A vision on a methodology for the application of an Intrusion Detection System for satellites

Sébastien Gios
UCLouvain
Louvain-La-Neuve, Belgium
sebastien.gios@uclouvain.be

Charles-Henry Bertrand Van Ouytsel
UCLouvain
Louvain-La-Neuve, Belgium
charles-henry.bertrand@uclouvain.be

Mark Diamantino Caribé
Telespazio - ESA
Transinne, Belgium
mark.diamantino@telespazio.com

Axel Legay
UCLouvain
Louvain-La-Neuve, Belgium
axel.legay@uclouvain.be

ABSTRACT

The security of satellites has become critical in recent years due to their important role in modern society. However, numerous challenges, including limited computing resources, evolving cyber threats, and the isolated nature of satellites, hinder the development of effective security solutions. Different solutions should be implemented and combined to protect space assets: encryption, access control, zero-trust architecture, etc. This vision presents the challenges and aspects to consider for implementing an Intrusion Detection System (IDS) tailored to improve the security of satellite systems. Our approach uses a multi-level structure to define rule-based and machine-learning security approaches that address the challenges associated with different mission types. By strategically placing IDS components and considering the trade-offs of each location, we improve detection reliability. Additionally, we present an ontology-based method for visualizing the IDS configuration, which provides clear insight into system capabilities, enhances situational awareness, and facilitates identification and response to potential threats. We also provide strategies for updating the IDS while maintaining efficiency and security. This vision helps improve the cybersecurity measures of satellite operations and increase their resilience to cyberattacks.

KEYWORDS

Cybersecurity, Satellite, Methodology, Intrusion detection, Signature-based security, Anomaly-based security, Machine Learning

ACM Reference Format:
Sébastien Gios, Charles-Henry Bertrand Van Ouytsel, Mark Diamantino Caribé, and Axel Legay. 2024. A vision on a methodology for the application of an Intrusion Detection System for satellites. In *39th IEEE/ACM International Conference on Automated Software Engineering (ASE '24), October 27-November 1, 2024, Sacramento, CA, USA.* ACM, New York, NY, USA, 5 pages. https://doi.org/10.1145/3691620.3695314

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than the author(s) must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.
ASE '24, October 27-November 1, 2024, Sacramento, CA, USA
© 2024 Copyright held by the owner/author(s). Publication rights licensed to ACM.
ACM ISBN 979-8-4007-1248-7/24/10
https://doi.org/10.1145/3691620.3695314

1 INTRODUCTION

In recent years, we have become increasingly dependent on satellites for Earth observation, communications, the Global Positioning System (GPS), military strategy, and more. In 2023, more than 200 launches deployed thousands of satellites into various orbits. These numbers are expected to continue to increase every year [8, 23, 28]. The dual use of satellites by civilians and the military renders them a primary target in the event of a conflict. This is because they can be used to steal important information and cause disruptions (e.g., disruption of communications, prevention of humanitarian aid, etc.). Since 1970, it is estimated that there have been 250 instances of cyberattacks on space-based systems [22], with an increasing number of such attacks occurring each year. This trend is likely to continue, given the growing number of satellites in space [11]. Moreover, new reconnaissance capabilities offered by artificial intelligence, such as modulation identification [16], greatly enhance the capabilities of attackers to target satellites. Although satellites are only remotely accessible, the attack surface is still significant. Threats may target the ground station, control center, customer terminal, or the satellite itself. These attacks can be launched from anywhere on the ground, from the ground station communicating with the satellite, or even from another space unit. The communications system is usually the preferred entry point. From there, it is possible to access the satellite's sensors, actuators, and the onboard computer. A wide variety of attacks have been identified (e.g.: jamming, spoofing, denial of service, etc.), and TTP (Tactics, Techniques and Procedures) Frameworks such as ESA Space Shield [12], MITRE ATT&CK [9] and SPARTA [30] cover some of them. While satellites are progressively recognized as critical assets (e.g.: see the ViaSat cyberattack during the Ukraine-Russia conflict [6]), there is still a lack of regulations and laws regarding space. Many companies, including space agencies, are reluctant to use open-source solutions (e.g.: WolfSSL [34]) and generally design their own software for their satellites. The lack of coordination and information sharing between entities leads to a *security by obscurity* approach for most satellites. The difficulty in obtaining real data from satellite communications hinders the development of new security approaches and methodologies for incident detection and responses [20]. As there is no publicly available information on the operation of these systems, it is only possible to consult recommendations (e.g., CCSDS guide [5, 7]) and to assume that some of them are applied by space actors.

