Secure ROS
Imposing secure communication in a ROS system

Aravind Sundaresan and Leonard Gerard
aravind@ai.sri.com, leolchat@gmail.com

SRI International, Menlo Park, CA 94025

Abundant Robotics

September 21, 2017

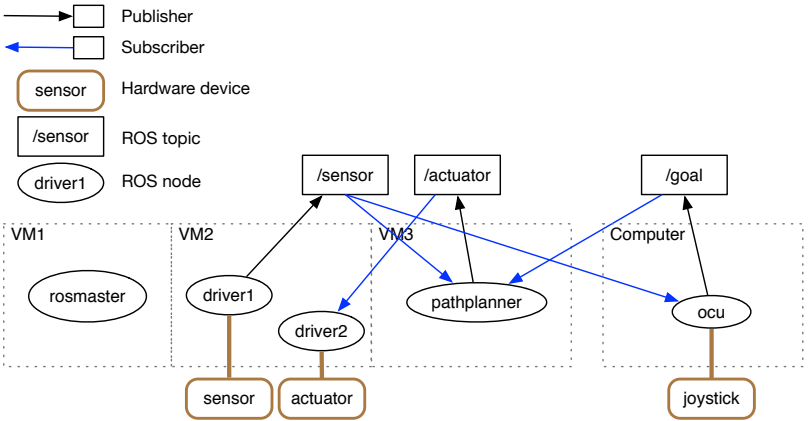
SRI International

ROS is insecure

ROS was built as an open robust, general-purpose robotics platform to encourage collaborative development. As such, ROS has no security.

- Any node on the network can publish or subscribe to any topic.
- Any process on the network can kill any node.
- Any new process with *nodename* will supersede existing node.

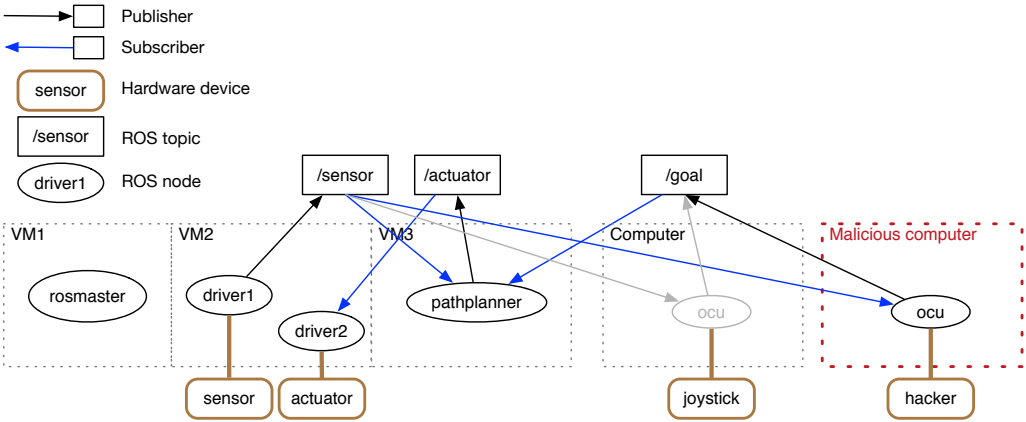
An example ROS system



Joystick controlled robot with a planner

SRI International®

An example ROS system: malicious attack



Joystick controlled robot with a planner

SRI International®

Secure ROS

Secure ROS is an overlay "package" that enables secure communication between ROS nodes.

- ROS public API is the same so that existing nodes can be re-used.
- A security configuration file can be used to specify authorized **publishers**, **subscribers**, **node names**, etc based on **IP address**.
- (If no file is specified, Secure ROS behaves like regular ROS.)
- Modified ROS master and client libraries (C++, Python) to enforce security rules.
- IPSec is used to ensure that IP packets are encrypted and authenticated.

