
# Managed Web Scraping Defense

Whitepaper

Matt Taylor SVP Business Development

## Introduction

Web scraping is a technique for extracting information from web sites that often uses automated programs, or bots (short for web robots), opening many sessions, or initiating many transactions while data is harvested. As web scraping methods evolve and become more sophisticated a variety of controls are recommended to provide holistic protection against web scraping. These start with generic traffic sanitation controls. On top of that layered rate monitoring, signature based detection, header analysis and dynamic detection methods; such as client challenges, source Geo/IP or other suspicious user criteria monitoring is recommended.

This document outlines the controls RedShield can implement and manage for you to protect your websites from such attacks.

## High-Performance Security is Not a Simple Task

Cybersecurity is more than just a technical problem – it's a people problem. It's common knowledge that bad guys have organized and business has not. The ease and convenience of programmed machines and today's as-a-service model is not enough to stay ahead of the human ingenuity behind malicious intent.

When compared to do-it-yourself or traditional cloud tool software-as-a-service, RedShield offers far more protection in web application security. We address the problem of how to use the tools effectively in a model called "with-a-Service."

## At Your Service. Not as-a-Service.

RedShield experts running a suite of integrated industry leading tools within a mature vulnerability management operational process framework is what separates our "with-a-service" model from the traditional "as-a-service" model. Integrating with your change management, incident management and operational reporting procedures, RedShield's tools and teams are an extension of your cybersecurity operation.

Risk acceptance of any vulnerability is becoming increasingly dangerous. You need tools and a cybersecurity team that is as skilled, organized and persistent as your adversary. RedShield's turn-key solution gets your enterprise web application security as skilled, organized and diligent as the bad guys.

## RedShield Managed Generic Traffic Sanitization

As part of the standard RedShield deployment a range of managed generic controls are deployed to enforce fair use policies and stop malicious connections.

These controls will immediately stop a range of current scraping bots and make scraping more difficult to implement against your website.

| Shield Types | Managed Shield Type Description |
|---|---|
| Hardening | These shields improve the security posture of the web application and often protect against attacks that would not be visible to the server side. Examples include adding HTTP security headers such as X-Frame_Options, adding cookie flags (Secure and HTTPOnly). These add security controls that harden the application rather than shielding a specific exploit. These are typically added to every response from the server and therefore alerting and reporting on their use is not meaningful. |
| Transformation | Some shields transform every request and/or response to prevent, for example, information leakage or the use of weak passwords. Alerting and reporting on their use is not meaningful. |
| Correlation | These shields mitigate a specific vulnerability that exists in the web application. Malicious requests and attack traffic can be correlated with vulnerabilities and shields and alerted and reported on in transparent and blocking mode. Examples of these include SQL Injection, Cross Site Scripting, Direct Object Reference, Authorisation Bypass etc |
| Anomalous Traffic | Traffic that causes anomalous traffic spikes can be alerted and reported on. These spikes can be from specific IPs, against specific URLS (from distributed client IPs), against a whole site and against all a client's sites. Thresholds can be configured and tuned for IPs, URLs and sites before alerting or more proactive shields are triggered. Proactive shields can include rate limiting, client side javascript challenges , CAPTCHAs etc. These shields are normally triggered by aggressive scanning or denial of service attacks. |
| Anomalous HTTP Response Codes | Alerting and reporting on anomalous numbers or patterns of HTTP error codes returned by the server is available. HTTP error response codes are not blocked by default. Often these are early indicators of attack or reconnaissance. Error codes reported on by default are 500 and 403. This is configurable for the client. |
| Scanner Traffic | Scanner traffic is reported on monthly where it is easily identified from one source IP, by User Agent or traffic characteristics (frequency, specific packet probes etc). |
| Attack Probes | Reporting and alerting on all attack probes whether relevant to vulnerabilities or not is part of the RedShield service. All malicious traffic such as directory traversal, multiple encoding bypass techniques, null byte, injection attacks, authentication |

| | |
|---|---|
| | bypass techniques, brute force are alerted and reported on in transparent or blocking mode. |
| System Abuse | Attacks such as card washing are also able to be detected and reported on. Typically this class of attack is system specific. RedShield conducts ongoing research into novel abuses of applications and is always keen to engage with clients on research to understand how they may be vulnerable to more subtle business abuses via their application because of complex system interactions, architectural or design flaws. (As an example see Cloudseeding). RedShield can then develop defense mechanisms for these weaknesses. |

