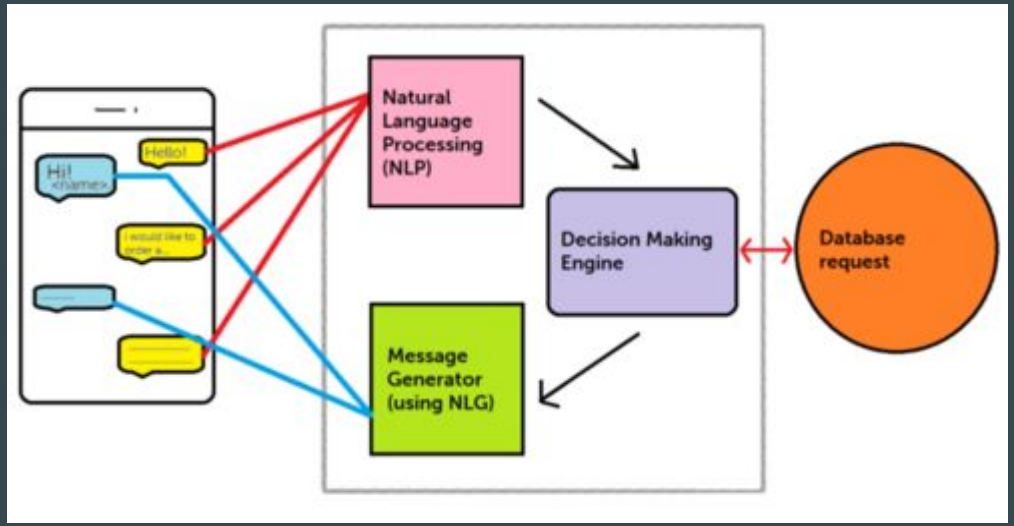
Chatbot Security and Privacy in the Age of Personal Assistants

By Winson Ye and Qun Li
College of William and Mary

Introduction

A. Client Module
B. Communication Module
C. Response Generation Module
D. Database Module



Client Module: Unintended Activation Attacks

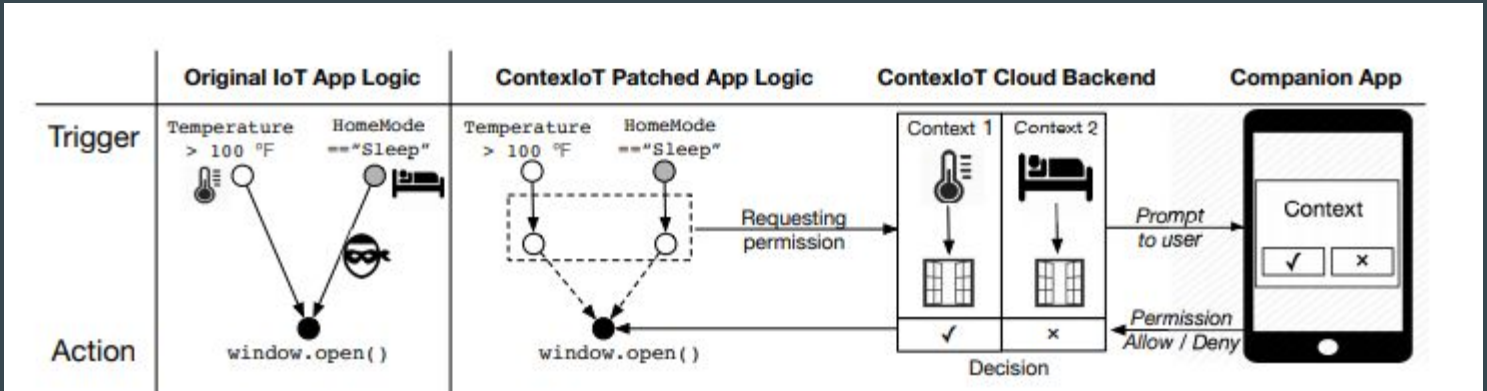
- Wake-up phrases are used to activate the smart device
  - What if the device confuses words?
  - What if the device records other people in the same room?
  - What if the device is tricked by a recording?
- What are some solutions?
  - Use Wifi to detect human motion
  - Detect whether user is talking to human or device

