DDIFT: Decentralized Dynamic Information Flow Tracking for IoT Privacy and Security

N. Sapountzis, R. Sun, D. Oliveira,

University of Florida,

February 2019

UNIVERSITY of FLORIDA

Warming Up


spyware
spam
phishing
trojan
data
security
virus
attack
malware
virus alert!
virus detected
25%

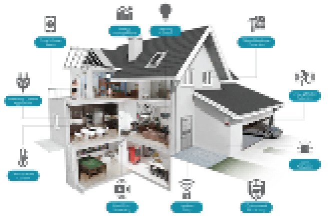
Overview

1. *Intro*: IoT era and DIFT
2. *DDIFT 1st step: Mobile phone running DIFT [fast timescale]*
3. *DDIFT 2nd step: Cloud running forensics [slow timescale]*
4. *DDIFT: Overview*
5. *Simulation* of DDIFT
6. *Conclusions*

1. Intro

2019: The dominance of IoT in human life is a reality

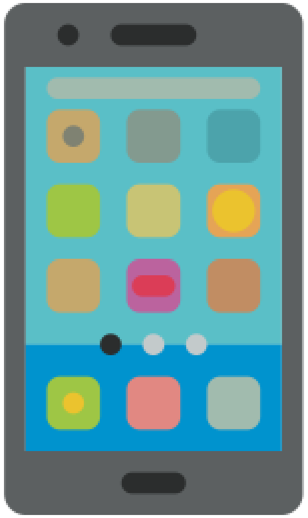
- Wearables
- Smart homes
- Healthcare
- Greening ecosystem



1. Intro

… usually *applications*, or *applets*, running at the mobile phone, is the interface to manage these devices

- Wearables *--> applet to upload to Doogle Dr. all new pics*
- Smart homes *--> open thermostat when I approach home*
- Healthcare *--> notify my doctor if my heart rate has improper impulses*
- Greening ecosystem *--> provide PV array analytics*



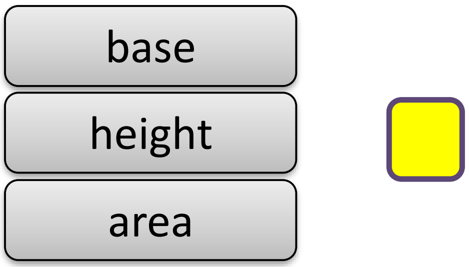
1. Intro

1. Intro: traditional DIFT

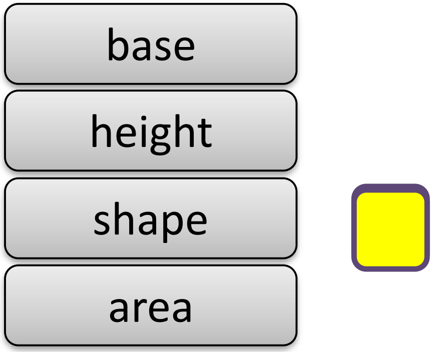
DIFT: works with 2 processes

- PROCESS A: **Tag insertion: insert tags to variables (or memory bytes)**
- PROCESS B: **Tag propagation: propagate tags during system execution**
  - *Direct Flow Propagations (DFP)*
  - *Indirect Flow Propagations (IFP)*

1. int base = 5; //the base of the triangle
2. int height = 4; //the base of the triangle
3. int area = base * height; //the area



1. int base = 5; //the base of the triangle
2. int height = 4; //the base of the triangle
3. int shape = “triangle”; //type of shape
4. (If shape == “triangle”)
5. area = base * height //the area



