IRT

SAINT
EXUPÉRY

IRT SAINT EXUPÉRY

A Satellite Constellation Simulator for Space Systems Cybersecurity Research and Development

Simone Urbano, Jacques Girard, Louis Lolive,
Vincent Nicomette, Pierre Bacquet, Patrick Hebrard,
Guillaume Auriol, Ludovic Cintrat, Benjamin Deporte,
Julien Airaud, Matteo Merialdo.

The CSS (Cybersecurity for Space Systems) project

- Technology transfer cooperative program
- Co-funded by French research, academic and industry partners
- Operated by IRT ("Institut de Recherche Technologique") Saint Exupery

3.3 M€

36 months

6 FTE

Objectives

- Apply cybersecurity state of the art methods to civilian space.
- Investigate the added value of new technologies (cyber range, AI) to civilian space cybersecurity.

Content

01. Goals

02. Reuse of open-source NASA NOS3

03. Nexova CITEF cyber range

04. Contributions

05. Constellation scenario

06. Modular and scalable architecture

07. Threat model

08. Attack library

09. Countermeasures

10. Onboard IDS

11. On ground probes

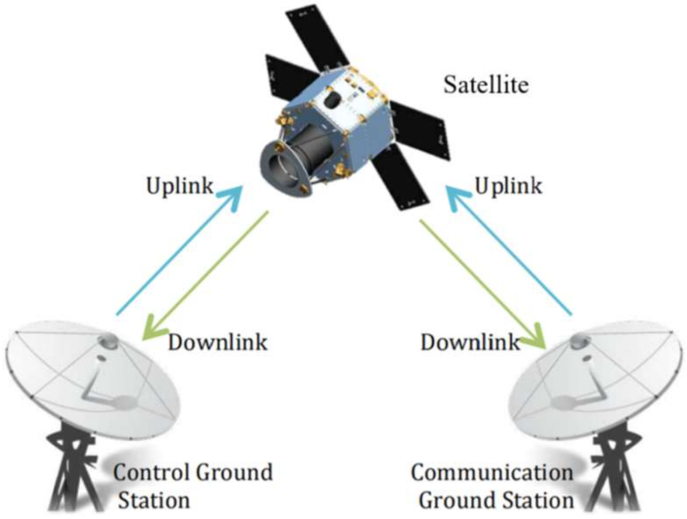
12. Example of Hijacking

13. Conclusions

01. Goals

Main goals of the simulator:

- Generate realistic CCSDS traffic for a constellation of CubeSats
- Automatic attack generation
- Automatic threat detection/mitigation
- Generate realistic CCSDS data sets for surrogate models and/or AI
- Create benchmark scenarios to compare detection performance

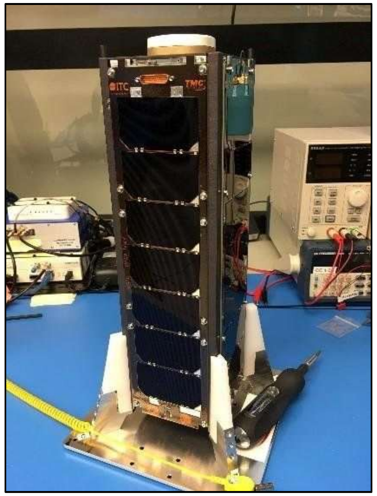


Hypothesis: no radiofrequency (RF) layer simulation.

02. Reuse of open-source NASA NOS^3

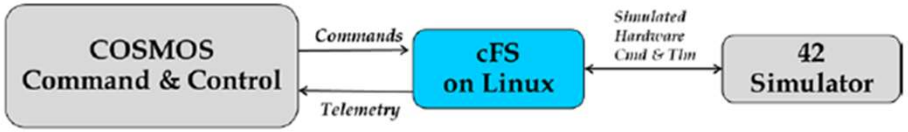
- Emulation based on **NASA NOS^3 v1.6.2**.
- Reference mission is **STF-1**.
- The emulator uses **CryptoLib** for SDLS-EP.

nos^3
NASA Operational Simulator
for Small Satellites



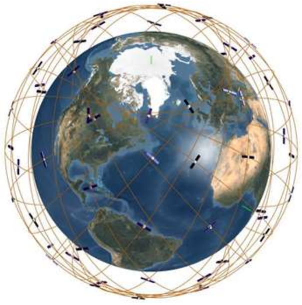
De-orbited in February 2024

CryptoLib

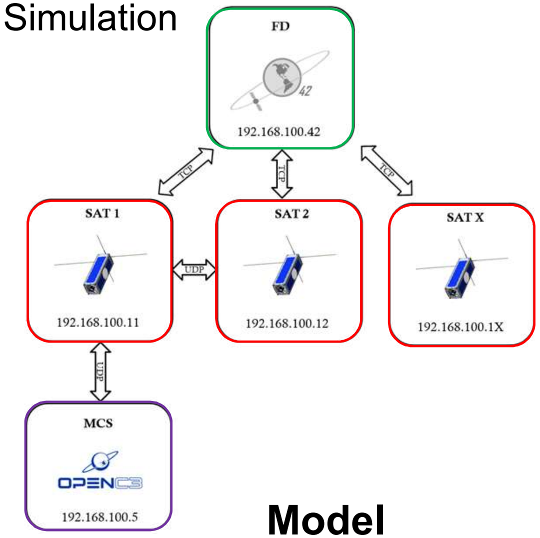


03. Nexova CITEF cyber range

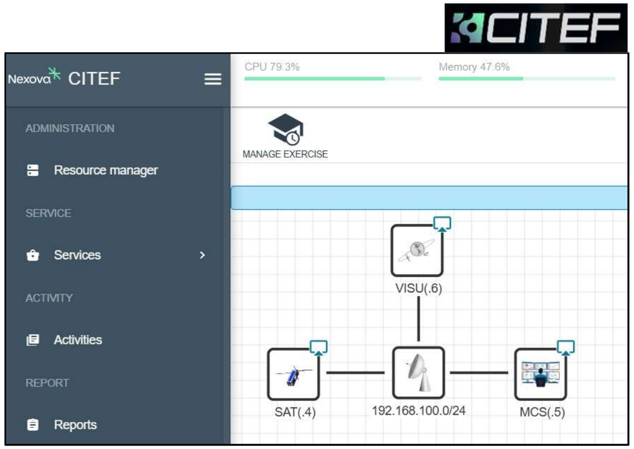
- Multiple VMs (Virtual Machines) hosted in CITEF
  - 1 **VM** for each SAT in the constellation
  - 1 **VM** for Mission Control System (MCS)
  - 1 **VM** for Space Environment Simulation



