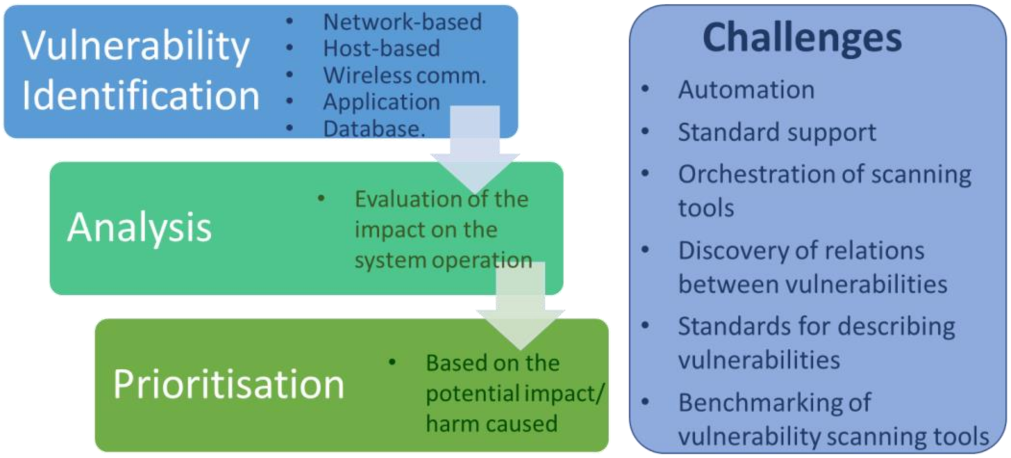
Vulnerability Assessment

Vulnerability assessment represents a critical component of the vulnerability management and IT risk management lifecycles in the companies of any size, protecting systems and data from any unauthorized access and data breaches; and improving overall security of companies' systems. Vulnerability assessment includes identifying, quantifying and prioritizing (or ranking) any potential vulnerabilities in computer networks, systems, hardware, applications and other parts of IT ecosystems. Not all threats are equally harmful for the companies, and due to a lack of time and resources, organizations often have to decide which threats to address first. They rank them based on the risk they represent for the company. This process is called prioritization and is very important for a successful implementation of new vulnerability management programs.


Vulnerability Identification
• Network-based
• Host-based
• Wireless comm.
• Application
• Database.
Analysis
• Evaluation of the impact on the system operation
Prioritisation
• Based on the potential impact/ harm caused
Challenges
• Automation
• Standard support
• Orchestration of scanning tools
• Discovery of relations between vulnerabilities
• Standards for describing vulnerabilities
• Benchmarking of vulnerability scanning tools

Existing technologies include vulnerability scanners which are automated tools, checking for security weaknesses or abnormalities in the companies' networks, systems and applications. They can be categorized into 5 types, based on the assets they scan - network-based, host-based, wireless, application and database scanners. As the name already suggests web application scanners scan web applications to detect security vulnerabilities such as cross-site scripting (XSS), SQL injection, command injection, path traversal and insecure server configuration. Unlike other types of scanners, web application scanners can find previously unknown vulnerabilities, unique to the tested application. These tools are categorized as Dynamic Application Security Testing (DAST), which is the process of testing applications or software products in an operating state. They are used together with Static Application Security Testing (SAST) tools that analyse web application's source code during the development phase. The Open Web Application Security Project (OWASP) provides a list of both, DAST and SAST tools available on the market, including their name, owner, license, platforms and some additional notes. The most popular DAST tools are general web vulnerability scanners, covering test cases for various types of web application vulnerabilities. Examples include AppSpider, Burp Suite, Nessus, Qualys Web Application Scanning (offered as a service), and open-source: OpenVAS, Arachni, OWASP ZAP, and w3af.

At the research forefront, vulnerability assessment faces several challenges: vulnerability scanning should be as automated as possible and support standard references to discovered vulnerabilities which is important, especially in terms of interoperability with other tools or their agglomeration. Using standards like CVE, CWE, or CVSS enables merging results from various scanning tools. Related, there is also a need for orchestration of scanning tools with the purpose of their integration into custom interfaces or automation frameworks or combining tools with similar or complementary functionalities to increase the accuracy of results or to extend the detection scope. Beside the mere detection of vulnerabilities, discovery of relations between vulnerabilities should be captured as well. Standards for describing vulnerabilities (e.g. SCAP) could be exploited to include mitigation specifications which could be, at least to some degree, automatically executed when the vulnerabilities are detected with a high confidence level. Another research topic includes benchmarking of vulnerability scanning tools for various infrastructure and application kinds with the objective of finding the most useful scanners depending on the situation and to explore optimal vulnerability coverage by using complementing scanning tools. Based on this, vulnerability scanners could also be dynamically deployed and agglomerated depending on the infrastructure, which can be automatically explored with network reconnaissance tools or recognized by machine-readable Infrastructure as Code (IaC) definitions.

FISHY includes two solutions: Zed Attack Proxy (ZAP) and w3af, both open source and developed as a result of WISER and CYBERWISER EC-funded projects. W3af is a web application attack and audit framework that helps users to secure their web applications by finding and exploiting all vulnerabilities. The framework is licensed under GPLv2.0., developed using Python and is easy-to-use and extend. ZAP is a flexible and extensible web application testing tool that stands between the tester's browser and the web application, and as such intercept and inspect messages between browser and web application, if necessary, modify the contents and forward packets on to the destination.

For more information, visit https://fishy-project.eu/library/deliverables and check D2.1!