TECHNICAL SPECIFICATIONS

The information in this document applies to **version 6.9** of Syhunt Hybrid Platinum.

This document presents the technical specifications of Syhunt Hybrid, an augmented HAST (*Hybrid Application Security Testing*), DAST (*Dynamic Application Security Testing*), SAST (*Static Application Security Testing*) and MAST (*Mobile Application Security Testing*) tool of high accuracy, depth and coverage that identifies vulnerabilities and weaknesses in applications.

A. LICENSE RIGHTS

- Syhunt Hybrid software licenses usually have a **term of validity of 12 months**.
- The licenses include version update and technical support rights, in addition to providing technical documentation in English in PDF and HTML formats.
- The licenses allow an unlimited number of scans against an unlimited number of Assets per year, which must be launched from a single or more Customer machines.

B. ARCHITECTURE

Syhunt Hybrid is an application security assesment software tool, which is hybrid, on-premise, modular, asynchronous and multi-process:

- It is hybrid, that is, it combines different analysis techniques, such as DAST, OAST, SAST, MAST and FAST (detailed below).
- It is on-premise, that is, the solution is deployed on the Customer's premises, so that the analyzed source code and analysis results do not leave its internal network.
- It is modular with a single management console that aggregates the configuration management features of its modules and analysis results presentation features.
- It is asynchronous, that is, the solution works with several scans simultaneously, without the need of waiting for the end of each analysis.
- It is multi-tabbed and multi-process, that is, each browser tab or scan is a different process in the operating system.
- It comes with a command line interface (CLI) that allows the launching of dynamic and source code scans and other tasks.

SUPPORTED TYPES OF ANALYSIS

Syhunt Hybrid performs the following types of analysis:

SAST	Static analysis of the security of a web application's source code
MAST	Static analysis of the security of a mobile application's source code (Android & iOS) and Android APK file
DAST	Dynamic analysis of the security of a web application security with deep crawling and parameter injection
OAST	Augmented dynamic analysis of the security of a web application security with out-of-band (OOB) techniques
HAST	Hybrid-Augmented analysis of the security of web applications, on which the results of the static analysis are automatically used to enhance its augmented dynamic analysis.
FAST	Forensic analysis of the security of a web application through the analysis of web server log files

VULNERABILITY DATABASE

Syhunt currently comes with an internal vulnerability database that contemplates the sets of publicly available vulnerabilities and regulations listed in the table below. The tool's vulnerability database is updated periodically, ensuring that the solution is always up-to-date with new vulnerabilities published by international vulnerability databases.

CWE/SANS Top 25 2019	Most Dangerous Software Errors: 2019 version or later
OWASP Top 10	Top 10 Web Application Security Risks: 2017 version or later
OWASP Mobile Top 10	Mobile Top 10 Risks: 2016 version or later (PLUS)
OWASP PHP Top 5	
CWE	Common Weakness Enumeration
CVE	Common Vulnerabilities and Exposures

WASC	The Web Application Security Consortium Threat Classification
WAVSEP	Web Application Vulnerability Scanner Evaluation Project
NIST SAMATE	Software Assurance Metrics And Tool Evaluation Project
PCI DSS	Payment Card Industry Data Security Standard: version 3.2, 3.2.1 or later

INTEGRATIONS

Syhunt Hybrid comes with features for integration with the following systems:

Ticket Management	JIRA GitHub GitLab
Version Control	Public and private GIT repositories Azure Repos (using GIT) GitHub and GitLab Branches
APIs & Scripting	Lua 5.1 API Web API (REST) PowerShell JSON and XML
Application Vulnerability Management	BIG-IP Application Security Manager (ASM) Imperva SecureSphere ModSecurity CRS OGASEC WAF
Development Process Control	GitLab CI and Security Dashboard Jenkins Pipeline
Web Browsers	Syhunt Sandcat (Built-In) Google Chrome Mozilla Firefox
Email	SMTP

The examples below show how to perform the analysis of a GIT repository through the various available integration options:

```
-- from the command prompt
scancode git://sub.domain.com/repo.git
scancode https://github.com/user/repo.git -rb:master

-- from GitLab CI Script
- Start-CodeScan -pfcond 'fail-if:risk=mediumup' -output 'report.pdf' -outputex 'gl-sast-report.json'

-- from Jenkins pipeline script
syhunt.scanCode([target: 'https://github.com/someuser/somerepo.git', branch: 'master', pfcond: 'fail-if:risk=mediumup'])

-- from PowerShell
$MyScan = @{
  target = 'https://github.com/someuser/somerepo.git';
  branch = 'master';
  pfcond = 'fail-if:risk=mediumup';
  output = 'report.pdf'
}
- Start-CodeScan @MyScan

-- from Lua script
code:scanurl('https://github.com/someuser/somerepo.git', 'master')

-- from Web API (REST, raw JSON)
-- POST /syhunt/launch.lua
{
  sourcetarget: "https://github.com/someuser/somerepo.git",
  sourcebranch: "master",
  apikey: "YOUR_API_KEY"
}
```

