Scanning the Internet for ROS: A View of Security in Robotics Research

Nicholas DeMarinis, Stefanie Tellex, Vasileios Kemerlis, George Konidaris, Rodrigo Fonseca
Computer Science Department
Brown University
Providence, Rhode Island, 02912
Email: {ndemarin, stefie10, vpk, gdk, rfonseca}@cs.brown.edu

I. Abstract

Security is particularly important in robotics platforms. A robot can sense the physical world using sensors, or directly change the physical world with its actuators. Thus, a robot can leak sensitive information about its environment if accessed by an unauthorized party, or even cause physical harm if operated unsafely. As robots become more common in our daily lives, these concerns become more important.

Security issues in robotic platforms have been studied in existing work, particularly in home and industrial environments. Vulnerabilities have been identified in various robotic platforms for use in home [1, 3] and research [7] environments, including weak or non-existent authentication procedures that allow an attacker to control robots, perform firmware updates, or access sensor data. Broadly, these issues seem to stem from lack of consideration in the platform design. Quarta et al. [8] and Maggi et al. [6] surveyed domain experts from both academia and industry, and found that 30% had robots accessible from the Internet, while 76% had never performed a professional cybersecurity assessment. They also used Internet search engines like Shodan [11] to identify industrial robotic devices exposed to the public Internet, identifying 28 industrial robots and thousands of "industrial routers" that enable remote access to devices.

Our goal is to add to this conversation by investigating the state of security in robotics research platforms deployed in practice, which have not been measured by previous works. Specifically, we conducted several Internet-wide scans to identify hosts using the Robot Operating System (ROS) [9], a widely-used platform in robotics research. ROS operates as a publish-subscribe service to distribute data among *nodes* in a system. Nodes publish or subscribe to *topics* by advertising or querying a central master node to send or receive data. Like many research platforms, ROS was not designed for security: the ROS master node trusts all nodes that connect to it, and thus should not be exposed to the public Internet. While several emerging approaches can provide authentication and authorization mechanisms to ROS, including SROS [12], Rosbridge [2], and ROS2 [10], none appear to be widely adopted in practice at this time.

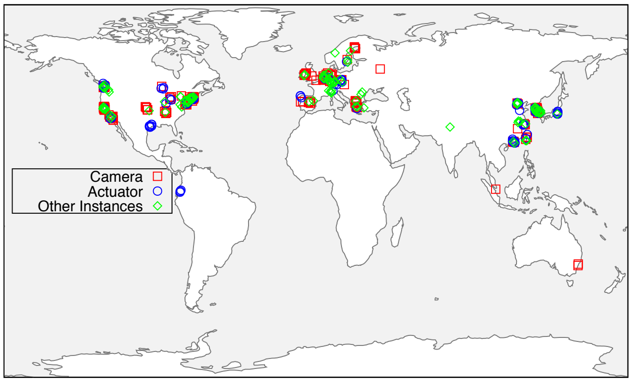

Fig. 1: Locations (slightly jittered to show multiple points) of identified ROS masters across all scans. Red indicates a host that showed evidence of publishing camera information. Blue indicates a host that showed evidence of a robot that could be actuated. Other hosts are in green.

We searched for ROS masters connected to the IPv4 Internet address space by performing a scan of all public addresses (roughly 3.7 billion IPs) on port 11311, the default ROS master port. Our scans were performed using ZMap [4], a research tool for performing Internet-wide port scans, as well as custom tools to identify ROS hosts and query available topics. While port scans are very common on the public Internet, conducting Internet-wide scans poses some inherent risks. When designing our scanning framework, we made efforts to minimize potential disruptions by sending probe packets at a low rate and using a series of minimally-invasive probes to confirm the presence of a ROS master before collecting host information. Critically, sending commands to active robots may pose a safety hazard. We selected a minimal set of *passive* commands designed to confirm that the host was in fact running ROS, and gather data on the topics and parameters available on each ROS instance. *At no time did we attempt to modify the state of the ROS master, or connect to any nodes.*

We conducted three scans on the ROS master port between December 2017 and January 2018. We refer to these scans as Master 1–3, respectively. Each ROS master scan observed over 100 ROS instances, spanning 28 countries, with over 70% of the observed instances using addresses belonging to various university networks or research institutions. A map of all hosts observed is shown in Figure 1. We performed one scan for Rosbridge instances in November 2017 and identified 15 total

TABLE I: Scan results summary.

Category	**Master 1**	**Master 2**	**Master 3**	**Rosbridge**
Identified robots	19	13	12	4
Simulation only	37	32	21	2
Empty ROS cores	37	29	26	0
Only sensors	24	28	18	2
Only actuators	2	1	3	0
Only identified services	11	8	12	6
Unclassified	14	11	10	1
Total ROS Instances	**144**	**122**	**102**	**15**

instances, with 11 instances located in networks recognizable as cloud service providers.

