ROS - Security risk?

James Gosling, Matthias Rottensteiner, Muhammet Yildiz, Olivér Jakab, Clemens Hausegger, Alexander Müllner
Technical secondary college
Department of computer science
2700 Wiener Neustadt Austria
Corresponding author email: gosling.james@student.htlwrn.ac.at

***Abstract*—In recent years, there has been an explosion of new techniques concerning the programming of robots, hence also the sudden influx of popularity, especially among younger people. Unfortunately, wherever there is progress, there are also dangers to be found. Therefore, the need for explaining the security risks of robot operating systems, or in our case more specifically, the ROS, (an abbreviation for Robot operating system) arises. In this paper, we will explore the numerous problems that seem to plague this OS, and some solutions.**

***Index Terms*—robotics, cybersecurity, ROS, hacking attacks**

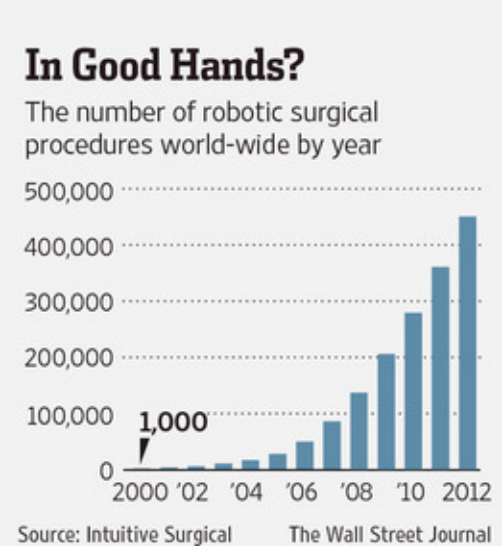


Figure 1. The number of robot surgeries every year up to 2012 - we trust robots with our very lives

I. Introduction

The most used operating system for robots, is called ROS . Unfortunately tough, it isn't very secure. ROS started as a a personal project of two Stanford students, Keenan Wyrobek and Eric Berger in 2006. [1] It was originally called the Stanford personal robotics program. In 2008 the first robot running with this OS was made, and the name was changed to the curent form. 2009 marks the advent of the first public ROS Distribution, named "Mango Tango". The eccentric names would be here to stay. From thereon, every year, one or two distros would be published, including the alphas and betas of its successor, until finally, in 2018 we got to see the full first version of ROS2. [2] When this paper refers to ROS, it will only mean ROS1, unless otherwise stated. Although ROS2 is by a wide margin better than ROS1 in most aspects, it is still not industry standard, and therefore there isn't enough data or significant users to accurately and justify a report on it. [3]

Our topic revolves around the security risks of robots and software exploits. Most modern robots are connected to the internet, so numerous risks arise. [4] Robots are often seen as a typical example of the pinnacle of cybersecurity, but nonetheless they are not perfect. Today, the typical robot usually includes the following components: (a) a control system, and (b) the physical components, e.g: Sensors, Cameras and similar things, that enable it to navigate and move. [5] Additionally, it is usually equipped with (c) networking elements. Of these three, c (we will henceforth call the networking elements), is the most prone to attacks, as there is often valuable data stored on connected systems. Networking elements tend to get attacked the most as one isn't required to get a hold of the actual hardware.Hardware is a very important part of cybersecurity, however we are only going to mention hardware in little detail as its out of the scope of this paper.

In 2013, a group of young computer scientists in Spain conducted a series of experiments regarding the state of cybersecurity in robots. [6] As a result of their studies, more people started to become aware of the problems that this popular operating system faced. Many people started proposing ideas for defensive mechanisms, most notably the work "Enhancing security in ROS: Advanced computing and systems for security" [7]

Figure 2. The Turtlebot 4 - a ROS2 flagship robot

II. Data

In 2021, there was a study, in which 93% of participants stated that they thought their robots could be hacked. [6] The (for us,) unexpected thing about this was, that only 48% of respondents admitted to having taken measures to protect their robots. Also, at least 77% stated that they controlled at least 1 robot that was running a distro of ROS, with 13% even answering that they controlled over 50 robots with it. All of this naturally leads us to the question: Are Robots really that insecure? To answer that, we must first delve a little bit deeper into how robots work. To do this, we will shortly show you two graphics.

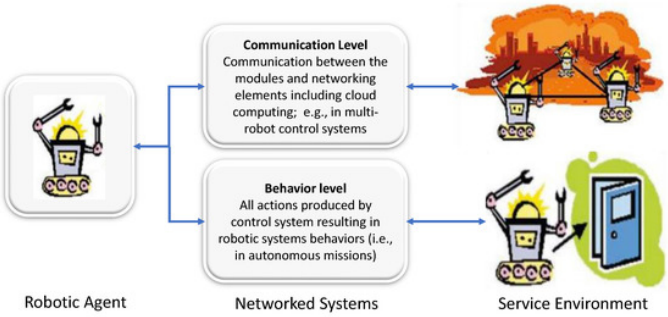


Figure 3. Description of the communication- and the behaviour level of robots.

