Designing a Bot Securely from the Ground Up

Learn RPA (/company/blog/categories/learn-rpa)

Daniel Yinanc (/company/blog/author/daniel-yinanc) Tuesday, 19 May 2020 320 Views

5 Minute Read

Tags: Security (/company/blog/tags/security) IT (/company/blog/tags/it) RPA (/company/blog/tags/rpa)

To many, security (/solutions/rpa-security) is an afterthought. Something that needs to be implemented before a piece of software is shipped or even after. The priorities of software developers are often:

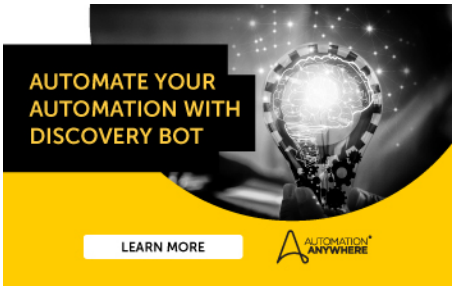
1. Make it work
2. Make it profitable
3. Make it secure

In that order.

The problem with that is if you have to re-engineer your entire application/bot after it has been developed because of a security defect, you've failed at priority 1 and are now eating into the profitability of priority 2. In this blog post, I will teach you how to implement security into your Automation Anywhere bot design from the start.

SHARE THIS:

CONTACT US > (/contact-us)



(/products/discovery-bot)

MOST POPULAR:

How RPA Enables New Ways of Collaboration (/company/blog/product-insights/how-rpa-enables-new-ways-of-collaboration)

Category: Product Insights (/company/blog/categories/product-insights) | 5 Minute Read

'Record' Is Critical for Enterprise-Grade RPA (/company/blog/product-insights/record-is-critical-for-enterprise-grade-rpa)

Category: Product Insights (/company/blog/categories/product-insights) | 5 Minute Read

I will walk you through my thought process of designing a bot and thinking about all the security controls one will need to implement as part of a bot's development. Hopefully, my thought process and thinking strategies will help you in building your secure bots.

One IoT bot to rule them all

Whether you are building bots for the enterprise or personal usage, security and privacy are critical. I recently moved into a brand-new house that came with a ton of IoT-related technology. Coupled with some other personal gadgets, there are currently a ton of IoT connected devices that can be controlled via apps and other services. The goal here is to build a bot to connect with the smart lock, thermostat, garage door opener, smart TV and the home security system and manage all of these services with one bot. It will allow me to automate tasks such as gathering usage data, analytics, report outages, or regulate my house temperature at specific times.

Most of these devices are managed by mobile apps that have APIs allowing me to connect to them. I will be able to write my own clients for the APIs so that my bot can interact with those servers directly. This bot might also be handy for building managers or larger facilities where multiple instances of each of these devices need to be managed. This bot will be designed with scalability in mind.

Initial planning

The phase of planning involves brainstorming some of the features I might want for the bot, the inputs and outputs it will produce, and a preliminary data flow diagram defining the bot's operation. I will also think about what sensitive data or assets my bot will be responsible for handling.

Feature Ideas:

- Collect and parse all device logs into a central location
- Perform daily actions on particular devices (setting the alarm on/off every day, setting the temperature, making sure the door is locked regularly)
- "Notify Me" of errors or incidents of note

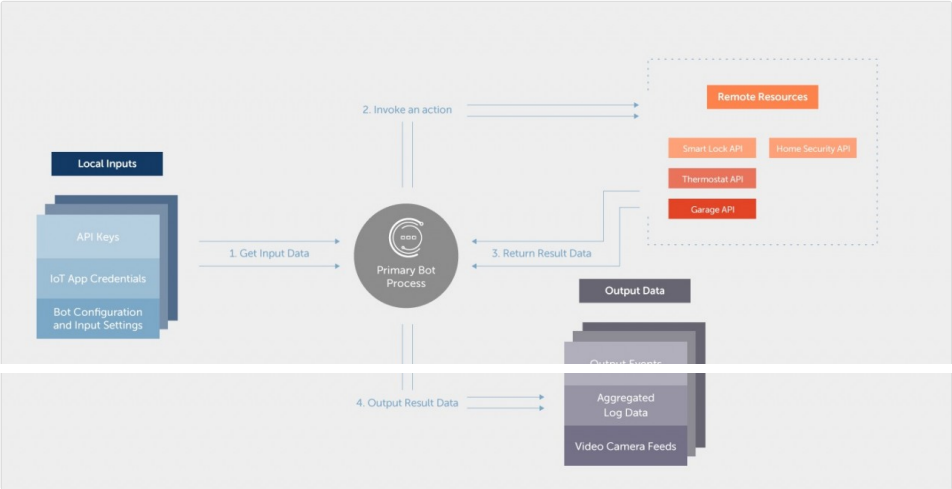
Inputs:

- Credentials for the various services and the associated API information
- Actions to be performed, with time intervals or frequencies

Output:

- Collected analytics, uptime, or log data
- Reported completed actions in a centralized log file
- Incident/Alert notifications to text messages/email

Initial data flow diagram



Bot assets

The bot will store and manage resources, some of which are sensitive (account credentials). It is important early in the bot design to properly enumerate all assets that the bot will be responsible for so that proper defensive planning can take place to protect these assets.

