigation Strategy Comparison

Measure	Used at	Technical mechanism	Reported effectiveness	User-access tradeoff	Evidence quality
Server scaling	<u>UNC</u> , <u>GLAM-E</u> respondents	Increase resources, load balancers, caching	New resources consumed by bots; temporary relief only	No direct user impact; high cost	Medium
User-agent blocking	<u>UNC</u> , <u>GLAM-E</u> respondents	Block known AI bot UAs (e.g., ClaudeBot)	Effective until bots switch to generic browser UA	Minimal for legitimate users	Medium
IP/subnet blocking	<u>UNC</u> , <u>GLAM-E</u> , <u>COAR</u> respondents	Block individual IPs, /24 subnets, cloud provider ranges	Temporary; traffic shifts to adjacent IPs	Risk of overblocking legitimate users	Medium
Geographic blocking	<u>UNC</u> , <u>GLAM-E</u> respondents	Block countries/regions (China, Brazil, Singapore, etc.)	Short-term only; "whack-a-mole"	Blocks legitimate users in targeted regions	Medium
<u>fail2ban</u> with progressive bans	<u>UNC</u> , <u>COAR</u> respondents	Log monitoring, regex pattern matching, escalating ban durations (1 day to 1 year)	Foundation for other layers; insufficient alone	Minimal with proper on-campus/VPN exemptions	Medium
Request throttling	<u>UNC</u> , <u>COAR</u> respondents	Block after N requests in time window (80/5 min, then 60/5 min)	Brief respites only; bots adapt with slower rates	Risk of blocking power users	Medium

Facet-based bot detection	UNC	Regex filter for 8–12+ facet combinations; treat discovery layer as honeypot	81% request reduction; effective ~1 month before evasion	1 user complaint in 5 months	High—quantitative metrics
Commercial WAF	UNC , GLAM-E respondents (Cloudflare)	Client trustworthiness ratings, traffic filtering before origin	Initial relief; bots bypassed after ~1 month	False positives (school blocked); VPN integration issues	Medium
AI-driven bot detection (commercial)	UNC (evaluated, not deployed)	WAF component using AI/ML for bot classification	Appeared effective; prohibitively expensive (2x+ WAF price)	Not assessed	Low—evaluated but not deployed
Cloudflare Turnstile	UNC	In-browser JS challenge; browser fingerprinting; behavioral analysis (Code4Lib); token validation via Siteverify API	Server load normal within hours; 2 user disruptions in 4 months; very few outages on protected endpoints	Max 1 challenge/day; on-campus/VPN exempt; 2 incidents in 4 months	High—quantitative outcomes
Anubis (proof-of-work)	Duke , UCLA , Columbia , Connecticut Archives Online, SourceHut/git.sr.ht	SHA-256 proof-of-work; dynamically adjustable difficulty (Code4Lib); JS execution	Duke: "much faster" websites (qualitative); SourceHut: "robust and reliable," sufficient to mitigate LLM traffic	Drawing appears <5 sec average; higher risk for older browsers, JS/cookie disabled	Medium—qualitative at Duke; limited quantitative

robots.txt / AI bot lists	GLAM-E , COAR , UNC , ai-robots-txt	RFC 9309 robots exclusion; community-maintained AI agent lists	Generally unsuccessful; no significant traffic reduction	No user impact; voluntary compliance only	Medium—widely used but ineffective
Cloudflare AI Labyrinth	COAR respondent	AI generates meaningless content to waste bot time/bandwidth	Not quantified	Minimal—human users recognize bogus content	Low—single mention
Community-shared patterns	UNC , Code4Lib #bots, shared repos	Facet detection code, IP/UA lists, mitigation documentation	Enabled rapid cross-institutional adoption	No direct user impact	Medium
API access provision	Wikimedia (GLAM-E Lab)	Machine-optimized API for structured access	Wikimedia: successful for paid API users; bots may still prefer human pages	Requires API maintenance; uncertain bot adoption	Low—single example
Login/authentication gates	GLAM-E respondents (considered)	Require login for sensitive collections	Significantly reduces bot traffic from public sites	Reduces public accessibility; may block search indexing	Low—mostly considered, not deployed

Evidence Gaps and Reliability Assessment

Quantified Success Rates

Only UNC provides quantitative success-rate data for specific mitigation strategies. The 81% request reduction from facet-based blocking, the return to normal server load within hours of Turnstile deployment, and the low user-disruption counts (1 facet-limit complaint in 5 months, 2 Turnstile incidents in 4 months) represent the strongest evidence in the literature. Duke's

Anubis deployment reports only qualitative improvement. The GLAM-E Lab and COAR surveys provide sector-level prevalence and impact data but mostly qualitative mitigation effectiveness. No percentage success rate is provided for any individual strategy in either survey.