[1] Figure 3 gives a good, if incomplete overview over how robots communicate. [2]To elaborate on this, imagine a scenario, where a robot is being controlled remotely. First, a robot would have to "communicate" with the remote control. This would be the communication level. But the Robot actually processing the data and bringing itself to move would fall into the behaviour category. Of the two levels shown above, Communication level is where cyberattacks occur. The Application-layer security column seems daunting at first, but it actually means that a dedicated Node Authentication Server is being used, instead of ROS' default node authentication method. According to literature, Robots are now evolving towards the IoE (the Internet of everything), with cloud services becoming much more prominent. [9] This in turn means, that through becoming more connected, more and more threats appear with every passing minute. Now, this Internet of everything seems to be a big security hurdle, however, there have been a lot of new security protocols that have been set in place to prevent any form of robot hacking from happening. But recent experiments that show that this isn't enough though, like for example the case of the staged attack on a robot named DoRIS with ARP Poisoning. [10] ARP Poisining is basically the act of sending an answer to a question meant for another device in the same network. For a more in-depth explanation, please read: [11]. However, as the following table shows, its possible to make ROS more secure by customising the communication-channel (and the application-layer security). [12]

	Effect		
Attacker action	**Unmodified ROS**	**Application-layer security**	**Secure communication channel**
Subscribe	Subscription is performed	Subscription is performed but message contents cannot be interpreted	Without valid certificate, communication is cancelled due to failed (D)TLS handshake.
Publish	Data is received and interpreted	Message is received but will be rejected due to invalid signature	Subscriber will not perform subscription since (D)TLS handshake fails
Consume service	Service can be called normally	Service result cannot be consumed.	(D)TLS handshake fails.
Advertise service	Service is consumed regularly	Result cannot be consumed by other nodes.	(D)TLS handshake fails.
Unauthorized access to topic (with valid certificate)	Access granted	Topic key not transferred by AS.	Communication canceled due to access to unauthorized topic.

Figure 4. Comparison of reaction to attacks between unmodified ROS, application-level secured ROS and ROS with secure communication channel.

[3] This is achieved by using two very important protocols: the Transport layer security(TLS) and the Datagram Transport layer security(DTLS). Additionally, authorization for each node (a process which performs computation) is performed on a per-topic (Topics are buses over which nodes exchange messages) basis. If we compare it with the components mentioned at the beginning, the application-layer would be equivalent to b, and the communication-layer would be c.xFor further in-depth explanation, we advise [12]. So, all isn't as bleak as is it is made out to be. This leads us neatly on to the results:

III. Results

Everything we have discussed up to now, leads us to think, that ROS has a very serious security problem. But, as already mentioned in the Introduction, there is a light at the end of the tunnel: ROS2 has now been out for a couple of years, and it's shaping solve most of the problems that the original ROS had. (Except for the horrible advertisement.). The official docs provide an overseeable list of security changes, one of the biggest of which is the change to private keys. Here follows a quote from the documentation :

> "The identity and permissions certificates also have associated private key files. Add new enclaves to the domain by signing their Certificate Signing Request (CSR) with the identity certificate's private key. Similarly, grant permissions for a new enclave by signing a permissions XML document with the permission certificate's private key."

It now looks like ROS has a rather bright future ahead of it. But, let us now look at the ethical ramifications, that this has.

[1]This image belongs to [5]

[2]If you want to go into more detail on the topic of robot communication specifically, we advise [8]

[3]This table belongs to [12]

IV. Discussion

The ethical implications of robot access control are complex and wide-ranging, as robots play an increasingly important role in many areas of society, such as every individuals home, healthcare, manufacturing etc. Here we will talk about three very important topics that need to be talked about, these will be privacy, autonomy and transparency and biases.