SUPPORTED LANGUAGES & ENVIRONMENTS

Syhunt Hybrid identifies vulnerabilities in applications with the following languages, environments and frameworks:

Language	Environment / Framework
C#	ASP.Net

Java	JEE JSP Android Spring Framework
JavaScript	Client-Side Server-Side Node.js (Barebone, Express.js & Koa.js) Angular (version 2 or higher) AngularJS JScript (ASP Classic) ElectronJS (Desktop)
Lua	ngx_lua mod_lua CGILua Lua Pages
Objective-C, C & C++	iOS
Perl	
PHP	
Python	CGI mod_python PSP WSGI Django
Ruby	Rails ERB mod_ruby
Swift	iOS
TypeScript	Angular
VB	VB.Net (ASP.Net) VBScript (ASP Classic)

C. SOURCE CODE ANALYSIS

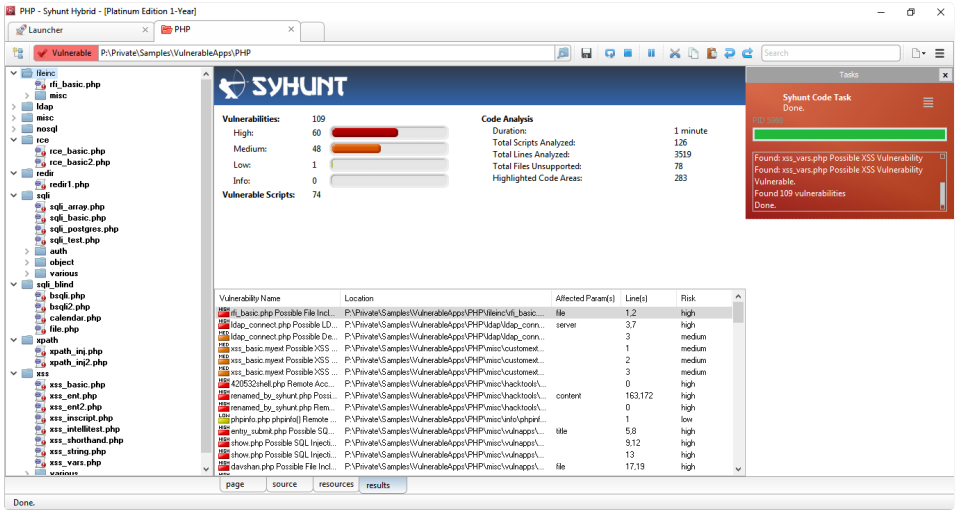
Module Name	Syhunt Code; Syhunt Mobile
Description	Analyzes the source code of web and mobile applications for security vulnerabilities
Type of Analysis	SAST (Static Application Security Testing) MAST (Mobile Application Security Testing)
Target Applications	Web Applications (including *Web Services*) Mobile Applications (Android & iOS) Desktop Applications
Target Languages & Platforms	APK (Android) ASP.Net & ASP Classic HTML Java (including Android) JavaScript Lua Objective-C, C & C++ (iOS) Perl PHP Python Ruby Swift (iOS) TypeScript
Vulnerabilities Detected	1300+
Vulnerability Categories Covered	40+
GIT Integration	Yes

- It comes with known application vulnerabilities in all programming languages and environments supported by the module.
- Supports code embedded in HTML and print shorthands.
- Identifies client-side and server-side vulnerabilities.
- Identifies the use of outdated vulnerable scripts, local or remote, such as vulnerable versions of

AngularJS, jQuery, fullPage, Bootstrap and momentjs.

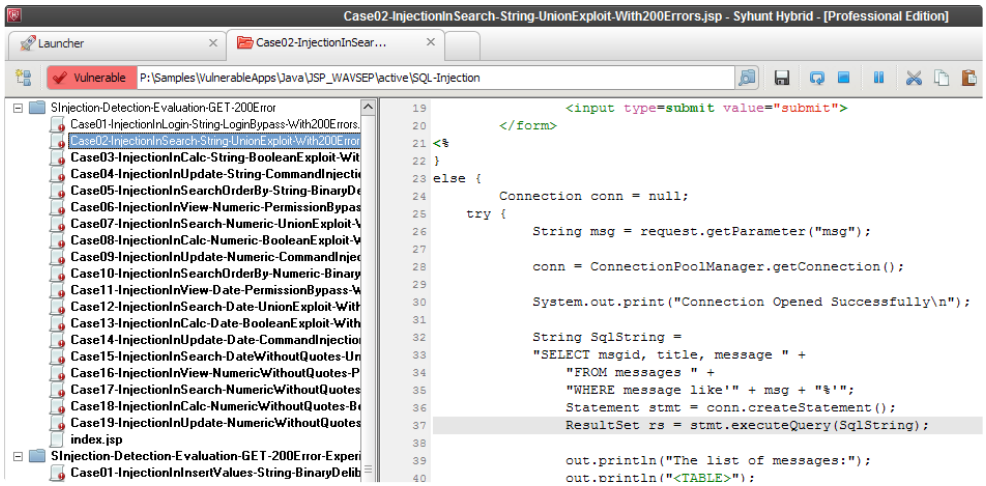
- Performs vulnerability analysis on complete source codes, code snippets and configuration files.
- Supports multi-auditing of several programming languages in the same scan session.
- Identifies and allows to navigate through key areas of code, such as specific HTML markers, JavaScript, XHR requests, entry points and interesting keywords.
- Identifies vulnerabilities in poorly designed code, that is, programming errors that expose the system to attack risks.
- Analyzes configuration files to assess security threats and identify appropriate countermeasures at the stage of server, environment or application configuration.
- Recognizes cases of input filtering and validation, providing accurate and false-positive free results.