## Generic Vulnerability Managed Shield Descriptions

| Vulnerability Type | Vulnerability Description | Shield Type |
|---|---|---|
| Volumetric DDoS | Traffic Flood detection | Anomalous Traffic |
| | High capacity attack absorption | Anomalous Traffic |
| | DNS based introduction of additional capacity | Anomalous Traffic |
| | IP based attack mitigation | Anomalous Traffic |
| | BGP introduction of mega scrubbing centers | Anomalous Traffic |
| Asymmetric DDoS | Half open L4 | Anomalous Traffic |
| | L4 Connection limits per IP | Anomalous Traffic |
| | L4 Connection limits per URL | Anomalous Traffic |
| | TLS renegotiation limits | Anomalous Traffic |
| | Slow HTTP session detection | Anomalous Traffic |
| RFC Violations | General HTTP RFC Compliance Checks | Anomalous Traffic |
| | Cookie not RFC-compliant | Anomalous Traffic |
| Evasion technique detected | Directory traversals | Attack Probes |
| | %u decoding | Attack Probes |
| | IIS backslashes | Attack Probes |
| | IIS Unicode codepoints | Attack Probes |
| | Bare byte decoding | Attack Probes |
| | Apache whitespace | Attack Probes |
| | Bad unescape | Attack Probes |
| Vulnerable Webserver, HTTP protocol compliance issues | Header name with no header value | System Abuse |
| | Several Content-Length headers | System Abuse |
| | Chunked request with Content-Length header | System Abuse |
| | Bad multipart parameters parsing | System Abuse |
| | No Host header in HTTP/1.1 request | System Abuse |

| | | |
|---|---|---|
| | CRLF characters before request start | System Abuse |
| | Content length should be a positive number | System Abuse |
| | Bad HTTP version | System Abuse |
| | Null in request | System Abuse |
| | Check maximum number of headers | System Abuse |
| | Bad host header value | System Abuse |
| | Check maximum number of parameters | System Abuse |
| | Mandatory HTTP header is missing | System Abuse |
| Input Violations | Brute Force: Maximum login attempts are exceeded | Attack Probes |
| | Illegal method | Anomalous HTTP Response Codes |
| | Illegal redirection attempt | System Abuse |
| | Request length exceeds defined buffer size | System Abuse |
| | Failed to convert character | System Abuse |
| | Illegal static parameter value | System Abuse |
| Additional Blocking Elements | Illegal URL | System Abuse |
| | Modified TS cookie | System Abuse |
| Negative Security Signatures | Abuse of Functionality | System Abuse |
| | Authentication/Authorisation Attacks | Attack Probes |
| | Buffer Overflow | Attack Probes |
| | Command Execution | Attack Probes |
| | Cross Site Scripting | Correlation |
| | Denial of Service | Attack Probes |
| | Detection Evasion | Attack Probes |
| | Directory Indexing | System Abuse |
| | HTTP Response Splitting | System Abuse |
| | Information Leakage | Attack Probes |
| | LDAP Injection | Correlation |
| | Other General Application Component Attacks | Correlation |
| | Non-browser Client | Attack Probes |
| | Path Traversal | Attack Probes |
| | Predictable Resource Location | Hardening |
| | Remote File Include | Correlation |
| | SQL Injection | Correlation |
| | Server Side Code Injection | Correlation |
| | Trojan/Backdoor/Spyware | Correlation |
| | Vulnerability Scanner | Scanner Traffic |
| | Xpath Injection | Correlation |

# RedShield Managed Web Scraping Specific Controls

### Source IP Address Restrictions

RedShield recommends blocking source IP addresses that match Tor nodes and public cloud server centers such as AWS.

### Referer Header Enforcement

RedShield recommends implementing refer header enforcement. If Cross-Origin Resource Sharing is required then a bespoke configuration is required.

### Signature Based Bot Detection

Bot signatures carefully identify bots and have a low rate of producing false positive results. The signatures identify the type of bot for classification and investigative purposes, and can distinguish between benign and malicious bots.

Benign bots can be useful for providing Internet services such as search engine bots, index crawlers, site monitors, and those used to establish availability and response time. Some environments may not want to block benign bot traffic.

RedShield applies a large range of Bot signatures to identify web robots by looking for specific patterns in the headers of incoming HTTP requests.

RedShield receives feeds from industry sources, plus is able to create custom signatures.

### Dynamic Bot Detection

Dynamic Bot detection adds additional defense by injecting challenges into the traffic flow that bots are unable to respond to. Dynamic detection can be configured to occur on the following conditions:

1) for all clients accessing the site
2) for particular source Geo IPs
3) on suspicious user criteria including:
    a) HTTP requests per second
        - Total requests per second;
        - Rate of requests per second increase
    b) Server side latency:
        - Total latency;
        - rate of latency increase
    c) Session opening anomalies:

- too many sessions are opened from an IP address;
- the rate of increase in sessions exceeds a threshold from an IP address;
- when the number of inconsistencies or session resets exceeds the configured threshold within the defined time period;
- requests that do not include the expected cookies

d) Session transactions anomalies:
- sessions that request too much traffic, compared to the average amount observed in the web application. This is based on counting the transactions per session and comparing that to the average amount observed in the web application.

e) Rapid surfing:
- If a specific number of different URLs are accessed in a specific amount of time;
- If one page is refreshed a specific number of times in a specific amount of time.

4) on anomalous site activity including:
   a) HTTP requests per second
      - Total requests per second;
      - Rate of requests per second increase
   b) Server side latency:
      - Total latency;
      - rate of latency increase

When triggered RedShield does not pass the next HTTP request from the client to the webserver, but instead replies with an HTTP response that includes a JavaScript computational challenge that the client must execute. A legitimate browser will automatically return an HTTP post request to RedShield containing a new TS cookie. A script will not.

