On a Collision Course: Unveiling Wireless Attacks to the Aircraft Traffic Collision Avoidance System (TCAS)

Giacomo Longo, *DIBRIS, University of Genova;* Martin Strohmeier, *Cyber-Defence Campus, armasuisse S + T;* Enrico Russo, *DIBRIS, University of Genova;* Alessio Merlo, *CASD, School of Advanced Defense Studies;* Vincent Lenders, *Cyber-Defence Campus, armasuisse S + T*

https://www.usenix.org/conference/usenixsecurity24/presentation/longo

This artifact appendix is included in the Artifact Appendices to the Proceedings of the 33rd USENIX Security Symposium and appends to the paper of the same name that appears in the Proceedings of the 33rd USENIX Security Symposium.

August 14–16, 2024 • Philadelphia, PA, USA

978-1-939133-44-1

Open access to the Artifact Appendices to the Proceedings of the 33rd USENIX Security Symposium is sponsored by USENIX.

USENIX Security '24 Artifact Appendix: On a Collision Course: Unveiling Wireless Attacks to the Aircraft Traffic Collision Avoidance System (TCAS)

Giacomo Longo
DIBRIS
University of Genova, Italy
giacomo.longo@dibris.unige.it

Martin Strohmeier
Cyber-Defence Campus
armasuisse S + T, Switzerland
martin.strohmeier@armasuisse.ch

Enrico Russo
DIBRIS
University of Genova, Italy
enrico.russo@unige.it

Alessio Merlo
CASD
School of Advanced Defense Studies, Italy
alessio.merlo@ssuos.difesa.it

Vincent Lenders
Cyber-Defence Campus
armasuisse S + T, Switzerland
vincent.lenders@armasuisse.ch

Version 2 - 9 August 2024

A Artifact Appendix

A.1 Abstract

In this appendix we provide the means to repeat the analysis performed in the article and reproduce all of the figures found within it. Furthermore, it provides additional visual proof to some claims found in the related manuscript [2].

A.2 Description & Requirements

A.2.1 Security, privacy, and ethical concerns

Building and running this artefact are not expected to cause security or privacy risks to the artifact user.

According to the ethical concerns expressed in the article, as attempting to reproduce the attacks outside of a controlled environment could have potentially catastrophic impacts, great care was taken to ensure the absence of any attack-enabling detail in this document and associated dataset.

A.2.2 How to access

The dataset, scripts, and the source code associated with this appendix can be found hosted on Zenodo at URL https://zenodo.org/doi/10.5281/zenodo.11351913 with DOI 10.5281/zenodo.11351913 [1].

A.2.3 Hardware dependencies

None

A.2.4 Software dependencies

In order to ensure reproducibility, all of the artifacts can be produced by leveraging a containerized environment.

As such, the only software dependencies are a *POSIX shell*, *GNU Make*, and *Podman* version 4 or greater.

A.2.5 Benchmarks

None

A.3 Set-up

We have included additional instructions in `README.md` files scattered thoughout the artifact directories. Below we describe just the minimal steps.

We recommend using a Fedora Linux 40 system with `make` and `podman` installed from the default repositories.

Please ensure that at least 10 GiB of storage is available.

When trying to run these instructions on Ubuntu 22.04, please notice that its default Podman version (3.4) is not sufficient for performing the remaining steps from this guide, which requires Podman version 4 and onwards.

A.3.1 Installation

1. Download and extract the `tcas-analysis.zip` file
2. Open a shell inside the extracted directory
3. Run `make build-container`

A.3.2 Basic Test

To the the functionality of the environment, we provide a command to automatically assert whenever the container is working.

Execute

```
make run-check
```

Below, an example output

```
Checking environment
--- Programs ---
  [OK] make found at /usr/bin/make
  [OK] mediainfo found at /usr/bin/mediainfo
  [OK] python3 found at /usr/bin/python3
  --- Paths ---
  [OK] found directory /data
  [OK] found directory /data/dataset
  [OK] found directory /data/scripts
Press any key to continue...
```

Six *OK*s indicate a successful execution.

A.4 Evaluation workflow

A.4.1 Major Claims

(C1): Section 8 - *"Every sensitivity alteration succeeds immediately upon the TCAS unit receiving the message."* This is proven by experiment (E1).

(C2): Section 9.1 - *"Our implementation consistently and accurately meets the response time requirements."* This is proven in the original article figures 11 and 12, which can be obtained from experiments (E3) and (E4).

(C3): Section 9.1 - *"Therefore, current cutting-edge COTS hardware has made such attacks feasible."* This is proven by experiment (E2).

(C4): Section 9.2 - *"These customizations were crucial in fully exploiting the hardware capabilities to overcome the challenges from . . . "* This is proven in Figure 13, obtainable from experiments (E3) and (E4).

(C5): Section 9.3 - *"Our tests indicated that the attacker's signals must be high enough to be received correctly"* This is proven by experiment (E5).