Constellation



Model



Virtualization

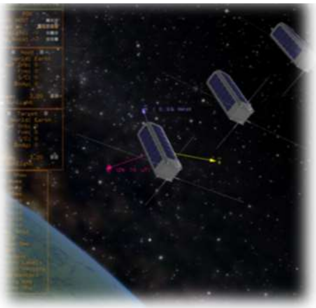
04. Contributions

- Use of Transfer Frames for TC and TM
- New components (ISL/RM, Front End) for constellation deployment
- Multipurpose Input generator
- Customizable mission
- Library of exploits
- On board IDS/IPS
- Hardware-in-the-loop (HIL) testing with on-ground probes
- Additional features for realism (ADCS modes, payload camera simulation, etc…)

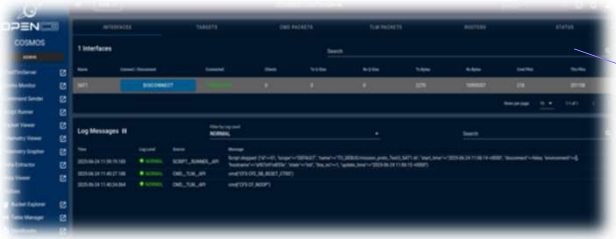
→ The user can simulate a realistic mission with realistic CCSDS traffic for a satellite/constellation with optical payload + the user has a baseline for attacks and countermeasures development.

