frontiers in Robotics and AI

OPINION
published: 29 September 2015
doi: 10.3389/frobt.2015.00023

Cryptobotics: why robots need cyber safety

Santiago Morante, Juan G. Victores and Carlos Balaguer*

Robotics Lab, Automation and Engineering Systems Department, Universidad Carlos III de Madrid, Madrid, Spain

Keywords: cryptography, robotics, cyber safety, communications, cyber-physical, cryptobotics, cyber security

Introduction

With the expected introduction of robots into our daily lives, providing mechanisms to avoid undesired attacks and exploits in robot communication software is becoming increasingly required. Just as during the beginnings of the computer age (Pfleeger and Pfleeger, 2002), robotics is established in a "happy naivety," where security rules against external attacks are not adopted, assuming that robotics knowledgeable people are well intended. While this may have been true in the past, the mass adoption of robots will increase the possibilities of attacks. This fact is especially relevant in defense, medical and other critical fields involving humans, where tampering can result in serious bodily harm and/or privacy invasions. For these reasons, we consider that researchers and industry should deploy efforts in cyber safety and acquire good practices when developing and distributing robot software. We propose the term *Cryptobotics* as a unifying term for research and applications of computer and microcontrollers' security measures in robotics.

Stating the Problem

The problems that the field of robotics will face are similar to those the computer revolution faced with the widespread of the Internet 30 years ago. Among the common attacks computers may suffer, there are: denial-of-service, eavesdropping, spoofing, tampering, privilege escalation, or information disclosure for instance. To these problems, robots add the additional factor of physical interaction. While taking the control of a desktop computer or a server may result in loss of information (with its associated costs), taking the control of a robot may endanger whatever or whoever is near.

As robots become more integrated on the communications networks, it seems appropriate to reuse the tools designed for web applications in order to controls the robots. However, the authors consider there are differences between regular computers communicating through the network, and robots performing the same actions. Mohanarajah et al. (2015) states differences between web and robotic applications: "Web applications are typically stateless, single processes that use a request-response model to talk to the client. Meanwhile, robotic applications are stateful, multiprocessed, and require a bidirectional communication with the client. These fundamental differences may lead to different tradeoffs and design choices and may ultimately result in different software solutions for web and robotics applications." To these differences, we could also add the real-time constraints that characterize robotics applications. Despite other sources of issues, like software bugs or vulnerabilities [buffer overflow, command injection, etc. (Tanenbaum and Bos, 2014)], we consider that communications currently are one of the main vulnerabilities in robotics.

A number of fields in robotics where security and privacy are particularly relevant can be addressed.

- Defense and Space: The military field should be very aware of the best practices in cyber security to be followed regarding its robots. Unmanned aerial vehicles, commonly called "drones," are being destined to surveillance and also to combat missions. Common sense dictates that any communications with these vehicles should be encrypted (Javaid et al., 2012), but reality shows us

OPEN ACCESS

Edited by:
Lorenzo Natale,
Istituto Italiano di Tecnologia, Italy

Reviewed by:
Fulvio Mastrogiovanni,
University of Genoa, Italy
Emanuele Ruffaldi,
Scuola Superiore Sant'Anna, Italy
Ali Paikan,
Istituto Italiano di Tecnologia, Italy

****Correspondence:***
Santiago Morante
smorante@ing.uc3m.es

Specialty section:
This article was submitted to Humanoid Robotics, a section of the journal Frontiers in Robotics and AI

Received: *04 June 2015*
Accepted: *11 September 2015*
Published: *29 September 2015*

Citation:
Morante S, Victores JG and Balaguer C (2015) Cryptobotics: why robots need cyber safety. Front. Robot. AI 2:23. doi: 10.3389/frobt.2015.00023

differently. For example, in the year 2012 it was reported that only between 30 and 50 percent of America's Predators and Reapers (two of the most used drones in US) were using fully encrypted transmissions.[1]

> *Situation: a non-authorized entity eavesdrops surveillance images of drones, takes its control, exploiting a non-encrypted connection, and crashes it into a populated area.*
> *Situation: a non-authorized entity takes control of a robot inside International Space Station and sabotages an ongoing experiment.*

- Telemedicine and Remote surgery: This exciting field can make remote surgery become an everyday reality, where experts can operate patients from the other side of the world. While this is beneficial to society, we must consider the potential dangers. In 2009, the Interoperable Telesurgery Protocol (ITP) (King et al., 2009) was proposed as a preliminary specification for interoperability among robotic telesurgery systems. Recently, the fact that ITP does not use any form of encryption or authentication was discovered.[2] This is an obvious system exposure to exploits using a man-in-the-middle attack for taking control of the robot (Bonaci et al., 2015).