- Usernames and passwords to various IoT services
- API keys for IoT services
- Generated log data and alert information
- Incidents or alert information
- Sensitive device actions (disable alarm, unlock door, open garage)
- Video camera feed data

Initial security considerations

With the creation of the data flow diagram and the enumeration of assets, brainstorming can now begin regarding some of the threats affecting the bot and the defenses needed. Threat modeling can be a great way of accomplishing this goal. In deriving your own security requirements, you will want to consider each asset and then brainstorm all the ways that asset could be compromised. From there, you can then think of each defense you will need to protect the assets.

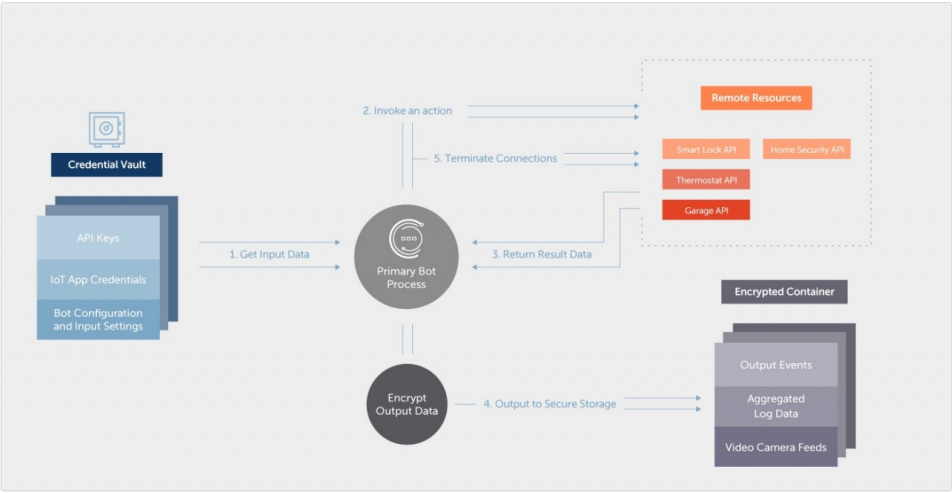
Preliminary security requirements

The following is a list of all the security considerations I will be taking while developing my IoT bot. This information will be used to revise my data flow diagram and develop all additional features alongside each functional feature.

1. TLS communication for all bot traffic (TLS 1.2 or better)
2. Use the Automation Anywhere Credential Vault for all sensitive data
3. Encrypt sensitive output files
4. Terminate all sessions and remote connections on completion of each bot execution
5. All access to local files will be on a whitelist only basis. (i.e. the bot will only have permission to access the specific files it needs to perform its duties)
6. All input to the bot, including data returned from remote API's will be treated as hostile and data validation will be performed on all input.
7. Prevent data leakage and logging of sensitive information. All sensitive data needs to be masked or not logged in log files.

Revised data flow diagram

Some of the security requirements will affect my data flow and bot process, as such I have revised my data flow diagram to include the new features.



Now that I've come up with some of the key security requirements alongside my bot features, it's time to sign up for Secure Bot Developer Learning Trail (https://university.automationanywhere.com/rpa-learning-trails/automation-anywhere-secure-bot-developer/) from

Automation Anywhere University (https://university.automationanywhere.com/)so that I can design, develop, and deploy my bot securely. Hopefully, I've given you some ideas to design your bots securely.

Get started on bot design for security.

ACCESS THE TRAIL (HTTPS://UNIVERSITY.AUTOMATIONANYWHERE.COM/RPA-LEARNING-TRAILS/AUTOMATION-ANYWHERE-SECURE-BOT-DEVELOPER/)

SHARE THIS:

(/company/blog/author/daniel-yinanc)

About Daniel Yinanc: (/Company/Blog/Author/Daniel-Yinanc)

Daniel Yinanc is a principal engineer, data scientist, and application security architect. As the creator and lead architect of the Automation Anywhere Bot Security program, he's a subject matter expert in RPA application security concerns.

LinkedIn (http://https://www.linkedin.com/in/danielyinanc/) | Subscribe to Posts

| View All Posts (/company/blog/author/daniel-yinanc)

Author's Recent Posts

More posts from author (/company/blog/author/daniel-yinanc)

10 Best Practices for Secure Bot Design (/company/blog/learn-rpa/ten-best-practices-for-secure-bot-design)

Wednesday, 22 April 2020

RELATED POSTS: