cFS Cryptography Library

Aman Thanvi (GSFC-5960)

Summer 2021

Mentor: John Lucas

Acknowledgements

- It has been an honor and privilege to be a summer intern with the GTOSat team.
- I would like to convey a special thanks to my mentor John Lucas for his mentorship throughout this summer.

Presentation Background

- The objective of this project was to provide a software-only solution using the latest CCSDS Space Data Link Security Protocol – Extended Procedures (SLDS-EP) to secure communications between a spacecraft running the core Flight System (cFS) and a ground station.
- The focus of this session's efforts was to streamline and refine the CryptoLib platform, bringing it closer to a mission-operational state ahead of its first major application with the launch of the Geostationary Transfer Orbit Satellite (GTOSat).
- Throughout the development process, new builds were tested and verified using a virtualized environment running the NASA Operational Simulation for Small Satellites (NOS^3) and COSMOS Ground Station.

Presentation Background

- What is NOS^3?
  - NOS^3 is an open-source, software-only testbed for small satellites licensed under the NASA Open-Source Agreement.
  - It is a collection of Linux executables and libraries.
  - It is intended to easily interface with flight software developed using the NASA Core Flight Software (cFS).
  - Current simulations are based on commercial-off-the-shelf (COTS) hardware that is being used on the Simulation-to-Flight 1 (STF-1) CubeSat.

nos^3
NASA Operational Simulator
for Small Satellites

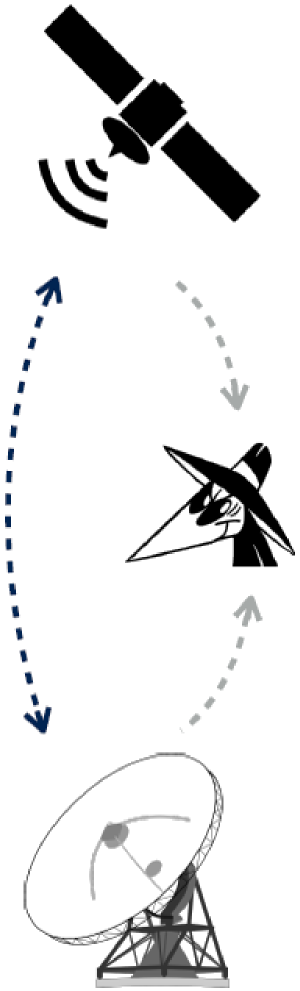
- What is COSMOS?
  - COSMOS is an open-source ground system software which is used to provide command and control of the flight software.
  - COSMOS is comprised of 17+ applications, that allow control of hardware that can be anything from test equipment (power supplies, oscilloscopes, switched power strips, UPS devices, etc.), to development boards, to instruments and complete satellites.
  - COSMOS can be easily setup and configured on Windows, Linux, and Mac OSX operating systems, allowing for greatly reduced project costs.
  - It uses unique telemetry presentations to provide detailed displays and on-demand information.
  - All data that passes through COSMOS is logged allowing for spacecraft health assessments, post-test analysis, anomaly investigation, and data archiving.

Ball Aerospace
COSMOS

Why CryptoLib for Security?

Generally, a mission's approach to comply with communication security requirements from applicable NASA Procedural Requirements (NPR) and/or Standards (NASA-STD-XXX), is driven by hardware-based solutions that need design and implementation efforts from early phases and sometimes require coordination with external agencies (e.g., NSA).