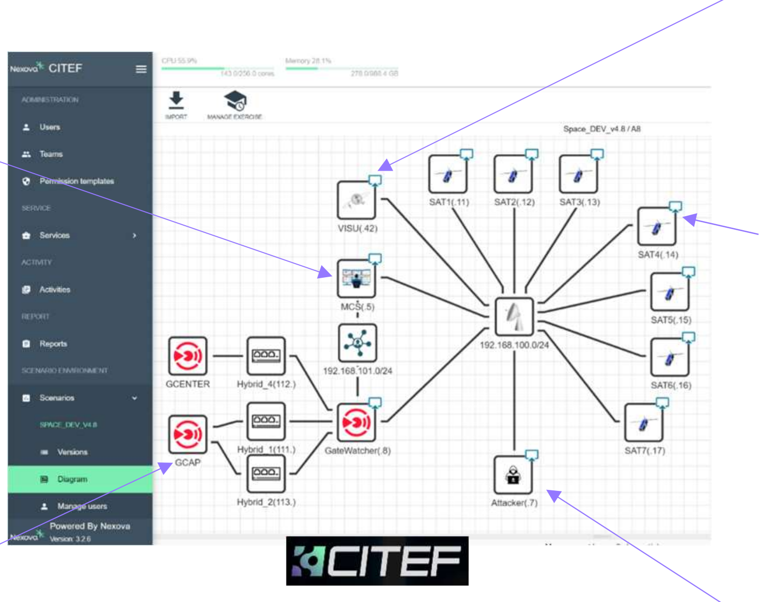
05. Constellation Scenario

Space Environment

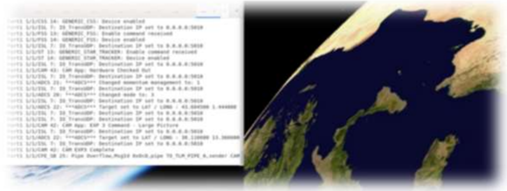


Ground Software





Flight Software and Payload



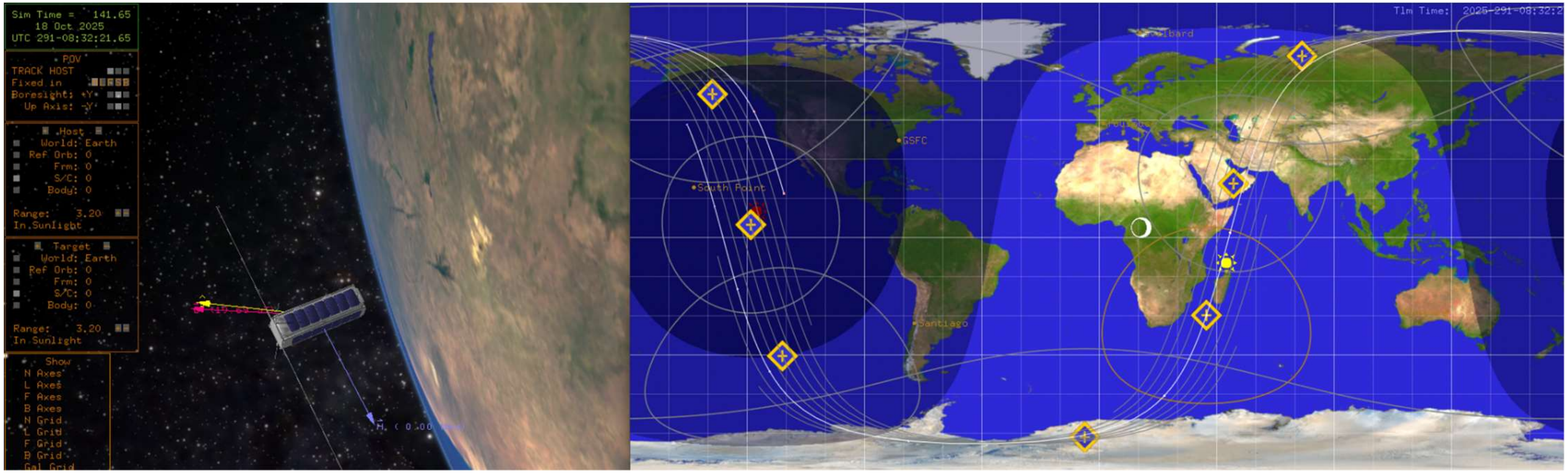
Ground Probes

Attacks Library

FRENCH INSTITUTES OF TECHNOLOGY

05. Constellation Scenario

Example from NASA 42

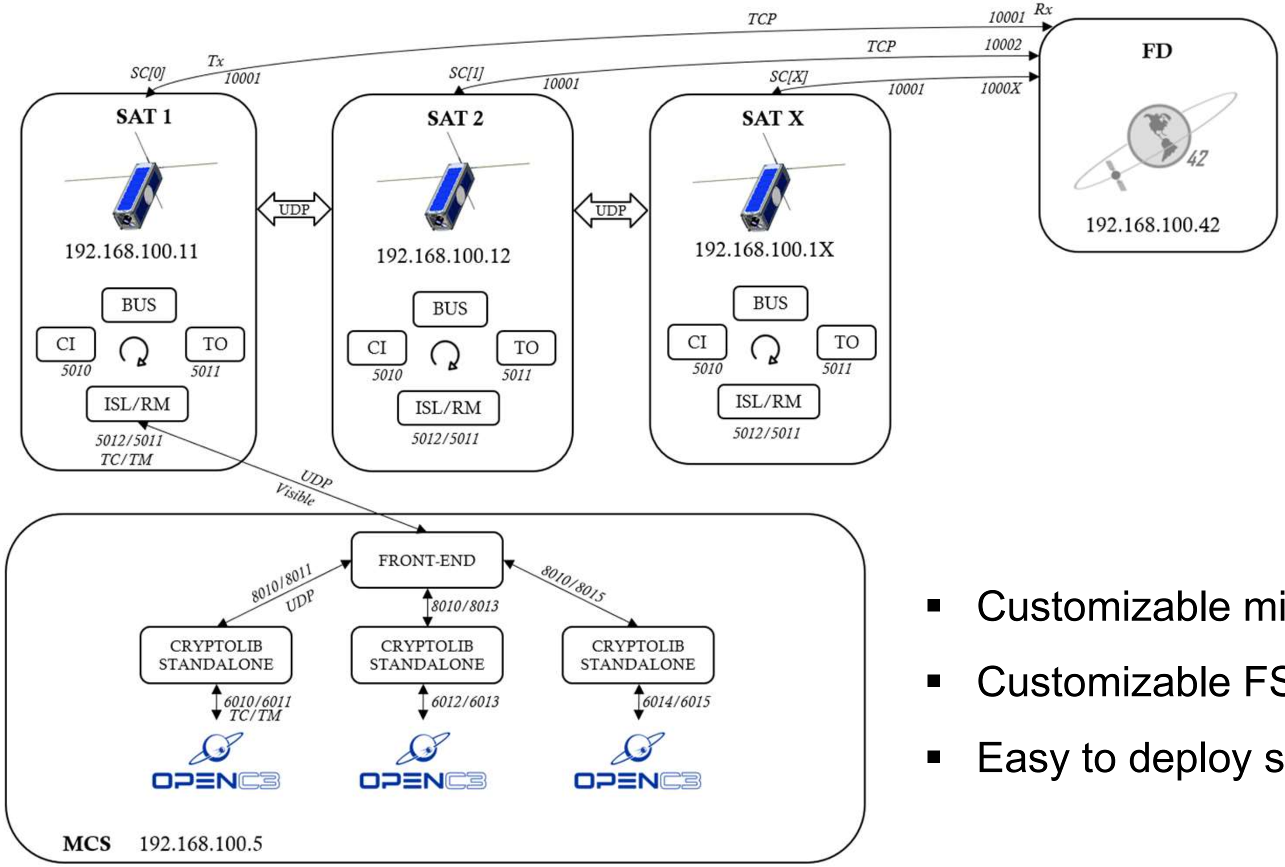


```
Sim Time =   141.65
   18 Oct 2025
UTC 291-08:32:21.65

       POV
TRACK HOST
Fixed in
Boresight: +Y
   Up Axis: -Y

      Host
      World: Earth
    Ref Orb: 0
        Frm: 0
        S/C: 0
       Body: 0

Range:      3.20
In Sunlight

     Target
      World: Earth
    Ref Orb: 0
        Frm: 0
        S/C: 0
       Body: 0

Range:      3.20
In Sunlight

      Show
   N Axes
   L Axes
   F Axes
   B Axes
   N Grid
   L Grid
   F Grid
   B Grid
   Gal Grid
```

```
Tlm Time:  2025-291-08:32:2
```

Example: 7 satellites constellation in Sun synchronous orbit. MCS is at Svalbard station.

