OWASP
Open Web Application
Security Project

AUTOMATED THREATS

Web applications

The OWASP Automated Threats to Web Applications Project is creating information and other resources for architects, developers, testers, and others to help web application owners defend against automated threats

Issue

There is a significant body of knowledge about application vulnerability types, and some general consensus about identification and naming. But issues relating to the misuse of valid functionality, which may be related to design flaws rather than implementation bugs, are less well defined. Yet these problems are seen day-in, day-out, by web application owners.

Excessive abuse of functionality is commonly mistakenly reported as application denial-of-service (DoS) attacks such as HTTP-flooding or application resource exhaustion, when in fact the DoS is a side-effect. Some examples are blog & comment spam, fake account creation, password cracking, web scraping, etc. Most of these problems seen regularly by web application owners are not listed in any OWASP Top Ten or other top issue list or dictionary.

This has contributed to inadequate visibility, and an inconsistency in naming such threats, with a consequent lack of clarity in attempts to address the issues.

OWASP Project

The OWASP Automated Threats to Web Applications Project has completed a review of reports, academic and other papers, news stories and vulnerability taxonomies/listings to identify, name and classify these attacks – threat events to web applications that are undertaken using automated actions. The initial aim is to produce an ontology providing a common language for developers, architects, operators, business owners, security engineers, purchasers and suppliers/vendors, to facilitate clear communication and help tackling the issues.

The project also intends to identify symptoms, mitigations and controls in this problem area. Like all OWASP outputs, everything is free and published using an open source license.

Use Cases

The ontology and supporting materials are expected to be useful for:

- Defining application security requirements
- Sharing intelligence within a sector
- Exchanging threat data between CERTs
- Labelling penetration test findings
- Documenting service acquisition needs
- Characterising vendor services

These are documented further on the project site.

Discussion Briefing

Overleaf we have summarised the work to date. This draft is the outcome of reading 150 information sources, analysing and assessing the information from these sources, and ongoing discussions with other people.

The project would like to hear your thoughts about the threats and their names, particularly if you believe it to be incomplete. We also want to receive real-world experience on the prevalence of such threats, especially if you are responsible for the ongoing operation of web applications.

> “Can you please contribute your experience by email or using the mailing list?
> Feel free to speak to me about this OWASP project in Amsterdam during the AppSec EU 2015 conference in May.”

Colin Watson
Project leader
colin.watson@owasp.org

https://www.owasp.org/index.php/OWASP_Automated_Threats_to_Web_Applications

OWASP Automated Threats to Web Applications

Discussion briefing

Names and summary text extracted from early draft of the ontology (8th May 2015)

Which of the following threats do you recognise, and which affect your web applications?

Many are sector-specific; some are functionality-specific. The magnitude of the business risk from each item is not equal. Additionally, the names and summaries are not finalised - please provide suggestions and comments by email or using the project's mailing list provided at the foot of this page.

Credential Stuffing

Mass log in attempts used to verify the validity of stolen username/password pairs.

Carding

Multiple small payment authorisations used to verify the validity of bulk stolen payment card data.

Scraping

Collect application content and/or data for use or republication elsewhere.

Sniping

Last minute bid or offer, for goods or services.

Credential Cracking

Identify valid log in credentials by trying different values for usernames and/or passwords.

Card Cracking

Identify missing payment card details by trying different values for expiry date and security code.

Spamming

Information addition that appears in content, databases or email messages.

Token Code Cracking

Mass enumeration of coupon numbers, voucher codes, discount tokens, etc.

CAPTCHA Bypass

Solve anti-automation tests.

Cashing Out

Buy goods or obtain cash from stolen payment cards.

Click and Impression Fraud

False clicks and fraudulent display of web-placed advertisements.

Skewing

Repeated link clicks, page requests or form submissions intended to alter some metric.

Fake Account Creation

Create multiple accounts for subsequent misuse.

Web Application Denial of Service

Target resources of the application and database servers, or individual user accounts, to acheive denial of service (DoS).

Something Confusing or Missing?

What are your thoughts and suggestions?

Scalping

Obtain limited-availability goods or services by unfair methods.

Fingerprinting

Illicit information about the supporting web, application and database server and framework types and versions.

Footprinting

Probe and explore application to identify constituents and properties of the application.

Vulnerability Scanning

Crawl and fuzz application to identify weaknesses and possible vulnerabilities.

User Bot

Perform tedious or time-consuming actions on behalf of a person.

Email colin.watson@owasp.org   Mailing list https://lists.owasp.org/mailman/listinfo/automated_threats_to_web_applications