INDIANA UNIVERSITY

SCHOOL OF INFORMATICS AND COMPUTING

Center for Security Informatics
Bloomington

Toward Securing Sensor Clouds

Apu Kapadia, Steven Myers,
XiaoFeng Wang and Geoffrey Fox

School of Informatics and Computing
Indiana University, Bloomington

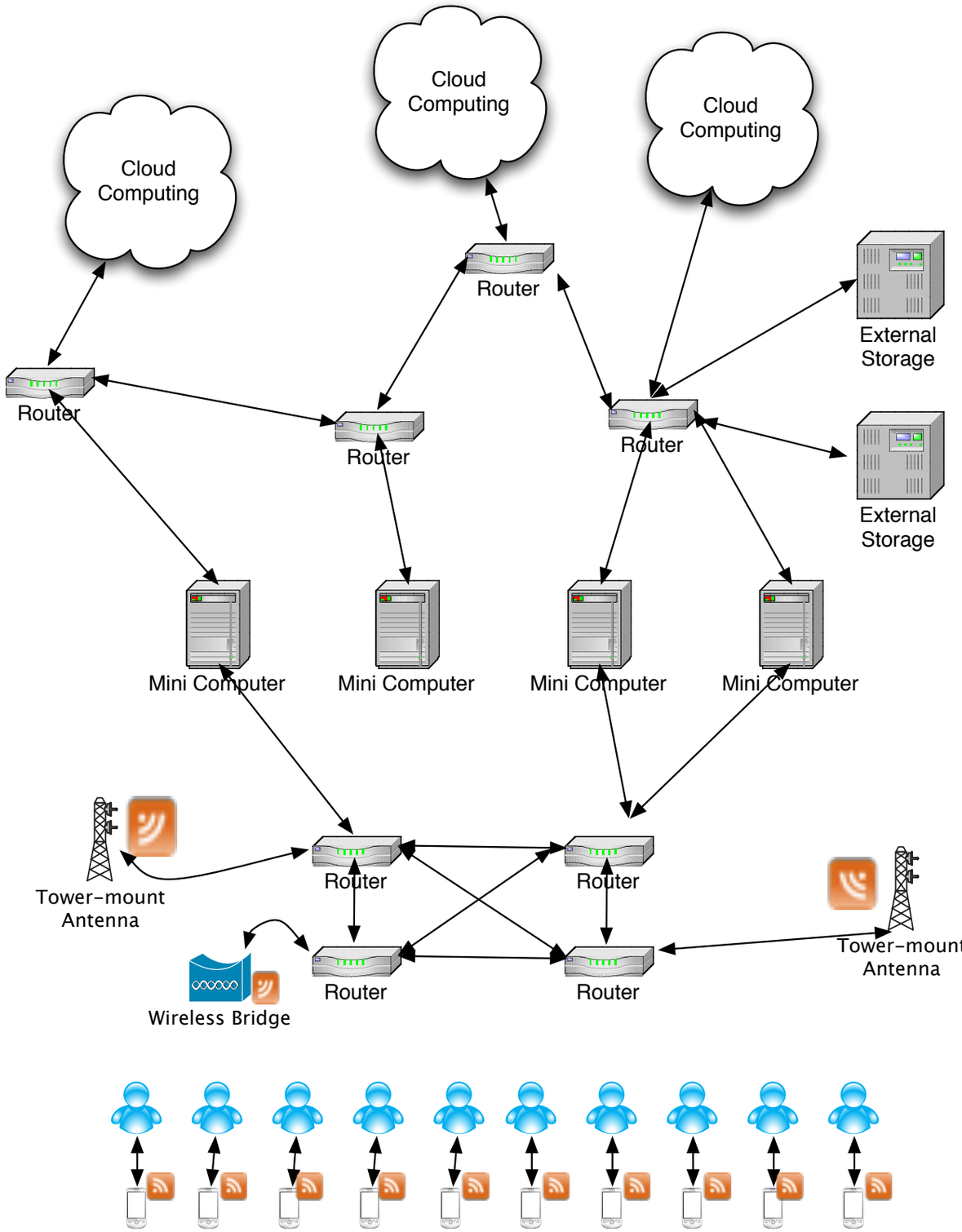
Sensor Model: (Not a Mote)

- Motorola Droid X, X 2
- Android OS
- 1 GHz Cortex-A8 processor
- 512 MB Ram
- Sensors
  - WiFi 802.11b/g
  - Bluetooth
  - Accelerometer
  - GPS
  - Touch Screen
  - Camera (8 MP), 720p HD
  - Audio
  - Magnetometer
  - Proximity
  - Ambient light
- 32 GB microSDHC storage

Image from http://hothardware.com/News/Leaked-Motorola-DROID-X-2-Daytona-Features-1GHz-Tegra-2-HigherRes-Screen/

INDIANA UNIVERSITY

Our model of a "sensor cloud"



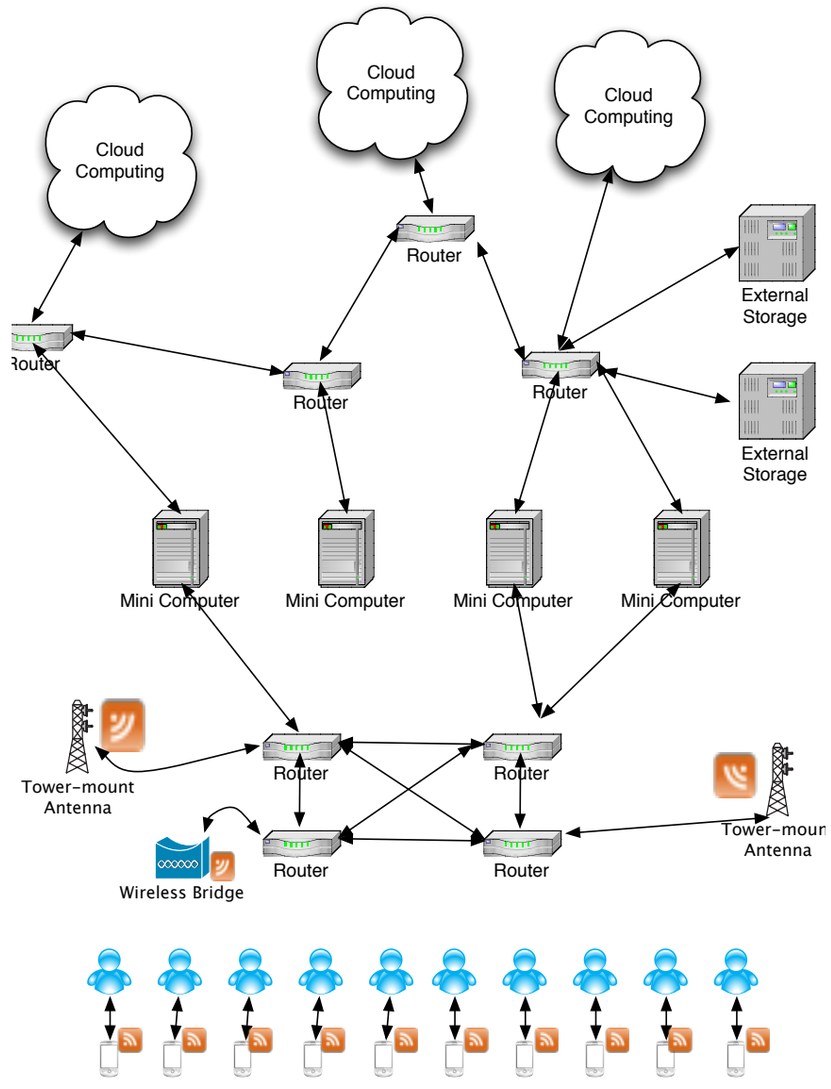
INDIANA UNIVERSITY

Opportunistic Sensing

- Leverage human mobility for broad coverage
- Monitor traffic
- Track military assets
- People flow (amusement parks, campuses)
- Health monitoring
- Social computing: micro surveys, amber alerts

Security Threats

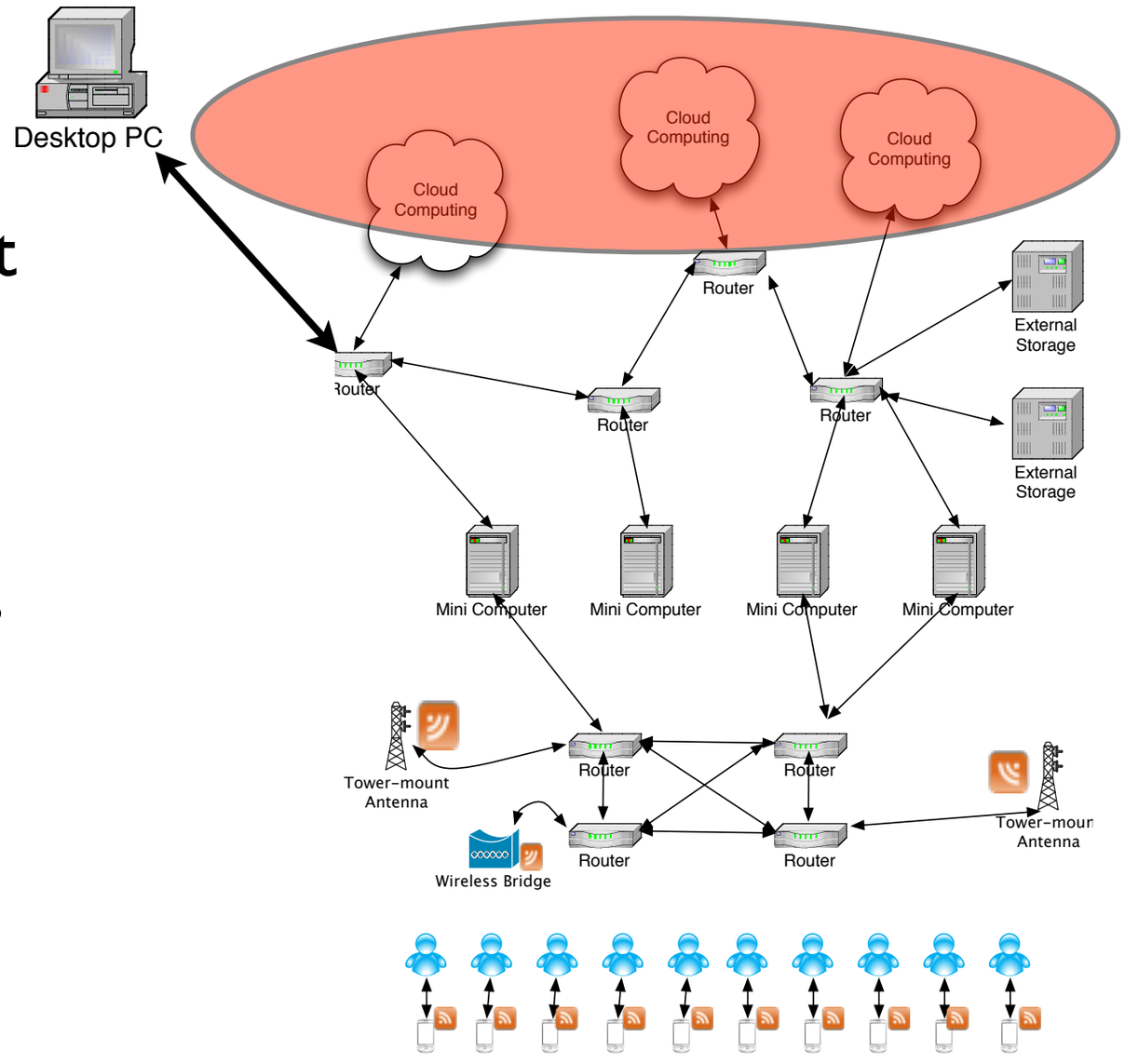
1. Cloud or Grid
2. Communication Channels
3. Client
4. Sensor
5. Environment



INDIANA UNIVERSITY

Threats to the Cloud

1. Cloud or Grid
    1. Information Theft
    2. Malware
    3. Covert Channels (shared CPU/ Resources)
    4. Proof of Computation?