The development of cybersecurity solutions for satellites and space in general is complicated by different challenges [2]. While satellites face national-level threats with important computational resources [33], the satellites themselves are generally designed to operate on a limited amount of energy provided by solar panels. As a result, engineers optimize the use of resources for the satellite's specific mission. Therefore, if they want to add a layer of cybersecurity, the security solution must be carefully considered and highly optimized at the design stage. Many satellites are designed to remain in orbit for an extended period and adapted to a specific mission. Consequently, they vary greatly from each other (e.g. different protocols, eras, capabilities, etc.). For example, a satellite sent to an asteroid deep in space differs greatly from a geostationary communications satellite; both require dedicated security protection. Moreover, our distance with satellites, isolated from the rest of the ground network, hinders communication [14] by causing longer delays, higher latency, and higher error rates than regular network traffic. Satellites should also be able to handle errors induced by solar radiation or sensors [32]. In this context, a security solution should distinguish benign traffic (with potential transmission errors) from malicious traffic and be as accurate as possible with a very low false positive rate (i.e. a commonly used value is less than 5%). Finally, the entire satellite supply chain should be considered in a *secure by design* approach: from the ground station to the satellite. For example, a poor choice of cryptographic algorithm can leave a satellite vulnerable during the mission.

Monitoring and analyzing satellite data also presents certain challenges. The security solution must be able to handle a significant amount of data (e.g., gigabytes per day) efficiently to respond quickly to an intrusion. Depending on the situation, information from different sources (e.g., different sensors or different satellites) may be analyzed separately or combined to identify anomalies. However, visualizing logs, configuring security rules, and understanding their dependencies is complicated. Even when an attack is detected, it's not always easy to understand its provenance across multiple data sources to confirm the presence of an attack.

Given the importance and complexity of securing satellites, more than one solution is needed to protect them. To ensure the confidentiality of data, it is possible to implement encryption and access control, to this end, some measures could be implemented. These include the use of encryption and access control to ensure data confidentiality, a zero-trust architecture and segmentation of the internal network to limit the propagation of breaches, and the deployment of an Intrusion Detection System (IDS) to provide security at execution. This paper focuses on the IDS as an efficient method of monitoring information such as network logs, Operating System (OS) configuration, or data files. While it usually employs approaches such as signature-based, anomaly-based, and heuristic-based detection against threats, new models based on artificial intelligence have emerged to complement or even replace these methods with more complex and powerful models [13].

2 IDS FOR SATELLITES: OUR VISION

Protecting satellites is challenging; each security solution must be tailored to the satellite's capabilities, mission type, and associated vulnerabilities. In addition, the solution should work on the satellite's specific architecture without causing additional disruption [27]. When designing an intrusion detection system (IDS), it is essential to tailor the system to address different potential attack vectors. Depending on the available resources and needs, the IDS solution can be placed in different locations: ground station, satellite, and network. All of these solutions offer different trade-offs in terms of protection, deployment, and resource usage. This choice should be made carefully; it is advisable to distribute the IDS among the various locations according to the specific requirements. Once the IDS location is decided, the IDS technology should be decided (rule-based, anomaly-based, machine learning, etc.) and an IDS configuration should be defined. Deploying and managing such a configuration is generally challenging; the technology should allow analysts to easily understand, monitor, and troubleshoot the configuration. Finally, the IDS should be updated to handle new attacks and optimized to limit resource consumption while maintaining the same level of security.

