International Journal of Advances in Computer Science & Its Applications – IJCSIA 2018
Copyright © Institute of Research Engineers and Doctors , SEEK Digital Library
Volume 8 : Issue 2- [ISSN : 2250-3765] - Publication Date: 28 December, 2018

Design of a Traffic Generation Platform for Offline Evaluation of NIDS

P.Velarde-Alvarado, R. Martínez-Pelaez, L. Mena-Camare, A. Ochoa-Brust, and A. Iriarte-Solís

***Abstract*— In this paper, the design of a platform for the generation of NIDS evaluation datasets is proposed. The increasing sophistication of network attacks requires high volumes of high quality and realistic test data that can faithfully represent today's security threats to information systems. Our platform is based on the generation of aggregate traffic through the reproduction of real offline traffic and the injection of anomalous traffic. The use of free software tools in Linux environment is proposed. This solution provides an answer to the lack of datasets suitable for the study of Intrusion Detection Systems.**

***Keywords*— Datasets generation, NIDS, snort, test-bed**

I. Introduction

The field of intrusion detection (ID) has become increasingly important as more sensitive data is stored and processed through networked systems. A Network Intrusion Detection System (NIDS) monitors network traffic for timely alerting malicious activity related to unauthorized attempts to access a network resource. According to the detection technique used, NIDSs can be classified as Signature-based Network Intrusion Detection System (S-NIDS) and Anomaly-based Network Intrusion Detection System, (A-NIDS), [1]. An S-NIDS employs pattern recognition techniques to identify a specific attack. These systems are relatively easy to implement and are capable of detecting known attacks; a drawback is that they are not able to detect zero-day attacks. On the other hand, an A-NIDS does not require prior knowledge of the attacks, which allows it to detect novel attacks. A-NIDSs are based on building a model of typical network behavior called a behavioral profile, any deviation of monitored traffic from the accepted behavioral model is considered an anomaly, which could be related to an attack or any other unusual activity. In this paper, we propose the generation of traffic traces for the training and evaluation of NIDSs.

P. Velarde-Alvarado
Universidad Autónoma de Nayarit

R. Martínez-Peláez
Universidad De La Salle Bajío
México

L. Mena-Camere
Universidad Politécnica de Sinaloa
México

Alberto Manuel Ochoa Brust
Universidad de Colima
Mexico

A. Iriarte-Solis
Universidad Autónoma de Nayarit
Mexico

In the intrusion detection research area, it is essential to have traffic traces or datasets that are suitable for the design and study of NIDSs. There are several public repositories of traffic traces that provide this type of data, such as the MIT-DARPA project from the Massachusetts Institute of Technology [2], the CAIDA project (The Cooperative Association for Internet Data Analysis) [3], the MAWI project (Measurement and Analysis on the Wide/Internet) [4], among others. The benchmark traffic traces of MIT-DARPA, have been artificially generated and used in a large number of research papers, [5], [6]. However, their creation dates back to 1998, which implies a disadvantage since they do not reflect the new trends in security threats facing today's networks. For example, they do not contain instances of network worms, web application attacks like SQL injection and cross-site scripting (XSS), botnets, etc. CAIDA and MAWI are public repositories of real traffic captured from major Internet backbones. However, due to privacy policies, these traffic traces have been modified to hide the identity of the source and destination IP addresses through a process called anonymization, [7]. The problem with anonymization is that it not only removes privacy-sensitive information but also alters information that is important for traffic characterization using traffic feature distribution methods found in traffic packet headers.

Datasets scarcity for network intrusion detection research is then identified. In this paper, we propose the design of a Linux-based platform and the use of free software tools for the generation of NIDS training and evaluation datasets. In this approach, a network will be able to generate its evaluation datasets, thus providing up-to-date data on security threats and avoiding the problem of anonymization, as long as the information is kept within a restricted environment, i.e., under the control of a network administrator. In this research work, a significant volume of traffic captures is available from the campus network of the Autonomous University of Nayarit, which has already been cleaned up through a sanitization procedure, this traffic will act as background traffic. Traffic sanitization is a procedure for filtering out abnormal traffic, [8]. The use of tools based on Information Theory algorithms to detect anomalies in network traffic has recently been proposed, [9], [10]. The background traffic mentioned above has been obtained through the procedure based on the use of Method of the Entropy Spaces (MES), which through a spatial-temporal representation of traffic, allows the identification and filtering of anomalous traffic patterns, [11]. This background traffic is used to generate new datasets using controlled replay and injection of network attacks. This platform will provide new datasets for the evaluation of ID algorithms.