INDIANA UNIVERSITY

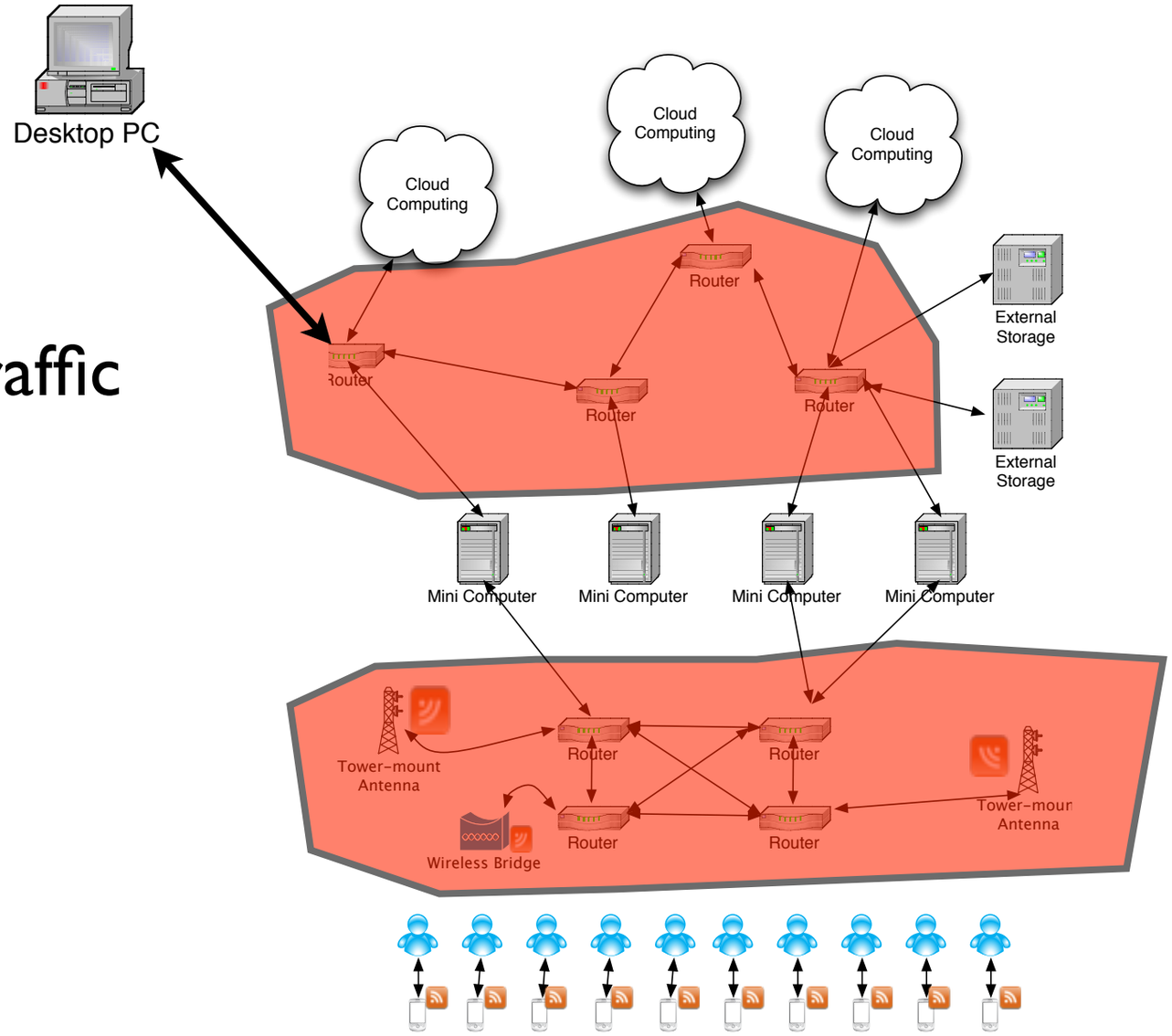
Communication Threats

2. Communication Channels

1. Eavesdropping, traffic analysis

2. Manipulation of packets

3. Denial/Delay Of Service



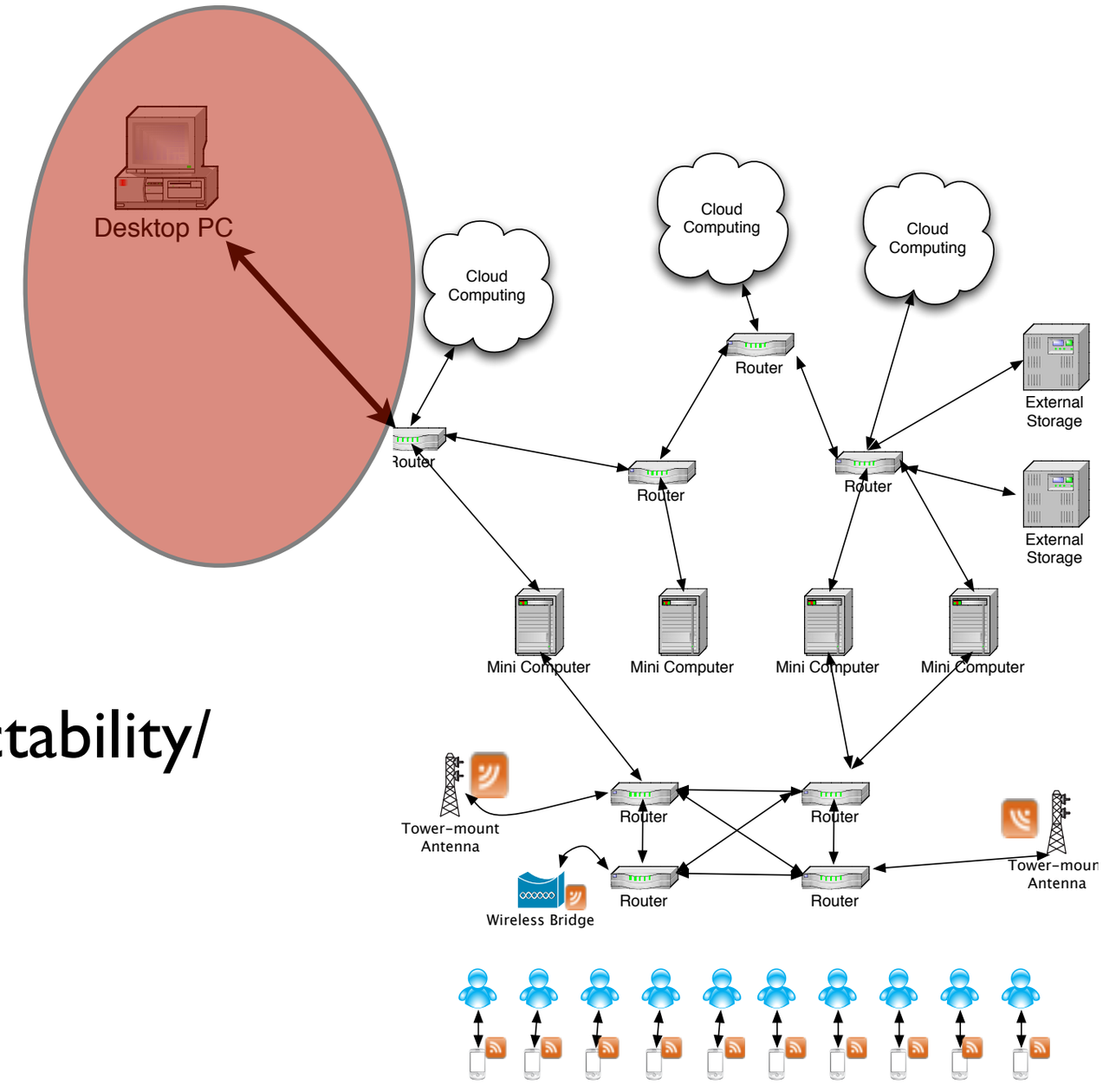
INDIANA UNIVERSITY

Client Threats

3. Client

1. Malware

2. Human Predictability/ Fallibility



INDIANA UNIVERSITY

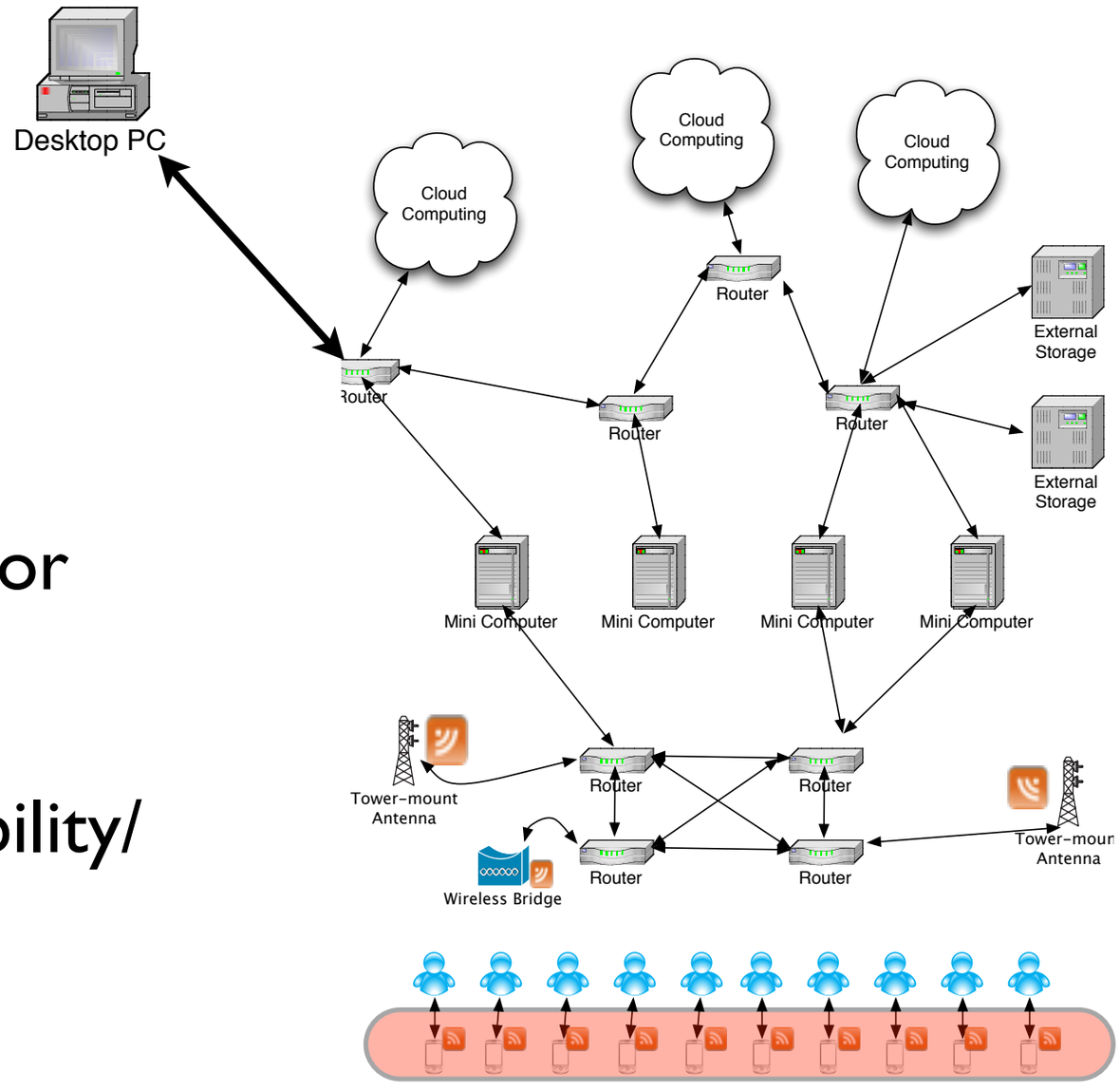
Sensor Platform Threats

4. Sensor

    1. Malware/Viruses

    2. Sensor data lost or stolen

