Secure UAV Communications System Design (Write-up)

MITRE eCTF 2021
Team Cacti
University at Buffalo

We spent a lot of time making our design pass the test. The lessons we learned were categorized into the following items.

1. **Teamwork**
   We divided the system implementation into several small tasks and assigned each task to our team members. Another team member will handle the whole business logic. This was very helpful and moved our work fast. But there was a problem that not all of our team members have a full understanding of the whole system that made it hard to identify the bug when the system got errors at the final stage.

2. **Find a suitable open-source lib**
   In our design, to protect the communication, we used asymmetric crypto (RSA). The issue we meet was that the key format of the C implementation of RSA was different from the standard key format such as ASN or PKCS#1 that python can provide. Due to the time, we did not find another lib. And we fixed this issue by simply utilizing the file system to connect the python script and C applications. This may introduce a race conditions problem.

3. **Plan B is important**
   System design always changes through implementation. Not all security designs can be deployed due to various reasons. Unfortunately, we did not sign the RSA public key because of unknown issues and exposed our system to attack. So a plan B will help a lot.

4. **Debug is important**
   Docker was used to set up the whole system which made debugging more difficult for us. We prefer to rely on IDEs to help us to debug. At this moment, all codes were tested by judging the output strings. This was a very bad and inefficient choice. We did not check every line of code we wrote. Then we spent a lot of time finding out the bugs after failing the submission. So,

   - EVERY line of code should be debugged and confirmed.
   - If the system can not pass the test, it must be somewhere the code you write has a bug.

5. **Test all possibilities**
   Before submission, there should be a comprehensive test framework to evaluate the functionality of the system. Do not just wait for the organizer's test result.

6. **Don't be shy to ask the organizer's help**
   Well, we emphasize this because we wasted a lot of time checking the incomplete logs. If you find that you can not identify the bug your system encounters, please ask for help.