CIS 4930: Secure IoT

Prof. Kaushal Kafle

Lecture 9

Adapted from slides by Adwait Nadkarni

Smart Home

Smart Home Platforms

- Many platforms are now *programmable*
- Developers can use the API to build *apps that*
    - Get status updates from devices
    - Send commands to devices
    - Interface with other services (SMS, Web Services)
- Prior work has looked at: devices, the cloud, **the Platform OS**

nest

HomeKit

hue

PHILIPS

SmartThings

Recall: Vulnerabilities (attack vectors)

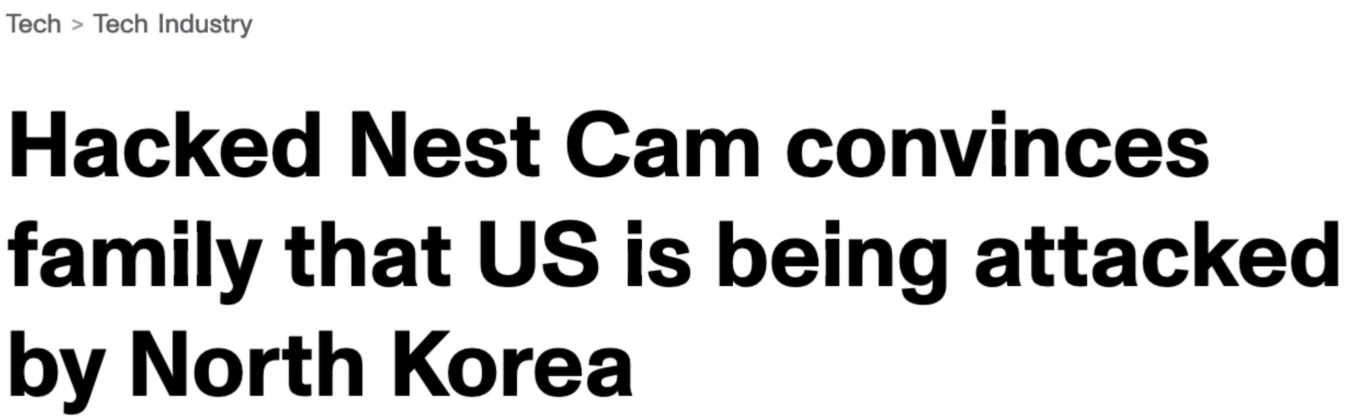
- A *vulnerability* is an artifact that an attacker can leverage to execute a threat
- e.g., unauthenticated network communication, network-facing services with known vulnerabilities, default passwords and other unsafe configurations
- What are the sources of vulnerabilities?
  - Bad hardware
  - Bad software
  - Bad design/requirements
  - Bad security policy/default config?
  - Unintended (mis)use (e.g., bad combinations of routines)

IoT Attack Vectors

- Many components: device, mobile apps, cloud endpoints, platform OS, automations
- Large attack surface! A *large & diverse set of attack vectors for each component!*