  > *Situation: a non-authorized entity takes control of a surgery robot during an operation, endangering the life of the patient.*

- Household robots: This market is growing both in research and commercially available robots. Robots will be used as assistants at home. For instance, one of these projects is Care-O-bot (Hans et al., 2002), a robotic assistant in homes. In one of the available versions, this robot is equipped with microphones, cameras and 3D sensors. This set of sensors can collect a huge amount of information, which must be protected (Denning et al., 2009). Service robots may one day also collect data about the health status of a person; law regulations require that this data is handled with extra care.

  > *Situation: a non-authorized entity takes control of a household robot and obtains streams of images with private data.*

- Disaster robots: Since the Fukushima Daiichi nuclear disaster in 2011, the robotics community has increased its efforts to build and deploy robots for disaster scenarios. One of the expected tasks these robots will have to face in a disaster scenario is related to accessing and repairing/disconnecting dangerous systems. Due to the potential danger that may arise in these situations (Vuong et al., 2014), robots should not be able to be externally modified by an external attack.

  > *Situation: a non-authorized entity takes control of a robot deployed to disconnect a nuclear platform that may suffer a partial meltdown, and can thwart the disconnection operation.*

Current State of Security in Mainstream Robotic Software

Robots are a combination of mechanical structures, sensors, actuators, and computer software that manages and controls these devices. Mainstream practices in robotics involve component-based software engineering. Each component is designed as an individual computer program (e.g., a motor moving program) which communicates with other components using predefined protocols. While a large quantity of software libraries for communication already exist, the robotics community has developed a number of "software architectures." Currently, one of the most popular robotics-oriented architecture is ROS (Robot Operating System) (Quigley et al., 2009). Another co-existing architecture is YARP (Yet Another Robot Platform) (Metta et al., 2006). Both systems work similarly: a system built using ROS or YARP consists of a number of programs (nodes or modules), potentially on several different hosts, connected in a peer-to-peer topology.

According to ROS documentation[3]: "Topics are named buses over which nodes exchange messages. Topics have anonymous publish/subscribe semantics (.) In general, nodes are not aware of who they are communicating with." From the point of view of security, this anonymous communication scheme is a welcome sign toward exploits (McClean et al., 2013). Messages are sent unencrypted through TCP/IP or UDP/IP. The default check performed is an initial MD5 sum of the message structure, a mechanism used to assure the parties agree on the layout of the message. Some researchers have developed an authentication mechanism for achieving secure authentication for remote, non-native clients in ROS (Toris et al., 2014). While it can increase the security of the overall system, without data encryption, an eavesdropper could acquire non-encrypted information.

Part of the ROS community is dedicating efforts to integrating OMG's DDS (Data Distribution Service) as a transport layer for ROS 2.0.[4] A preliminary alpha version has just been released. DDS is a standard specification followed by several vendors for a middleware providing publish-subscribed communications for real-time and embedded systems. RTI provides plugins which comply with the DDS Security specification including authentication, access control and cryptography. It would be a big step forward for securing our robots if ROS 2.0 aimed to comply with the DDS Security specification as well.

YARP states among its documentation[5]: "A [default] new connection to a YARP port is established via handshaking on a TCP port. So everyone who can access this TCP port can connect to your YARP port. So if you are not behind a firewall, you are exposing your YARP infrastructure to the world (.) And if your application is vulnerable to corrupted data, it is a security

[1] Most U.S. Drones Openly Broadcast Secret Video Feeds: http://www.wired.com/2012/10/hack-proof-drone/
[2] Interoperable Telesurgery Protocol (ITP) Plaintext Unauthenticated MitM Hijacking: http://osvdb.org/121842
[3] http://wiki.ros.org/Topics
[4] http://design.ros2.org
[5] http://wiki.icub.org/yarpdoc/yarp_port_auth.html

leak." Other YARP documentation reads clearly[6]: "If you expose machines running YARP to the Internet, expect your robot to 1 day be commanded to make a crude gesture at your funders by a script kiddie in New Zealand." However, an authentication mechanism can be activated in YARP, which adds a key exchange to the initial handshaking in order to authenticate any connection request. It has been enabled by default so it is always compiled. However, to preserve backward compatibility, the feature is skipped at runtime if the user does not configure it by providing a file that contains the authentication key.