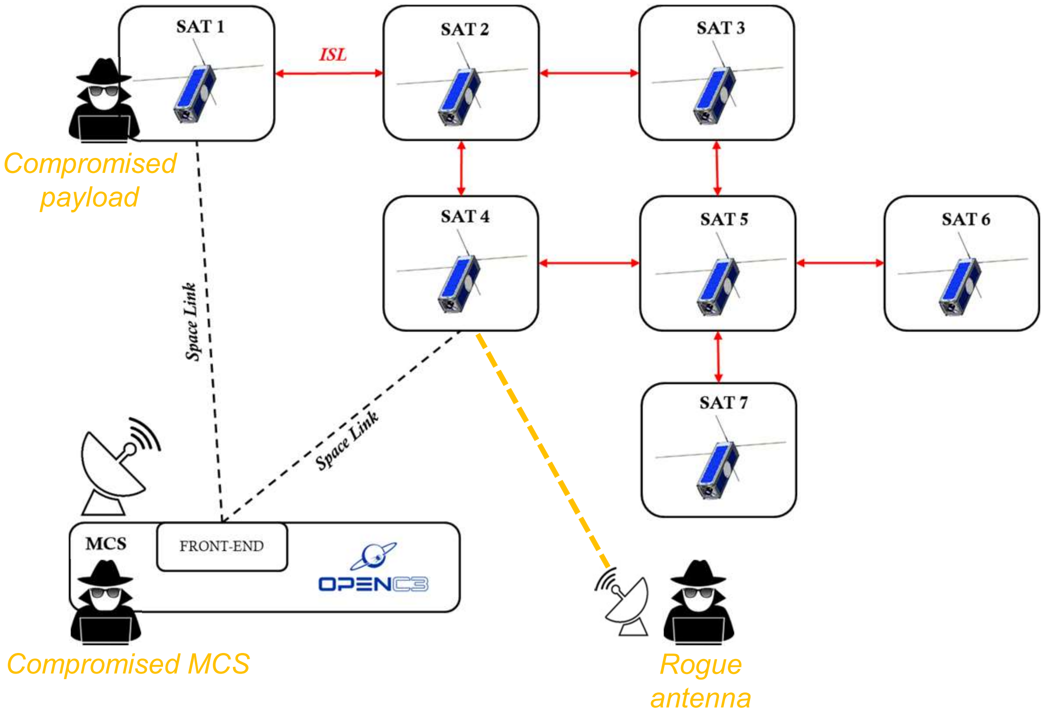
06. Modular and scalable architecture



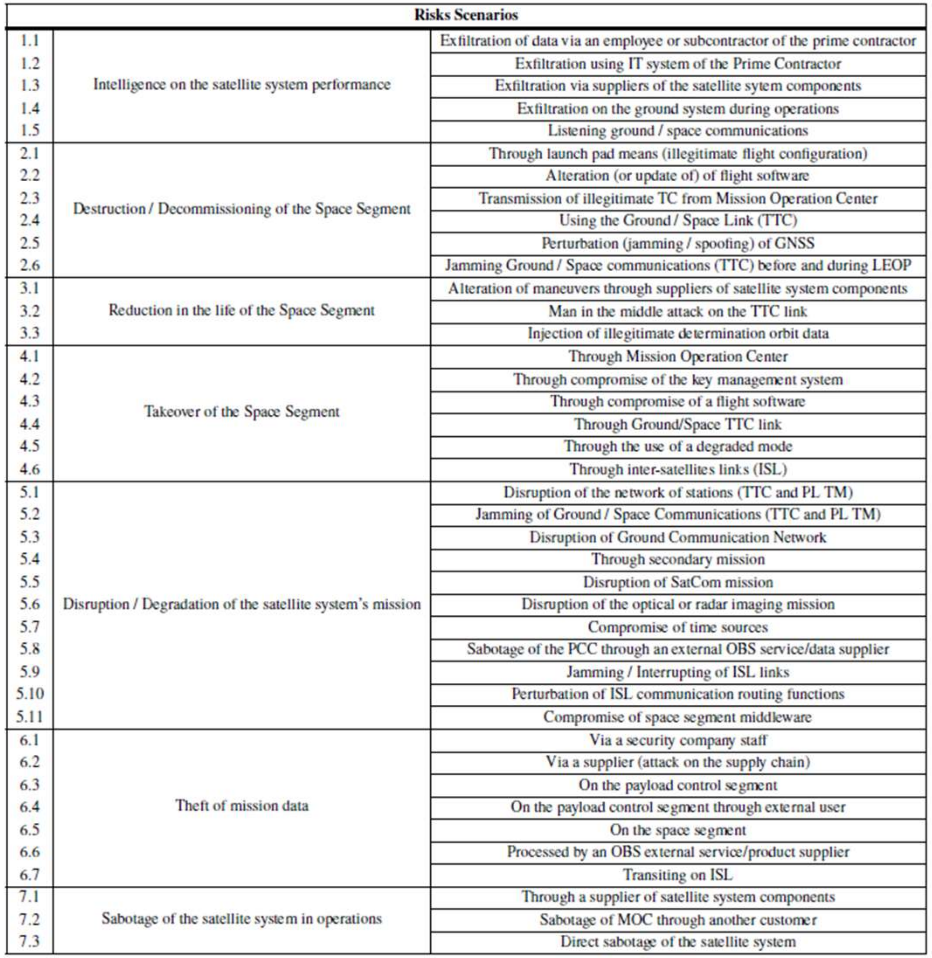
- Customizable mission
- Customizable FSW
- Easy to deploy small constellation