SYHUNT CODE USER INTERFACE



The Syhunt Code's user interface allows real-time viewing of the status of running scans, including:

1. Tree of analyzed and vulnerable files:
2. Scan duration;
3. Total scripts analyzed and vulnerable;
4. Total vulnerabilities found and by severity level;
5. Total lines analyzed;
6. List of vulnerabilities found.



In addition, it identifies and displays points in the source code where it is possible with just one fix, to remediate vulnerabilities found in the application.

C2 MOBILE ANALYSIS

Syhunt Code performs MAST (Mobile Application Security Testing) in Android and iOS applications and Android APK files for pressing OWASP Mobile Top 10 and CWE/SANS Top 25 risks, such as:

Improper Platform Usage
Insecure Data Storage
Insecure Communication
Insecure Authentication
Insufficient Cryptography
Insecure Authorization
Client Code Quality
Code Tampering
Reverse Engineering
Extraneous Functionality

D. DYNAMIC ANALYSIS

Module Name	Syhunt Dynamic Augmented
Description	Detects security vulnerabilities in dynamic web applications and web servers
Type of Analysis	DAST (Dynamic Application Security Testing) OAST (Out-of-Band Application Security Testing)
Target Applications	Web Applications & Web Servers
Target Languages & Platforms	ASP.Net & ASP Classic HTML & JavaScript Java Lua Perl PHP Python Ruby
Optimized Analysis for HTTP Servers	Apache Apache Tomcat Microsoft IIS Nginx
Vulnerabilities Detected	7000+
Vulnerability Categories Covered	68+
Injection Checks	570+
Supported Protocols	HTTP version 1.0 & 1.1 HTTPS (SSL 2/SSL 3/TLS 1) IPv6 Proxy (HTTP, Socks 4 & Socks 5)
Supported HTTP Methods	Keep-Alive GZIP Compression Authentication (Basic & Form-Based)

- It comes with known web application vulnerabilities in the programming languages and environments supported by the module.

- Performs augmented dynamic analysis, a combination of DAST and OAST test methods.
- Performs hybrid-augmented analysis, a combination of DAST, OAST and SAST test methods.
- Identifies the use of outdated vulnerable scripts, server software and other components.
- Identifies client-side and server-side vulnerabilities.
- Identifies in-band, inferential and out-of-band (OOB) vulnerabilities.
- Identifies known, vulnerable applications in all programming languages supported by the solution, including ColdFusion, Flash & Server Side Includes (SSI).
- Automatically performs structural, login form and HTTP authentication brute-force attacks.
- It comes with invasive and non-invasive scanning capability.
- Allows to turn off denial of service (DoS) tests that may affect the availability of the web application.
- Allows to configure the number of retries and timeout used to connect to the web application.
- Supports analysis of applications built on top of content management systems, such as Drupal, Joomla, WII and WordPress.
- Supports analysis of SPAs (Single Page Applications).
- Identifies the technologies used in the application and optimizes the scanning time based on the detected technologies.
- Identifies the hidden versions of server software and components (Hunter-Sense™), such as Apache, Nginx, PHP, mod_ssl, OpenSSL and Phusion Passenger.
- Performs data injection and manipulates parameters in the target application in URLs and forms (both GET and POST).
- Performs injection mutations, in order to cover all programming languages and target platforms supported by the solution.
- Identifies vulnerabilities such as SQL injection,NoSQL injection, OS Command Injection, exposure and code injection through techniques such as inferential (response time), in-band (error message or print), out-of-band (OOB) and Passive Analysis.