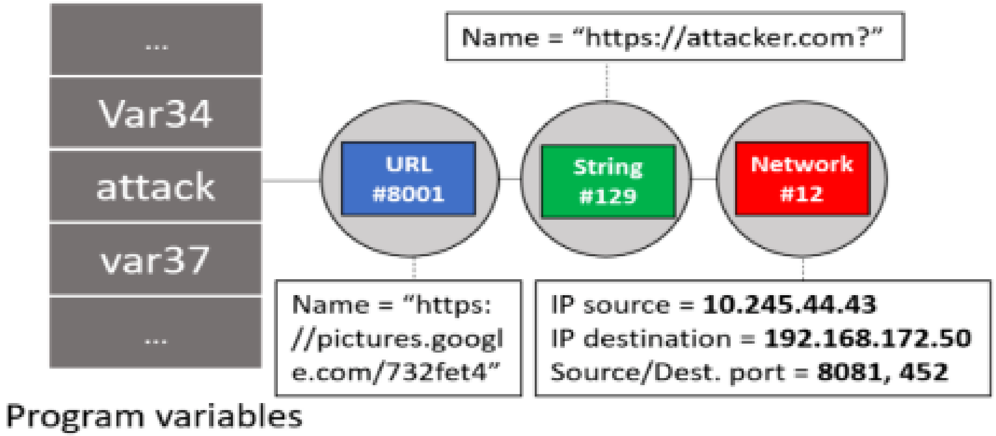
2. DDIFT Algorithm at device

What about DIFT in IoT?

2. Device: running DIFT at the app level

Challenges while applying DIFT in IoT [Tag Insertion]: :

1. **Different system activities add heterogeneous knowledge in the information flow and should add different level of security concerns**
   - **We consider tag differentiation (many colors), e.g., network, file, RV tags**
   - **Each variable has a list that accommodates up to N #of tags**
2. **Ability to reverse engineer back to the inputs**
   - **We keep provenance information (e.g., network tag coming from IP 13.2.3.5)**



2. Device: running DIFT at the app level

Challenges while applying DIFT in IoT [Tag Propagation]: (cont')

3. Limited (memory, battery resources).

- **Algorithm 1 (next slide): How to optimally allocate the limited resources to the vast #tags that attempt to propagate?**

4. Minimize false alarms that DIFT usually bring.

- **Algorithm 1 (next slide): Should all indirect flows be propagated?**

2. DDIFT Algorithm at device

Algorithm 1 Optimal resource allocation and indirect flow decisioning.

Input. $\mathcal{C}$: the objective metric we attempt to optimize.
Output. $\Delta(n_{t,i})$: drop or schedule the tag.

1: Define the objective metric we attempt to optimize based on the per-tag metric value $c_{t,i}$, the tag type weights λ, and the tag weights μ.

$$\mathcal{C} = \sum_t \lambda_t \sum_i \mu_{t,i} \cdot c(n_{t,i}) \tag{1}$$

2: In order to decide about the potential propagation of a tag, the DIFT system should consider which decision offers the best gain for $\mathcal{C}$. To that end, we differentiate $\mathcal{C}$ with respect to $n_{t,i}$ (number of copies of the i-th tag belonging at the t-th type), we discretize and obtain:

$$\begin{aligned}\Delta(\mathcal{C}) &= \sum_t \lambda_t \sum_i \mu_{t,i} \frac{\partial c(n_{t,i})}{\partial n_{t,i}} \Delta(n_{t,i}) = \\ &= \sum_t \sum_i U_{t,i} \Delta(n_{t,i}),\end{aligned} \tag{2}$$

where:

$$U_{t,i} = \lambda_t \cdot \mu_{t,i} \cdot \frac{\partial c_{t,i}}{\partial n_{t,i}} \text{ is the utility of tag \{t,i\}} \tag{3}$$

$$\Delta(n_{t,i}) = \begin{cases} -1, & \text{if the tag \{t,i\} is dropped} \\ 0, & \text{if no action for the tag \{t,i\} is taken} \\ +1, & \text{if the tag \{t,i\} is scheduled} \end{cases} \tag{4}$$

3: For *resource allocation*: the DIFT should (i) schedule (i.e., keep or propagate) the tags in the order of decreasing $U_{t,i}$, and (ii) drop (i.e., delete or not propagate) the tags with the lowest $U_{t,i}$ (i.e., to ensure that the tags "carrying" more information are prioritized).

4: For *indirect flow* propagation decisioning: the DIFT should propagate a tag if this tag has $U_{t,i} > 0$ (i.e., the indirect flow propagation brings information to the DIFT).