07. Threat model

Simplified threat model



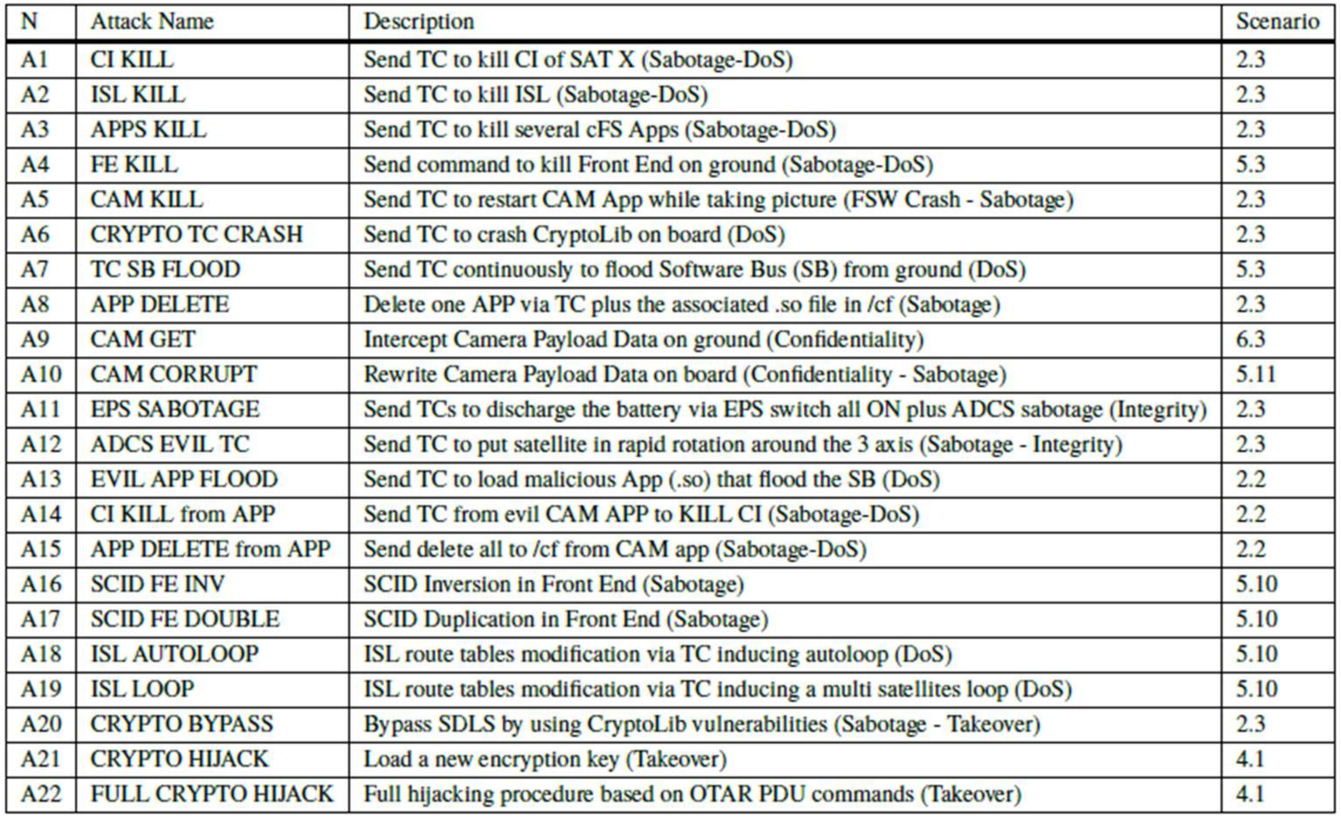
EBIOS RM Approach



Risks Scenarios		
1.1	Intelligence on the satellite system performance	Exfiltration of data via an employee or subcontractor of the prime contractor
1.2		Exfiltration using IT system of the Prime Contractor
1.3		Exfiltration via suppliers of the satellite sytem components
1.4		Exfiltration on the ground system during operations
1.5		Listening ground / space communications
2.1	Destruction / Decommissioning of the Space Segment	Through launch pad means (illegitimate flight configuration)
2.2		Alteration (or update of) of flight software
2.3		Transmission of illegitimate TC from Mission Operation Center
2.4		Using the Ground / Space Link (TTC)
2.5		Perturbation (jamming / spoofing) of GNSS
2.6		Jamming Ground / Space communications (TTC) before and during LEOP
3.1	Reduction in the life of the Space Segment	Alteration of maneuvers through suppliers of satellite system components
3.2		Man in the middle attack on the TTC link
3.3		Injection of illegitimate determination orbit data
4.1	Takeover of the Space Segment	Through Mission Operation Center
4.2		Through compromise of the key management system
4.3		Through compromise of a flight software
4.4		Through Ground/Space TTC link
4.5		Through the use of a degraded mode
4.6		Through inter-satellites links (ISL)
5.1	Disruption / Degradation of the satellite system's mission	Disruption of the network of stations (TTC and PL TM)
5.2		Jamming of Ground / Space Communications (TTC and PL TM)
5.3		Disruption of Ground Communication Network
5.4		Through secondary mission
5.5		Disruption of SatCom mission
5.6		Disruption of the optical or radar imaging mission
5.7		Compromise of time sources
5.8		Sabotage of the PCC through an external OBS service/data supplier
5.9		Jamming / Interrupting of ISL links
5.10		Perturbation of ISL communication routing functions
5.11		Compromise of space segment middleware
6.1	Theft of mission data	Via a security company staff
6.2		Via a supplier (attack on the supply chain)
6.3		On the payload control segment
6.4		On the payload control segment through external user
6.5		On the space segment
6.6		Processed by an OBS external service/product supplier
6.7		Transiting on ISL
7.1	Sabotage of the satellite system in operations	Through a supplier of satellite system components
7.2		Sabotage of MOC through another customer
7.3		Direct sabotage of the satellite system

08. Attack Library

- Threat model + vulnerability analysis* → simple list of attacks (as of October 2025, to be enriched)



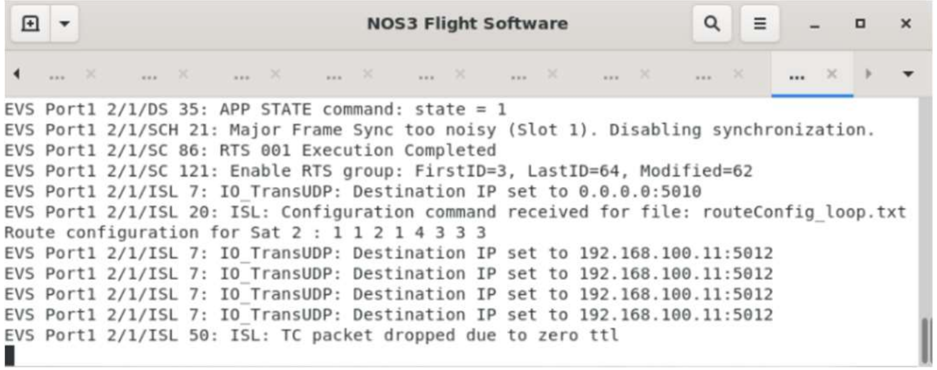
N	Attack Name	Description	Scenario
A1	CI KILL	Send TC to kill CI of SAT X (Sabotage-DoS)	2.3
A2	ISL KILL	Send TC to kill ISL (Sabotage-DoS)	2.3
A3	APPS KILL	Send TC to kill several cFS Apps (Sabotage-DoS)	2.3
A4	FE KILL	Send command to kill Front End on ground (Sabotage-DoS)	5.3
A5	CAM KILL	Send TC to restart CAM App while taking picture (FSW Crash - Sabotage)	2.3
A6	CRYPTO TC CRASH	Send TC to crash CryptoLib on board (DoS)	2.3
A7	TC SB FLOOD	Send TC continuously to flood Software Bus (SB) from ground (DoS)	5.3
A8	APP DELETE	Delete one APP via TC plus the associated .so file in /cf (Sabotage)	2.3
A9	CAM GET	Intercept Camera Payload Data on ground (Confidentiality)	6.3
A10	CAM CORRUPT	Rewrite Camera Payload Data on board (Confidentiality - Sabotage)	5.11
A11	EPS SABOTAGE	Send TCs to discharge the battery via EPS switch all ON plus ADCS sabotage (Integrity)	2.3
A12	ADCS EVIL TC	Send TC to put satellite in rapid rotation around the 3 axis (Sabotage - Integrity)	2.3
A13	EVIL APP FLOOD	Send TC to load malicious App (.so) that flood the SB (DoS)	2.2
A14	CI KILL from APP	Send TC from evil CAM APP to KILL CI (Sabotage-DoS)	2.2
A15	APP DELETE from APP	Send delete all to /cf from CAM app (Sabotage-DoS)	2.2
A16	SCID FE INV	SCID Inversion in Front End (Sabotage)	5.10
A17	SCID FE DOUBLE	SCID Duplication in Front End (Sabotage)	5.10
A18	ISL AUTOLOOP	ISL route tables modification via TC inducing autoloop (DoS)	5.10
A19	ISL LOOP	ISL route tables modification via TC inducing a multi satellites loop (DoS)	5.10
A20	CRYPTO BYPASS	Bypass SDLS by using CryptoLib vulnerabilities (Sabotage - Takeover)	2.3
A21	CRYPTO HIJACK	Load a new encryption key (Takeover)	4.1
A22	FULL CRYPTO HIJACK	Full hijacking procedure based on OTAR PDU commands (Takeover)	4.1