    3. Human Predictability/ Fallibility



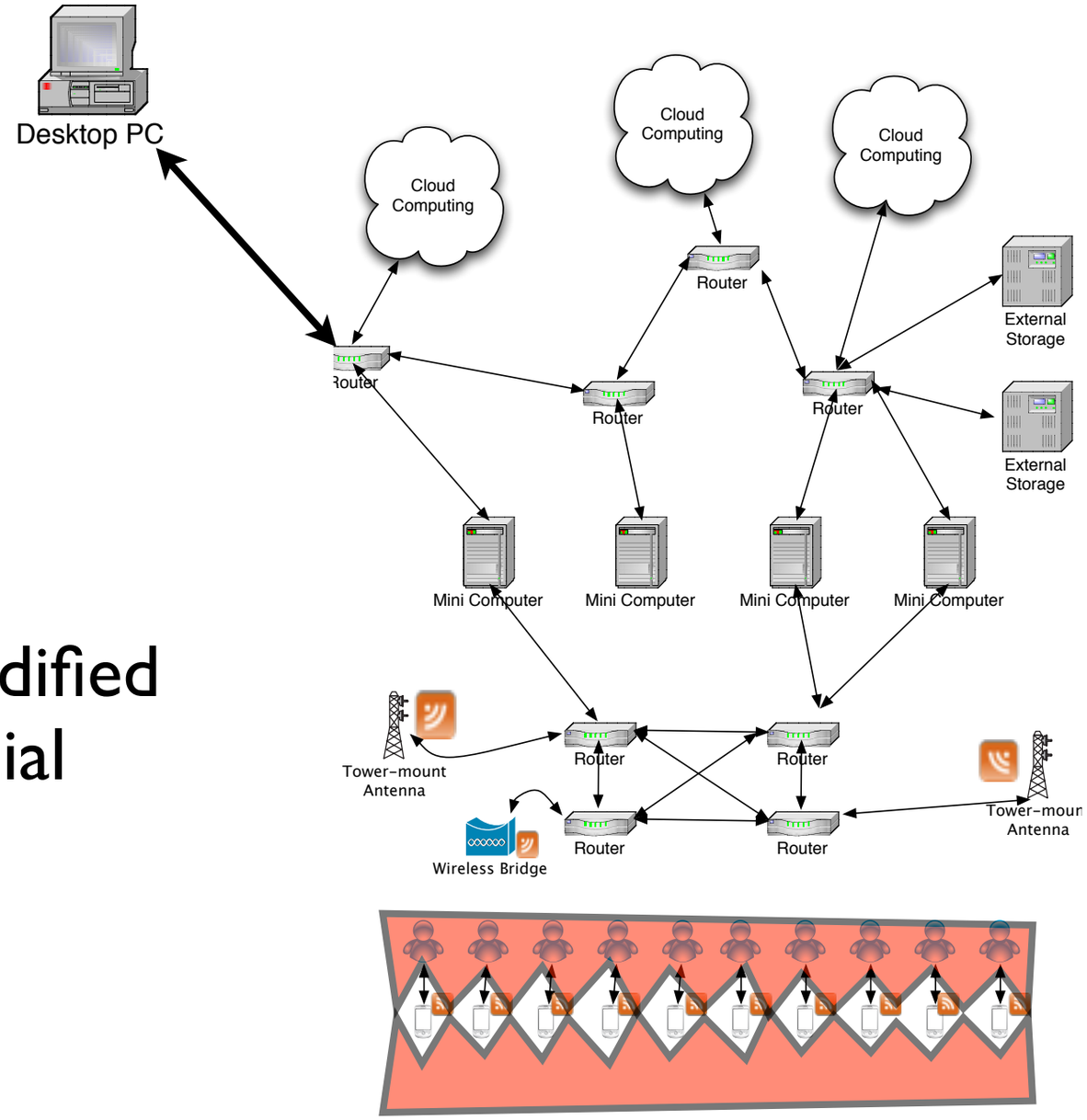
INDIANA UNIVERSITY

Environment Threats

5. Environment

    1. Sensor stolen or repositioned

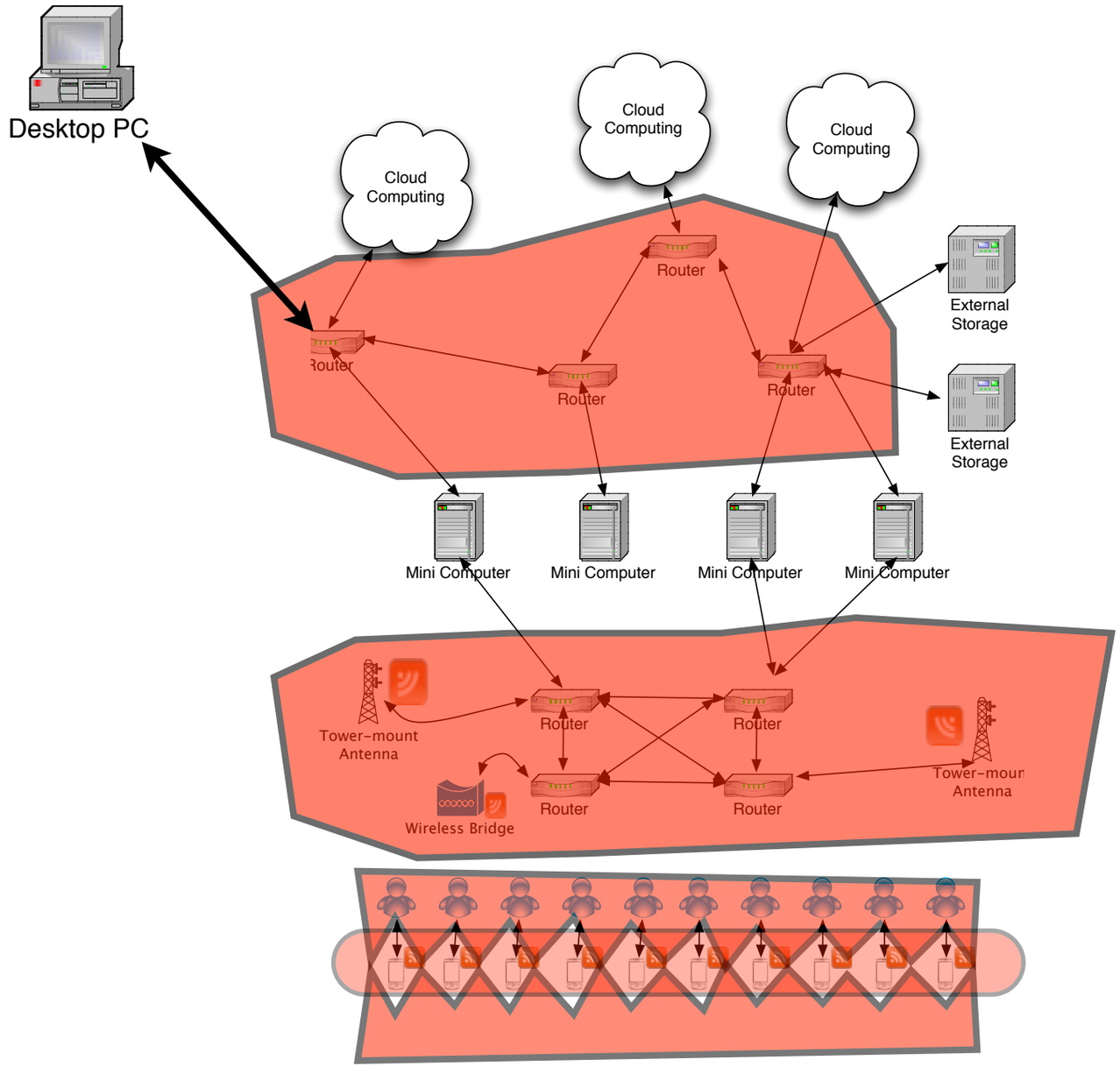
    2. Environment modified to provide artificial sensor readings



INDIANA UNIVERSITY

Focus of our Research

1. Cloud or Grid
2. Communication Channels
3. Client
4. Sensor
5. Environment



INDIANA UNIVERSITY

Threats to sensor platform

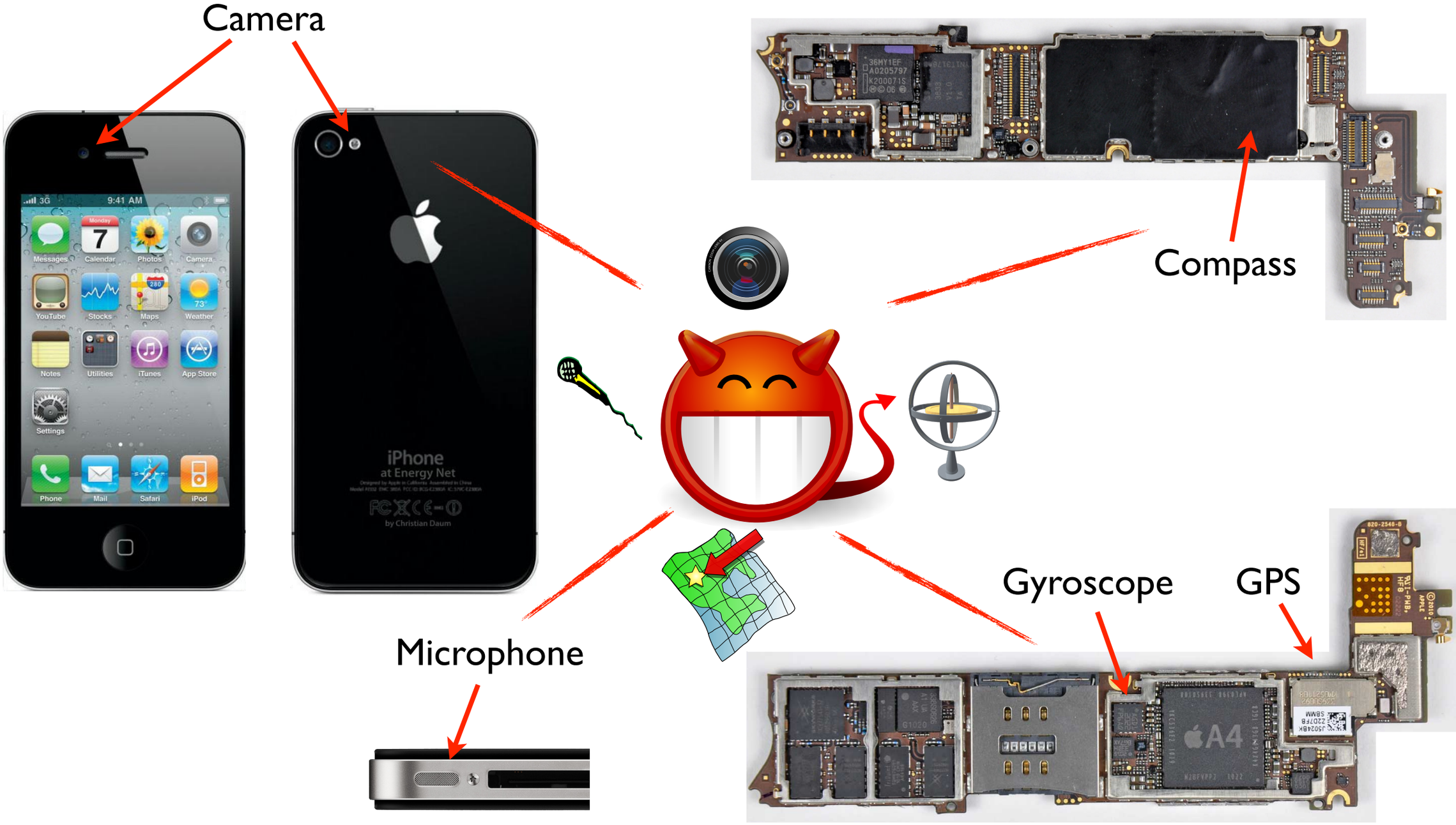
"Sensory Malware"

