Request bots are a common means of bypassing legacy bot mitigation solutions. They exploit architectures with poorly obfuscated client-side JavaScript by applying automated scripts that replay legitimate requests against static detection scripts.

kasada

HOW ATTACKERS USE REQUEST BOTS TO BYPASS LEGACY BOT MITIGATION SOLUTIONS

DYNAMIC CHALLENGE

Some anti-bot providers also apply "dynamic" challenges to continually revalidate that the requests are being sent from within a browser. For example, a commonly used dynamic challenge is to monitor mouse clicks, movements, and screen interactions.

REVERSE ENGINEER CLIENT SCRIPT

Attackers reverse engineer client-side JavaScript code used by anti-bot providers to collect telemetry. Bypass methods are shared within communities including GitHub and Discord groups.

CHALLENGE BYPASS

A bot operator generates or buys gesture data. This can be easily accomplished using a self-generating randomized human activity script, recording legitimate activity, or by paying for a subscription. They submit the activity data via their request bot.

1

3

4

5

GENERATE REQUEST BOT

2

The bot operator builds a request bot with non-browser based scripts to submit telemetry to the bot provider based on what's been learned from reverse engineering. This will replay the anti-bot system with seemingly legitimate data and tricks the system to validate the request as human.

RESIDENTIAL PROXY NETWORKS

The use of residential proxy networks allow bot operators to randomize the submission of data from their request bot and fly beneath the radar of detection methods that rely on known bad IP addresses and user agents.

6

OUTCOME

Adversaries monetize attacks and/or extract valuable data before most businesses are aware their bot mitigation solution has been bypassed.