BROWSING AND CRAWLING CAPABILITIES

Syhunt Hybrid maps the structure, including all links and entry points of the target application, while emulating a modern browser with support for the following standards and features:

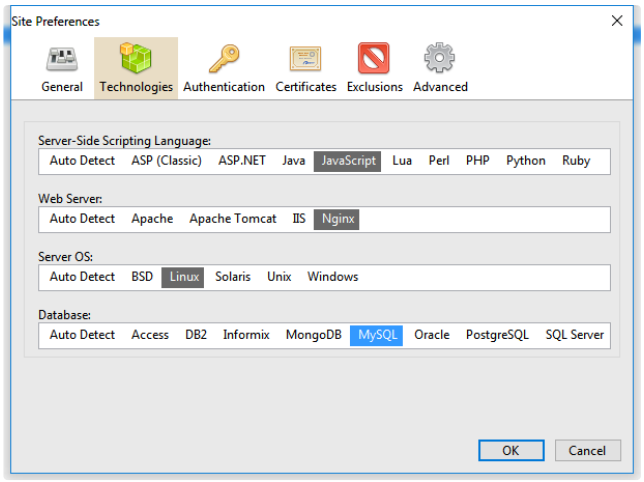
Latest Web Standards	HTML 5 CSS 3
Intelligent HTML Analysis	Relative Paths Standard HTML Non-standard or deformed HTML Form Recognition

Browser Emulation	Google Chrome Mozilla Firefox Microsoft Edge Referer Submission
JavaScript Support	JavaScript analysis and execution DOM Emulation Browser Behavior XHR Calls Support for external JavaScript files
User Interaction Simulation	Key press Mouse clicks Autofill forms Automatic login
Cookies Support	Cookie management Session management
Follow Redirects	HTTP Meta refresh JavaScript
Process Isolation	Each browser or scan tab is a different process in the system
Robots.txt File Analysis	If the file is available

Syhunt intelligently handles large, complex websites with dynamic content generation and includes mechanisms to prevent loop situations during application mapping. In addition, it allows to limit the depth of scans, including:

- Maximum number of links per server and links per page
- Maximum URL size in bytes and HTTP response in kilobytes

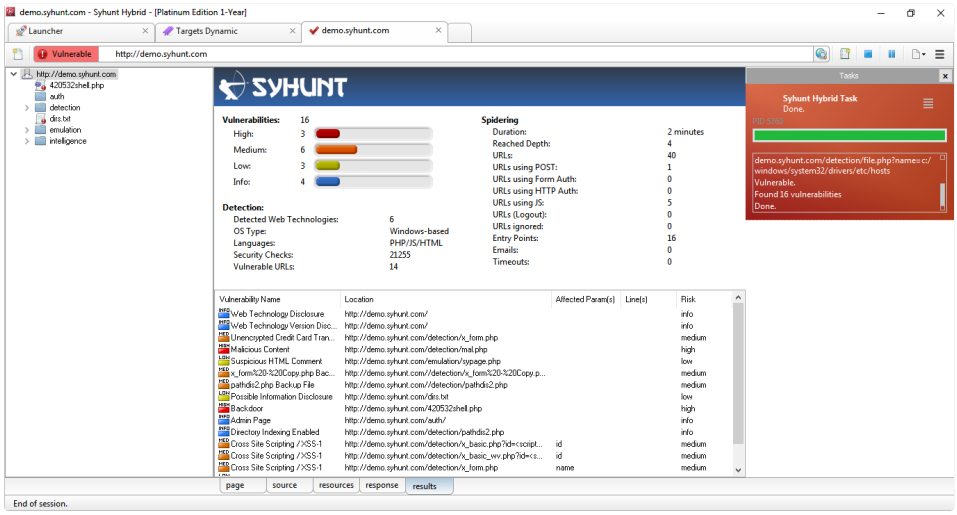
SITE PREFERENCES



Syhunt Dynamic allows the creation of scan profiles for Assets, which specify:

- Technologies used by the application in order to optimize the scanning time, such as the server-side language, the web server, the operating system and the database of the target application;
- Start URLs;
- SSL certificates and credentials for authentication allowing to perform the analysis logged in the target system (basic and web form-based);
- Exclusion of objects from scanning, such as specific paths, forms and vulnerabilities;
- Depth and layer limitation
- Signatures for detecting custom 404 error pages;
- Manual configuration of cookies and session token.

SYHUNT DYNAMIC USER INTERFACE

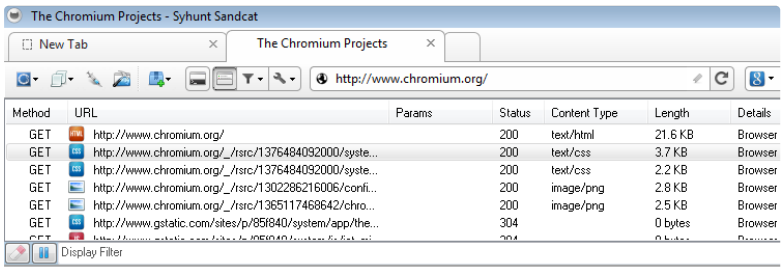


The Syhunt Dynamic's user interface allows real-time viewing of the status of running scans, including:

- Site tree of found and vulnerable paths;
- Scan duration;
- Total vulnerable URLs;

- Total URLs using POST, authentication and JavaScript;
- Total entry points;
- Total timeouts;
- Total vulnerabilities found and by severity level;
- Total security checks performed;
- Detected web technologies;
- Target operating system type;
- Target programming languages;
- Depth Reached;
- List of vulnerabilities found.