Sensor platform is compromised

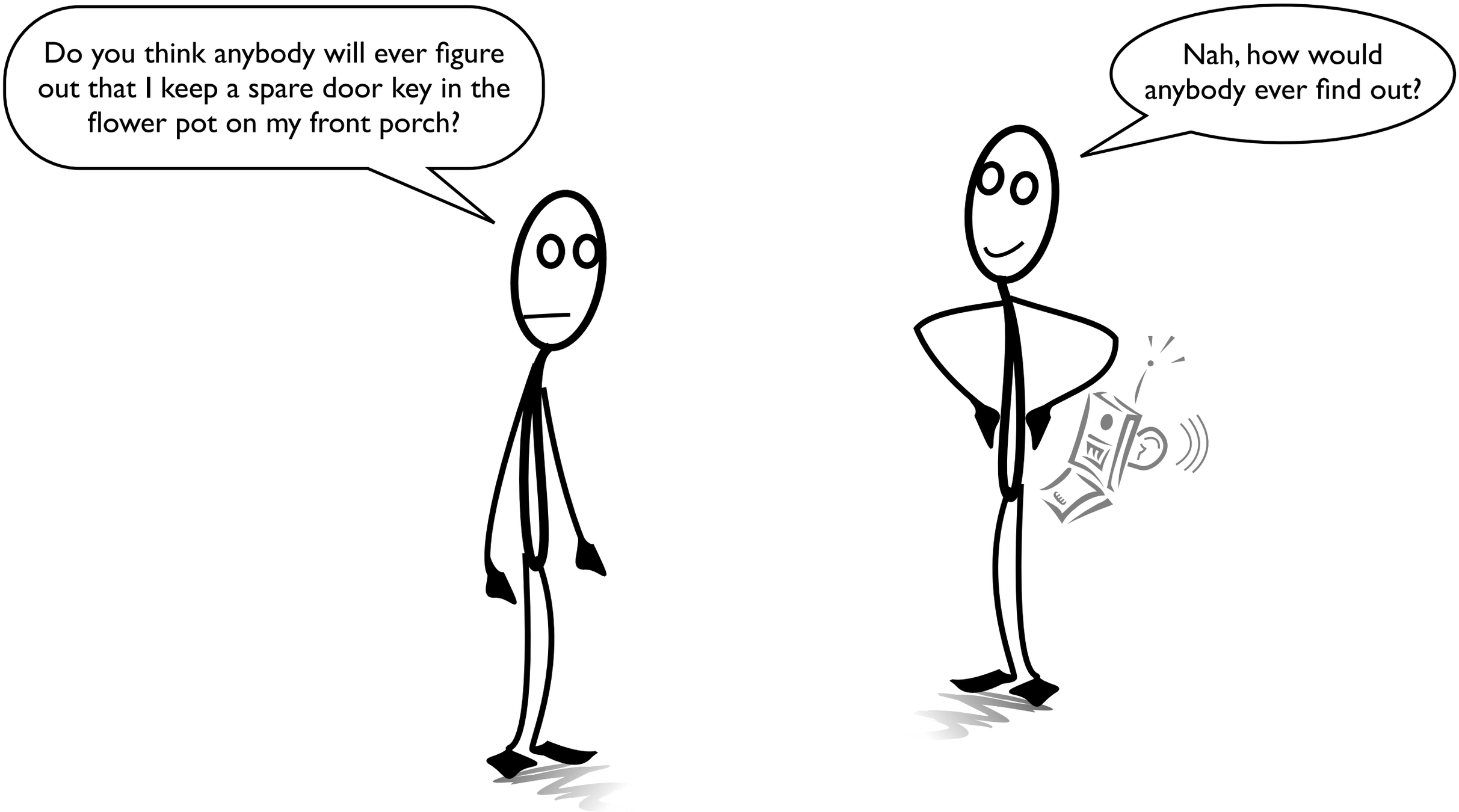
Roman Schlegel, Kehuan Zhang, Xiaoyong Zhou, Mehool Intwala,
Apu Kapadia, XiaoFeng Wang

NDSS 2011

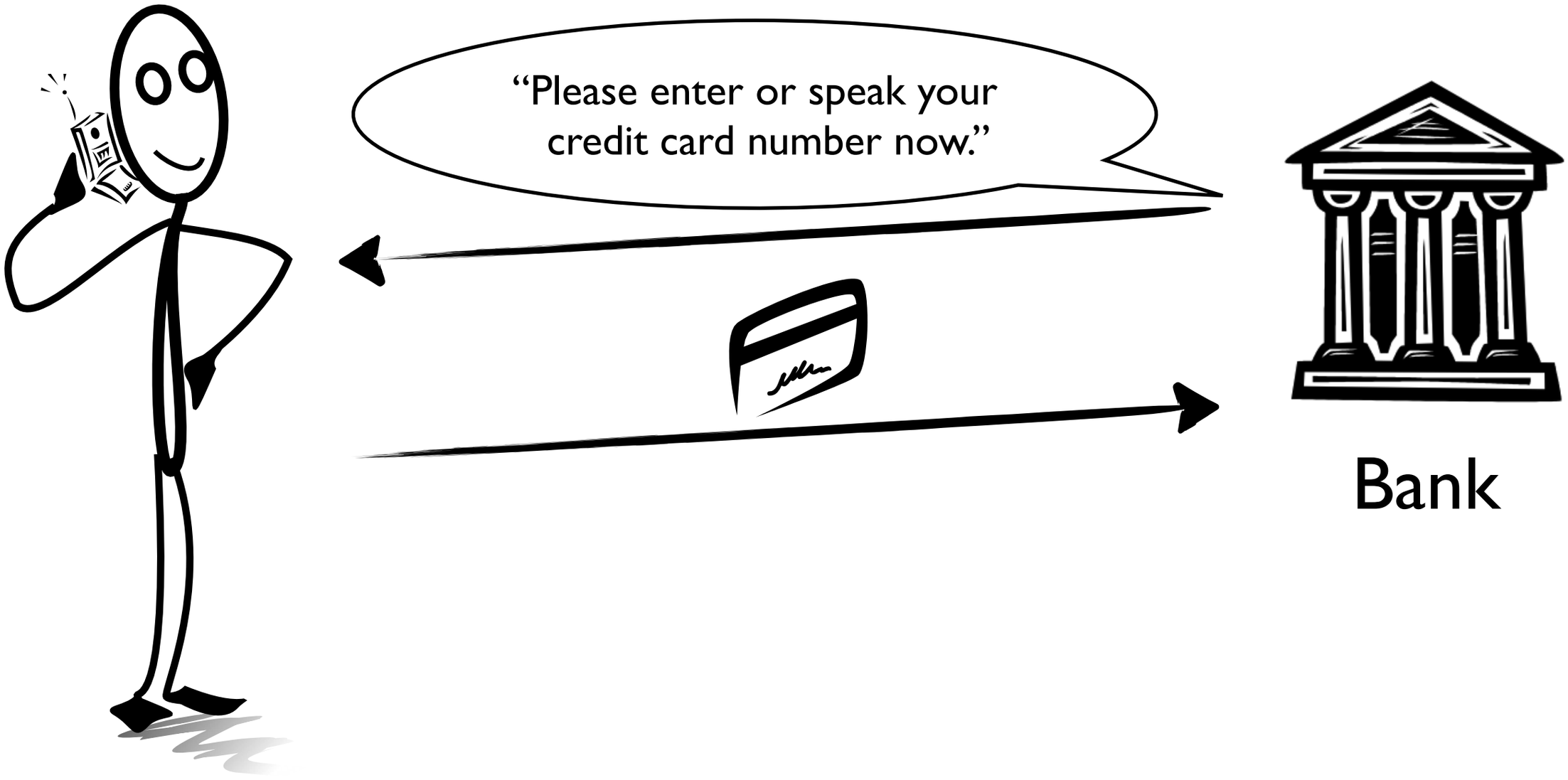
Threat of sensory malware



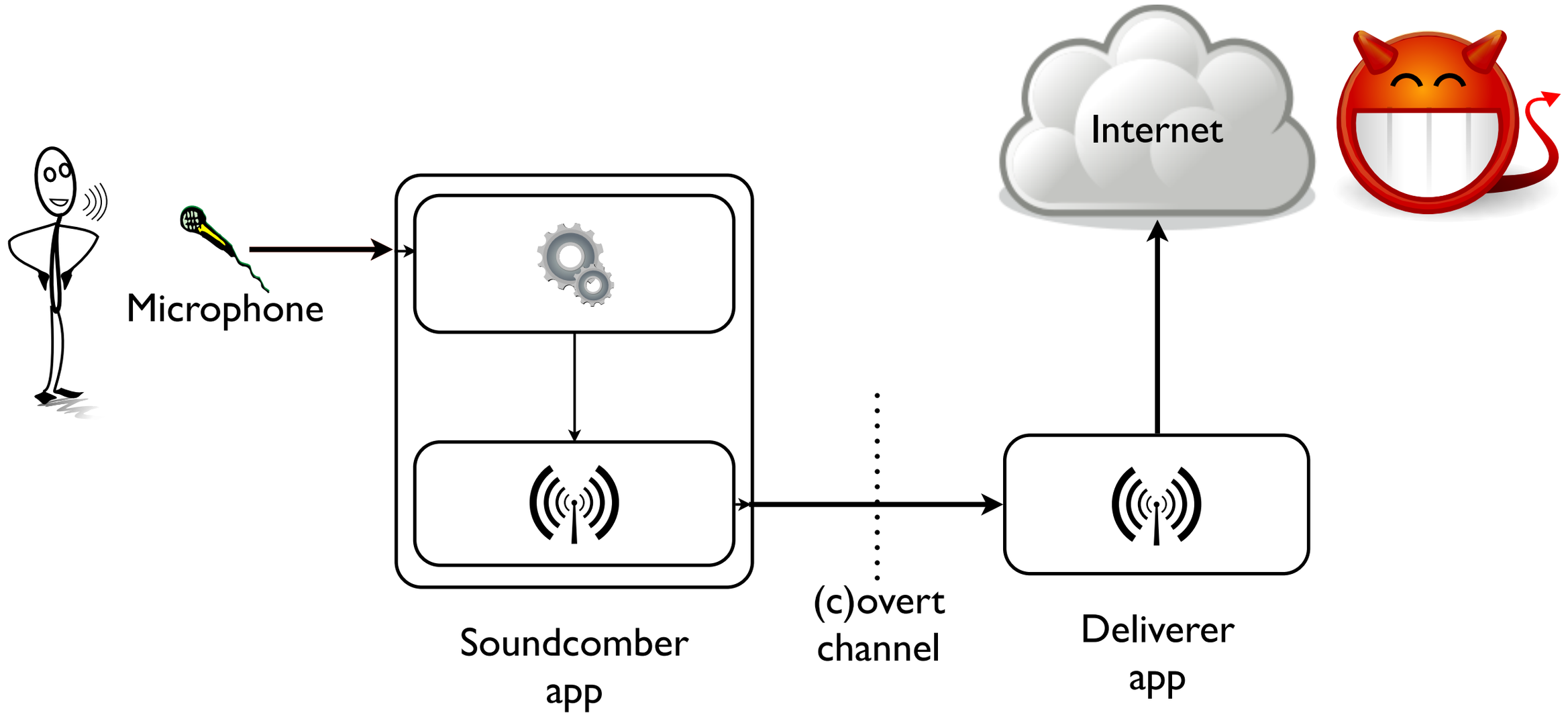
What can malware overhear?



