AACT – Executive Summary Report (ESR)

Prepared by	**Soumya Paul and Tom Leclerc**
Reference	**SA-DOPS-STU-RP-0027**
Issue	**1**
Revision	**1**
Date of Issue	**19/09/2024**
Status	**Draft**
Document Type	**Deliverable**

Revision control

Issue	Revision	Date	Author	Comments
1	0	01/08/2024	Soumya Paul	Initial document
1	1	19/09/2024	Soumya Paul	Revised after AR RIDs

Approval

Issue	Revision	Date	Approved by	Comments

Table of Contents

1 Introduction

The current document provides an executive summary for the ESA activity AO/1-10464/21/D/AH. The contract was performed by Proximus Luxembourg S.A. and Telespazio Germany. The activity designed and developed a custom automated security testing tool capable of comprehensive testing of both standard IT systems and space mission ground segment systems and associated protocols and interfaces. The developed tool was deployed and validated in a representative space mission ground segment environment.

2 Terms, definitions and abbreviated terms

2.1 List of Acronyms

Acronym	***Full Text***
AACT	Advanced Automated Cybersecurity Testing
SUT	System Under Test
SRA	Security Risk Assessment
UX	User Experience

3 Description of the work performed

The work was performed through several tasks.

The first task consisted of performing a thorough study of the current state of the art for automated penetration testing. This included the identification of suitable tools, tactics, techniques and procedures suited to security testing of mission ground segment systems, protocols and interfaces and the definition of associated test cases, attack trees and space mission-relevant scenarios. Further, a lightweight Security Risk Assessment (SRA) was performed. An assessment of the suitability of the tools proposed in the first part for testing space applications and protocols was also carried out during this task.

The second task consisted of setting up the testing and development environments and conducting a manual penetration testing campaign in this environment. An end-to-end test environment was set up in order to meet the AACT needs and requirements and to support both internal and external penetration tests. The test environment supports penetration testing at network level as well as at space application level (software component interfaces, data flow etc.).

The top-level software architectural design of the AACT framework was carried out in the third task following which the development of the AACT software was performed in the fourth task in order to produce the AACT tool. The development executed out in an agile manner by iterating as necessary the architectural and conceptual design, software development, test and validation, documentation as well as the deployment, delivery and acceptance.

4 Outcome of the activity

The outcome of the AACT activity is a ‘plug-and-play’ penetration testing automation suite which can be used on the different segments of a space mission to discover security vulnerabilities and to ensure security of the missions. This brings several benefits for ESA.

AACT not only decreases the cost of (automated) penetration testing but also increases its efficiency. This means, that AACT can be integrated into the standard testing process at ESA for both ground and space segments with very little additional time and cost. Thus, although there is no substitute for full-fledged manual penetration tests carried out by security experts, the AACT framework nonetheless helps in improving the security by a significant amount. Furthermore, to facilitate such an integration, the design of the framework has been kept generic and flexible with added emphasis on its usability, simplicity, maintainability and an intuitive user experience.

This has been achieved by, first, deploying and using every tool and every module as separate docker containers. The principal advantage of using docker containers for every module of the architecture is that it offers tremendous flexibility in terms of maintenance and upgrade. It is sufficient to upgrade or swap one container image with another according to specific needs. This helps make the maintenance easier, especially the images of the hacking tools that are not straightforward to use. This also allows to segregate executions ensuring that each execution starts from a clean slate. Secondly, the deployment of the framework itself is straightforward. In order to deploy the AACT framework, only a docker server with the AACT docker images is needed.

Thirdly, the AACT attack scenarios are highly configurable. They are defined using the scenario graphical editor that allows to determine the tool chain and associated logic nodes. Scenarios can be based on pre-defined templates, but can then be configured for each session independently. The configuration can range from adjusting the parameters of the attack scenario (its noise, disruptiveness, time limit etc.) to editing and extending the scenario itself (e.g. selecting specific tools, defining requirements verifications and execution rules). Such configurability is very helpful for fine tuning the testing scenarios for different SUTs and different parts of the space mission.

Finally, The AACT framework provides support for standard attack catalogs like MITRE ATT&CK, SPARTA etc. On top of this, the framework can be expanded by adding support for other testing standards by adding additional tools and catalogs. This also helps to maintain and keep the framework up-to-date with the rapidly evolving security landscape which is especially relevant for space systems given their extended lifecycle (typically 5-10 years).