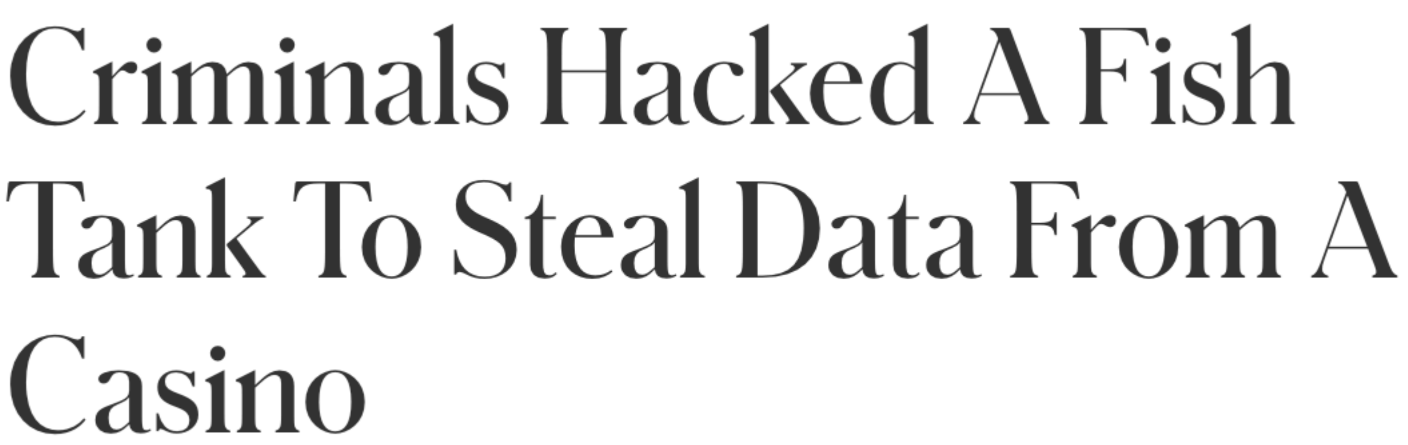
Some bad news

- IoT is no different



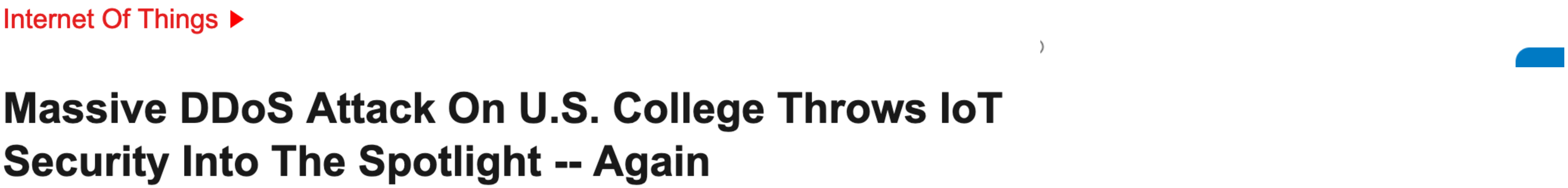
Tech > Tech Industry

Hacked Nest Cam convinces family that US is being attacked by North Korea



> CYBERSECURITY

Criminals Hacked A Fish Tank To Steal Data From A Casino



Internet Of Things ▶

Massive DDoS Attack On U.S. College Throws IoT Security Into The Spotlight -- Again

Motivation

Key question: *Is the IoT platform and its API secure?*

Integrity

Can attackers manipulate devices? (e.g., insert lock codes)

Availability

Can attackers disable devices? (e.g., turn OFF a camera)

Privacy

Can attackers learn private information? (e.g., the user’s schedule)

Authenticity

Can attackers spoof messages? (e.g., event spoofing, using stolen OAuth tokens)

Confidentiality

Can attackers learn sensitive information (e.g., lock codes)

Motivation

Key question: *Is the IoT platform and its API secure?*

Platform OS

nest

hue
PHILIPS

SmartThings

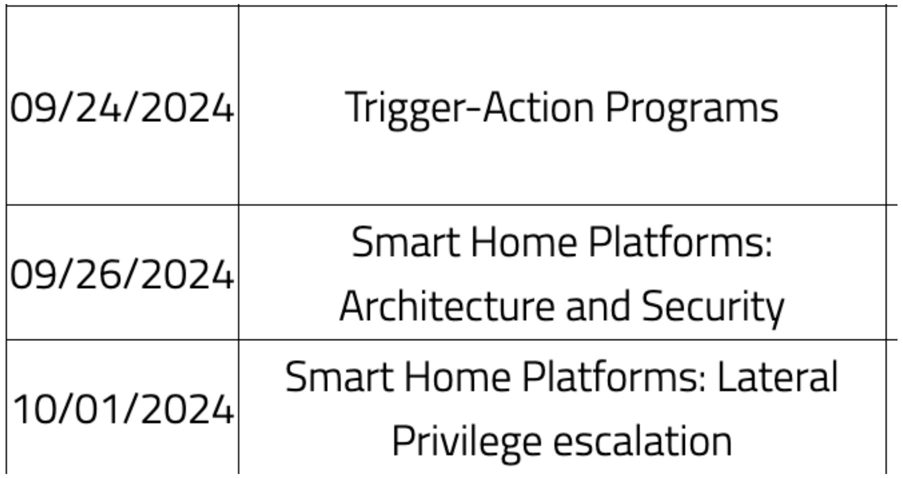
Trigger-Action Programs

yeti.