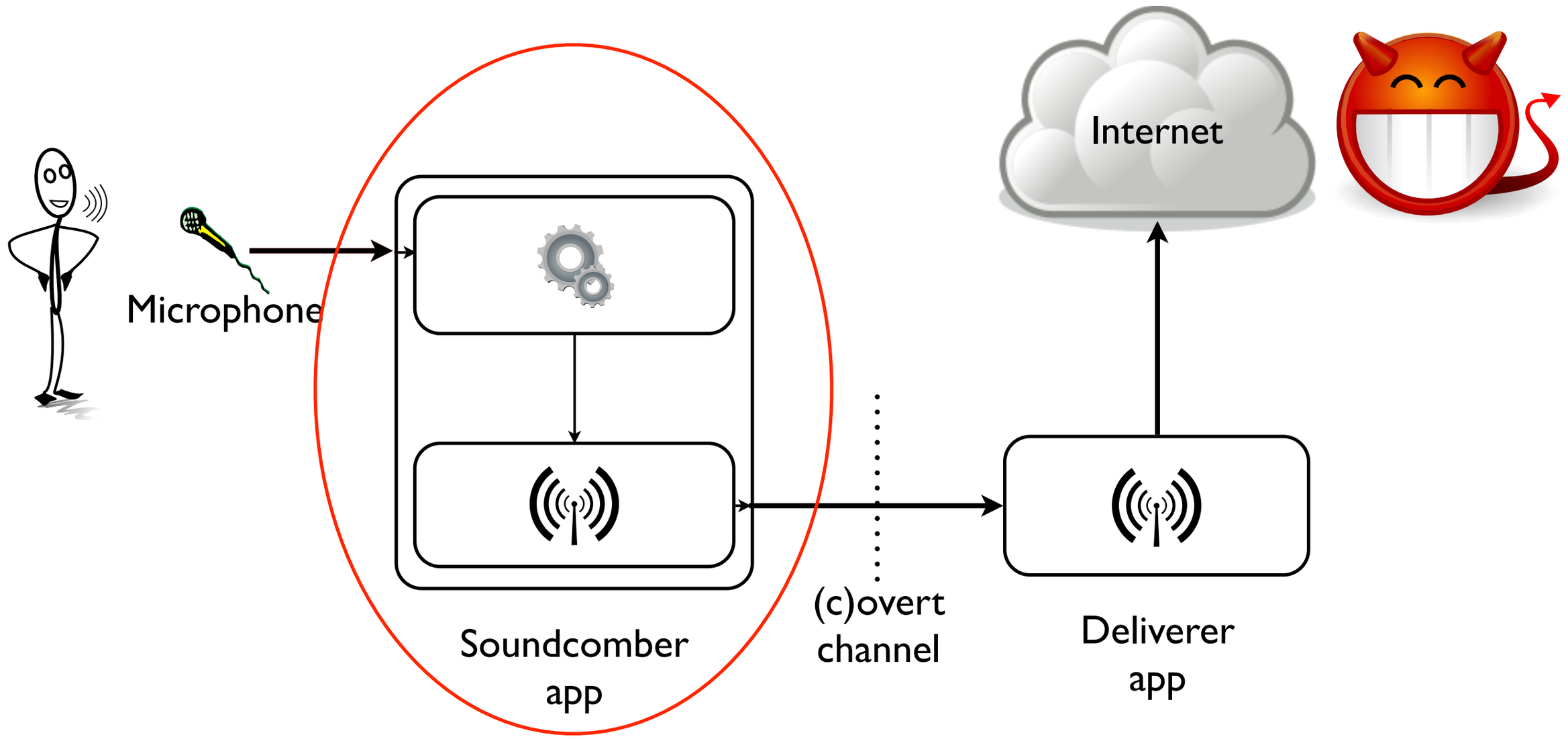
Some situations are easy to recognize



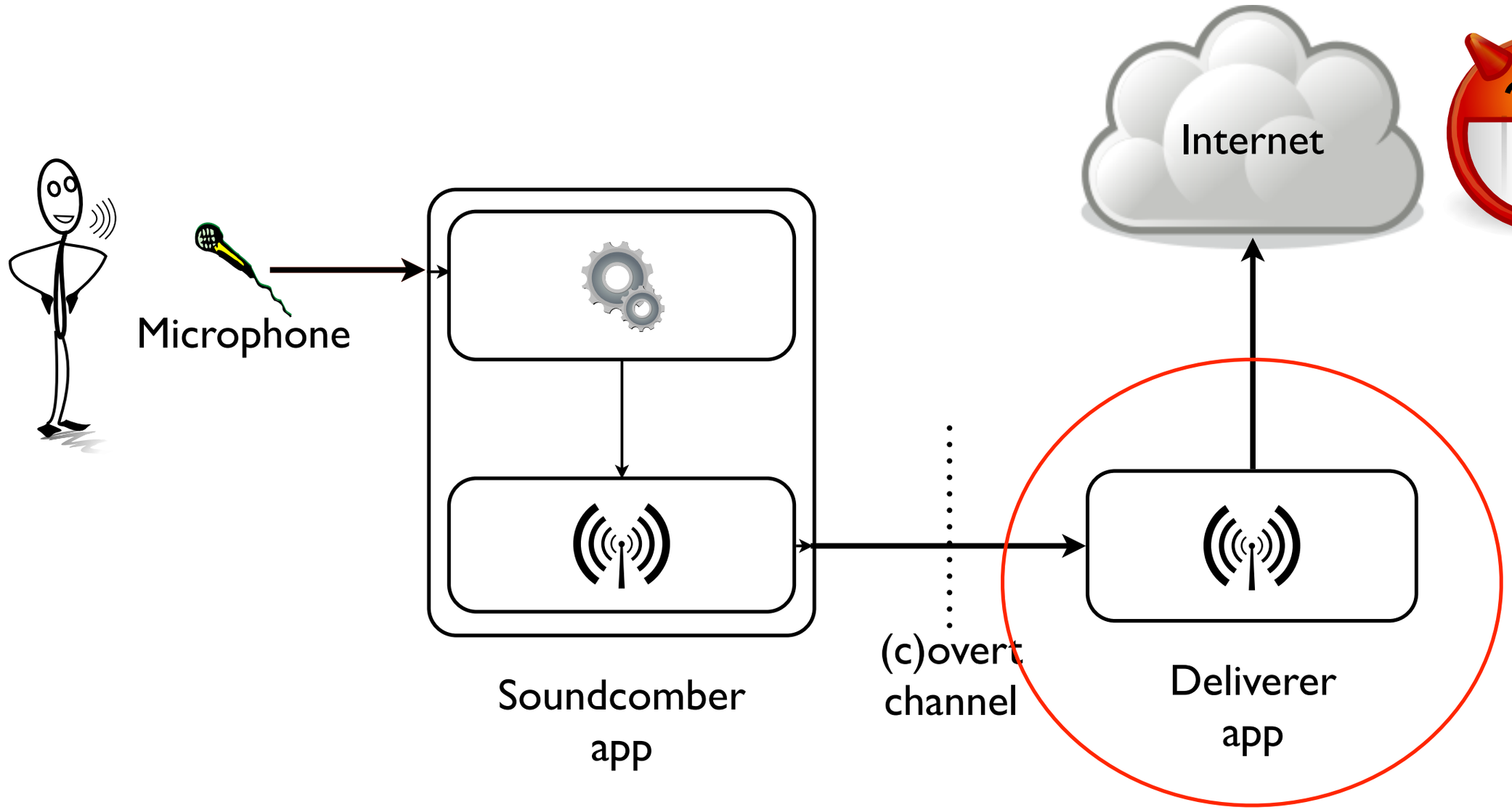
Soundcomber:
Two trojans are stealthier than one



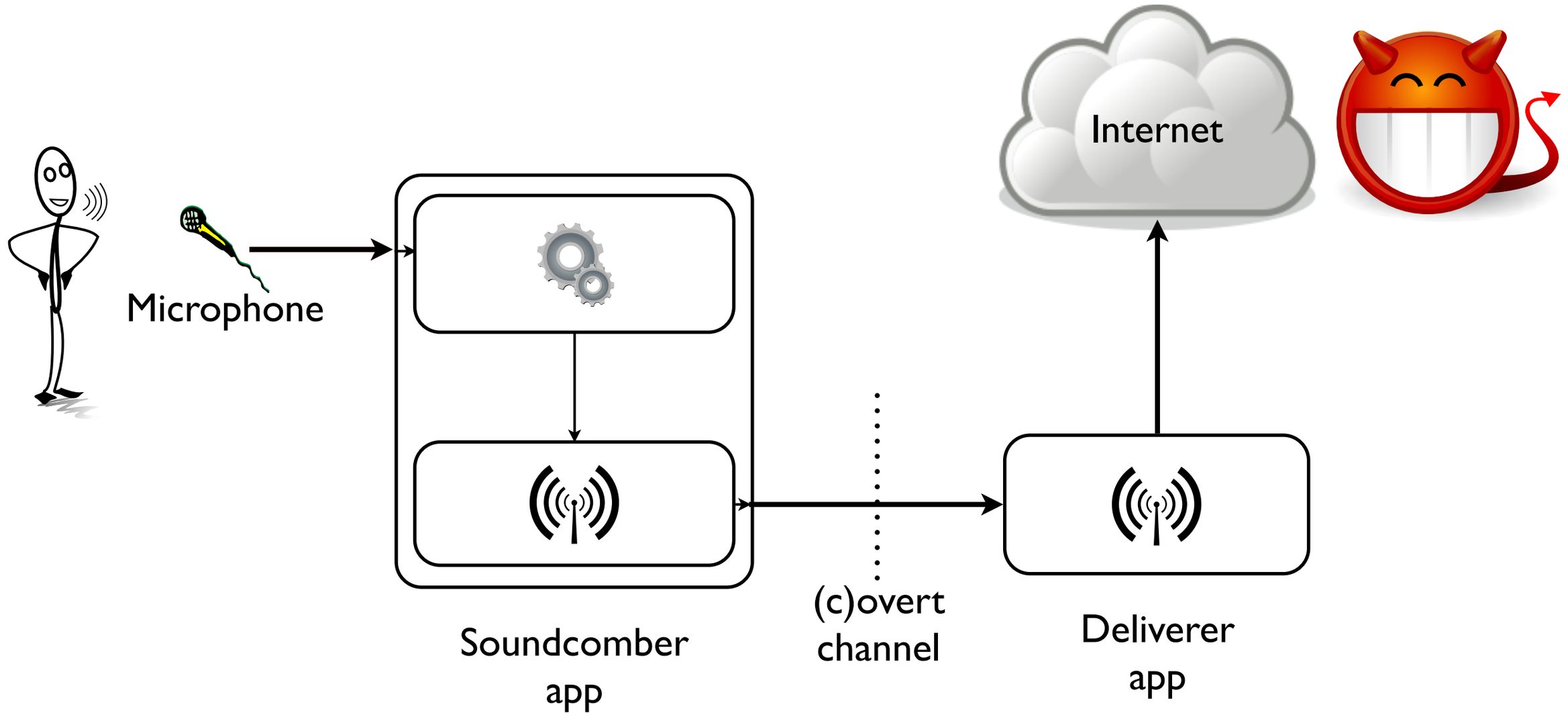
Soundcomber:
Two trojans are stealthier than one