A.4.2 Experiments

(E1): *SLC effect visual inspection [8 human-minutes]:*
Preparation: Steps 1-2 from A.3.1
Execution: From the artifact root directory, go inside `dataset/slc`. Assert in the video that the indicated TCAS mode transitions between the "TA/RA" and "TA" modes, as commanded by the attacker terminal on the right.
Results: This inspection should assert that the RA DoS attack succeds as soon as the SLC command is received.

(E2): *Attack effects visual inspection [30 human-minutes]:*
Preparation: Steps 1-2 from A.3.1
Execution: From the artifact root directory, go inside `dataset/explanatory-and-promotional/reliability`. Watch the video called "super_ta" or its sped up version "super_ta_timelapse". Assert that the intruder is indicated as dangerous by the unit for the entirety of the video.
Results: This inspection should assert that the attack can reliably work over extended periods of time under laboratory environment.

(E3): *Automated dataset analysis [10 compute-minutes]:*
Preparation: Steps from A.3.1
Execution: From the artifact root directory, run `make run-analysis`.
Results: The analysis should generate the following files under `analysis/data`

encounters-distances-x.csv Each file contains three columns corresponding to the intruder desired distance, measured distance (by means of OCR), and error between the two.

encounters-summary.json This file contains summary statistics about the previous files, the only values of interest are the number of datapoints (222936) and the sum of experiment lengths (ts_s_sum = 13053s = 217.55 minutes).

encounters-ta-ra-distances.csv This file contains the cumulative distribution function (CDF) between distance and how many encounters had either a Traffic Advisory (TA) or Resolution Advisory (RA) at that point.

precision-calibration-data.json This file contains the suggested value for calibrating the testbed following a latency test, this value is mentioned in Section 7 of in [2] *"T_p is measured by calibrating the system beforehand with self-interrogations."*

precision-cdf.csv The CDF used for Figure 11 in [2].

precision-jitter.csv The CDF used for Figure 12 in [2].

precision-summary.json This file reports statistics for each requested delay, and is used to produce Figures 9 and 10 in [2].

precision.csv This file reports statistics for each requested delay, and is used to produce Figures 9 and 10 in [2].

ta-2-ra-summary.json This file summarizes the results for the experiments found under dataset/power-scaling, and is used to produce Figures 16 and 17 in [2].

time-to-ra.csv CDF for Figure 17 in [2].

time-to-ta.csv CDF for Figure 17 in [2].

time-to-ta2ra.csv CDF for Figure 17 in [2].

tuning-x.csv CDF for Figure 13 in [2]. `unopt` refers to the baseline measurement. `opt2` refers to the measurements taken after optimizations have been

applied.

tuning-x-summary.json Aggregated statistics for Figure 13 in [2].

(E4): *Automated plotting [10 compute-minutes]:*

Preparation: Steps from A.3.1 and Experiment E3

Execution: From the artifact root directory, run `make run-plots`.

Results: The analysis should generate the following files under `analysis/plots`:

in-time-replies.pdf Figure 9 in [2]

precision.pdf Figure 10 in [2]

precision-cdf.pdf Figure 11 in [2]

reply-jitter.pdf Figure 12 in [2]

opt-vs-unopt.pdf Figure 13 in [2]

encounters-distance-delta.pdf Figure 15 in [2]

encounters-ta-ra-dist.pdf Figure 16 in [2]

ta-ra-time.pdf Figure 17 in [2]

(E5): *Power effects visual inspection [30 human-minutes]:*

Preparation: Steps 1-2 from A.3.1

Execution: From the artifact root directory, go inside `dataset/power-scaling`. Each folder is named according to a normalized power value, with 1 indicating full transmission power and 0.05 indicating 5% of it. Assert each folder contains at least a file with its name ending with `_ta` and one ending in `_ra`.

Results: This inspection should assert that as long as the attacker transmitter is being heard its success rate is not influenced by the received power.

A.5 Version

Based on the LaTeX template for Artifact Evaluation V20231005. Submission, reviewing and badging methodology followed for the evaluation of this artifact can be found at https://secartifacts.github.io/usenixsec2024/.

References

[1] Giacomo Longo, Strohmeier Martin, Enrico Russo, Alessio Merlo, and Vincent Lenders. Dataset for "On a Collision Course: Unveiling Wireless Attacks to the Aircraft Traffic Collision Avoidance System (TCAS)". May 2024. https://doi.org/10.5281/zenodo.11351913.

[2] Giacomo Longo, Strohmeier Martin, Enrico Russo, Alessio Merlo, and Vincent Lenders. On a Collision Course: Unveiling Wireless Attacks to the Aircraft Traffic Collision Avoidance System (TCAS). In *33rd USENIX Security Symposium (USENIX Security 24)*, Philadelphia, PA, August 2024. USENIX Association.