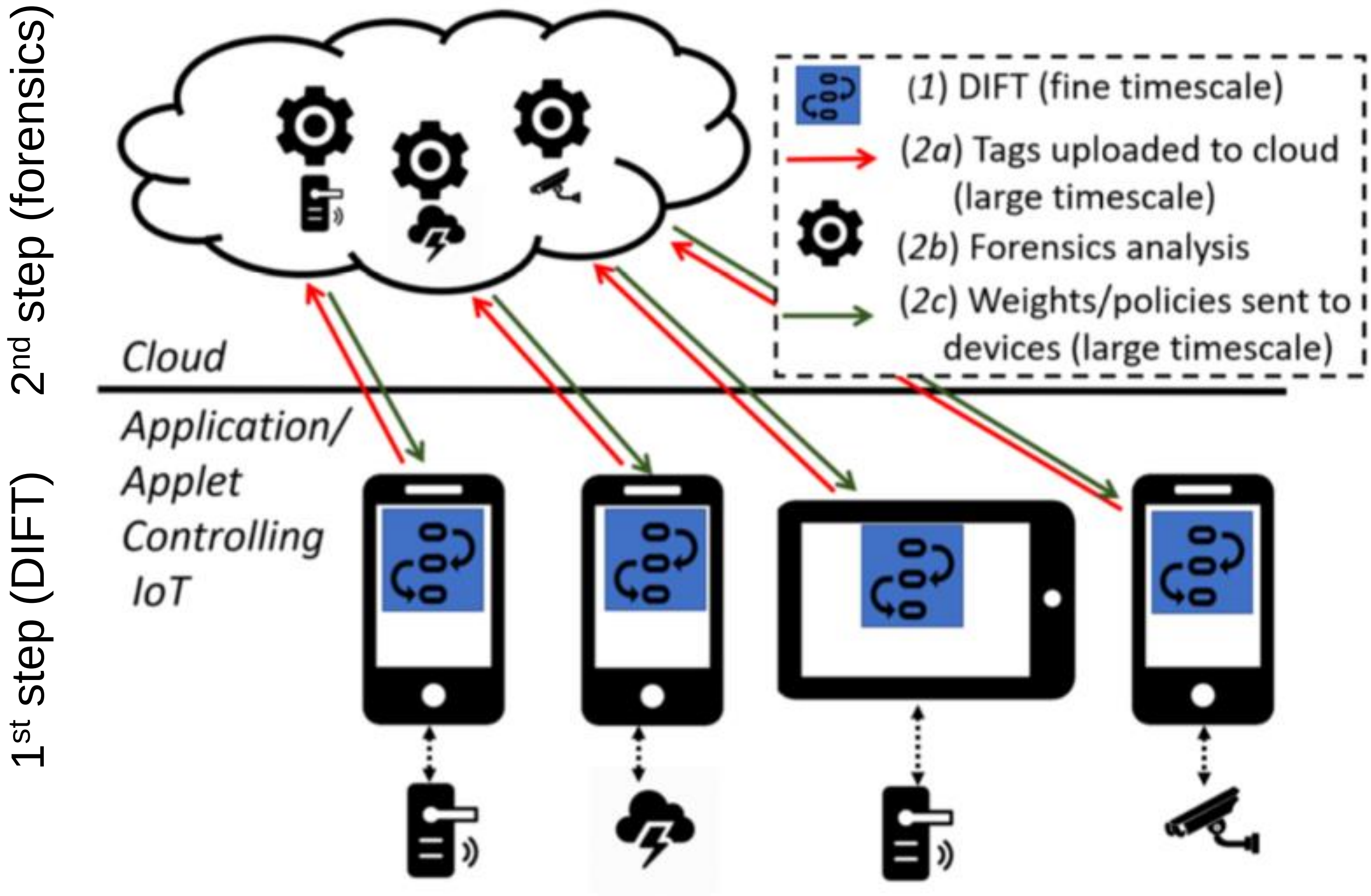
3. Cloud: running forensics

- Cloud performs heavy forensics analysis in a slow timescale, relying on a continuous analysis of a large volume of tags
- Cloud's main objective is to:
  - dictate the best values for the weighting parameters λ, μ, so e.g. the devices boost the important tags

$$\lambda = \lambda_{prev} + \zeta \cdot \sum_i \mu_{t,i} \cdot c(n_{t,i}),$$

  - develop privacy and security policies,
    - e.g. URL tag + String tag + netflow tag = URL Attack
    - learn a priory what strings, network connections etc. are suspicious