To address these choices, our vision is to develop a multi-level decision and feature configuration structure illustrated in Figure 1. This structure is based on three levels of inputs and offers three levels of outputs [31]. The input levels include all the specifics of the mission and the available resources and needs. The first level defines the nature of the mission, the second level outlines the available resources in terms of computing power and budget, and the third level addresses security needs by defining the level of data criticality and acceptable latency for threat detection. Unfortunately, it is not always possible to have all the necessary information when designing the IDS solution. In such cases, artificial intelligence could be used to suggest possible options for the missing information, based on the decisions already made, to select the best configuration for the mission. The output of the configuration structure shows the optimal configuration of IDS features at different levels. The first level specifies how the IDS should be distributed across different locations. The second level delves deeper into the IDS configuration, detailing the type of IDS, its data source, and any preprocessing requirements. The third level focuses on maintenance and update requirements, specifying the types of tasks to be performed and how they should be verified and validated. The following sections present the theories behind the output presented at the last three levels, the IDS characteristics.

The application of our configuration structure is illustrated in Figure 1 using a CubeSat Earth Observation constellation and geostationary satellite communications values for comparison. CubeSats are small satellites with **low computing power** that are often deployed in groups and interact with each other to distribute tasks. In contrast, their associated ground station has more computing power. Although the **security budget is small** in our scenario, the **data is not critical** because it is for research purposes only. The system still requires **near real-time** threat detection and response. This CubeSat scenario will be further explored as a running example in the rest of this paper.

2.1 Identifying the best IDS location

While the location of the intrusion detection system affects the resources required, it also provides access to different data for detecting malicious activity [17]. At the ground station, there is

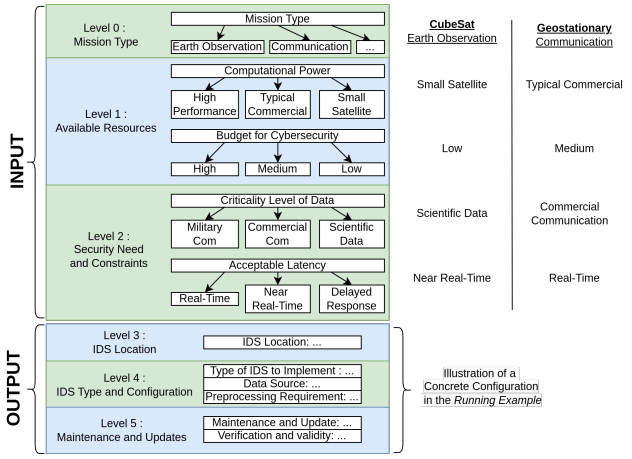


Figure 1: Decision process to identify best IDS solutions

typically access to configuration files, incoming network traffic, and a subset of satellite logs. On the network, all encrypted traffic between the ground station and the satellites can be observed. Finally, the satellite has access to all of its logs and incoming network traffic. The advantages and disadvantages of each IDS location are explored below.

Ground station-based IDS. The ground station does not face computational power limitations as satellites. It has access to OS information (e.g.: user management, file system, running process, etc.) and to a subset of the satellite logs (i.e., only a subset of essential logs is sent back from the satellite to the ground), and has also access to the ground-to-ground network. While the diversity of data available could be challenging to handle, it allows ground stations to have an overview of all activities related to satellite communications. Nevertheless, an attack that targets the satellite hardware or onboard computer directly, without passing through the ground station, will not be detected by an IDS at that location. In addition, if jamming is used to prevent a legitimate signal from being received or transmitted, the IDS will be ineffective.

Network-based IDS. Network-based IDS can be located on a separate entity that is listening to the network or directly on the ground station. It is not typically situated on a satellite, as the cost of monitoring the network from a satellite is often too high. Network-based IDS have been widely studied [18] since monitoring network traffic generally allows the detection of a wide variety of attacks. However, these solutions must be adapted to operate on satellite communication, as they are typically designed for use on the ground network. If there is only an IDS that monitors the network between the satellite and the ground station, any attack that targets one of them directly, without affecting the communication channel, is not detectable. Moreover, an attack based on specific payloads could bypass the IDS if the traffic is encrypted.