Additionally, a new port monitoring and arbitration (Paikan et al., 2014) functionality inside YARP has been used to implement a LUA encoder/decoder of data.[7] Data are passed through a Base64 encoder before being sent, and decoded upon reception at the target port. A similar mechanism could potentially be used to encrypt and decrypt the data.

A limited amount of other works has also focused on securing robot communications. In Groza and Dragomir (2008), they implement an authentication protocol to assure the authenticity of the information when controlling a robot via TCP/IP. However, they do not implement encrypted communications. In Coble et al. (2010), they implemented a hardware system that verifies integrity and health of the system software (to avoid tampering) in telesurgical robots. Regarding the previously mention ITP protocol, some researchers are working on security enhancements (Lee and Thuraisingham, 2012). One commercially available robot that does take cyber security into account is BeamPro, a telepresence robot[8] where secure protocols, symmetric encryption, and data authentication are used, thus providing security and privacy.

Secure communications are even more important in new trends in robotics which aim at outsourcing computation, namely *Cloud Robotics*. In this paradigm, robots use their sensors to collect data, and then upload the information to a remote computation center, where the information is processed, and may be shared with other robots. Rapyuta (Mohanarajah et al., 2015) is an example of this paradigm where the technologies used (e.g., WebSockets) allow to secure the information.

Another usual way of communications between robot's devices is through communication buses (CAN, EtherCAT, etc.). Currently, none of the traditional field buses offers security features against intentional attacks (Dzung et al., 2005). However, those based on ethernet could potentially make use of the security measures included in TCP/UDP/IP. For instance, secure routers (e.g., EDR-G903), include firewalls and VPNs, and support EtherCAT.

Discussion

A big market of opportunities for research regarding cyber safety in robotics exists. Most robots are not yet prepared, from a security point of view, to be deployed in daily life. The software is not prepared to protect against attacks, because communications are usually unencrypted.

Regarding the dates of the exploits presented, and the current hype in deployment of daily robotics (vacuum cleaners, amateur drones, etc.), *Cryptobotics*, understood as a mix of cyber security and robotics, comes just in time to prepare these systems to be safely used.

An important issue to be discussed is whether the implementation of encrypted communications may affect the performance, especially in real-time systems. The question about performance is highly dependant on the hardware, the software and the network used. Encrypted communications on the Internet (https, ssh) show us that it is possible to perform secure communications and offer remote services. For instance, Adam Langley (Google Senior Staff Software Engineer) has stated: "when Google changed Gmail from http to https (.) we had to deploy no additional machines and no special hardware. On our production front-end machines, SSL/TLS accounts for less than 1 of the CPU load.[9]" From our experience in humanoid robotics, a 1% overhead (while respecting determinism in time) can be acceptable if it means our devices can be less vulnerable to cyber attacks. Could an 8 MHz microcontroller perform real-time encryption? Is it reasonable to implement authentication mechanisms along field buses in time-constrained scenarios? This article intends to raise awareness for developers to determine whether it is viable to integrate these mechanisms depending on each specific use case.

Some may ask why these problems have not been addressed previously. In recent years, intrinsically safe industrial robots, the rise of domestic robots, and the use of mobile robots in public spaces, have arisen issues that the robotics community did not have to face in its previous 60 years of existence. Researchers are now focused on developing applications to make robots useful, which may have made cyber safety a low priority.

Author Contributions

SM discovered the potential issue in mainstream robotics software and wrote part of the paper. JV found the technical support behind the security issues and wrote and improved the text. CB provided context for the topic, reviewed the text, defined the criteria to evaluate the work, and contributed to the text.

References

Bonaci, T., Herron, J., Yusuf, T., Yan, J., Kohno, T., and Chizeck, H. J. (2015). "To make a robot secure: an experimental analysis of cyber security threats against teleoperated surgical robots," in *arXiv Preprint arXiv:1504.04339*.