II. NIDS Evaluation Considerations

Once a NIDS is developed it is necessary to evaluate its performance. There are two approaches to evaluating a NIDS: online evaluation and offline evaluation. Online evaluation requires that a real physical network be built, where real users generate both background and attack traffic. The NIDS is then deployed on the network so that its ability to detect malicious activity can be tested by measuring the correct event identification rate and false alarm rate. On the other hand, offline evaluation only requires the existence of traffic traces that represent the traffic that the NIDS would commonly find on a production network. An important issue is that the dataset is labeled in order to assess the classification capability of NIDS. The traffic can then be reproduced to the NIDS to measure its performance based on the correct event identification rate and false alarm rate.

One of the challenges facing intrusion detection research is to create scenarios where researchers can evaluate and improve the proposed ID algorithms through testing and refinement. In this sense, to verify the effectiveness of an ID algorithm, tests can be performed online. However, it is not practical to conduct tests or experiments on an organization's network as network functionality could be compromised. One possible solution is the creation of physical test networks, also called test beds, in which traffic equivalent to an original network is generated. Traffic generation for the test bed should consider diversity factors that affect the characteristics of typical traffic on real networks. The following are listed below:

- **Users**. User behavior determines the periods of network activity and inactivity.
- **Applications**. The type of applications used defines the application protocols, their types and versions of the observed traffic.
- **Operating systems**. Traffic parameters such as time to live (TTL) and initial sequence number (ISN) depend on the operating system used on the network.

The traffic capture is a fundamental step to create the datasets required for research. The study carried out from them can be directed to different topics related to traffic engineering, for example, the modeling of packet loss and delay, the characterization of long-range dependence (LRD), self-similar processes, etc. They can also be used for network security analysis in the field of network intrusion detection, for example, training and evaluation of NIDS.

The use of realistic traffic traces for testing purposes faces several drawbacks. Firstly, these datasets may be difficult or impossible for many researchers to obtain. Sharing this type of data can pose security and privacy issues, or even be prohibited by the organization's policies, [12]. These policies require that the captured traces of an organization have to be modified before they can be shared with the research community. This process is called anonymization. Its objective is to preserve the characteristics of the traffic and at the same time comply with the information privacy policies. Some tools and techniques have been implemented for trace anonymization, such as Crypto-Pan, Anontool, ip2anononip, FLAIM, IP-Anonymous, and TCPdprive. These tools use a variety of algorithms for anonymization, such as black-marker, random permutations, truncation, pseudo-anonymization, and prefix preservation.

III. Network traffic management tools

In this section, the tools used in the proposal are discussed. These tools are organized as follows: injection tools, capture tools, and support tools.

A. Injection tools

The injection tools selected in this paper are hping3, tcpreplay, scapy, and nmap. They require libnet or libpcap libraries to access various protocols. These tools are explained below:

Hping3, [13] is a command level TCP/IP packet analyzer and assembler. It is inspired by the UNIX ping command, although unlike this one, hping3 is not only capable of sending ICMP packets but can also send TCP, UDP, and RAW-IP packets. Other features include: traceroute mode, the ability to send files over an encrypted channel, etc.

Tcpreplay, [14], is a suite of free open source utilities for editing and replaying network traffic captured in the libpcap format.

Scapy, [15], [16] is a packet manipulation tool written in Python. Scapy is capable of sending and decrypting packets for a wide range of protocols, analyzing transmitted packet responses, performing most typical tasks such as scanning, tracerouting, or network discovery.

Nmap (Network Mapper), [17] is a free tool for network auditing and enumeration. With nmap, it is possible to find out the hosts that are active on a network, discover open ports with the corresponding services and software versions, the operating system running, whether a firewall is enabled, etc.

B. Capture and Analysis Tools

In this work, the traffic capture tools used are tcpdump, Wireshark, ipsumpdump, and snort.

Tcpdump, [18] is a command-line tool whose primary utility is to analyze the traffic passing through the network. Allows the user to capture and display in real time the transmitted and received packets arriving at the capture interface.

Wireshark, [19] is a protocol analyzer, whose primary objective is traffic analysis, and is also an excellent educational application for the study of communications and the resolution of network problems. Wireshark implements a wide range of display filters that facilitate the definition of search criteria for the more than 1,100 protocols supported.

Ipsumdump, [20] is a tool that can capture traffic as well as dump specific data from a tcpdump file into a text file. The content of the text file will depend on the options indicated by the user. The available options are as follows:

timestamp, source IP addresses, TCP/UDP port addresses, protocol, payload size, etc.

Snort, [21] is an IDS/IPS, capable of performing real-time traffic analysis and packet authentication over TCP/IP networks. Snort analyzes packet content and can be used to detect a variety of attacks and intrusion activities. Events detected by Snort can be logged in a Syslog or a database such as Mysql or Postgres, for later analysis by a network administrator.

C. *Support tools*

There is a set of support tools that are widely used in network security research projects. Virtualbox was used to deploy virtual machines. Kali provides a set of applications for performing network penetration testing, and Metasploitable2 is an intentionally vulnerable Linux virtual machine. Through virtual machines, victims and attackers are added to the scenario.

IV. Overview of the proposed platform

The design of the traffic generation platform is shown in Figure 1. It consists of a hardware and software infrastructure, which allows the creation of different attack scenarios. It has a virtualization component to generate a variety of operating systems. The corpus of the traces generated through the platform is the result of integrating all the components mentioned, along with a methodology explained as follows.

The main elements that form the platform are:

Attacking hosts, consisting of a physical host and a virtual host. The first is a Fedora Linux Box which has the private address 192.168.10.1 assigned to it, while the virtual host runs Kali over VirtualBox with the IP address 192.168.10.5.

Victim hosts, consisting of a physical and virtual host, running on Fedora and metasploitable 2 with addresses 192.168.10.2 and 192.168.10.6, respectively.

A NIDS, implemented with Snort, its purpose is to monitor the traffic exchanged between attackers and victims to detect malicious traffic. There are two interfaces for traffic sniffing, eth0 and eth1 configured in bonding mode, to create a single invisible logical interface bond0 on the network 192.168.10.0. It is desirable for NIDS to have an additional interface for NIDS management.

Traffic databases. There are three databases used in the platform. The DB_01 database contains traces of malicious traffic that can be replayed to generate aggregate traffic with the attacker-victim traffic. These traces were obtained from repositories or traffic samples from different Internet sites. The DB_02 database contains traffic traces from the campus network, which were previously processed through a sanitation procedure. This traffic is considered anomaly-free, and its replaying forms the background traffic. This simulates larger-scale network scenarios. In each attack test, background and attack traffic are merged to generate a new evaluation trace that will be stored in the database DB_03. The information provided by NIDS allows the identification of attacks perpetrated at the time they occur, the number of malicious packets, duration, among others. This information is relevant as it is used for the labeling of these traces.