BUILT-IN WEB BROWSER



Syhunt Dynamic incorporates its own modern browser, known as Syhunt Sandcat, with vulnerability analysis extensions and features for manual testing and support for automated testing, such as:

- Manual login to web applications
- Capture URLs in manual navigation mode
- Live HTTP headers
- Preview capability for the most common web file formats, such as CSS, Flash, HTML, common image formats (bmp, gif, ico, jpg, png and svg), JavaScript, JSON, text and XML
- Automatic JavaScript deobfuscation
- Request replay capability
- Fuzzer
- Script executor
- HTTP and XHR editors and request loader.

D2. AUGMENTED ANALYSIS

Module Name	Syhunt Signal
Description	Detects out-of-band security vulnerabilities in dynamic web applications

Type of Analysis	OAST (Out-of-Band Application Security Testing)
Target Applications	Web Applications

Syhunt Dynamic Augmented integrates with the online Syhunt Signal service to perform automated OAST (Out-of-Band Application Security Testing):

- Allows the scanner to detect otherwise invisible, high-risk out-of-band (OOB) vulnerability variants
- Listens to forced requests coming in from a vulnerable target web server over the course of a scan and signals back to the scanner.
- Automatically correlates the received alerts with attack requests it launched.
- Adds the identified OOB vulnerabilities to its report and user interface.
- Returns zero false positives.
- Automatically exfiltrates data from a vulnerable target, which gets added to the scan results, using different commands and techniques (environment and OS-specific).
- Needs an active Internet connection to work.

The following out-of-band vulnerability types are detected by Syhunt Signal:

Command Execution
Remote File Inclusion (RFI)
Server-Side Request Forgery (SSRF)
SQL Injection
XML External Entity (XXE) Injection

D3. HYBRID-AUGMENTED ANALYSIS

Syhunt Hybrid combines the results of Syhunt Dynamic, Syhunt Signal and Syhunt Code to perform Hybrid-Augmented Analysis, AKA augmented HAST (Hybrid Application Security Testing), on which the results of the static analysis are automatically used to enhance its augmented dynamic analysis:

- Combines SAST, DAST & OAST test methods.
- Performs hybrid client-side JavaScript code analysis (SAST-in-DAST).
- Dynamically detects and confirms vulnerabilities by simulating inferential, in-band and out-of-band (OOB) attacks and by using entry point and other information acquired through source code analysis.

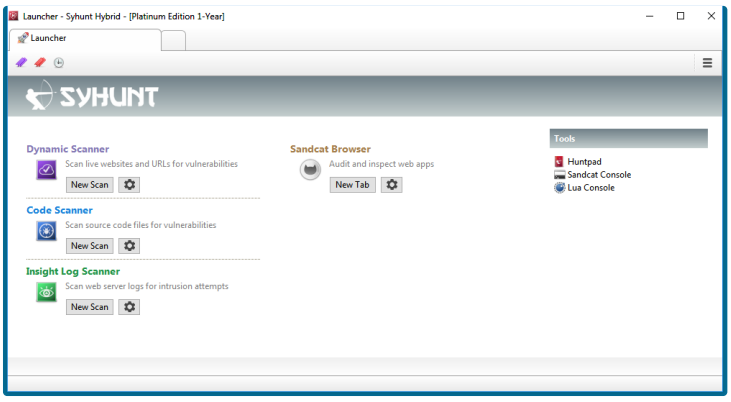
D4. FORENSIC LOG ANALYSIS

Module Name	Syhunt Dynamic Insight
Description	Detects security attacks in dynamic web applications
Type of Analysis	FAST (Forensic Application Security Testing)
Target Applications	Web Applications

Syhunt Dynamic Insight performs on-demand security heuristic analysis of web server log files to detect attacks, including:

- Identification of origin (IP address), country, type and methods used to attempt to compromise web applications;
- Reconstruction of the attack session, accurately differentiating legitimate traffic from malicious traffic, in addition to differentiating automated attacks from manual attacks;
- Intrusion detection, with backdoor installation, and used intrusion tools.
- Detection of attempts to exploit OWASP Top 10 vulnerabilities and the use of defense evasion techniques.
- Support for log files generated by Apache, Microsoft IIS and Nginx web servers, with automatic format detection.

E. MANAGEMENT CONSOLE



The management console aggregates the features for managing module settings and presenting the results of SAST, DAST and FAST scans:

- Allows to access stored results of scans performed or in progress.
- Does not require prior knowledge of information security and secure coding to use the console.

- Requires little or no user intervention before and during the progress of scans.
- Provides a graphical view that indicates the progress of the analysis and the risk level of the analysis being performed.
- Displays software license information, showing the license type, license expiration date, and supported programming languages.
- Allows the user to disable vulnerability detection rules and identify which rules have been disabled.
- Allows the user to configure to ignore specific or multiple vulnerabilities.
- Allows pause, resume and immediate scan cancellation.
- Allows to identify, remove, export and import the results of scans performed.
- Allows to add and manage assets.
- Allows to export and import target lists from files in CSV or list format.
- Allows to export and import the current tool settings to/from a file.
- Allows real-time viewing of the list of vulnerabilities found.
- Provides syntax highlighting for the programming languages supported by the solution.
- Includes its own notepad for manual testing, including a collection of common injection string generators, hash generators, encoders and decoders, HTML functions and text manipulation.
- Includes an Extension Development Kit (EDK) to allow the addition of new features.