*publicly available vulnerabilities

09. Countermeasures

“Threat based” approach (as of October 2025, to be enriched):

- Time To Leave (TTL) on top of CCSDS TF
- Routing tables verification on board
- “Critical TC” concept (additonal MAC)
- Watchdog messages for ISL
- IDS/IPS on board
- On ground IDS probes
- ...



```
EVS Port1 2/1/DS 35: APP STATE command: state = 1
EVS Port1 2/1/SCH 21: Major Frame Sync too noisy (Slot 1). Disabling synchronization.
EVS Port1 2/1/SC 86: RTS 001 Execution Completed
EVS Port1 2/1/SC 121: Enable RTS group: FirstID=3, LastID=64, Modified=62
EVS Port1 2/1/ISL 7: IO_TransUDP: Destination IP set to 0.0.0.0:5010
EVS Port1 2/1/ISL 20: ISL: Configuration command received for file: routeConfig_loop.txt
Route configuration for Sat 2 : 1 1 2 1 4 3 3 3
EVS Port1 2/1/ISL 7: IO_TransUDP: Destination IP set to 192.168.100.11:5012
EVS Port1 2/1/ISL 7: IO_TransUDP: Destination IP set to 192.168.100.11:5012
EVS Port1 2/1/ISL 7: IO_TransUDP: Destination IP set to 192.168.100.11:5012
EVS Port1 2/1/ISL 7: IO_TransUDP: Destination IP set to 192.168.100.11:5012
EVS Port1 2/1/ISL 50: ISL: TC packet dropped due to zero ttl
```

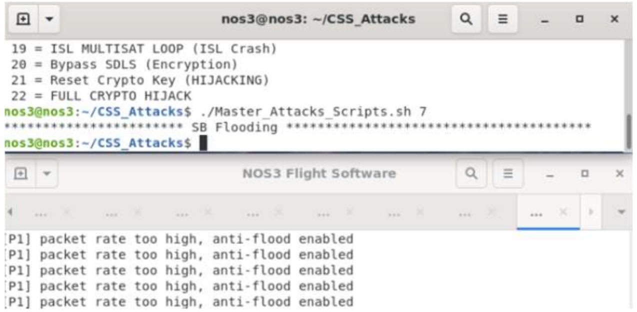
Example: FSW output on the Satellite 2 during the attack A19
if the TTL is activated.

10. Onboard IDS/IPS

Multilayer approach with:

- set of probes (in different locations)
- set of detection modules

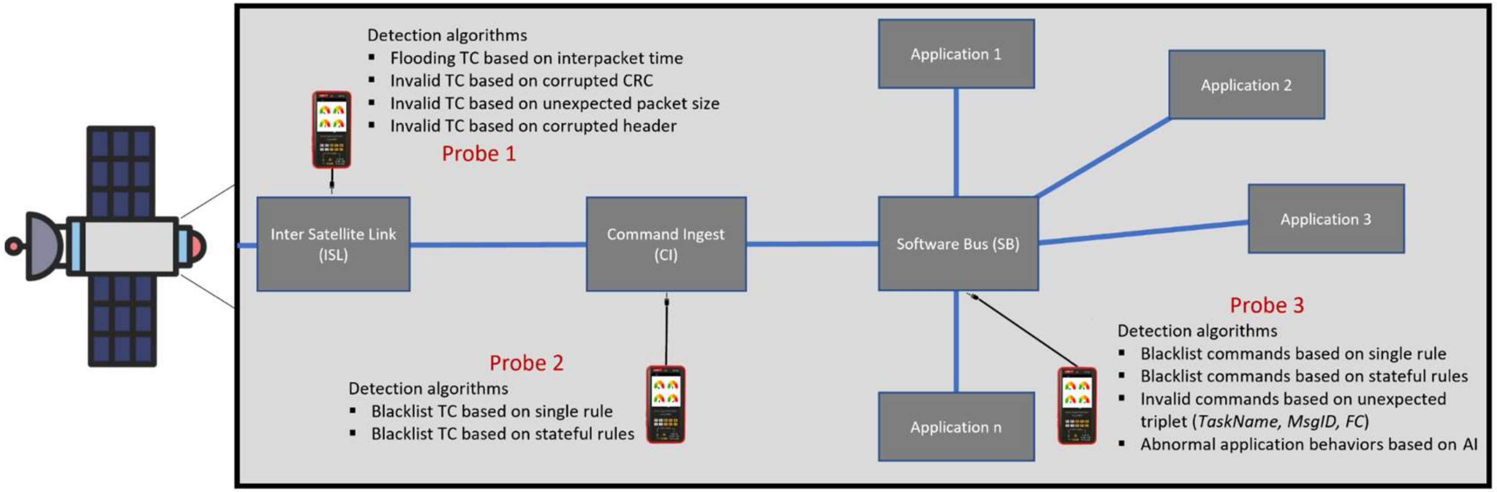
The IDS is customized to NASA cFS architecture and it is deployed as embedded FSW component.



```
 19 = ISL MULTISAT LOOP (ISL Crash)
 20 = Bypass SDLS (Encryption)
 21 = Reset Crypto Key (HIJACKING)
 22 = FULL CRYPTO HIJACK
nos3@nos3:~/CSS_Attacks$ ./Master_Attacks_Scripts.sh 7
*********************** SB Flooding **************************************
nos3@nos3:~/CSS_Attacks$ 
```

```
[P1] packet rate too high, anti-flood enabled
[P1] packet rate too high, anti-flood enabled
[P1] packet rate too high, anti-flood enabled
[P1] packet rate too high, anti-flood enabled
[P1] packet rate too high, anti-flood enabled
```

Example: FSW output on the Satellite 1 during attack A7 if the IDS is activated.