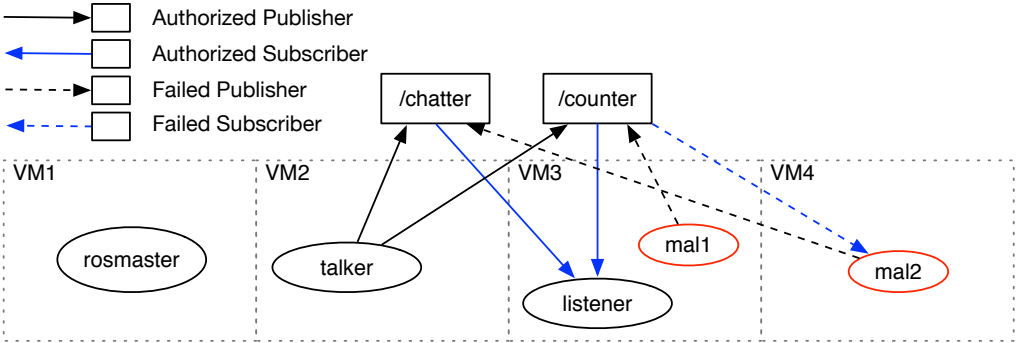
SRI International®

Authorization configuration example

```
aliases:
  vm2: [192.168.10.202]
  vm3: [192.168.10.203]
topics:
  /chatter:
    publishers: [vm2]
    subscribers: [vm3]
  /counter:
    publishers: [vm2]
    subscribers: [vm3]
nodes:
  /talker: [vm2]
  /listener: [vm3]
```

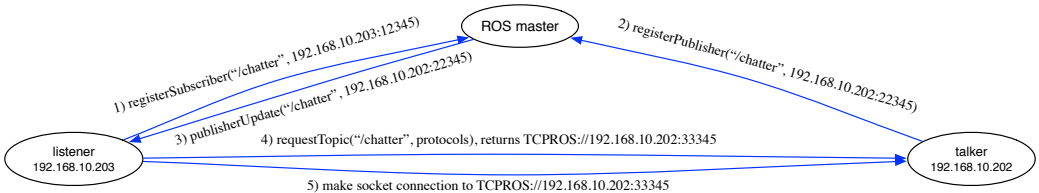
SRI International®

Secure publish-subscribe example



SRI International®

Secure publish-subscribe steps



1. Master: Is 192.168.10.203 allowed to subscribe to /chatter?
2. Master: Is 192.168.10.202 allowed to publish to /chatter?
3. Client: Is publisherUpdate() coming from Master?
4. Client: Is 192.168.10.203 allowed to requestTopic(/chatter)?
5. Client: Is 192.168.10.203 allowed to make socket connection to tcpros://192.168.10.202:33345?

SRI International®

More Secure ROS features

- You can also specify parameters and services in authorization file.

```
parameters:
  /param/a:
    setters: [vm2]
    getters: [vm3]
services:
  /service/b:
    providers: [vm2]
    requesters: [vm3]
```

- Only “master” node can kill nodes.
- Each machine can only get information that it needs (*rostopic*, *rosparam*, *etc.*).

SRI International®

Secure ROS status

- Secure ROS can be merged into ROS.
  - Pull request 1080 being reviewed.
  - https://github.com/ros/ros_comm/pull/1080
- Secure ROS deb packages available for download.
  - secure-ros-lunar-secure-ros_0.9.3-1_amd64.deb
  - secure-ros-kinetic-secure-ros_0.9.3-1_amd64.deb
- Install as overlay.
  - /opt/secure_ros/lunar/

SRI International®

Configuring IPsec (1)

- Install secure_ros_tools

```
sudo apt-get install racoon ipsec-tools python-pip
sudo pip install git+https://github.com/SRI-CSL/secure_ros_tools.git
```

- Create YAML file (e.g. “hosts.yaml“)

```
machine1: 192.168.10.201
machine2: 192.168.10.202
machine3: 192.168.10.203
```

- Generate configuration files for IPsec.

```
create_ipsec_conf -i hosts.yaml
```

SRI International®

Configuring IPsec (2)

- Copy to appropriate machines. E.g., on “machine1“,

```
sudo tar xzf machine1.tgz -C /
```

- List of files on each machine. E.g., on “machine1“,

```
/etc/ipsec-tools.conf
/etc/racoon/certs/machine1
/etc/racoon/certs/machine1.pub
/etc/racoon/certs/machine2.pub
/etc/racoon/certs/machine3.pub
/etc/racoon/racoon.conf
```

SRI International®

Resources

- Secure ROS: http://secure-ros.csl.sri.com/
- Download: http://secure-ros.csl.sri.com/download/
- Documentation: http://SRI-CSL.github.io/secure_ros
- Github:
  - https://github.com/SRI-CSL/secure_ros.git
  - https://github.com/SRI-CSL/secure_ros_tools.git
  - https://github.com/SRI-CSL/ros_comm.git
  - https://github.com/SRI-CSL/nodelet_core.git

SRI International®