Client Module: Faked Response

- User misconceptions abound
  - 30% of users have trouble turning off smart device
  - 78% did not use LED to check for proper termination
- What if a malicious skill tricks the user into thinking they switched to a different app?
- What if a malicious skill fakes termination?
- What are some solutions?
  - Check smart device responses against a black list

Client Module: Access Control Attacks

- Some apps may grant very broad permissions to the user
  - What if a hacker can take advantage of this to break into the house?
- What are some possible solutions?
  - Defensive coding strategies
  - Security profilers

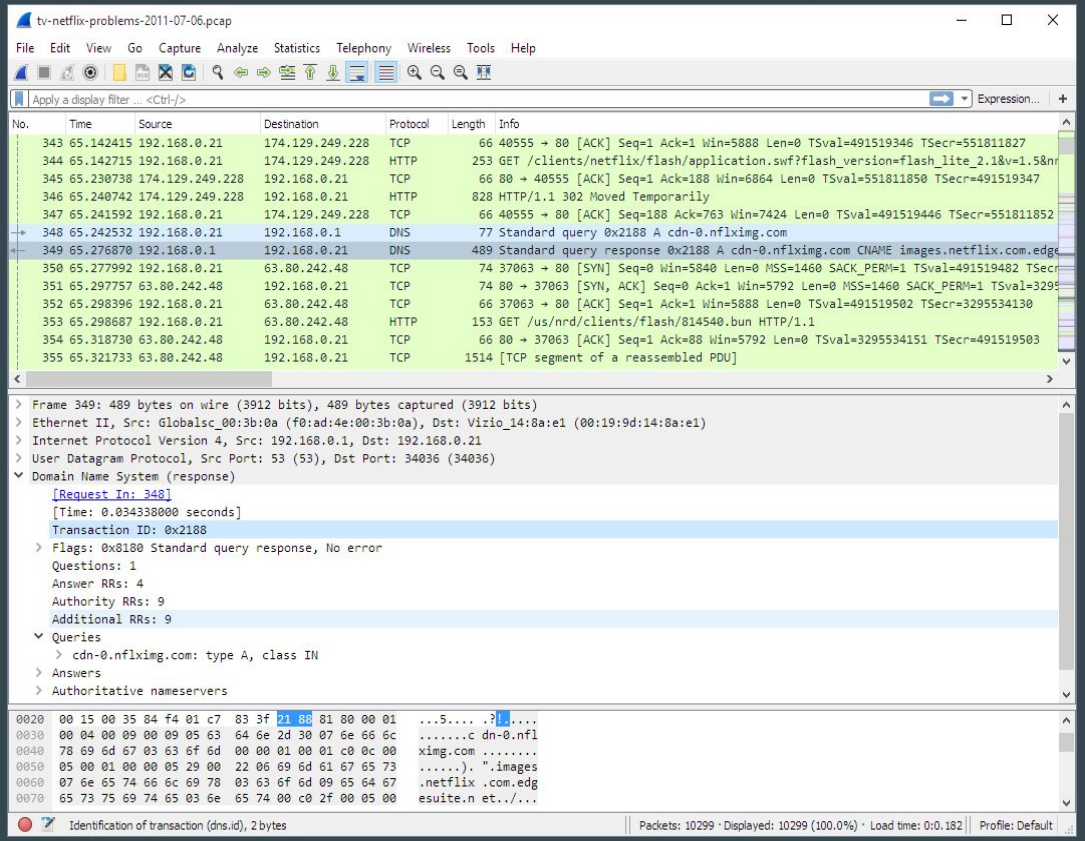


Client Module: Adversarial Voice Samples

- Voice recognition technology is essential for personal assistants
  - What if we can perturb the voice command such that the personal assistant misinterprets it?
  - What if we can hide voice commands in songs?
- What are some possible solutions?
  - Retrain the model
  - Keep the architecture secret

Communication Module

- DDos Attacks
  - Flood the server with as many requests as possible
- Wiretapping
  - Use packet metadata to predict voice command
- MitM Attacks
  - Intercept messages and delete/modify them



Response Generation Module: Out of Domain Attacks

- Chatbot is generally very adept at a select few domains
  - What if we make out of domain requests?
- What are some potential solutions?
  - Train a classifier to detect out of domain requests
  - Improve network's ability to quantify uncertainty