IoT Mobile Apps

Motivation

- Many platforms are now *programmable*
- Developers can use the API to build *apps that*
  - Get status updates from devices
  - Send commands to devices
  - Interface with other services (SMS, Web Services)
  - Facilitate automations
- Prior work has looked at: devices, the cloud, the Platform OS and…



09/24/2024	Trigger-Action Programs
09/26/2024	Smart Home Platforms: Architecture and Security
10/01/2024	Smart Home Platforms: Lateral Privilege escalation

Trigger-Action Programs

Let’s look at trigger-action programs!

Motivation

- Questions:
    - RQ1: *What are programs about?*
        - *Common themes, triggers, actions, …*
    - RQ2: *Are end-users really creating these programs?*
    - RQ3: *What do these characteristics mean for research?*

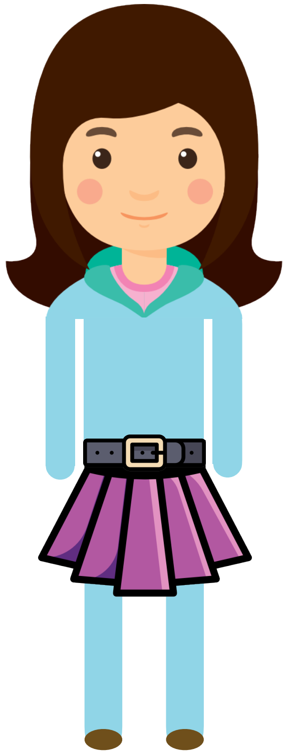
- Questions: From the information flow perspective
    - RQ1: *Are individual programs safe?*
    - RQ2: *Do programs trigger each other? Are such chains safe?*

What are they?

trigger — *action* programs

Trigger

Alice is home



Action

Turn the security camera OFF

Heating / Off

Camera / On

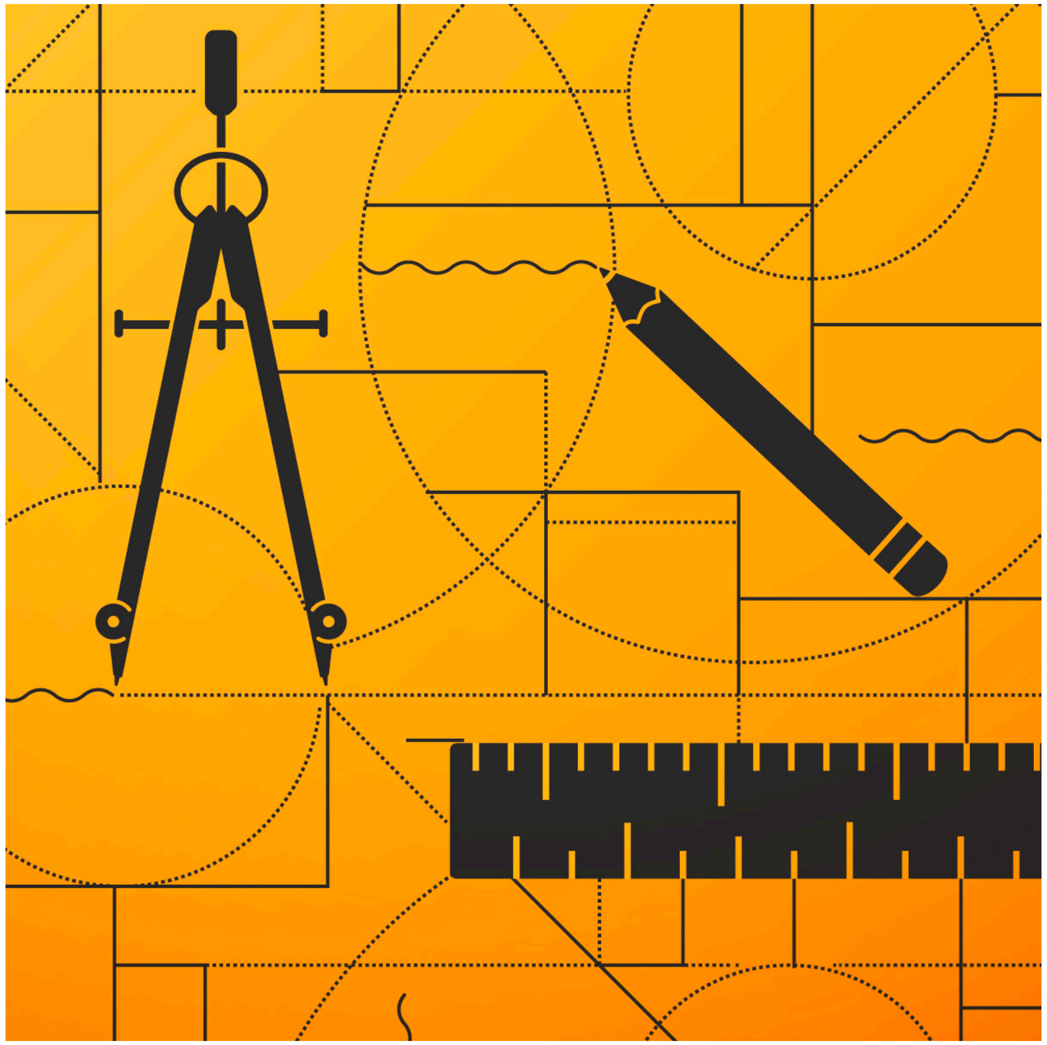
End-user programming

What does this mean for IoT?

- i.e., programming by end users
- Trig-action programming is *conceptually simple*
  - >=1 triggers, >=1 actions, a conditional relationship
  - Users *can* do this!   Challenges?

Measurement studies!

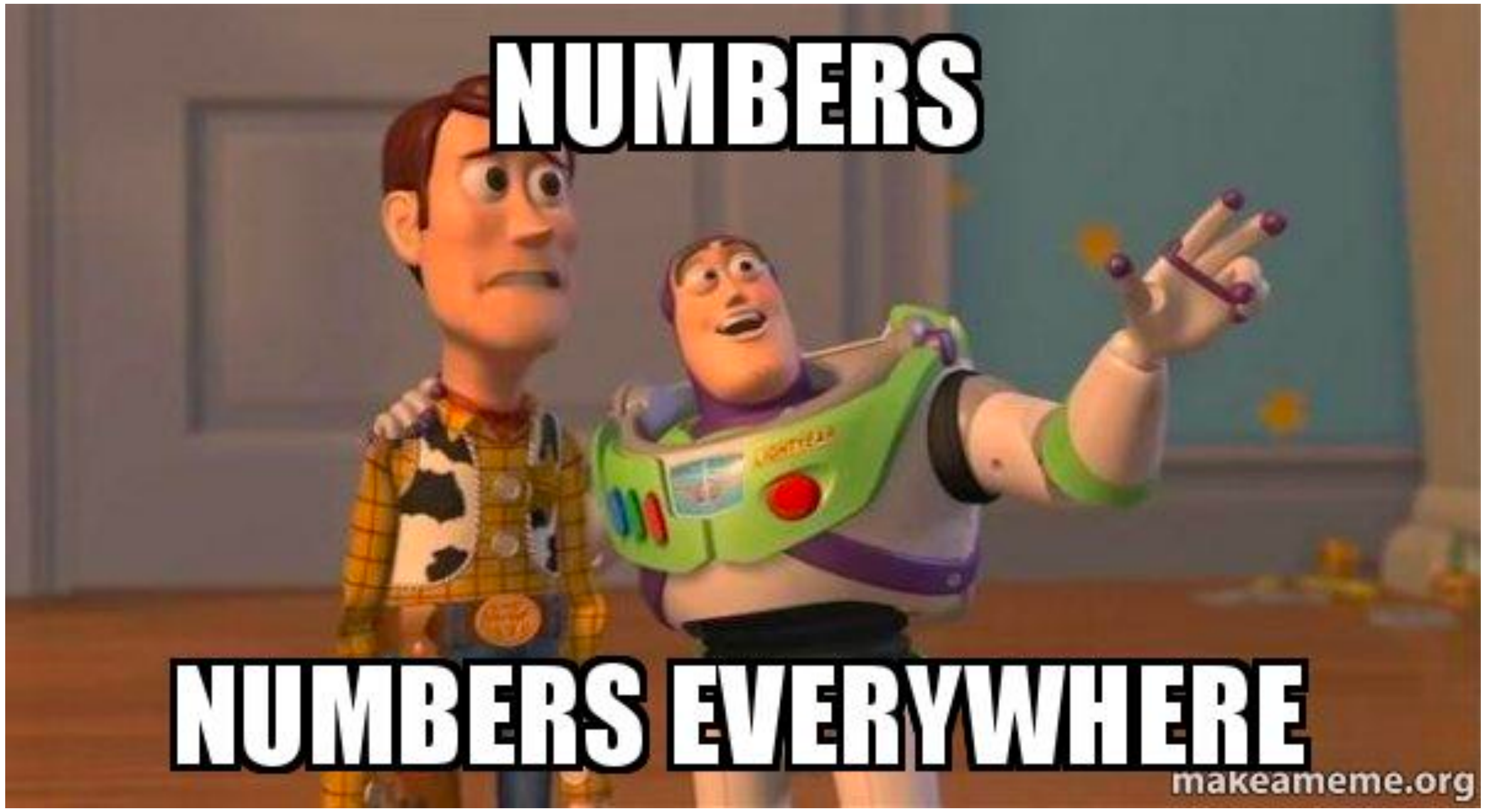
- Study IFTTT
  - Most popular platform
  - Contains Applets created by end-users/ software developers (*recipes* until recently)