Server Load Metrics

UNC does not provide numerical CPU, memory, bandwidth, request-queue, or infrastructure-capacity metrics. The GLAM-E Lab provides one concrete server-utilization example (below 25% to 100%) and one DDoS-pattern example (once daily, ~3 minutes). The COAR survey reports excessive CPU use, high server temperature, and energy costs qualitatively. No institution has published comprehensive before-and-after server performance benchmarks.

Anubis Technical Details

The Anubis GitHub README is sparse (35 lines, 2.23 KB), lacking configuration options, deployment methods, computational difficulty specifications, performance measurements, and integration documentation. The TecharoHQ solutions page provides deployment examples (SourceHut, kernel.org) but no formal benchmarks. Duke's deployment page does not specify the proof-of-work algorithm, Anubis version, deployment architecture, protected hostnames, percentage of traffic challenged, block rates, or false-positive rates.

Cloudflare Turnstile Internals

Cloudflare's documentation describes the widget types and token mechanism but does not specify which browser signals are collected, the behavioral analysis methodology, the proof-of-work component (mentioned by UNC but not in Cloudflare docs), or accessibility testing details beyond the WCAG 2.1 Level AA claim.

Long-Term Effectiveness

No source provides longitudinal data beyond UNC's approximately one-year experience. The sustainability of each mitigation strategy against evolving bot behavior remains uncertain. The

COAR survey explicitly characterizes the current state as a "cat-and-mouse game" with measures that "slow bots down" but require more permanent solutions.

Recommendations for University Libraries

Immediate Actions

1. **Deploy layered defenses from day one.** UNC's experience demonstrates that no single measure suffices. Combine fail2ban with progressive bans, request throttling, and IP/subnet blocking as a baseline.
2. **Implement library-specific detection.** UNC's facet-based bot detection exploited a pattern unique to discovery layers. Libraries should analyze their own catalog and repository query patterns to identify anomalous bot behavior that generic WAFs cannot detect.
3. **Deploy in-browser client verification selectively.** Cloudflare Turnstile (free, WCAG 2.1 AA compliant) or Anubis (open-source, proof-of-work) applied to resource-intensive endpoints while exempting on-campus and VPN users provides the strongest protection with minimal user impact.
4. **Establish cross-departmental response teams.** UNC's effective response required libraries IT, campus security, and campus networking staff. Libraries should establish these relationships before incidents occur.

Medium-Term Strategies

5. **Join community mitigation networks.** The Code4Lib Slack #bots channel, shared code repositories, and community-maintained AI bot lists ([ai-robots-txt](#)) enable rapid knowledge transfer. Over 95% of UNC survey respondents expressed interest in collective knowledge sharing ([Code4Lib Journal](#)).
6. **Implement bot-aware analytics.** Bot traffic massively distorts usage statistics. The IEEE finding that ~89% of recorded traffic may be non-human ([IEEE](#)) underscores the need for

bot-filtering analytics before traffic reaches crisis levels. Libraries should not interpret raw traffic increases as human engagement growth.

7. **Evaluate WAF cost-benefit carefully.** UNC's WAF provided temporary relief but was bypassed within a month. Commercial AI-driven bot detection was prohibitively expensive (2x+ WAF price) ([Code4Lib Journal](#)). Libraries should negotiate trial periods and evaluate effectiveness before committing to annual licenses.
8. **Consider proof-of-work for open-source infrastructure.** Anubis offers a no-cost alternative for institutions that cannot or prefer not to use Cloudflare. Its deployment at Duke, UCLA, Columbia, and Connecticut Archives Online demonstrates feasibility in library environments. However, the search-engine deindexing tradeoff and limited technical documentation should be weighed.

Long-Term Directions

9. **Engage with sector-wide policy development.** COAR launched a "Repositories and AI Bots Task Force" in summer 2025. The GLAM-E Lab report calls for sustainable rules governing access by entities building AI-training datasets, practical mechanisms for signaling AI-training preferences, and privileged access for compliant entities.
10. **Plan for sustained adaptive threats.** Bot operators rapidly adapt to each countermeasure—shifting IPs, user agents, geographic origins, and query patterns within days or hours. Libraries should build adaptive defense architectures rather than static blocking rules, and budget for ongoing staffing costs (COAR estimates ~10% FTE average, up to 50% FTE in severe cases) ([COAR](#)).
11. **Address the open-access paradox.** Regional blocking and authentication gates conflict with libraries' professional commitment to open access to information. Tiered access models that prioritize an institution's primary service areas during high load, while allowlisting known research partners, offer a compromise—but require careful equity considerations.

12. **Monitor for bot traffic proactively.** Many institutions did not realize they were experiencing bot traffic until it overwhelmed their systems. Regular monitoring of server logs, traffic patterns, and analytics anomalies should be standard practice, not a crisis response.