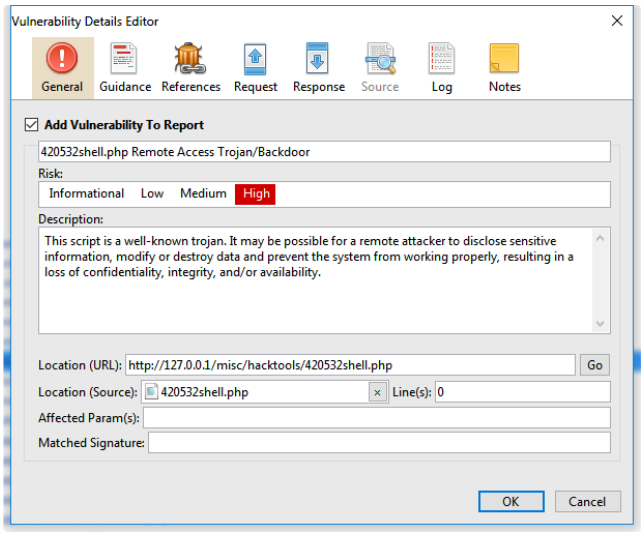
SCHEDULING SCANS

The management console allows to schedule scans with the following fields and options:

- Time, date or day of the week
- Type of Scan: Dynamic or Code
- Scan Target
- Report generation with selection of report template
- Sending a report by email after the analysis is finished
- Option to perform a hidden scan (without window)
- Option to export command line to be used in third party schedulers or other tools

VULNERABILITY ALERTS

The management console allows editing and viewing a vulnerability alert, including changing the state, severity, and inserting comments on the vulnerabilities found.



Such alerts contain the following properties:

- Vulnerability name
- Vulnerability description;
- Vulnerability location, which can be a URL or file.
- Reference code for known vulnerability bases, such as CVE, CWE, NVD, OSVDB, if any;
- CVSS score: version 2 and 3;
- Severity level (High, Low, Medium or Informational);
- Remediation guide;
- The affected parameters;
- Affected line numbers;
- A short extract from the vulnerable source code, when applicable;
- Remediation code examples, when applicable;
- Request, Headers and HTTP Response, in case of dynamic analysis;
- Exfiltrated data (if any);
- User notes

F. SCAN FEATURES

REPORTING AND RESULTS

Syhunt presents the scan results and generates reports on the vulnerabilities found:

- Allows to perform pass/fail testing based on the severity of the vulnerabilities identified.
- Allows the comparison between two scans performed against the same target or source code, presenting the differences through a report and the management console that indicates the differences, such as new, unchanged or removed vulnerabilities.
- Allows automatic sending of analysis results to a selected email address.
- Generates alerts for each type of unique vulnerability that is identified.

- Confirms whether an application's security breach has occurred from a trace.
- Supports the CVSS (Common Vulnerability Scoring System) standard version 2.0 and 3.0, to communicate the severity of a vulnerability and help determine the urgency and priority of the security response, including base score and other metrics.
- Generates reports in different formats, including the formats: PDF and HTML (for reading) and XML, JSON and CSV (to be processed by other tools).
- Sorts vulnerabilities based on CVSS3, CVSS2 (ranging from 0.0 None to 10.0 Critical) or Four Severity Steps (high, low, medium or informational).

REPORT TEMPLATES

Syhunt offers the possibility of generating reports using the following templates:

Standard	Standard management report
Comparison	Including the comparison of results of a scan with previous scans, indicating clearly the differences with evaluation graphs and tables
Compliance	Pass / fail report with items from OWASP Top 10, CWE / SANS Top 25 Most Dangerous Software Errors and PCI DSS in their most current versions
Compliance (Mobile)	Pass / fail report with items from OWASP Mobile Top 10, CWE / SANS Top 25 Most Dangerous Software Errors and PCI DSS in their most current versions
Complete	Including comparison of results, compliance and all necessary technical and vulnerability information

The reports generated by the solution come with following information:

- General scan details: start date, target, status, duration and scan method used;
- Graphs and Statistics: consolidated total vulnerabilities and by severity level;
- Vulnerability details: all details about each vulnerability alert, as displayed by the management console;
- Coverage Details: mapped structure, list of forms, emails, JavaScript files and other resources found, as well as technologies and platforms detected during the analysis;
- Version and brief description of the active license.