With the TS cookie correctly returned, RedShield will wait for predefined number of HTTP requests (the grace period) before injecting the Client-Side Human User Indicator (CSHUI) javascript (patent/US9420049).

This CSHUI performs the following functions and produces a CSHUI cookie that accompanies further HTTP requests:

- Mouse and Keyboard activity detection:
  - the coordinates of the mouse location, and relate it to the amount of time it takes for the mouse to move is measured
  - the time between the keydown and keyup keyboard actions.
  - the variance and standard deviation of every keyboard event is calculated
- Sequence enforcement
  - the sequence of events that the browser triggers, and detect irregular sequences.
- Monitoring of honeypot hidden links
  - a bot will navigate to these whilst a human operator will not

## RedShield Managed Custom Controls

In additional to the specific methods mentioned above, RedShield has developed an innovative microservices architecture that uses an application's existing, external messaging framework as a lightweight message bus to seamlessly integrate software objects to transform message flows and application logic. This architecture allows RedShield to develop new and increasingly sophisticated custom shields as the Web Scraping Arms Race continues.

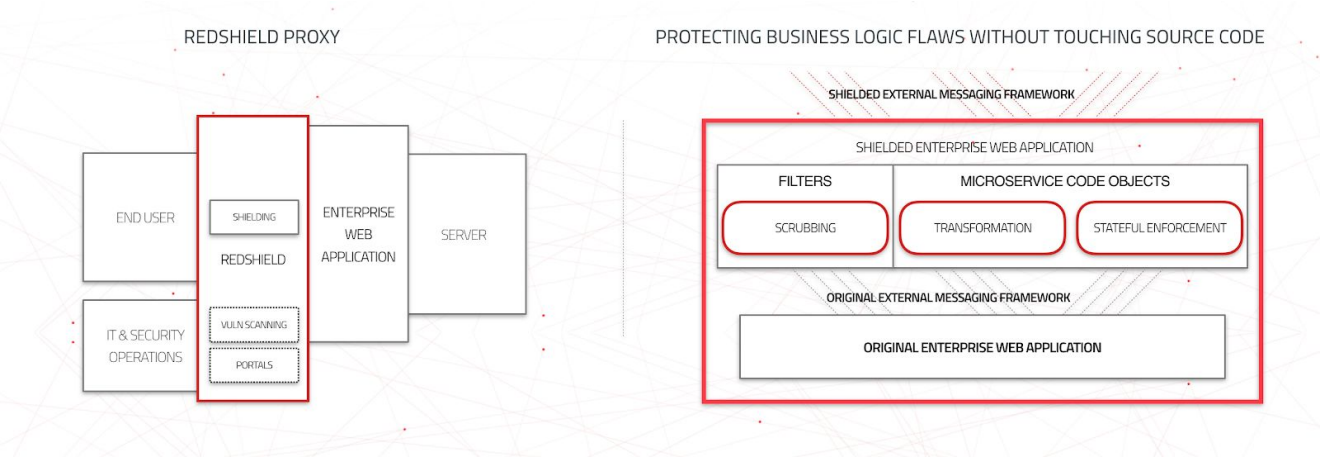


Figure 1: RedShield Shielding Architecture

## Managed Enforcement Methods

If suspicious traffic is identified, either dynamically or otherwise a number of managed enforcement methods are available:

- Block for a period or indefinitely
  - During an attack offending IP addresses can be blocked and subsequently released as many attacks are temporarily hidden behind proxies.
- Rate limit
  - Often severely limiting the bandwidth of an attacking IP address causes less disruption to services as well as makes the site less attractive to attackers.
- CAPTCHA
  - CAPTCHA options are highly effective enforcement methods, but do require user interaction.
- FunCaptcha (https://www.funcaptcha.com/)
  - A range of automated and sweatshop OCR and ICR brute force techniques can be implemented against classic CAPTCHA technology;
  - FunCaptcha uses the mathematical shadows in machine learning to provide the most advanced CAPTCHA option that is effective against even the most intelligent bots and brute force techniques.

## Managed Enforcement Scope

The selected enforcement method can be applied and managed by RedShield to either source or destination traffic in the following ways:

- Source
  - IP based - These controls can be configured against specific IP addresses that are defined by known blacklists as well as individual addresses found through run-time monitoring;
  - Geo IP based - Controls can also be defined by global geography of source address.
- Destination
  - URL based - The controls can be configured very granularly against specific site URL's;
  - Site wide - The controls can be applied across the entire site.

## Ongoing Service Management

RedShield solves the people and process problem by expertly deploying, maintaining systems, 24/365 monitoring, expert response and tuning, continual auditing and reporting.

Additionally, RedShield security researchers stay abreast of vulnerability advisories and new scraping techniques, whilst RedShield developers are on hand to develop custom controls when required.

Get ahead and stay ahead with RedShield's

# Managed Web Scraping Defense