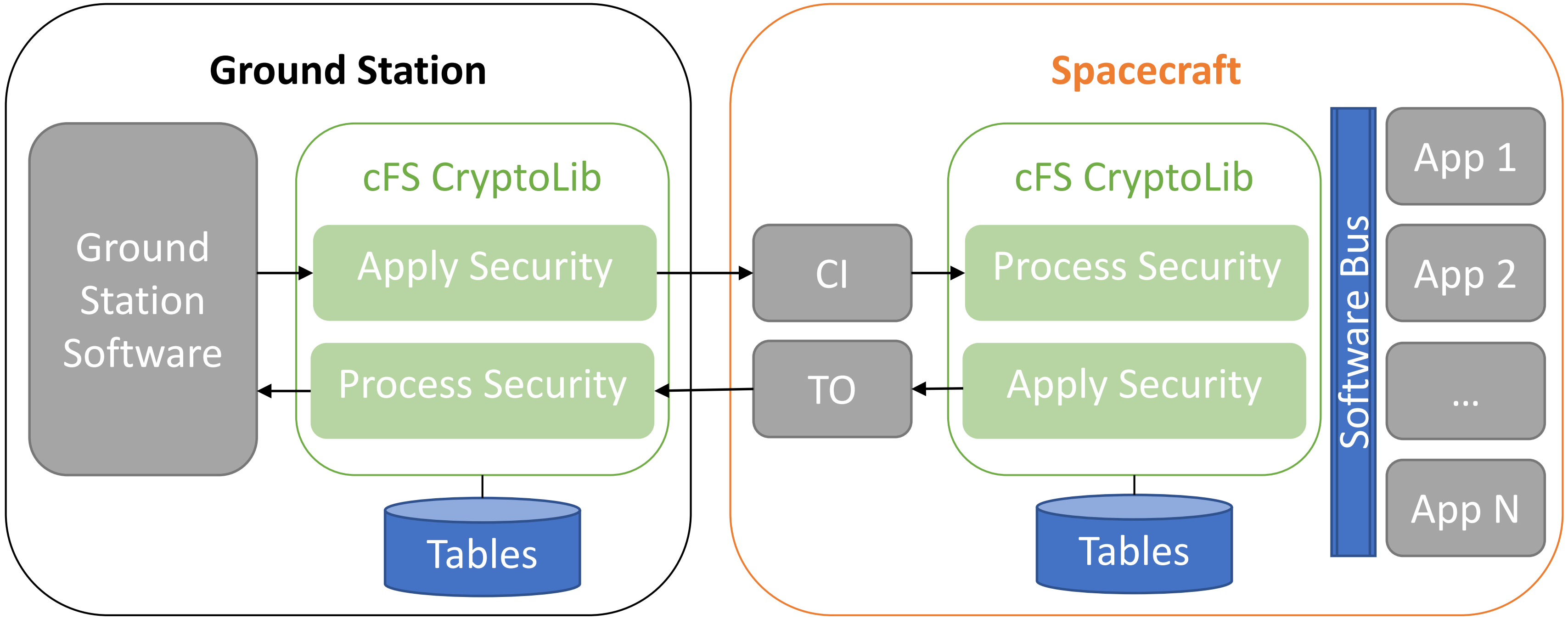
Also, the current hardware-based security platforms are expensive and non-modular, imposing challenges to seamless integration into pre-existing mission platforms and architecture.