VULNERABILITY SCANNING METHODS

Syhunt comes with the following pre-defined scanning methods:

Application Scan (Dynamic)	Maps the structure of a website and performs Passive Analysis and Active Attacks
Application Code Scan	Focused on all types of vulnerabilities in source code
Spider Only	Only maps the structure of a website without performing Passive Analysis or attacks
Passive Scan	Only maps the structure of a website with Passive Analysis, but without attacks
Top 25 CWE	Based on the CWE Top 25 Most Dangerous Software Errors list
Top 10 OWASP	Based on the OWASP Top Ten Web Application Security Risks list
Top 5 PHP	Based on OWASP PHP Top 5, but not just limited to PHP
Fault Injection	Focused on data injection flaws, such as XSS, SQL Injection, File Inclusion and Command Execution
Structure Brute-Force	Focused on finding common backup files, administrative pages and similar exposures
Backup Files	Focused on backup, hidden and obsolete files, but not as aggressively as the Structure Brute-Force method
Complete Penetration Test	Performs all dynamic tests in an extensive and time-consuming manner
SQL Injection	Focused on SQL Injection and NoSQL Injection vulnerabilities
XSS	Focused on Cross-Site Scripting (XSS) vulnerabilities and evasion of anti-XSS filters
File Inclusion	Focused on local or remote file inclusion (LFI and RFI) vulnerabilities
Malicious Content	Focused on malware, backdoors, hidden entry points and signs of intrusion
Unvalidated Redirects	Focused on redirect vulnerabilities

SQL INJECTION DETECTION

Syhunt identifies SQL and NoSQL injection vulnerabilities through dynamic analysis, using inferential (time-based), in-band (error-based) and out-of-band (OOB) techniques, as well as through source code analysis. The following databases and techniques are covered:

Database	DAST (Inferential)	DAST (In-Band)	SAST	OAST (OOB)
NoSQL Injection	✔	✔	✔	N/A
MongoDB	✔	✔	✔	N/A
SQL Injection	✔	✔	✔	✔
Firebird/InterBase	N/A	✔	✔	N/A
IBM DB2	N/A	✔	✔	N/A
Informix	N/A	✔	✔	N/A
MariaDB / MySQL	✔	✔	✔	N/A
Microsoft Access	N/A	✔	✔	N/A
Microsoft SQL Server	✔	✔	✔	✔
Oracle	✔	✔	✔	✔
PostgreSQL	✔	✔	✔	N/A
SQLite	✔	✔	✔	N/A
Sybase	✔	✔	✔	N/A

DETECTION OF CODE INJECTION AND CODE DISCLOSURE

Syhunt identifies code injection and exposure vulnerabilities through dynamic analysis, using inferential (time-based) and in-band (print-based and Passive Analysis techniques), through source code analysis. The following languages, environments and techniques are covered:

Language or Environment	DAST (Inferential)	DAST (Disclosure)	DAST (In-Band)	SAST (Injection Flaw)

ASP Classic	✔	✔	✔	✔
ASP.NET	✔	✔	✔	✔
Java	✔	✔	✔	✔
JavaScript	✔	N/A	✔	✔
Lua (Nginx, Apache, CGI-Lua, and so on)	✔	✔	✔	✔
Perl	✔	✔	✔	✔
PHP	✔	✔	✔	✔
Python	✔	N/A	✔	✔
Ruby	✔	N/A	✔	✔
Server Side Includes (SSI)	N/A	✔	N/A	N/A

OS COMMAND INJECTION DETECTION

Syhunt identifies OS command injection vulnerabilities through dynamic analysis, using inferential (time-based) and in-band (print-based) and out-of-band (OOB) techniques, as well as through source code analysis. The following OSes are covered:

OS	DAST (Inferential)	DAST (In-Band)	SAST	OAST (OOB)
Unix/Linux	✔	✔	✔	✔
Windows	✔	✔	✔	✔
BSD	✔	✔	✔	✔
Solaris	✔	✔	✔	✔
MacOS	✔	✔	✔	✔
iOS	N/A	N/A	✔	N/A
Android	N/A	N/A	✔	N/A

LIST OF VULNERABILITY CHECKS

Syhunt identifies the following types of vulnerabilities, weaknesses and exposures in web applications, as well as in mobile applications when applicable:

API Abuse & Misuse
Arbitrary File Manipulation
Backdoor (Web-based)
Bad Practices
Broken Authentication
Broken Cryptography
Buffer Overflow
Code Injection, EL (Expression Language) Injection & Regular Expression Injection
Command Execution
Common Backup Files and Folders, and Backup with Common or Double Extension
Common Form Weaknesses, including email form hijacking, hidden price field, auto-complete enabled and unencrypted credit card transaction
Cookie Manipulation
Cross-Site Scripting (XSS), including DOM-based XSS, HTML5 specific, Weak XSS Filter and Cross Frame Scripting (XFS)
Dangerous Methods
Debug Entry Points, including Hidden Debug Parameters
Default Content
Denial-of-Service (DoS): Client and Server-Side
Directory Listing