Host-based IDS. While the resources available on the satellite are more limited, the IDS has access to all the logs produced by all the satellite's modules and the network logs of communications with other satellites. This provides great defensive capabilities, but the IDS implementation must be highly customized to the satellite's mission and architecture (e.g. adapted to the satellite's bus). In addition, each module generates its own type of log with its syntax. Consequently, efficient preprocessing is required to fully utilize all available information [29]. The preprocessing options available will vary considerably depending on the circumstances [26]. The most significant factor influencing this is the availability of resources on the satellite, with the choice between processing the data directly on the satellite or sending it to a ground station. The range of operations that can be carried out therefore extends from data reduction to identifying significant events, to prepare the data for efficient transmission, storage, or analysis. Unfortunately, if a ground station is compromised, the IDS on the satellites will not be able to detect it and may continue to trust commands from the hijacked ground station. Similarly, in the event of a man-in-the-middle attack, an IDS is incapable of detecting any eavesdropping on the satellite transmissions. If the man-in-the-middle possesses sufficient information and the security measures in place are inadequate, they may even impersonate the ground base without compromising it.

Let us consider our CubeSat running example. Due to the limited budget, the IDS will only be distributed between the ground station and the constellation. The IDS on the ground station will secure both the ground and the network, while the IDS on the constellation will focus solely on securing the satellites. It is possible to choose between having a bit of security on each satellite, or an IDS that is only deployed on a subset of CubeSats. The optimal choice will depend on a multitude of factors, and in this precise running example, the second option is more optimal. These CubeSats will monitor the entire constellation and relay essential security information back to the ground station.

2.2 Representation of the IDS configuration

The configuration of an IDS is generally challenging for analysts as it should generally define a large set of rules to protect against attacks while avoiding false positives. Allowing the analyst to visualize, understand, and modify the IDS configuration is therefore critical to facilitate deployment and troubleshooting when new attacks or false positives are encountered. Without a formal structure, security engineers often struggle to understand the set of rules that trigger the detection of suspicious activity.

We propose to develop an ontology, a formal representation of the IDS information and relations in a graph structure. When a suspicious activity triggers a combination of rules, the ontology allows to easily understand which rules were concerned, interact with the configuration, and apply queries to answer analyst questions. Similarly to Web Application Firewall [1], it could allow to correct errors, vulnerabilities, and logical inconsistencies of the IDS, thanks to the interactions on the ontology to search rule condition and target. To illustrate, the following configuration is represented by the ontology on the right side of Figure 2, where rule 1.1 could allow access to the ground station remotely only if the IP address of the user is from the European Union.

Artificial intelligence IDS models are also an efficient way to detect anomalies, but they should also be interpretable, as ontologies are for rules, for the same reasons. Explainable AI technique [3] and feature reduction solutions [25, 35] could allow to understand

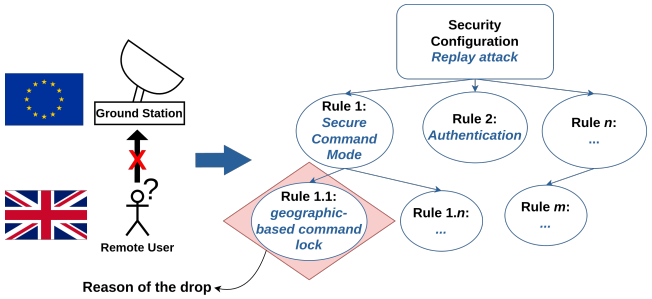


Figure 2: Example of ontology application

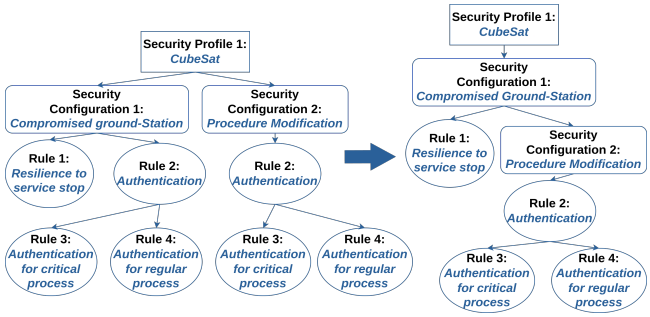


Figure 3: Ontology optimization example for CubeSat

and troubleshoot these IDS. Another possibility is to use models interpretable by design like decision trees. Finally, a hybrid approach using multiple types of IDS can address a range of potential attack vectors [24] by combining several signature-based algorithms that use diverse sources of information, such as OS information and data files, with an anomaly-based IDS that analyzes network logs [4].