Coble, K., Wang, W., Chu, B., and Li, Z. (2010). Secure software attestation for military telesurgical robot systems" in *Military Communications Conference, 2010 – Milcom 2010*, 965–970. doi:10.1109/MILCOM.2010.5679580

Denning, T., Matuszek, C., Koscher, K., Smith, J. R., and Kohno, T. (2009). "A spotlight on security and privacy risks with future household robots: attacks

[6]http://wiki.icub.org/yarpdoc/what_is_yarp.html
[7]http://wiki.icub.org/yarpdoc/coder_decoder.html
[8]http://www.suitabletech.com/
[9]http://www.imperialviolet.org/2010/06/25/overclocking-ssl.html

and lessons," in *Proceedings of the 11th International Conference on Ubiquitous Computing* (New York, NY: ACM), 105–114. doi:10.1145/1620545.1620564

Dzung, D., Naedele, M., Von Hoff, T. P., and Crevatin, M. (2005). Security for industrial communication systems. *Proc IEEE* 93, 1152–1177. doi:10.1109/JPROC.2005.849714

Groza, B., and Dragomir, T.-L. (2008). "Using a cryptographic authentication protocol for the secure control of a robot over TCP/IP," in *IEEE International Conference on Automation, Quality and Testing, Robotics, 2008. AQTR 2008*, Vol. 1, 184–189. doi:10.1109/AQTR.2008.4588731

Hans, M., Graf, B., and Schraft, R. (2002). "Robotic home assistant care-o-bot: past-present-future," in *Proceedings of the 11th IEEE International Workshop on Robot and Human Interactive Communication, 2002*, 380–385. doi:10.1109/ROMAN.2002.1045652

Javaid, A. Y., Sun, W., Devabhaktuni, V. K., and Alam, M. (2012). "Cyber security threat analysis and modeling of an unmanned aerial vehicle system," in *2012 IEEE Conference on Technologies for Homeland Security (HST)*, 585–590. doi:10.1109/THS.2012.6459914

King, H. H., Tadano, K., Donlin, R., Friedman, D., Lum, M. J., Asch, V., et al. (2009). "Preliminary protocol for interoperable telesurgery," in *International Conference on Advanced Robotics, 2009. ICAR 2009*, 1–6. Available at: http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=5174711&isnumber=5174665

Lee, G. S., and Thuraisingham, B. (2012). Cyberphysical systems security applied to telesurgical robotics. *Comput Stand Interfaces* 34, 225–229. doi:10.1016/j.csi.2011.09.001

McClean, J., Stull, C., Farrar, C., and Mascareñas, D. (2013). "A preliminary cyber-physical security assessment of the robot operating system (ROS)," in *Proceedings of SPIE 8741, Unmanned Systems Technology XV*, 874110. doi:10.1117/12.2016189

Metta, G., Fitzpatrick, P., and Natale, L. (2006). Yarp: yet another robot platform. *Int J Adv Rob Syst* 3, 43–48. doi:10.5772/5761

Mohanarajah, G., Hunziker, D., D'Andrea, R., and Waibel, M. (2015). Rapyuta: a cloud robotics platform. *IEEE Trans Autom Sci Eng* 12, 481–493. doi:10.1109/TASE.2014.2329556

Paikan, A., Fitzpatrick, P., Metta, G., and Natale, L. (2014). Data flow ports monitoring and arbitration. *J Software Eng Rob* 5, 80–88.

Pfleeger, C. P., and Pfleeger, S. L. (2002). *Security in Computing*. Prentice Hall Professional Technical Reference.

Quigley, M., Conley, K., Gerkey, B., Faust, J., Foote, T., Leibs, J., et al. (2009). "ROS: an open-source robot operating system," in *ICRA Workshop on Open Source Software*, 3, 5.

Tanenbaum, A. S., and Bos, H. (2014). *Modern Operating Systems*. Prentice Hall Press.

Toris, R., Shue, C., and Chernova, S. (2014). "Message authentication codes for secure remote non-native client connections to ros enabled robots," in *2014 IEEE International Conference on Technologies for Practical Robot Applications (TePRA)*, 1–6. doi:10.1109/TePRA.2014.6869141

Vuong, T., Filippoupolitis, A., Loukas, G., and Gan, D. (2014). "Physical indicators of cyber attacks against a rescue robot," in *2014 IEEE International Conference on Pervasive Computing and Communications Workshops (PERCOM Workshops)*, 338–343. doi:10.1109/PerComW.2014.6815228

Conflict of Interest Statement: The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.