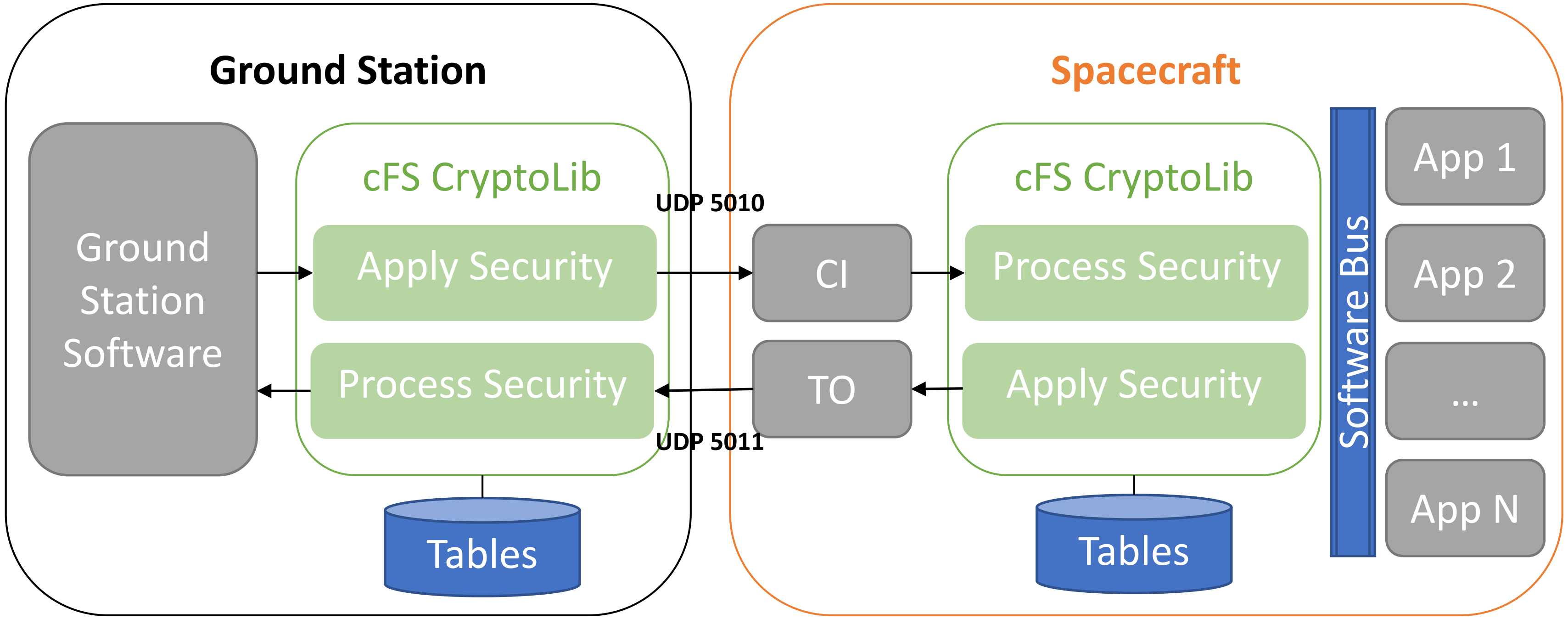
CryptoLib provides a software-only, fully compliant solution that can plug-and-play with the standard flight and ground software mission architecture while also upholding three fundamental pillars of security:
- Confidentiality
- Integrity
- Availability

Shifting to a platform like CryptoLib allows projects to adapt to increasing demands for communications security rapidly and cost-effectively; easily accessible, preexisting mission resources, like processing power, can be traded in exchange for operation on the CryptoLib platform within full compliance with NASA PRs and STDs.