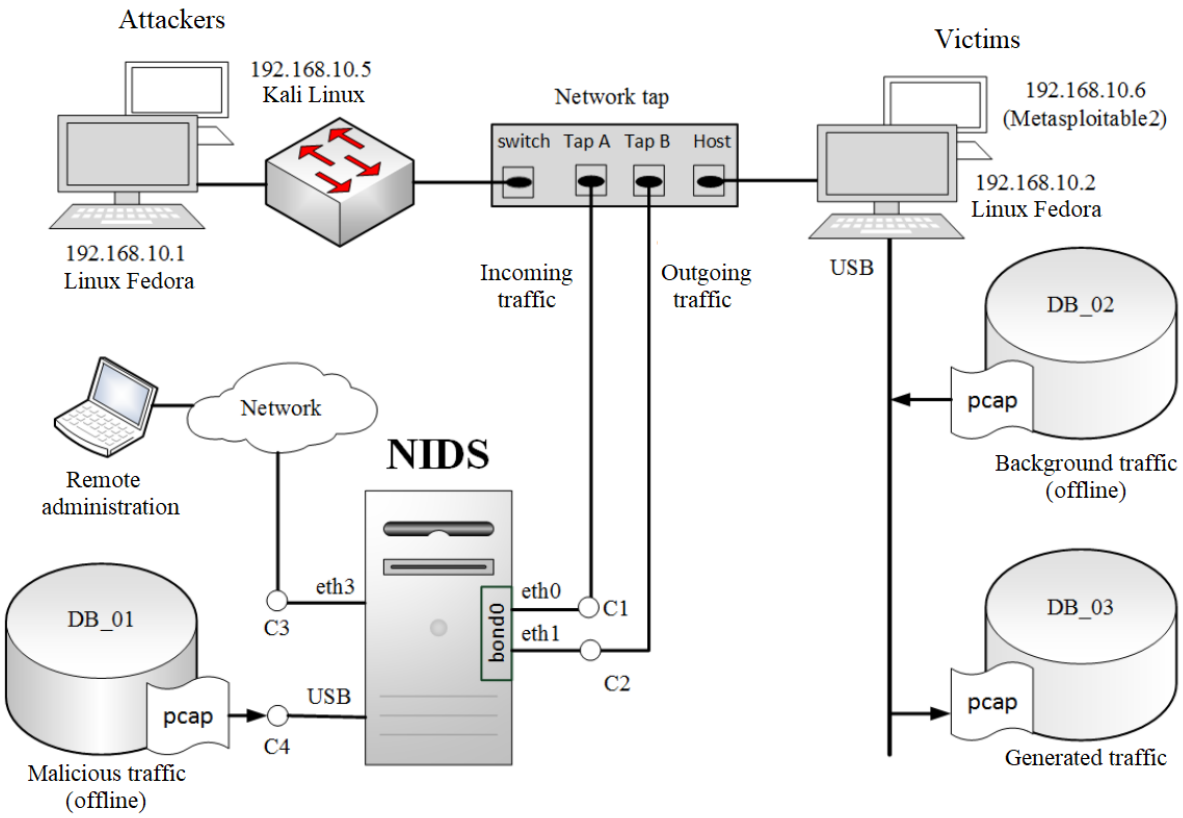

Figure 1. Configuration of the platform for traffic generation

Support components: some elements allow to interconnect the different parts, this is the case of the Network Tap, shown in figure 2. This device is fundamental for traffic capture on the platform. Its purpose is separate the attacker-victim traffic in incoming and outgoing traffic while isolating the segment of the network where the victim is located. It also makes NIDS transparent to both attacker and victim hosts.

Figure 2. Network tap

Additionally, to monitor the traffic through the bond0 interface, ntop was installed. Ntop is a tool that allows real-time control of users and applications that are consuming network resources, [22]. To access the monitoring information provided by ntop, open a Web browser at http://localhost:3000. A screenshot of the ntop web interface is shown in Figure 3.

Snort was configured to listen on the bond0 interface. The snort.conf file defines the network address, rules, and preprocessors. Snort is executed on the command line as follows:

```
#snort -i bond0 -c /etc/snort/snort.conf
```

In this manner, the bond0 interface listens to the traffic flowing in both victim and attacker host directions. As a signature is fulfilled, snort generates an alert.

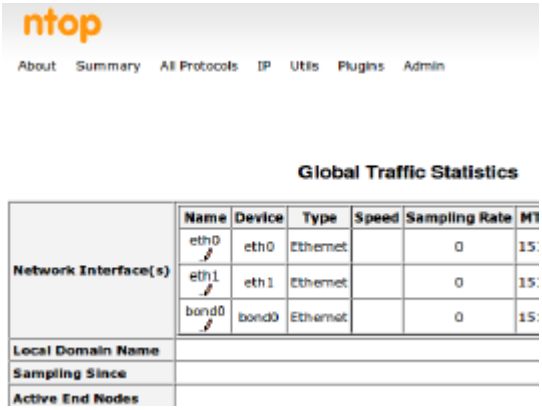

Figure 3. Traffic monitoring in the NIDS using ntop

V. Deploying attacks

Once the configuration of the platform was completed, traffic data is generated through the execution of attacks and the reproduction of malicious traffic and anomaly-free traffic.

Attack set #1. Port scanning attack. nmap was used to implement three scanning techniques: syn, xmas and tcp connect(). To create background traffic, the victim host replayed an anomaly-free traffic trace from DB_02 using tcpreplay. The capture was done using tcpdump in the NIDS. The result was a new traffic trace with the same length of the anomaly-free trace but now mixed with the scanning traffic. This trace was documented and stored in the DB_03 database. The alert generated by snort is shown below:

```
06/24-20:17:42.849648 [**][1:1228:7]SCAN nmap
XMAS[**][Classification:Attempted Information
Leak][Priority:2]{TCP}192.168.10.1:57351 ->
192.168.10.2:1594
```