Surbatovich, Milijana, et al. "***Some recipes can do more than spoil your appetite: Analyzing the security and privacy risks of IFTTT recipes.***" *Proceedings of the 26th International Conference on World Wide Web*. 2017.

Methodology

1. *Scrape* IFTTT applets and get related metadata
    1. Trigger event, trigger channel, action event, action channel
    2. E.g., *“if it gets too hot, then open the window”:*
        1. trigger channel: nest_thermostat,
        2. trigger event: temperature_rises_ above,
        3. action channel: Smartthings,
        4. action event: unlock.
2. *Execute* applets, to answer backend-related questions (e.g., how long does the execution take?)
3. *Characterize* applets based on individual labels.



Methodology

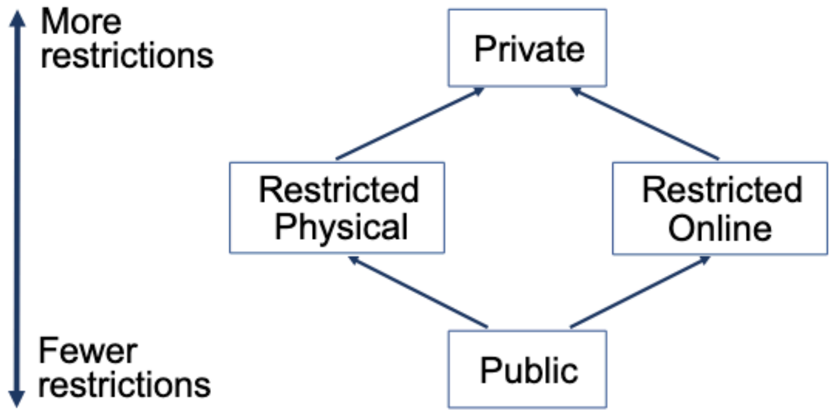


Figure 2: Secrecy lattice. A violation occurs when the corresponding labels of a trigger–action pair go from more restricted to less restricted or if they go between the middle groups.