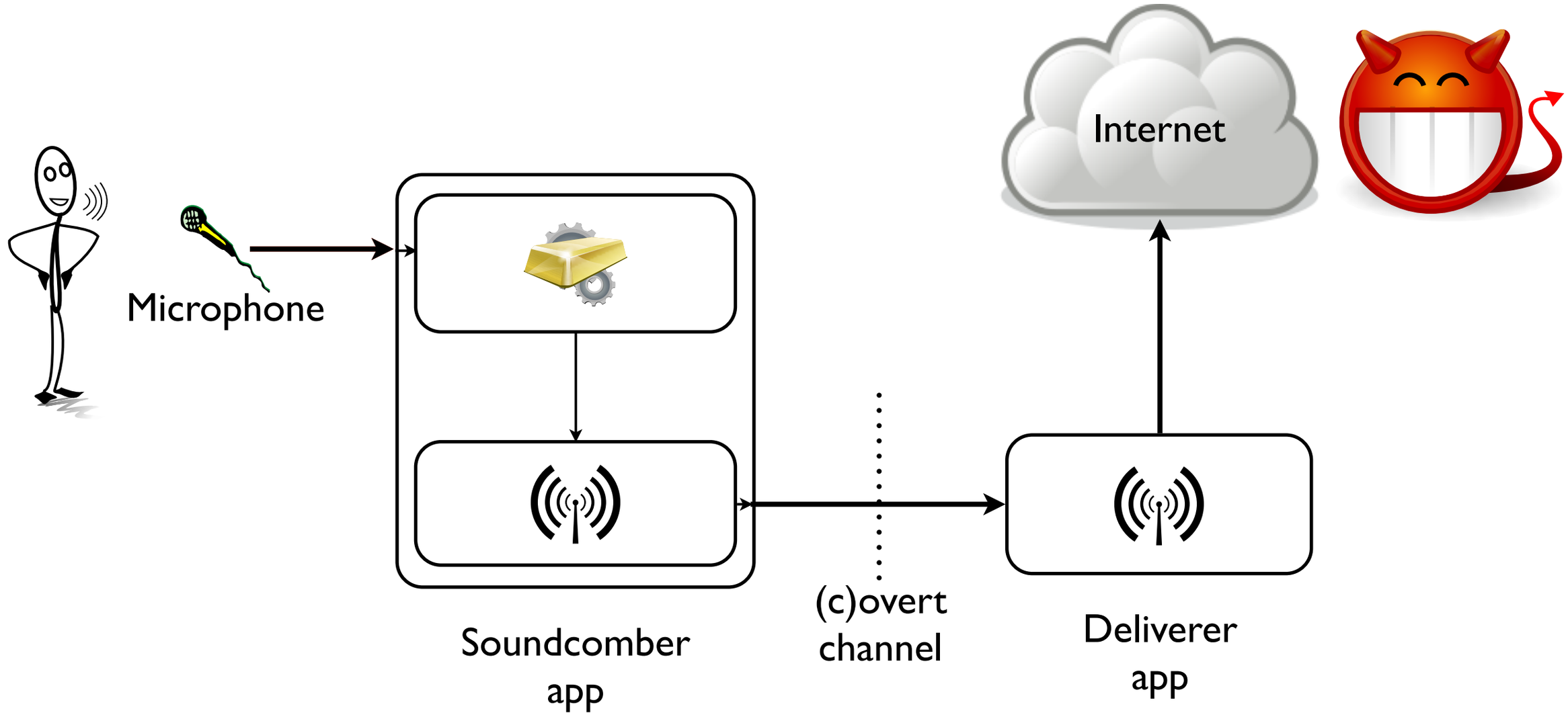
Soundcomber:
Two trojans are stealthier than one



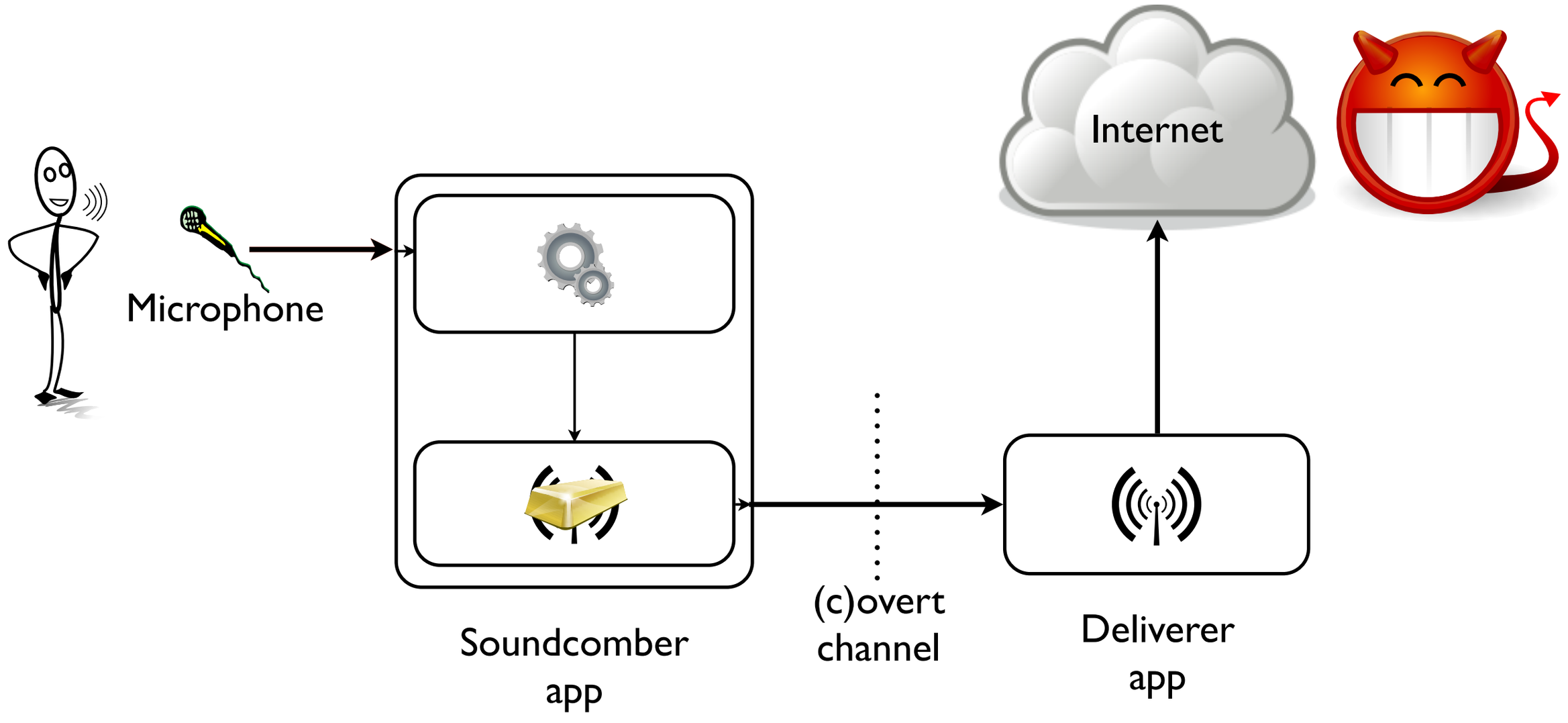
Soundcomber:
Two trojans are stealthier than one



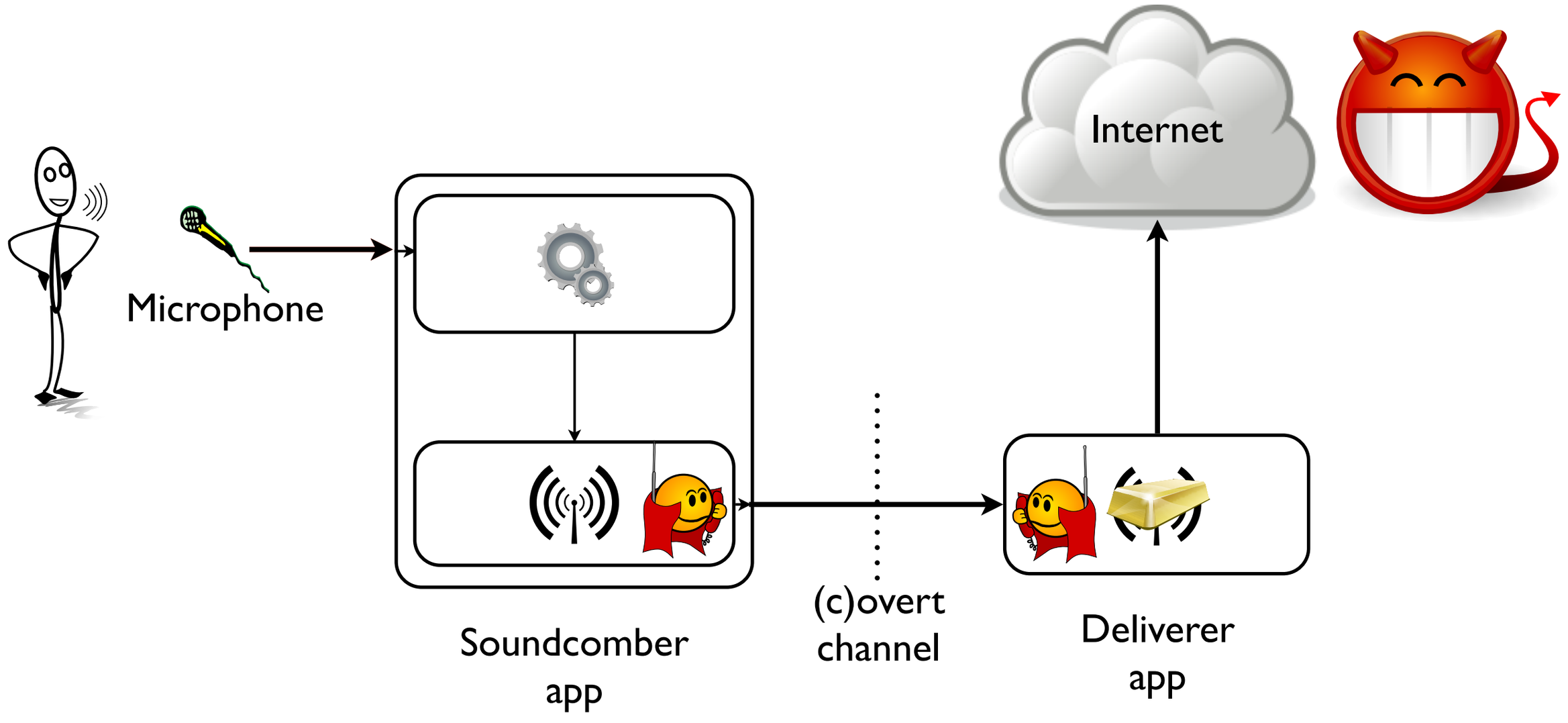
Soundcomber:
Two trojans are stealthier than one