Attack set #2. Slammer Worm attack. From database DB_01, a sample of traffic containing an instance of the Slammer Worm was taken to be replayed on the platform. Similarly, background traffic was used, and the resulting trace was stored in DB_03. The alert generated by snort is shown below:

```
07/03-17:05:04.602611[**][1:1000001:1] SQL Slammer
Worm [**][Priority:0]{UDP} 213.76.212.22:20199 ->
65.165.167.86:1434
```

Attack set #3. Teardrop. In this set of attacks, tcpreplay was used to inject a traffic trace containing the teardrop attack. The trace consists of 17 traffic packets and is part of DB_01. The alert generated by snort is shown below:

```
07/02-19:54:42.564547 [**][123:2:1](spp_frag3)
Teardrop attack [**][Priority:3]{UDP}10.1.1.1 ->
129.111.30.27
```

Attack set #4. DDoS attack. In this set of attacks, the hping3 tool was used to perform a distributed denial-of-service attack. On the victim host, anomaly-free traffic was played to provide background traffic. The alert generated by snort is shown below:

```
07/03-18:29:00.652674[**][1:504:7] MISC source
port 53 to <1024 [**][Classification: Potentially
Bad Traffic][Priority:2]{TCP}192.168.10.1:53 ->
192.168.10.2:80
```

Attack set #5. Nestea attack. In this set of attacks, scapy was used to implement a sample of traffic with packet fragmentation attack to be reproduced along with the background traffic. The alert generated by snort is shown below:

```
07/03-17:56:06.813614 [**][123:3:1] (spp_frag3)
Short fragment, possible DoS
attempt[**][Priority:3]{UDP} 192.168.10.1 ->
192.168.10.6
```

VI. Conclusions

In this paper, a test bed for traffic generation for the evaluation of NIDSs has been proposed. Through this platform a properly documented database has been created, consisting of five datasets, each dataset contains an anomaly-free traffic trace and a version contaminated with malicious traffic. The attacks considered are network worms, distributed denial-of-service attacks, probing attacks with different scanning techniques, and packet fragmentation attacks. For the documentation of the evaluation traffic databases, ipsumdump was used to generate files in text format with the time stamps and packet sizes. A Matlab script was then written to process the above dump files and generate traffic volume plots. The capinfos tool was also used to summarize the traces with information on duration, number of packets, bytes, etc.

The generated datasets were evaluated with the NIDS snort, allowing to confirm that the attacks integrated into the evaluation traces were detected correctly.

Acknowledgment

We would like to thank CONACyT for its support through its "SEP-CONACyT Ciencia Básica CB-2011" Research funding for Project number 167859.

References

[1] S. a Joshi and V. S. Pimprale, "Network Intrusion Detection System (NIDS) based on Data Mining," *Int. J. Eng. Sci. Innov. Technol.*, vol. 2, no. 1, pp. 95–98, 2013.

[2] "DARPA Intrusion Detection Data Sets," 1998. [Online]. Available: https://www.ll.mit.edu/ideval/data/.

[3] "CAIDA: Center for Applied Internet Data Analysis." [Online]. Available: http://www.caida.org/home/.

[4] "MAWI Working Group Traffic Archive." [Online]. Available: http://mawi.wide.ad.jp/mawi/.

[5] C. Thomas, V. Sharma, and N. Balakrishnan, "Usefulness of DARPA dataset for intrusion detection system evaluation," 2008, vol. 6973, p. 69730G.

[6] J. Sommers, P. Barford, and V. Yegneswaran, "Toward Comprehensive Traffic Generation for Online IDS Evaluation," 2005.

[7] A. Aleroud, Z. Chen, and G. Karabatis, "On the Move to Meaningful Internet Systems: OTM 2016 Workshops," vol. 10034, pp. 934–942, 2017.

[8] G. F. Cretu, A. Stavrou, M. E. Locasto, S. J. Stolfo, and A. D. Keromytis, "Casting out Demons: Sanitizing Training Data for Anomaly Sensors," in *2008 IEEE Symposium on Security and Privacy (sp 2008)*, 2008, pp. 81–95.

[9] Y. Gu, A. McCallum, and D. Towsley, "Detecting anomalies in network traffic using maximum entropy estimation," in *Proceedings of the 5th ACM SIGCOMM conference on Internet measurement*, 2005, pp. 32–32.

[10] Kuai Xu, Zhi-Li Zhang, and S. Bhattacharyya, "Internet Traffic Behavior Profiling for Network Security Monitoring," *IEEE/ACM Trans. Netw.*, vol. 16, no. 6, pp. 1241–1252, Dec. 2008.