- **Private**: Only the recipe creator should see
- **Restricted physical**: Privileged physical space e.g., home
- **Restricted online**: Events seen by restricted online audience e.g., Instagram posts
- **Public**: Seen by everyone

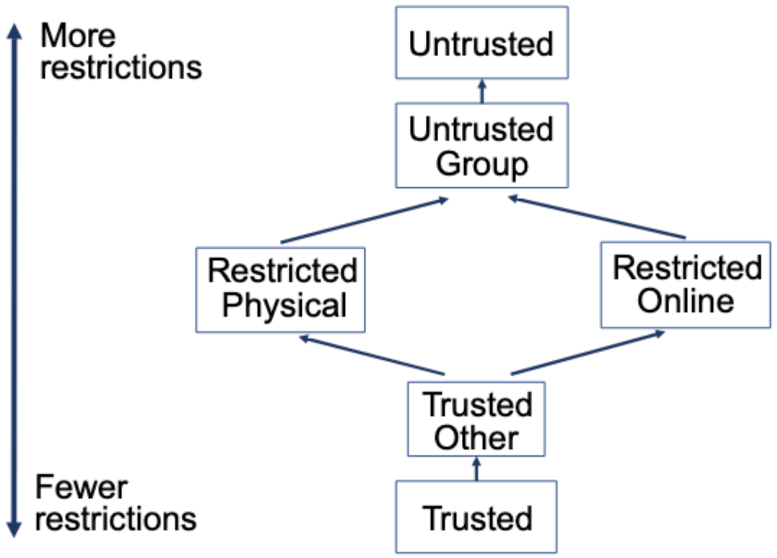
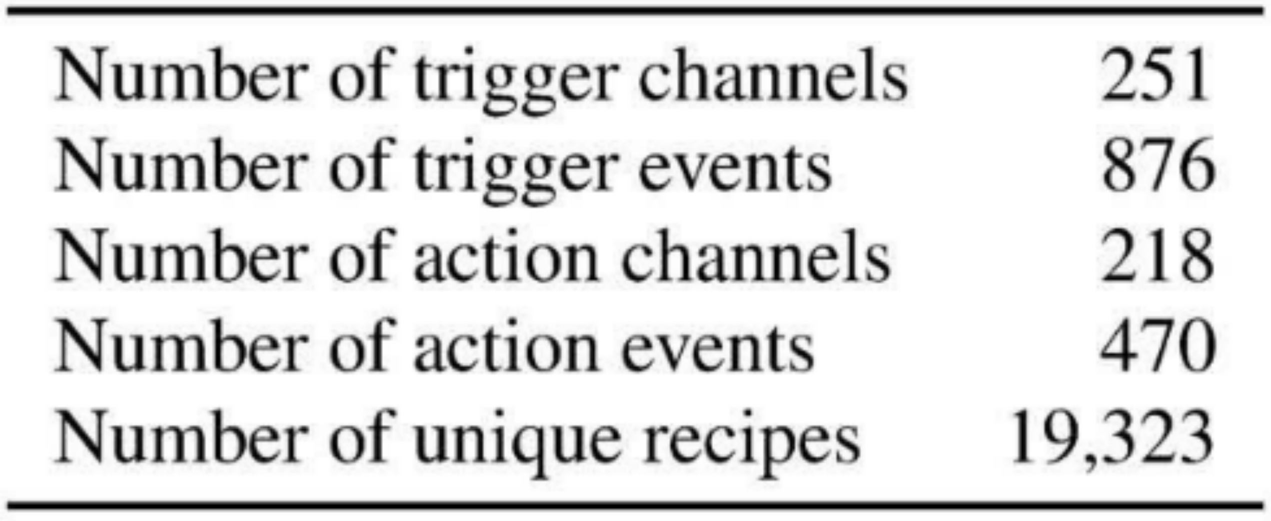


Figure 3: Integrity lattice. It has a similar structure as the secrecy lattice with additional variants of trusted and untrusted sources.

- **Trusted**: Only the recipe creator should cause
- **Trusted other**: Caused by trusted third-party e.g., weather, time
- **Untrusted group**: Caused by untrusted third-party e.g., trending topics on reddit..
- **Restricted physical**: E.g., motion sensors in user's home
- **Restricted online**: E.g., shared drive folders
- **Public**: Could be caused by anyone e.g., motion sensor outside home

Surbatovich, Milijana, et al. ***"Some recipes can do more than spoil your appetite: Analyzing the security and privacy risks of IFTTT recipes."*** *Proceedings of the 26th International Conference on World Wide Web*. 2017.