In our vision, for each scenario, a list of algorithms and machine learning models to be employed is presented. This is combined with the previous step, which indicates the optimal location for the solution.

Let us consider again our CubeSat running example. On the ground station, a complex rules-based IDS is optimal. When it blocks legitimate communication, the ontology shown in Figure 2, could be queried and analyzed to troubleshoot the problem directly. For example, if an overseas analyst's communication is blocked, it can be directly determined that the rule related to geographic location (i.e., rule 1.1 in the ontology) is responsible. For the CubeSats, simple machine learning anomaly-based models are employed. These models were trained on the regular execution of all mission procedures since regular telemetry procedures rarely deviate from the routine and even simple models can detect anomalies.

2.3 IDS improvements and updates

Since updating an IDS "over the air" after satellite deployment can be challenging, ensuring that it remains efficient and does not introduce new vulnerabilities is essential. For rule-based IDS, the security rules should be cleaned up when they are updated to remove unnecessary ones. The development of an ontology will facilitate the identification and modification of such rules, reducing the complexity of the IDS and the resources that may be wasted on duplicate actions.

Figure 3 illustrates how an ontology can be used to identify redundancy and implement an optimization within the set of rules. While this could be identified by inspecting the ontology, another solution is to apply algorithms to detect redundancy directly to the IDS rules. Consider the following set of rules:

```
1. alert tcp any any -> 192.168.1.0/24 143
(content: "|90C8|/bin/sh"; msg: "Malicious shell!";)
...
// Whitelisted IP range
31. suppress track by_src, ip 192.168.1.0 192.168.1.255
...
56. alert tcp any any -> 192.168.1.0/24 143
(content: "|FFFF|/bin/bash"; msg: "Malicious shell!";)
...
// Specific whitelisted IP
97. suppress track by_src, ip 192.168.1.150
98. suppress track by_src, ip 10.0.0.5
...
```

Both rules 31 and 97 allow the same IP address to be whitelisted, as the IP range 192.168.1.0 to 192.168.1.255 includes 192.168.1.150. To detect this redundancy and simplify the configuration, techniques based on relationship identification [19, 21] could be applied in the rule set. Such approaches allow the discovery of diverse dependencies between rules, such as equality, inclusion, and overlap. This could be used directly to identify the inclusion of rule 97 in rule 31.

When it comes to detecting configuration vulnerabilities after an update, the most common methodologies involve a combination of automated and non-automated testing. This process may begin with an evaluation of the IDS against a pre-defined set of attacks. Alternatively, a dedicated team could evaluate the IDS in a test environment before actual deployment, focusing on the differences between the ontologies of the old and new configurations. Solutions such as Space ISAC [15] also provide great opportunities for specialists to share their knowledge of new attacks. However, machine learning models should be evaluated on new datasets to ensure that they do not face temporal shifts and are still accurate [3]. If possible, new training should be performed and the parameters of the models updated accordingly.

Regarding the CubeSat running example, the signature-based IDS at the ground station contains a rule set based on attacks detected on other CubeSats and is updated when it is possible [10]. After an update, the configuration is examined to detect redundancy. The ontology from Figure 3 represents an IDS designed to provide security against two vulnerabilities: the compromised ground station and the illegal modification of the process. The second security configuration, which is designed to defend against the change of procedure, comprises all of the rules present in the first security configuration and they could therefore be merged. For update verification, a cyberattack on the ground station may be performed by a red team, and the ontology is monitored to detect whether the rules are being triggered correctly.