4. **DDIFT**: Decentralized Dynamic Information Flow Tracking



5. Simulations (N=5): traditional DIFT (propagate all IFs ☹)

1. na
2. nu
3. di
4. fo
5.
6.
7.
8. ex

exit

#89

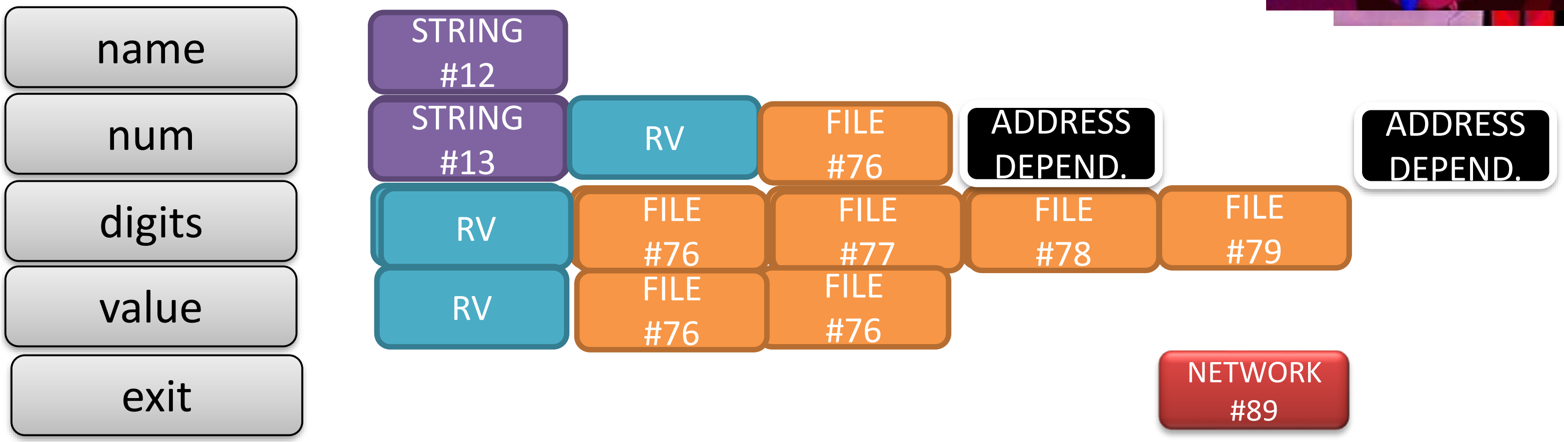
5. Simulations (N=5): traditional DIFT (propagate all IFs ☹)

All IFs propagated
=>
Over-tainting (+ memory pollution)

No one of the IF is propagated
=>
Under-tainting

Other over-tainting cases (when more space is available)
=>
Aggressively propagate tags

5. Simulations (N=5): DDIFT (α-fair utility) ☺



Simulated different provenance-list sizes scenarios:
Detection efficiency improvement 43%
Memory usage improvement 71%

6. Conclusions

- **DDIFT: dynamically tracks the information flow at the mobile device level, and adapts to the IoT challenges:**
  - ***Large #devices:*** Scalable scheme through the synergy of cloud (slow timescale) and device (fast timescale)
  - ***IoT limited resources:*** *optimally* prioritizing tags
  - **[open problem] Indirect Flow Propagation**: tackled with optimization theory
  - **Extendable to software and hardware**
  - **Able to design malware signatures through the tag confluwnces and further detect them**

Thank you!