Key results



Number of trigger channels	251
Number of trigger events	876
Number of action channels	218
Number of action events	470
Number of unique recipes	19,323

Does this say anything in particular about the trigger-action ecosystem?

Users are creating recipes themselves!

Many combinations possible; some more popular than others!

Key results

- Many recipes about social media and online services.
- 19323 recipes - 49.9% with either secrecy or integrity violations
- 22.9% with integrity, 16.7% with secrecy, 10.3% with both

Example violations..

- Remember: User created…….
- Secrecy violation (*private* → *public*):
    - If I take a new photo with the front camera of my phone (trigger), add it to Flickr as a public photo (action).
        - *(Why) is this harmful?*
- Integrity violation (*restricted_physical* → *(private, restricted_online*):
    - If there is a new Instagram photo by anyone in the area, turn my smart switch on, then off.
        - *(Why) is this harmful?*

Some takeaways..

- IoT is not here in full force
- Users are *creating* recipes, rather than searching for existing ones.
- Users are using recipes to fill gaps in functionality, and not inventing new functionality

Followup (from your readings):

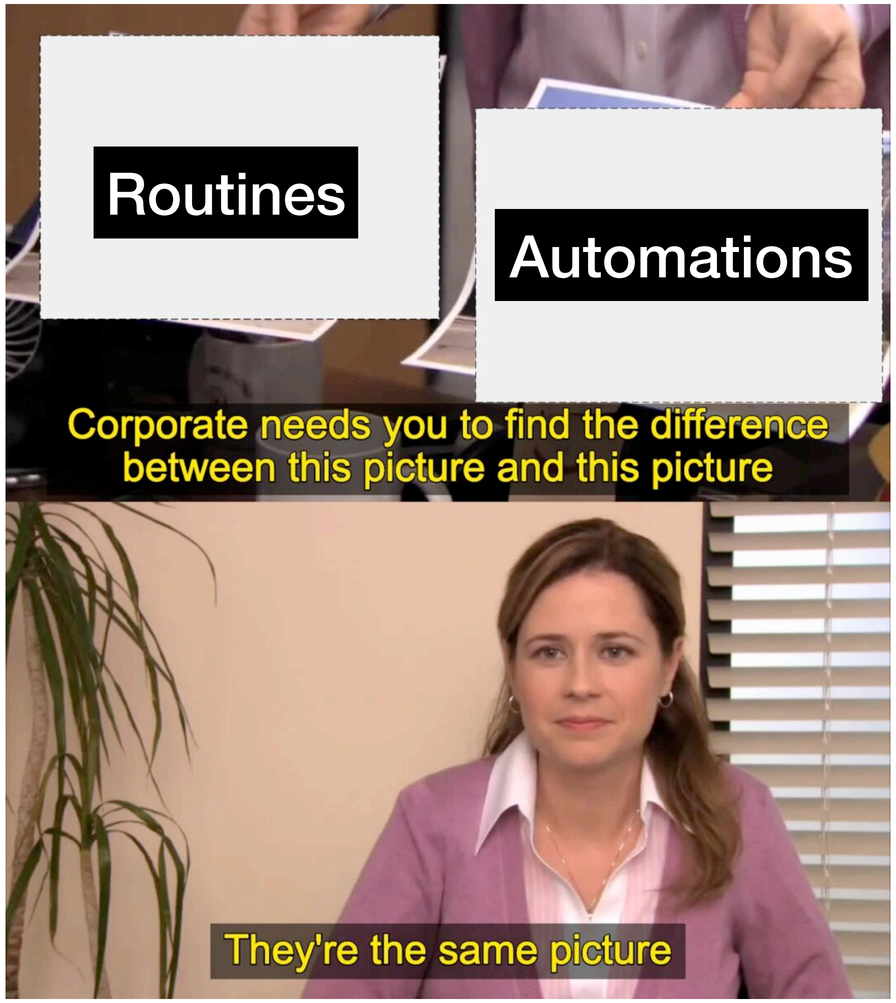
Cobb, Camille, et al. "***How Risky Are Real Users'{IFTTT} Applets?.***" Sixteenth Symposium on Usable Privacy and Security (SOUPS 2020). 2020.