Probe 1

Detection algorithms

- Flooding TC based on interpacket time
- Invalid TC based on corrupted CRC
- Invalid TC based on unexpected packet size
- Invalid TC based on corrupted header

Probe 2

Detection algorithms

- Blacklist TC based on single rule
- Blacklist TC based on stateful rules

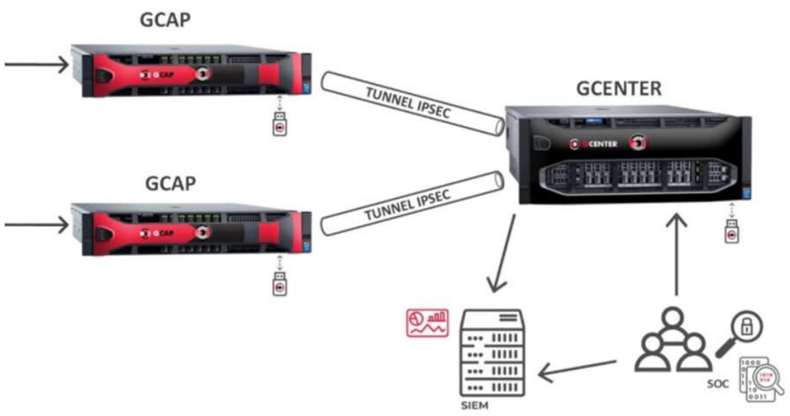
Probe 3

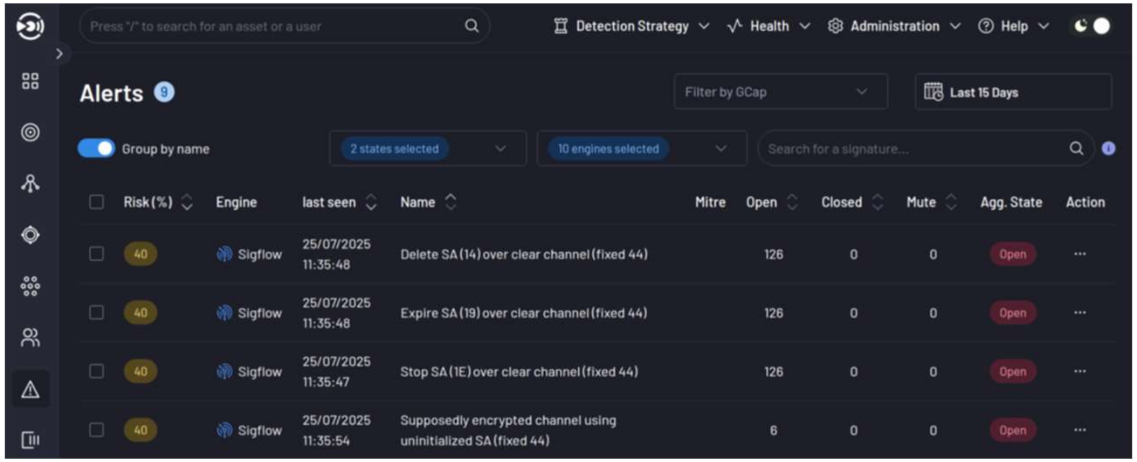
Detection algorithms

- Blacklist commands based on single rule
- Blacklist commands based on stateful rules
- Invalid commands based on unexpected triplet (*TaskName, MsgID, FC*)
- Abnormal application behaviors based on AI

11. On ground probes

- GCAP: Detection probe at network point; capture, analyse and send threat data.
- GCENTER: Central platform; manages probes, performs in-depth analysis, provides dashboards and long-term data storage (detection is based on Suricata engine).





Alerts 9

Filter by GCap

Last 15 Days

Group by name

2 states selected

10 engines selected

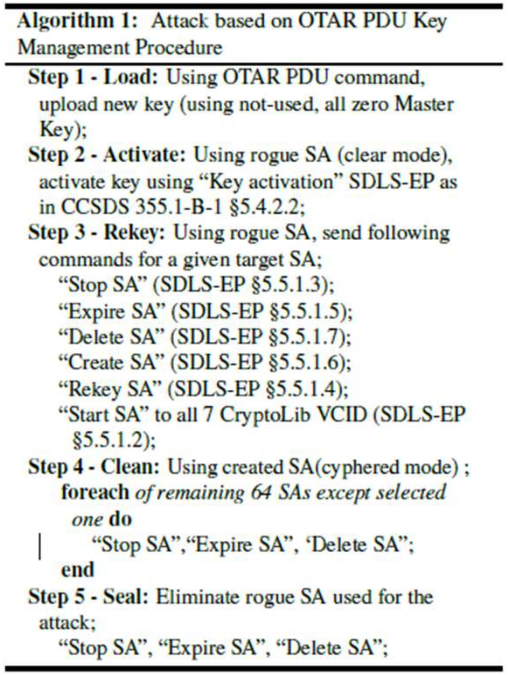
Search for a signature...

Risk (%)	Engine	last seen	Name	Mitre	Open	Closed	Mute	Agg. State	Action
40	Sigflow	25/07/2025 11:35:48	Delete SA (14) over clear channel (fixed 44)		126	0	0	Open	···
40	Sigflow	25/07/2025 11:35:48	Expire SA (19) over clear channel (fixed 44)		126	0	0	Open	···
40	Sigflow	25/07/2025 11:35:47	Stop SA (1E) over clear channel (fixed 44)		126	0	0	Open	···
40	Sigflow	25/07/2025 11:35:54	Supposedly encrypted channel using uninitialized SA (fixed 44)		6	0	0	Open	···

Example: type of alarms raised in GCENTER console during the attack A22.