3 CONCLUSION

We presented an overview of satellite security, highlighting the numerous challenges and constraints encountered in developing a cybersecurity solution. Our vision to tackle these challenges is a methodology for deploying an IDS using the features, mission specifications, resource constraints, budget, etc. Considering the different constraints, these parameters are used in a multi-level structure to discover the optimal configuration to distribute the IDS between ground, network, and satellite. We propose to use an ontology to visualize the IDS configuration, which helps engineers to understand the circumstances that triggered the IDS. Finally, we propose solutions to update the IDS and reduce redundancy in its rule set without compromising efficiency, either by analyzing the ontology or by applying techniques based on relationship identification.

ACKNOWLEDGMENTS

This research is supported by Telespazio.
This research is supported by the Walloon region's CyberExcellence program (grant #2110186).

REFERENCES

[1] Ali Ahmad, Zahid Anwar, Ali Hur, and Hafiz Farooq Ahmad. 2012. Formal reasoning of web application firewall rules through ontological modeling. In *2012 15th International Multitopic Conference (INMIC)*. IEEE, 230–237.
[2] Ayodeji James Akande, Ernest Foo, Zhe Hou, and Qinyi Li. 2023. Cybersecurity for satellite smart critical infrastructure. In *Emerging Smart Technologies for Critical Infrastructure*. Springer, 1–22.
[3] Daniel Arp, Erwin Quiring, Feargus Pendlebury, Alexander Warnecke, Fabio Pierazzi, Christian Wressnegger, Lorenzo Cavallaro, and Konrad Rieck. 2022. Dos and don'ts of machine learning in computer security. In *Proc. of the USENIX Security Symposium*.
[4] Jessa Bekker and Jesse Davis. 2020. Learning from positive and unlabeled data: A survey. *Machine Learning* 109, 4 (2020), 719–760.
[5] Blue Book. 2003. *Space packet protocol*. Technical Report. CCSDS 133.0-B-1.
[6] Nicolò Boschetti, Nathaniel G Gordon, and Gregory Falco. 2022. Space cybersecurity lessons learned from the viasat cyberattack. In *ASCEND 2022*. 4380.
[7] CCSDS. 2019. *SECURITY GUIDE FOR MISSION PLANNERS*. https://public.ccsds.org/Pubs/350x7g2.pdf
[8] Porsche Consulting. 2023. *The Race to Space*. https://www.porsche-consulting.com/sites/default/files/2023-03/race_to_space_c_porsche_consulting_2023.pdf
[9] MITRA ATTCK Contributor. 2024. *MITRE ATTCK®, knowledge base adversary tactics and techniques*. https://attack.mitre.org/
[10] Otman Driouch, Slimane Bah, and Zouhair Guennoun. 2023. Intrusion detection system for CubeSats: a survey. In *2023 International Wireless Communications and Mobile Computing (IWCMC)*. IEEE, 596–601.
[11] ENISA. 2024. *Foresight Cybersecurity Threats For 2030 - Update 2024: Extended report*. https://www.enisa.europa.eu/publications/foresight-cybersecurity-threats-for-2030-update-2024-extended-report
[12] ESA. 2024. *Space Attacks and Countermeasures Engineering Shield (SPACE-SHIELD)*. https://spaceshield.esa.int/
[13] Shilin He, Jieming Zhu, Pinjia He, and Michael R Lyu. 2016. Experience report: System log analysis for anomaly detection. In *2016 IEEE 27th international symposium on software reliability engineering (ISSRE)*. IEEE, 207–218.
[14] Zhé Hóu, Qinyi Li, Ernest Foo, Jin Song Dong, and Paulo de Souza. 2022. A digital twin runtime verification framework for protecting satellites systems from cyber attacks. In *2022 26th International Conference on Engineering of Complex Computer Systems (ICECCS)*. IEEE, 117–122.
[15] ISAC. 2024. *Information sharing and analysis center*. https://spaceisac.org/
[16] Jiahao Jiang, Zhelong Wang, Hongyu Zhao, Sen Qiu, and Jiaxi Li. 2021. Modulation recognition method of satellite communication based on CLDNN model. In *2021 IEEE 30th International Symposium on Industrial Electronics (ISIE)*. IEEE, 1–6.