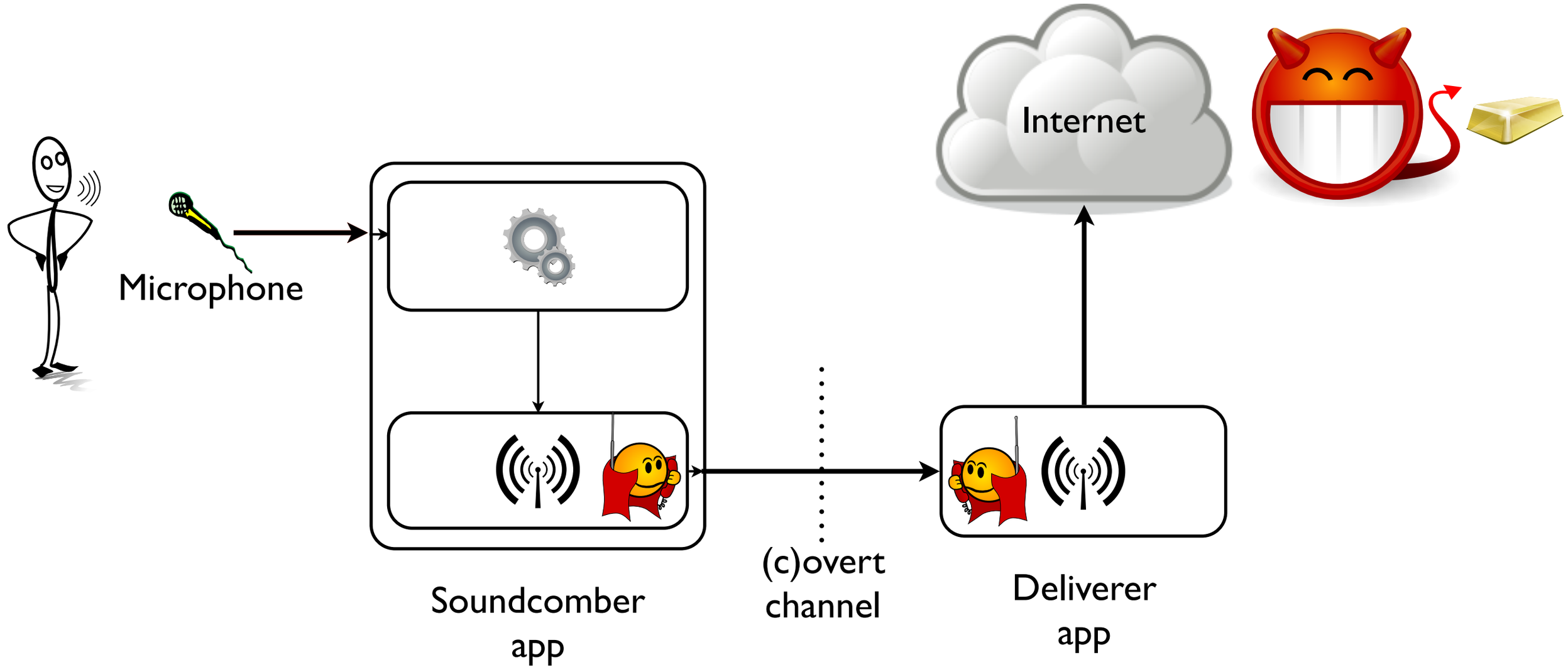
Soundcomber:
Two trojans are stealthier than one



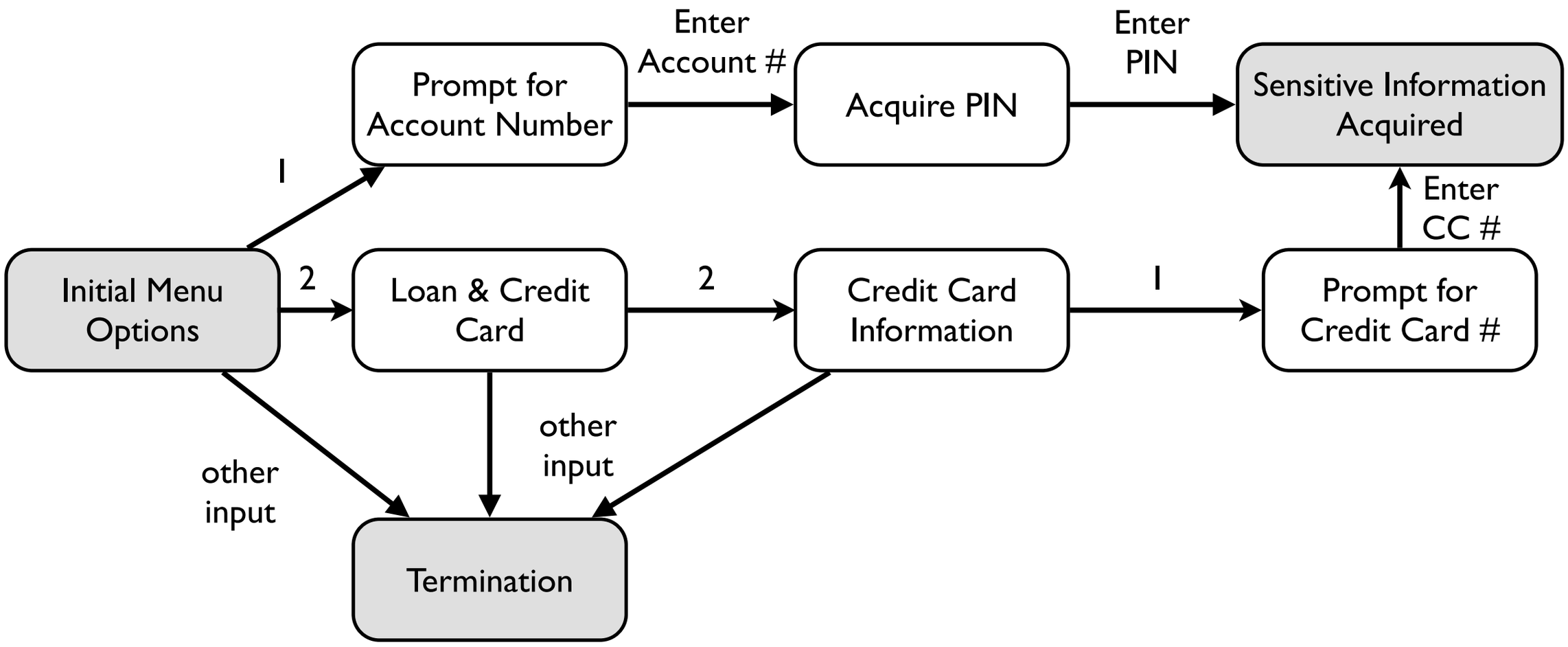
Soundcomber:
Two trojans are stealthier than one



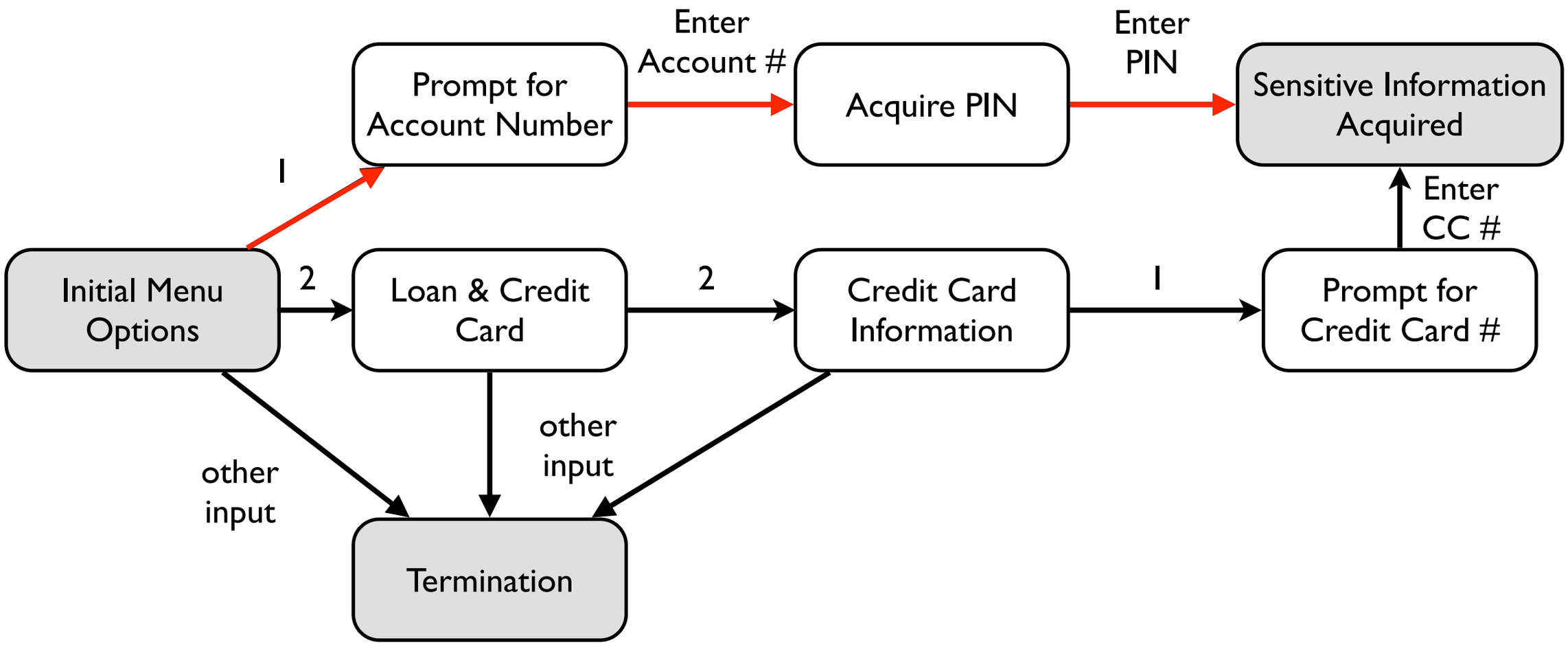
Soundcomber:
Two trojans are stealthier than one



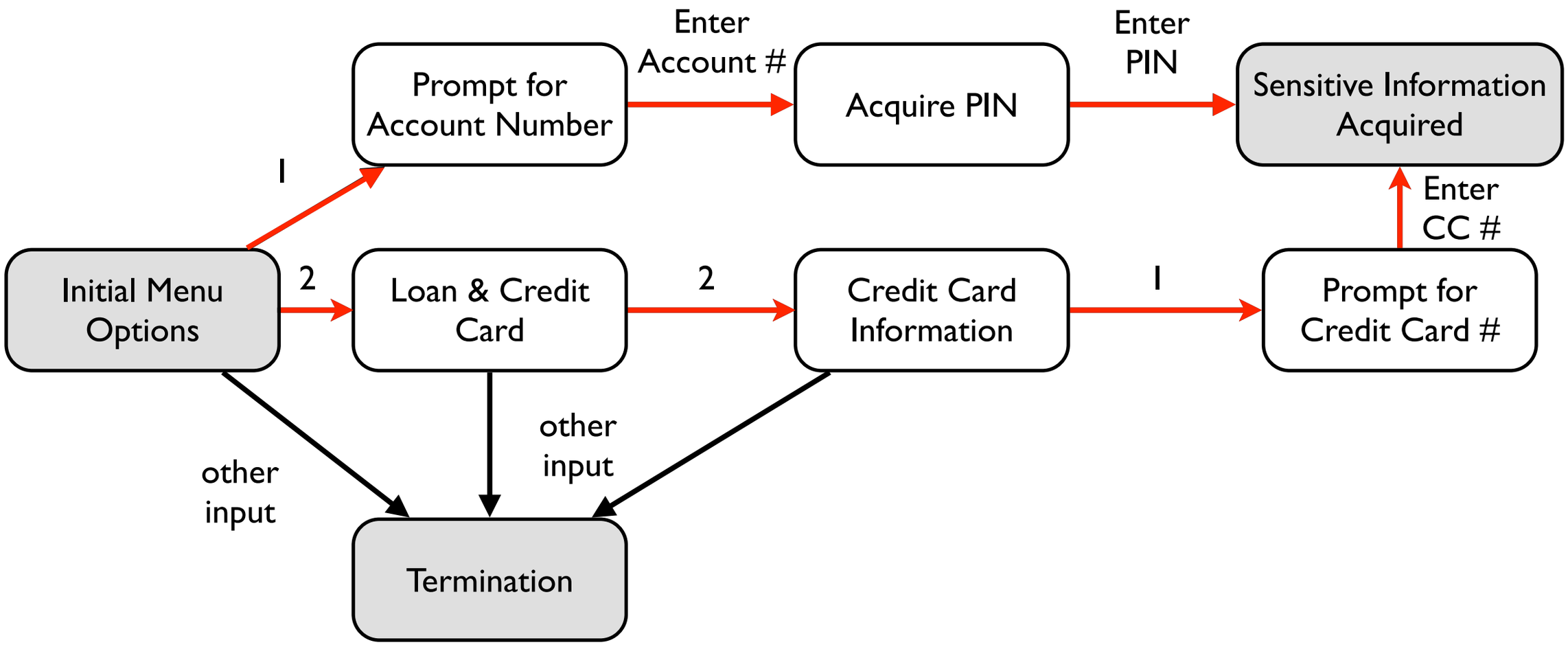
Profiles allow for context aware extraction