Response Generation Module: Adversarial Text Samples

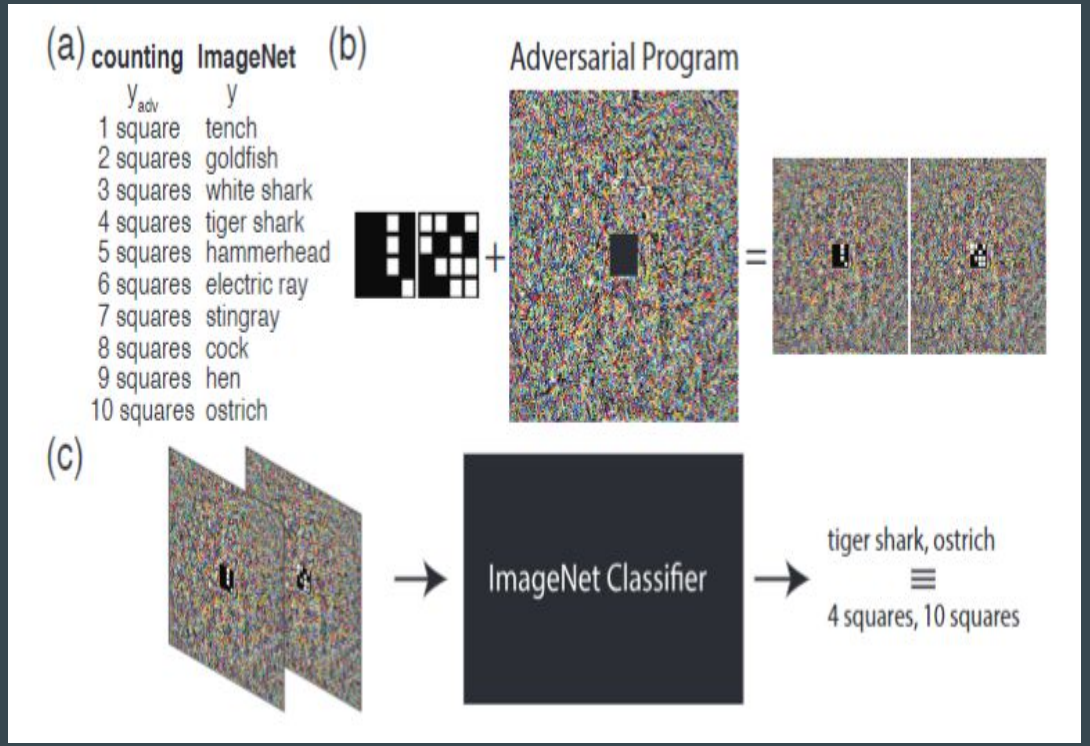
- Chatbot is constantly learning from its environment
  - What if we purposely poison the environment?
- What are some potential solutions?
  - Employ a hate speech detector

Response Generation Module: Language Model Attacks

- State of the art chatbots reply on language models like BERT
  - What if we can create malicious language models that sabotage the chatbot very discretely?
- What are some potential solutions?
  - Search for trigger words
  - Constantly vet language models

Response Generation Module: Adversarial Reprogramming

- The chabot replies on a number of different deep learning modules
  - What if we can repurpose these modules for our malicious tasks?
- What are some potential solutions?
  - Make it harder for the adversary to learn the weaknesses of the model



Response Generation Module: Feedback Engineering

- Chatbot usually gets a reward signal from the user
- The system improves itself through either:
  - Retraining
  - Reinforcement learning
- What if we can discretely retrain the chatbot to use offensive language after hearing certain trigger words?
- What if we can alter the reward signal and get the chatbot to adopt our malicious policy?
- What are some possible solutions?
  - Make it harder to query the model multiple times
  - Separate training examples and response generation module

Database Module

- Database module houses a lot of sensitive information
  - What if we launch an injection attack against it?
  - What if we manipulate the knowledge graph?
- What are some possible solutions?
  - Search database for injection vulnerabilities before deployment
  - Clean the data used to train the knowledge graph

Conclusion

- Contact Information
  - wye@email.wm.edu