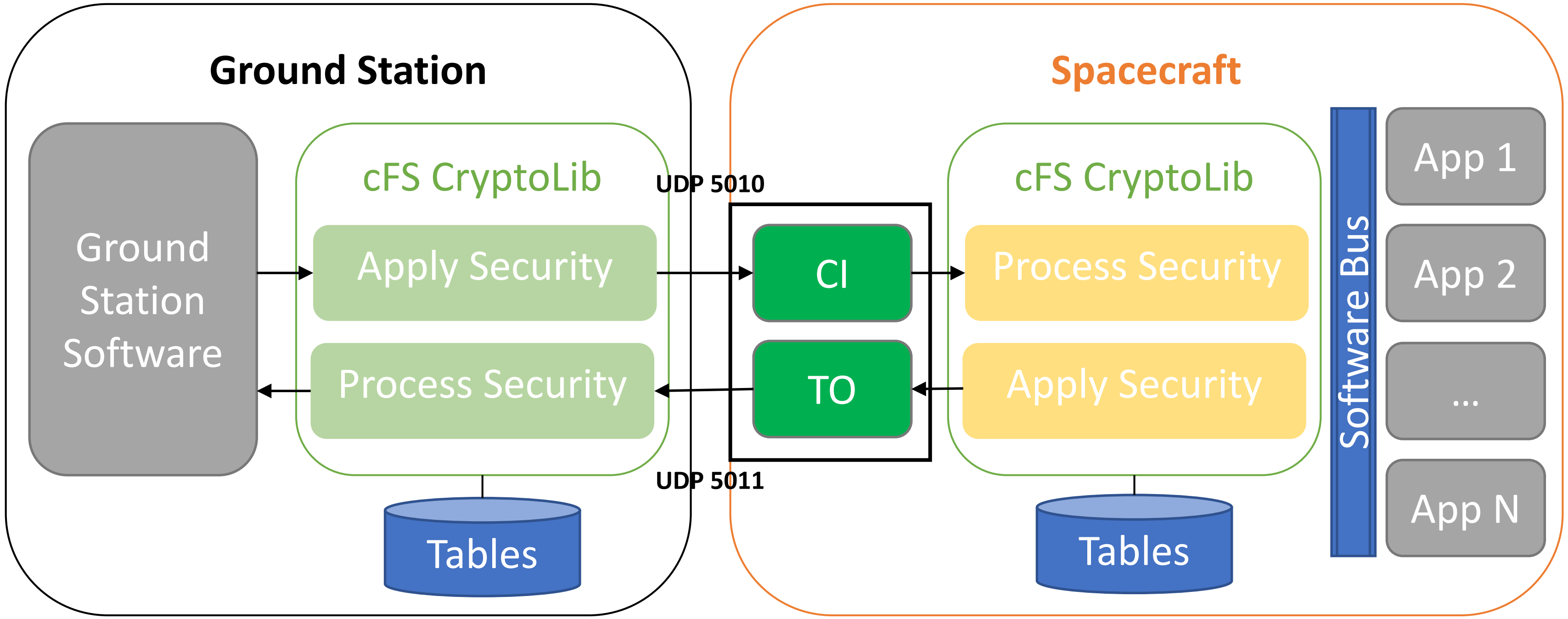
CryptoLib Overview – Initial Baseline



CryptoLib Overview - Verified baseline



CryptoLib Overview – Current functionality



Presentation Methodology - Input

- Started with a functional, but not mission-operational, build of CryptoLib. All basic functionalities were demonstrated and verified through test scripts simulating different flight conditions and operations.
- However, the existing codebase required refactoring to reach a more modular and polished structure to move toward a more mission-ready state.

Presentation Methodology - Process

- The Command Ingest (CI) and Telemetry Output (TO) applications were restructured first.
  - Large functional blocks were split into smaller, more manageable files that were all connected under dynamic header files.
  - Outdated standards and cipher integrations were removed and staged for replacement with newer cryptographies.
  - After restructuring these applications, a new functional baseline was established and verified before continuing with the refactor process.
- Next were the ‘Apply Security’ and ‘Process Security’ sublayers.
  - These modules proved to be the most intensive and time-consuming to restructure, as they make up a large portion of CryptoLib’s active functionality and have the most dynamic modules.

Presentation Methodology - Outcome

- CryptoLib’s initial codebase was significantly restructured to satisfy target functionality and requirements and mission-readiness guidelines while maintaining compliance with NASA-STD-1006.
- The Command Ingest (CI) and Telemetry Output (TO) functionalities were restructured entirely, while the Spacecraft ‘Process Security’ (TC_Process) and ‘Apply Security’ (TC_Apply) functionalities are in the process of being re-prototyped.

Continuing Development of CryptoLib

- Complete the re-prototyping of the Spacecraft ‘Process Security’ (TC_Process) and ‘Apply Security’ (TC_Apply) functionalities.
- Update the ‘Tables’ on both the Ground Station and Spacecraft sides of the architecture to leverage and integrate more modern standards and technologies.

Presentation Results and Conclusions

- CryptoLib's current state puts it on track to be fully operational prior to GTOSat's launch readiness milestones.
- CryptoLib has proven to be a modular software-only solution for missions to be compliant with applicable communication security requirements.
- Update the Ground Station layer to match Spacecraft layer updates to ensure seamless operation across a more modern software-only platform.

Future Plans

- Receive a Bachelor’s degree in Computer Science with a cybersecurity focus and a minor in Cybersecurity at the University of Maryland, College Park.
- Pursue additional security intern and research opportunities within the space communication and mission domain.
- Explore further computer science opportunities in the public and private sectors.
- Pursue a Master’s degree in Computer Science

References

References

Asbury, M. (2017, August 22). *NASA operational simulation for small satellites*. NASA. https://www.nasa.gov/centers/ivv/jstar/nos3.html.

Cosmos. Aerospace. (n.d.). https://www.ball.com/aerospace/programs/cosmos.

Nos3. (n.d.). http://www.stf1.com/NOS3Website/Nos3MainTab.php.

Questions?