

Inline web protection on CSIC 2010

61,000 HTTP requests to a single application, 25,065 of them attacks. The engine is induced on normal requests only and never sees an attack during learning.

91.69% of attacking sources blocked by their third request, with 0 of 6,000 genuine sources wrongly blocked

RESULTS

REQUESTS SEEN FROM A SOURCE	ATTACKING SOURCES BLOCKED	GENUINE WRONGLY BLOCKED
2 requests	73.38%	0 of 9,000
3 requests	91.69%	0 of 6,000
5 requests	99.68%	0 of 3,600
10 requests	100% (1,253 sources)	0 of 1,800

Bars run 0 to 100%.

HOW IT WAS MEASURED

Unit	The source – a run of consecutive requests from one client
Atom	Raw wire bytes of the request. No parsing, no field extraction, no signature
Learning	Normal requests only. No attack is labelled, and none is seen during induction
Level	Set on normal traffic alone, at one wrongly blocked genuine run in a thousand

CONDITIONS

Single-request ranking AUC	0.9221
Decision cost per request	~0.1 ms on one 2020 laptop core, ~10,000 requests/s
Memory	24 MB
Pre-trained artifact	None. Everything is learnt on your own box

THE PASS MARK, FIXED BEFORE THE RUN

Set before the run: ranking AUC at or above 0.85, and “could block inline” meaning 90% or more of attacks blocked at 1% or fewer wrongly blocked.

LIMITS OF THIS RESULT

CSIC 2010 carries no client addresses, so a source is a run of consecutive requests of one class; a real attacker may interleave normal requests. It is one application, generated traffic, from 2010. A pass-through device needs TLS terminated ahead of it. The byte-level method is protocol-agnostic in principle and is proven here on HTTP.

REPRODUCING IT

The dataset is public and the harness is the same binary a customer installs. Ask us for the harness and the exact dataset revision and run it on your own hardware. The run behind this sheet is recorded in `validate/csic2010/CSIC_FINDINGS.md`, `validate/BENCHMARKING.md`, including the pass mark, every figure, and the results that did not favour us.