Directory Traversal
Disclosure of Path, Source Code, Database, Password, Internal IP Address, Web Technology and others
Forgery of Log, Cross Site Request and Server Side Request (SSRF)
Hardcoding or Logging of Sensitive Information
HTTP header injection, HTTP response splitting
Inappropriate or Malicious Content
Inclusion of Local or Remote File
Information Leakage
Injection of LDAP
Injection of JSON, XML, XPath & XXE (XML External Entity)
Injection of NoSQL, SQL & HQL
Injection of SSI (Server-Side Includes)
Insecure Communication
Insecure Cryptographic and Hash Algorithms
Insecure Data Storage: missing or insufficient data protection cases
Insecure Randomness
Insecure Salting
Missing or Weak HTTP Security Headers
Security Misconfiguration
Sensitive Information on Client-Side
Suspicious Comments in Source Code and HTML
Uncontrolled Format String

Unencrypted Login
Unvalidated Redirects
Use of Local Storage, as well as confidential data stored in local storage
Weak Password Hashing
Weak Protocols

G. SYSTEM REQUIREMENTS

Syhunt Hybrid (including its Community Edition) can be installed on 64-bit Windows or 64-bit Linux, but it is able to analyze applications designed for any target platform, including Android, Apple iOS and MacOS, BSD, Linux, Windows, Solaris and Unix, independently of the platform it is executed from.

1. 4GB of available RAM (8GB recommended)
2. 1GB of free disk space*
3. Internet Connection (recommended for code scans and dynamic scans and some features)
4. One of the following compatible 64-bit operating systems:
    1. Windows 7, 8 or 10, or Windows Server 2008 to 2019
    2. Ubuntu Server or Desktop 18 or higher
    3. CentOS 7 or 8 (Minimal or Everything)
    4. Any unofficially supported Linux distribution such as the ones listed below.
5. (Optional) GIT on Linux or GIT for Windows (optional for GIT repository scans)
6. Java or Java Headless installed on Linux OS
7. If native binary is not available for your specific Linux distribution yet, Wine64 Stable (3, 4 or 5) is required to be installed.
8. (Optional) Java 8 or higher (optional for Android APK file scan)

* This does not include the space required to save scan session data, which varies depending on the website or source code being analyzed and the scan frequency.

COMPATIBLE LINUX DISTRIBUTIONS

Officially Supported:

✔ Ubuntu Server/Desktop 18.10 and later

✔ CentOS 7.7 and later (Minimal or Everything)

Unofficially (Successfully Tested):

✔ Kali Linux 2019 and later
✔ Parrot OS 4.1, 4.7 and later
✔ Debian 9.11 and later
✔ Linux Mint 19.2 and later
✔ OpenSUSE Leap 15.1 and later
✔ Fedora 32
✔ MX Linux 19.1 and later
✔ KDE Neon 2020.03 and later
✔ Deepin 15.9
✔ Manjaro 19
✔ Arch Linux 2019 and later

Unsupported:

⊘ Elementary OS 5.1 (Successfully Tested), 5.0 (Unsupported)
⊘ CentOS 6.1 (Successfully Tested)
⊘ Solus 4.1 (Unstable)

H. GLOSSARY/REFERENCES

1. **Active Attacks**: when a security analysis carries out attacks such as brute force, injection and denial of services.
2. **Augmented Dynamic Analysis**: the combination of the DAST and OAST test methods.
3. **Asset**: a URL, source code file or repository that can be analyzed.
4. **DAST**: Dynamic Application Security Testing is when a tool communicates with a web application to identify vulnerabilities and weaknesses in the application. Also known as a black-box test.
5. **DOM**: Document Object Model.
6. **FAST**: Forensic Application Security Testing.
7. **HAST**: Hybrid Application Security Testing is the combination of the SAST and DAST test methods.
8. **Hybrid-Augmented Analysis**: the combination of the SAST, DAST and OAST test methods.
9. **Injection**: when the tool submits data to application's entry points and analyzes its response to determine if the application's code is vulnerable.
10. **Lua**: lightweight and extensible scripting language created in Brazil, designed to expand applications.
11. **MAST**: Mobile Application Security Testing is when a tool analyzes the source code or package of an application to identify programming errors and conditions that indicate vulnerabilities.
12. **NVD**: National Vulnerability Database, the US government's vulnerability database.
13. **OOB**: Out-of-Band.
14. **OAST**: Out-of-band Application Security Testing is when a tool tries to force a web application to connect to other servers to identify otherwise invisible vulnerabilities in the application.
15. **OSVDB**: Open Source Vulnerability Database.
16. **OWASP**: Open Web Application Security Project.

7. **Passive Analysis**: when a security analysis identifies vulnerabilities and exposures without executing attacks or drawing attention.
8. PLUS: Indicates functionality available only in the Hybrid Platinum Plus license.
9. **Proof of Concept (PoC)**: Set of actions to demonstrate that a product will work as intended.
10. **Rule**: An option that allows to enable or disable one or more vulnerability checks.
11. **SAST**: Static Application Security Testing is when a tool analyzes the source code of an application to identify programming errors and conditions that indicate vulnerabilities. Also known as a white-box test.
12. **XHR**: XMLHttpRequest is an API used to send HTTP or HTTPS requests from JavaScript.

For additional product documentation, visit **syhunt.com/docs**