[17] Roshan Kumar and Deepak Sharma. 2018. Signature-anomaly based intrusion detection algorithm. In *2018 Second International Conference on Electronics, Communication and Aerospace Technology (ICECA)*. IEEE, 836–841.
[18] Satish Kumar, Sunanda Gupta, and Sakshi Arora. 2021. Research trends in network-based intrusion detection systems: A review. *IEEE Access* 9 (2021), 157761–157779.
[19] Magali Legast and Axel Legay. 2023. Assistance in the management of rule sets for rule-based expert systems. In *Proceedings of the 35th International Conference on Software Engineering and Knowledge Engineering*. 367–372.
[20] Kun Li, Huachun Zhou, Zhe Tu, Weilin Wang, and Hongke Zhang. 2020. Distributed network intrusion detection system in satellite-terrestrial integrated networks using federated learning. *IEEE Access* 8 (2020), 214852–214865.
[21] Charles Lohest and Axel Legay. 2024. Improving Security Analysis Rule Set by Relationship Identification. In *Proceedings of the 17th IEEE International Conference on Software Testing, Verification and Validation (ICST 2024)*.
[22] Mark Manulis, C. Bridges, R. Harrison, V. Sekar, and A. Davis. 2021. Cyber security in New Space: Analysis of threats, key enabling technologies and challenges. *International Journal of Information Security* 20 (06 2021). https://doi.org/10.1007/s10207-020-00503-w
[23] Union of Concerned Scientists. 2023. *In-depth details on the 7,560 satellites currently orbiting Earth, including their country of origin, purpose, and other operational details*. https://www.ucsusa.org/resources/satellite-database
[24] Yazan Otoum and Amiya Nayak. 2021. As-ids: Anomaly and signature based ids for the internet of things. *Journal of Network and Systems Management* 29, 3 (2021), 23.
[25] Azar Abid Salih and Maiwan Bahjat Abdulrazaq. 2019. Combining best features selection using three classifiers in intrusion detection system. In *2019 International Conference on Advanced Science and Engineering (ICOASE)*. IEEE, 94–99.
[26] Mohanad Sarhan, Siamak Layeghy, Nour Moustafa, Marcus Gallagher, and Marius Portmann. 2022. Feature extraction for machine learning-based intrusion detection in IoT networks. *Digital Communications and Networks* (2022).
[27] Matthew Scholl, Matthew Scholl, and Theresa Suloway. 2023. *Introduction to cybersecurity for commercial satellite operations*. US Department of Commerce, National Institute of Standards and Technology.
[28] Space Foundation. 2024. *The Space Report 2023 Q4 - Press release*. https://www.spacefoundation.org/2024/01/23/the-space-report-2023-q4/
[29] Zarrin Tasnim Sworna, Zahra Mousavi, and Muhammad Ali Babar. 2023. NLP methods in host-based intrusion detection Systems: A systematic review and future directions. *Journal of Network and Computer Applications* (2023), 103761.
[30] The Aerospace Corporation. 2024. *SPARTA: Space Attack Research and Tactic Analysis*. https://sparta.aerospace.org/
[31] John P Thebarge, Wayne Henry, and Gregory Falco. 2022. Developing scenarios supporting space-based ids. In *ASCEND 2022*. 4219.
[32] Jouko Vankka and Antti Kestilä. 2013. Sun outage calculator for satellite communications. In *2013 IEEE International Conference on Space Science and Communication (IconSpace)*. IEEE, 51–55.
[33] Ly Vessels, Kenneth Heffner, and Daniel Johnson. 2019. Cybersecurity risk assessment for space systems. In *2019 IEEE Space Computing Conference (SCC)*. IEEE, 11–19.
[34] WolfSSL Inc. 2024. *WolfSSL: Embedded SSL library*. https://www.wolfssl.com
[35] Rizgar Zebari, Adnan Abdulazeez, Diyar Zeebaree, Dilovan Zebari, and Jwan Saeed. 2020. A comprehensive review of dimensionality reduction techniques for feature selection and feature extraction. *Journal of Applied Science and Technology Trends* 1, 1 (2020), 56–70.