12. Example of Hijacking (CryptoLib vulnerability)

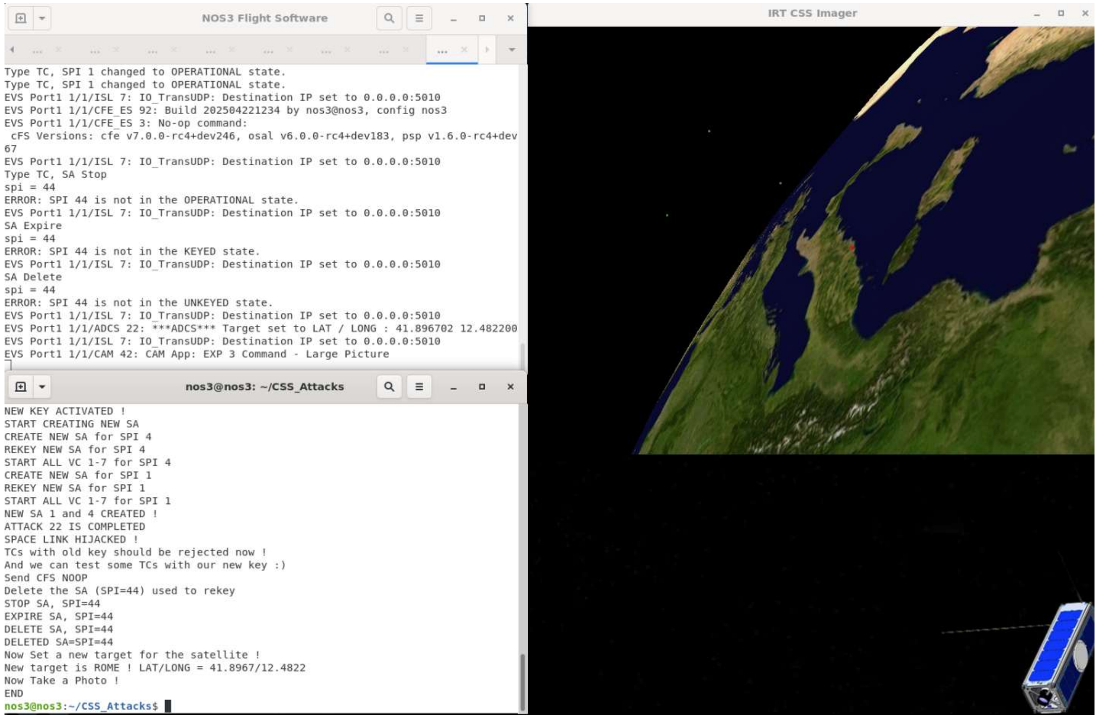
Create a set of malicious TC based on OTAR PDU procedure*



Algorithm 1: Attack based on OTAR PDU Key Management Procedure

```
Step 1 - Load: Using OTAR PDU command, upload new key (using not-used, all zero Master Key);
Step 2 - Activate: Using rogue SA (clear mode), activate key using "Key activation" SDLS-EP as in CCSDS 355.1-B-1 §5.4.2.2;
Step 3 - Rekey: Using rogue SA, send following commands for a given target SA;
    "Stop SA" (SDLS-EP §5.5.1.3);
    "Expire SA" (SDLS-EP §5.5.1.5);
    "Delete SA" (SDLS-EP §5.5.1.7);
    "Create SA" (SDLS-EP §5.5.1.6);
    "Rekey SA" (SDLS-EP §5.5.1.4);
    "Start SA" to all 7 CryptoLib VCID (SDLS-EP §5.5.1.2);
Step 4 - Clean: Using created SA(cyphered mode) ;
    foreach of remaining 64 SAs except selected one do
        "Stop SA","Expire SA", 'Delete SA";
    end
Step 5 - Seal: Eliminate rogue SA used for the attack;
    "Stop SA", "Expire SA", "Delete SA";
```

The attacker has full control of space link communications (encryption key ownership)



```
Type TC, SPI 1 changed to OPERATIONAL state.
Type TC, SPI 1 changed to OPERATIONAL state.
EVS Port1 1/1/ISL 7: IO_TransUDP: Destination IP set to 0.0.0.0:5010
EVS Port1 1/1/CFE_ES 92: Build 202504221234 by nos3@nos3, config nos3
EVS Port1 1/1/CFE_ES 3: No-op command:
 cFS Versions: cfe v7.0.0-rc4+dev246, osal v6.0.0-rc4+dev183, psp v1.6.0-rc4+dev
67
EVS Port1 1/1/ISL 7: IO_TransUDP: Destination IP set to 0.0.0.0:5010
Type TC, SA Stop
spi = 44
ERROR: SPI 44 is not in the OPERATIONAL state.
EVS Port1 1/1/ISL 7: IO_TransUDP: Destination IP set to 0.0.0.0:5010
SA Expire
spi = 44
ERROR: SPI 44 is not in the KEYED state.
EVS Port1 1/1/ISL 7: IO_TransUDP: Destination IP set to 0.0.0.0:5010
SA Delete
spi = 44
ERROR: SPI 44 is not in the UNKEYED state.
EVS Port1 1/1/ISL 7: IO_TransUDP: Destination IP set to 0.0.0.0:5010
EVS Port1 1/1/ADCS 22: ***ADCS*** Target set to LAT / LONG : 41.896702 12.482200
EVS Port1 1/1/ISL 7: IO_TransUDP: Destination IP set to 0.0.0.0:5010
EVS Port1 1/1/CAM 42: CAM App: EXP 3 Command - Large Picture
```

```
NEW KEY ACTIVATED !
START CREATING NEW SA
CREATE NEW SA for SPI 4
REKEY NEW SA for SPI 4
START ALL VC 1-7 for SPI 4
CREATE NEW SA for SPI 1
REKEY NEW SA for SPI 1
START ALL VC 1-7 for SPI 1
NEW SA 1 and 4 CREATED !
ATTACK 22 IS COMPLETED
SPACE LINK HIJACKED !
TCs with old key should be rejected now !
And we can test some TCs with our new key :)
Send CFS NOOP
Delete the SA (SPI=44) used to rekey
STOP SA, SPI=44
EXPIRE SA, SPI=44
DELETE SA, SPI=44
DELETED SA=SPI=44
Now Set a new target for the satellite !
New target is ROME ! LAT/LONG = 41.8967/12.4822
Now Take a Photo !
END
nos3@nos3:~/CSS_Attacks$
```

* Attack described by Antonin Boulnois, see https://securitybynature.fr/post/hacking-cryptolib

13. Conclusions

As part of CSS project, we propose a space system simulator that can be used for cybersecurity research and development and that we will publish online in 2026.

The main characteristic of the simulator are:

- Representative CCSDS traffic for constellation of CubeSats with optical payloads (including encryption)
- Representative Flight and Ground Software (NASA cFS and COSMOS OpenC3)
- Adapted for education and training (integration in CITEF)
- Mission and satellites can be customized (based on NASA cFS)
- Attack library is available (e.g. satellite takeover) based on public vulnerabilities
- Mitigations are available (TTL, routing tables checks, watchdogs, …)
- On board customized IDS/IPS available
- Up to 7 satellites with current hardware (can be extended)
- No radiofrequency layer by default (can be added, for example using GNU Radio)

13. Perspectives

- Inter Satellites Links: static routing by default, dynamic routing will be included in Q4’2025
- Extend the vulnerability analysis of the system (NASA cFS, COSMOS, …)
- Enrich the attack library with new attacks
- Integration and testing of AI algorithm with IDS for the detection of malicious payloads
- Correlation of on-board IDS information and on ground probes to enhance threats detection

IRT SAINT EXUPÉRY

Questions ?

Thank you