[11] P. Velarde-Alvarado, C. Vargas-Rosales, R. Martinez-Pelaez, H. Toral-Cruz, and A. F. Martinez-Herrera, "An unsupervised approach for traffic trace sanitization based on the entropy spaces," *Telecommun. Syst.*, vol. 61, no. 3, 2016.

[12] H. Djidjev, L. Alamos, L. Aleksandrov, and C. Technologies, "Generation of SSH Network Traffic Data for IDS Testbeds," in *6th Workshop on Cyber Security Experimentation and Test*, 2013.

[13] "hping3 | Penetration Testing Tools." [Online]. Available: https://tools.kali.org/information-gathering/hping3.

[14] "Tcpreplay." [Online]. Available: http://tcpreplay.synfin.net/.

[15] "Scapy." [Online]. Available: http://www.secdev.org/projects/scapy/.

[16] B. Burns, *Security power tools*. O'Reilly, 2007.

[17] P. Calderon, *Nmap : network exploration and security auditing cookbook*, Second Edi. Packt, 2017.

[18] J. A. (Joshua A. Davies, *Implementing SSL/TLS using cryptography and PKI*. Wiley, 2011.

[19] C. Sanders, *Practical Packet Analysis, 3rd Edition*. No Starch Press, 2017.

[20] "Ipsumdump and Ipaggcreate." [Online]. Available: http://www.read.seas.harvard.edu/~kohler/ipsumdump/.

[21] "Snort - Network Intrusion Detection and Prevention System." [Online]. Available: https://www.snort.org/.

[22] "ntop – High Performance Network Monitoring Solutions based on Open Source and Commodity Hardware." [Online]. Available: https://www.ntop.org/.

Currently, Dr. Velarde is a research professor at the Academic Unit of Basic Sciences and Engineering of the Universidad Autónoma de Nayarit. He received his Bachelor's degree in electronics engineering from the Universidad Autónoma de Guadalajara in 1993. Master's and doctoral degrees in science from the Centro de Investigación y Estudios Avanzados del IPN (CINVESTAV-IPN) in 2001 and 2009, respectively. Dr. Velarde is a researcher of the Sistema Nacional de Investigadores (SNI). His main lines of research are related to IP traffic modeling and the design of entropy-based statistical models for detection systems.

Rafael Martínez Peláez holds a Ph.D. from the Polytechnic University of Catalonia and a Computer Systems Engineering Bachelor's degree from the Universidad del Valle de México in 2010 and 2003, respectively. Currently, he is a full-time research professor at the University of La Salle Bajío and a member of the National System of Researchers (S.N.I.). He is co-author of more than forty scientific papers published in journals and conferences. His areas of interest are authentication, security in electronic services, and privacy in social networks.

Luis J. Mena Camere holds a bachelor's degree in computer science and a master's degree in applied computer science from the Universidad del Zulia, Venezuela. He later obtained his doctorate degree in computer science from the Instituto Nacional de Astrofísica, Óptica y Electrónica, México. He is currently a full-time researcher-professor in the academic programme of computer engineering and leader of the consolidated academic group "Information Technologies and Applied Communications." He is a National Researcher of the Sistema Nacional de Investigadores and Honorary Researcher of the Sistema Sinaloense de Investigadores y Tecnólogos. Among his principal scientific achievements are the development of new algorithms to measure blood pressure variability and to extract patterns from unbalanced data sets. He has also published more than 40 peer-reviewed articles in indexed journals and proceedings of prestigious national and international conferences, and his research interests include data mining for medical diagnosis and prognosis and the development of mobile applications for personal health monitoring.

Alberto Ochoa received a doctoral degree in Electronics. He is a professor and researcher at the Electronics Department of the Universidad de Colima. His academic interests include electronic and sensorial systems for mobile robots, ultrasonic signal processing, embedded systems and computing architectures for local positioning systems.

Adalberto Iriarte Solís is a Doctor of Education with a specialization in Instructional Technology and Distance Education at Nova Southeastern University (NSU) in Miami, Florida. His lines of research include the development of research projects in infrastructure security and the administration of computer servers, as well as the use of mobile devices and the development of virtual and augmented reality applications. He is responsible for the area of Administration of Servers and Security in Computing in the Directorate-General for Academic Infrastructure.