We present both a quantitative and qualitative overview of our findings. Quantitatively, we assessed the number of topics that appear to be types of sensors and actuators. A number of topics we observed indicated ROS hosts running simulators such as Gazebo [5], while others appeared to be connected to real sensors and actuators. We observed a few very common types, but a long tail of one-off sensors and actuators.

Table I provides a summary of our results, organized into types based on their topic data. We define a simulator to be a host that showed evidence of a topic consistent with a simulator. We define a robot as a host that shows evidence of a sensor and an actuator. Each type is mutually exclusive, so identified sensors, actuators, and robots, in this table, did not show evidence of being a simulator. These results must be taken as approximate, since we did not actually subscribe to any of the topics, consider their type, or verify connectivity with real hardware. However, given the standardization of topic names, it seems likely that many of the hosts we found were indeed running sensors, actuators, and other software. Empty ROS cores showed only the base topics and services for a master with no currently-connected nodes. Unclassified nodes did not fit into any of our other categories. We do not combine the results of each scan to form a “grand total” for each type, as many hosts appeared in more than one scan and returned different topic data each time.

Qualitatively, we present case studies of several types of robots we identified as well as commonly-observed types of sensors, actuators, and libraries. We also present a proof-of-concept “takeover” of one of the robots we identified from a research group in a US university.[1] With the consent of its owner, we were able to read image sensor data and move the robot in order to demonstrate the potential capabilities of an attacker accessing open ROS master.

Overall, our results identified that a number of hosts supporting ROS are exposed to the public Internet, thereby allowing anyone to access robotic sensors and actuators. Our goal is not to single out any researchers or robot platforms, but to promote security as an important consideration—not just in production systems, but in research settings as well. Instead, our aim is to provide information about a concerning situation and guidance about how the robotics community can improve their security.

Note that before the workshop date, we intend to reach out to the owners of all affected robots and provide them with a summary of our findings.

II. Acknowledgements

We would like to thank Sidd Srinivasa and Youngsun Kim for their help with our proof-of-concept hack.

References

[1] Cesar Cerrudo and Lucas Apa. Hacking robots before skynet. In *PacSec Applied Security Conference*, 2017.

[2] Christopher Crick, Graylin Jay, Sarah Osentoski, and Odest Chadwicke Jenkins. Ros and rosbridge: Roboticists out of the loop. In *Proceedings of the seventh annual ACM/IEEE international conference on Human-Robot Interaction*, pages 493–494. ACM, 2012.

[3] Tamara Denning, Cynthia Matuszek, Karl Koscher, Joshua R Smith, and Tadayoshi Kohno. A spotlight on security and privacy risks with future household robots: attacks and lessons. In *Proceedings of the 11th international conference on Ubiquitous computing*, pages 105–114. ACM, 2009.

[4] Zakir Durumeric, Eric Wustrow, and J Alex Halderman. Zmap: Fast internet-wide scanning and its security applications. In *USENIX Security Symposium*, volume 8, pages 47–53, 2013.

[5] Nathan Koenig and Andrew Howard. Design and use paradigms for gazebo, an open-source multi-robot simulator. In *Intelligent Robots and Systems, 2004.(IROS 2004). Proceedings. 2004 IEEE/RSJ International Conference on*, volume 3, pages 2149–2154. IEEE, 2004.

[6] Federico Maggi, Davide Quarta, Marcello Pogliani, Mario Polino, Andrea M Zanchettin, and Stefano Zanero. Rogue robots: Testing the limits of an industrial robots security. Technical report, Technical report, Trend Micro, Politecnico di Milano, 2017.

[7] Jarrod McClean, Christopher Stull, Charles Farrar, and David Mascareñas. A preliminary cyber-physical security assessment of the robot operating system (ros). *SPIE Defense, Security, and Sensing*, 8741:874110, 2013.

[8] Davide Quarta, Marcello Pogliani, Mario Polino, Federico Maggi, Andrea Maria Zanchettin, and Stefano Zanero. An experimental security analysis of an industrial robot controller. In *Security and Privacy (SP), 2017 IEEE Symposium on*, pages 268–286. IEEE, 2017.

[9] Morgan Quigley, Ken Conley, Brian Gerkey, Josh Faust, Tully Foote, Jeremy Leibs, Rob Wheeler, and Andrew Y Ng. Ros: an open-source robot operating system. In *ICRA workshop on open source software*, volume 3, page 5. Kobe, 2009.

[10] ros2. ros2, Accessed: January 19, 2018. https://github.com/ros2/ros2/wiki.

[11] Shodan. Shodan, Accessed: January 20, 2018. https://www.shodan.io/.

[12] SROS. Secure robot operating system, Accessed: January 20, 2018. http://wiki.ros.org/SROS.

[1] https://www.youtube.com/watch?v=haQXGn_wOd4