A. Privacy

Robots often collect and process sensitive personal information, and access control mechanisms must be in place to protect this information from unauthorized access or misuse as no one wants their sensitive information in untrustworthy hands. This may include information about your house, collected by a cleaning robot or voice recordings of some smart home devices. [13]

B. Autonomy and transparency

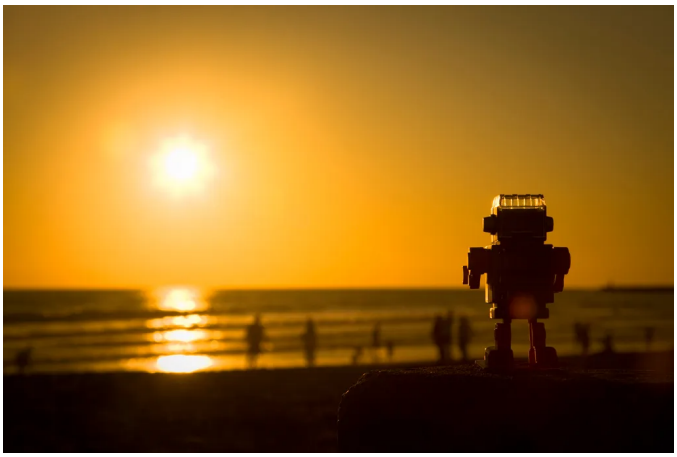
In many different industries every year more processes are automated via machines. Some may even be able to make decisions independently, these robots particular need access control mechanisms in place to ensure that these decisions align with ethical and legal guidelines set by the programmer(s) . With the ever increasing complexity of decisions made by said robots it can be difficult or near impossible to explain every result we see or pr edict what a robot may do in a given situation. This again highlights the importance of robot access control as social and ethical consequences are near impossible to predict we have a need for more transparency and accountability in the design and deployment of robots. It is crucial for researchers, developers, and policy makers to work together to ensure that robots are developed and used in ways that align with ethical principles and benefit society as a whole.

C. Bias

Robots may be programmed with, or have a dataset that is not neutral that results in biases that can have negative effects on people and communities. For example, the now very popular large language model "ChatGPT" by OpenAI is known to have some biases, these may be the result of non neutral training data

V. Conclusion

Now that we have discussed all these points, there remain still one question: Are Robots secure enough to trust them like we do ever day? They do a whole lot of things nowadays, and a lot of people are scared. As we previously described, Robots are prone to cyber attacks, and in the Discussion we also stated, that even the makers of robots could use them to malicious ends. Simultaneously, the last couple of years have been amazing years. There have been so many improvements in cybersecurity in general, that you couldn't list them all in the space provided for this paper a hundredfold. But then, you may find yourself asking, where is all this leading? What is the conclusion to be reached? And, quite frank, it's hard to decide. On the one hand, yes, robots out of the boy aren't very secure, but many people who work with them customise them somehow, to make them much safer. That's why our closing statement will be: Robot's are safe enough for even the most serious cases with the need for more than perfect precision. You just have to know how they work.



VI. Acknowledgements

We thank our professor, Michael Stifter, and our seniors for supporting us through our first year of robotics and giving us advice on how to participate in the PriaOpen.

References

[1] R. Tellez, "A history of ros." https://www.theconstructsim.com/history-ros/, 2019.

[2] C. Bedard, "Distributions." https://docs.ros.org/en/rolling/Releases.html, 2023.

[3] Y. Maruyama, S. Kato, and T. Azumi, "Exploring the performance of ros2." https://dl.acm.org/doi/pdf/10.1145/2968478.2968502, 2016.

[4] S. Malenkovich, "Industrieroboter hacken." https://www.kaspersky.de/blog/hacking-industrial-robots/14314/, 2017.

[5] V. Dutt and T. Zielińska, "Cybersecurity of robotic systems: Leading challenges and robotic system design methodology." https://www.mdpi.com/2079-9292/10/22/2850, 2021.

[6] "Robot cybersecurity, a review." https://conceptechint.net/index.php/CFATI/article/view/41/16, 2021.

[7] C. G., W. R., and C. A, *Advanced Computing and Systems for Security Volume Eight*, ch. Enhancing security in ROS: Advanced computing and systems for security. Springerlink, 2019.

[8] T. Z. Vibekananda Dutta, "Networking technologies for robotic applications." https://arxiv.org/ftp/arxiv/papers/1505/1505.07593.pdf, 2015.

[9] V. Authors, "Internet of robotic things intelligent connectivity and platforms." https://www.frontiersin.org/articles/10.3389/frobt.2020.00104/full, 2020.

[10] R. R. Teixeira, I. P. Maurell, and P. L. J. Drews-Jr, "Security on ros: analyzing and exploiting vulnerabilities of ros-based systems." https://ieeexplore.ieee.org/document/9307107, 2020.

[11] R. Grimmick, "Arp poisoning: What it is & how to prevent arp spoofing attacks." https://www.varonis.com/blog/arp-poisoning, 2021.

[12] B. Dieber, B. Breiling, and S. Kacianka, "Security on ros: analyzing and exploiting vulnerabilities of ros-based systems." https://www.researchgate.net/publication/320368007_Security_for_the_Robot_Operating_System, 2017.

[13] M. Astor, "Your roomba may be mapping your home, collecting data that could be shared." https://www.nytimes.com/2017/07/25/technology/roomba-irobot-data-privacy.html, 2017.