What about trigger-action programs in modern platforms?

Trigger-action Programs

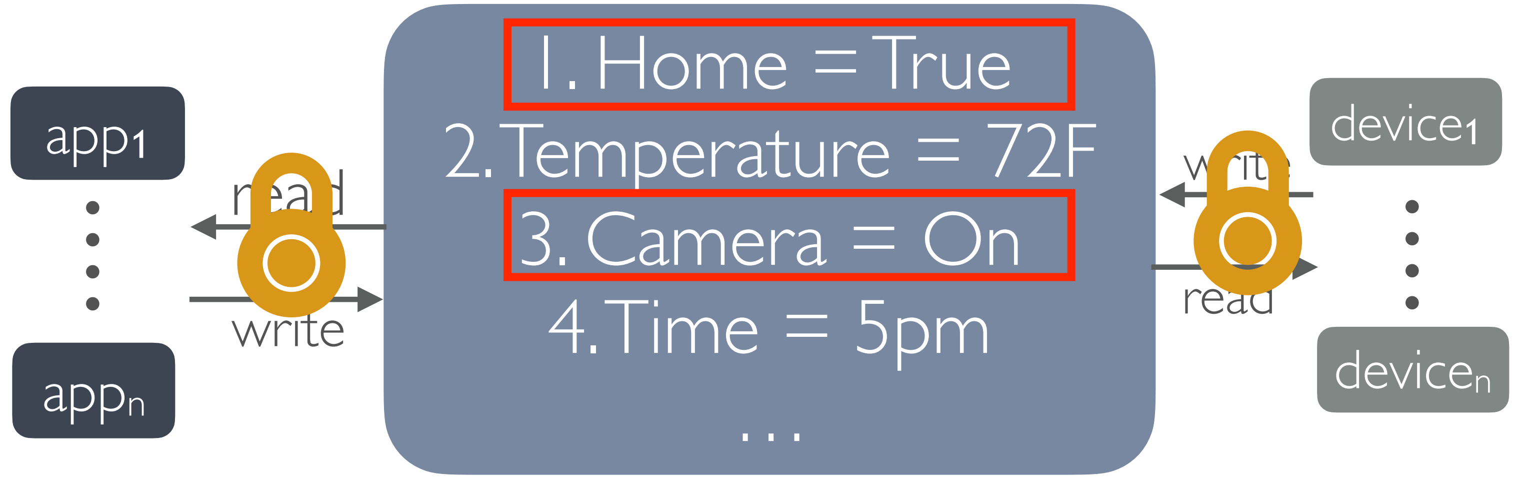
Different names:

- SmartThings - SmartApps
- Nest - Routines
- Philips Hue - Automations/ Scenes
- HomeAssistant - Automations


Routines
Automations
Corporate needs you to find the difference between this picture and this picture
They're the same picture

Overview

Data Store-Based (DSB) platforms



app$_1$

app$_n$

read

write

1. Home = True

2. Temperature = 72F

3. Camera = On

4. Time = 5pm

. . .

write

read

device$_1$

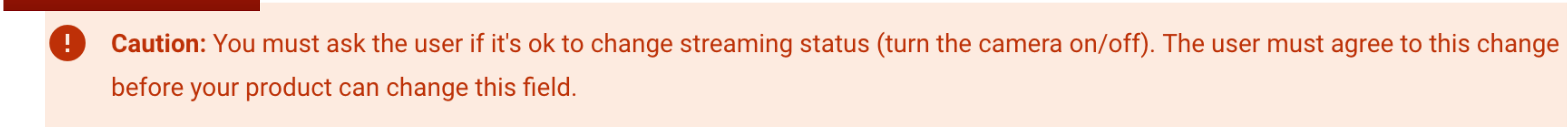
device$_n$

Data Store

Permissions protect reads/writes to high-security variables (e.g., Camera ON/OFF, user home/away)

Remember Access Control?

Nest Developer Documentation



Caution: You must ask the user if it's ok to change streaming status (turn the camera on/off). The user must agree to this change before your product can change this field.

Next..

- Chaining effects of automations
- Platform Defense Mechanisms

- Quiz #2!