Profiles allow for context aware extraction



Profiles allow for context aware extraction



DTMF tones are "dual tones"

8 frequencies

2 simultaneous frequencies for each digit

used to navigate hotline menus

	1209 Hz	1336 Hz	1477 Hz	1633 Hz
697 Hz	1	2	3	A
770 Hz	4	5	6	B
852 Hz	7	8	9	C
941 Hz	*	0	#	D

Soundcomber is fast and accurate

	No Error	1 Error	>= 2 Errors	1 missing	>= 2 missing
Speech	55 %	12.5 %	15 %	7.5 %	10 %
Tone	85 %	5 %	0	10 %	0

	Recording Length	Processing Time
Speech	20 s	7 s
Tone	45 s	8 s

香港城市大學 City University of Hong Kong

INDIANA UNIVERSITY

Soundcomber is fast and accurate

	No Error	1 Error	>= 2 Errors	1 missing	>= 2 missing
Speech	55 %	12.5 %	15 %	7.5 %	10 %
Tone	85 %	5 %	0	10 %	0

	Recording Length	Processing Time
Speech	20 s	7 s
Tone	45 s	8 s

香港城市大學 City University of Hong Kong

INDIANA UNIVERSITY

Soundcomber is fast and accurate

	No Error	1 Error	>= 2 Errors	1 missing	>= 2 missing
Speech	55 %	12.5 %	15 %	7.5 %	10 %
Tone	85 %	5 %	0	10 %	0

	Recording Length	Processing Time
Speech	20 s	7 s
Tone	45 s	8 s

香港城市大學 City University of Hong Kong

INDIANA UNIVERSITY

Soundcomber is fast and accurate

	No Error	1 Error	>= 2 Errors	1 missing	>= 2 missing
Speech	55 %	12.5 %	15 %	7.5 %	10 %
Tone	85 %	5 %	0	10 %	0

	Recording Length	Processing Time
Speech	20 s	7 s
Tone	45 s	8 s

香港城市大學 City University of Hong Kong

INDIANA UNIVERSITY

Soundcomber is fast and accurate

	No Error	1 Error	>= 2 Errors	1 missing	>= 2 missing
Speech	55 %	12.5 %	15 %	7.5 %	10 %
Tone	85 %	5 %	0	10 %	0

	Recording Length	Processing Time
Speech	20 s	7 s
Tone	45 s	8 s

香港城市大學 City University of Hong Kong

INDIANA UNIVERSITY

Soundcomber is fast and accurate

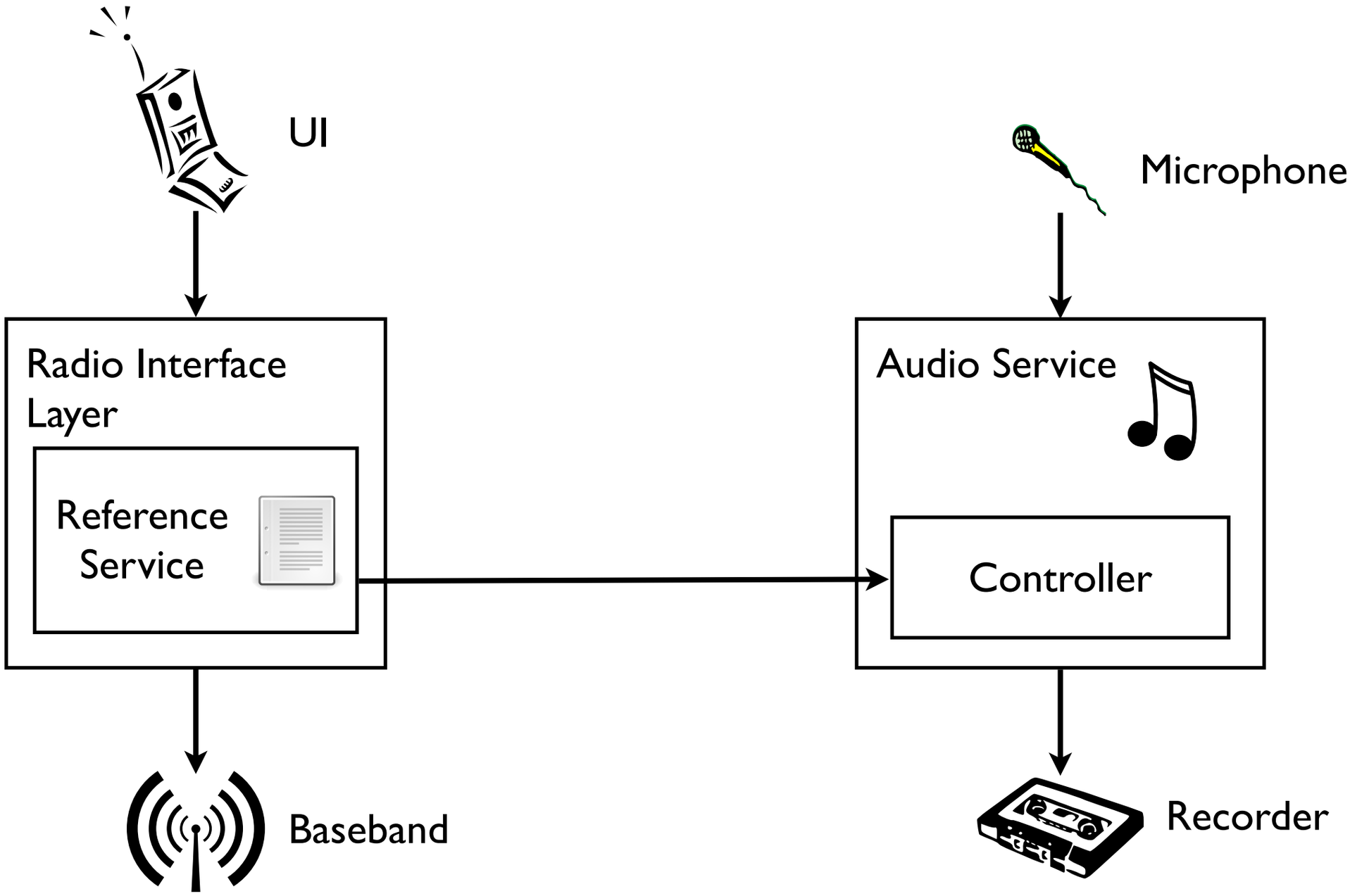
	No Error	1 Error	>= 2 Errors	1 missing	>= 2 missing
Speech	55 %	12.5 %	15 %	7.5 %	10 %
Tone	85 %	5 %	0	10 %	0

	Recording Length	Processing Time
Speech	20 s	7 s
Tone	45 s	8 s

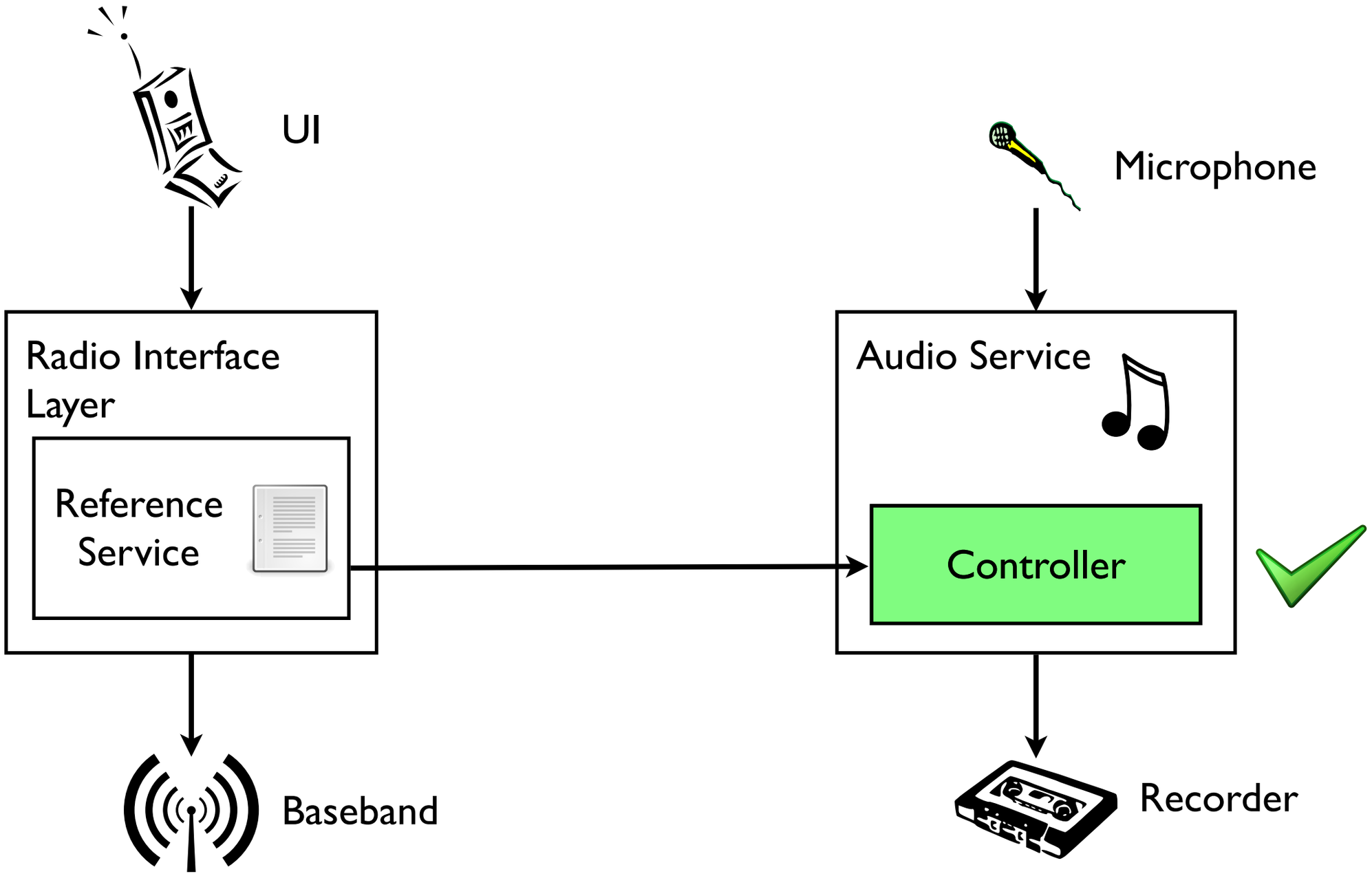
香港城市大學 City University of Hong Kong

INDIANA UNIVERSITY

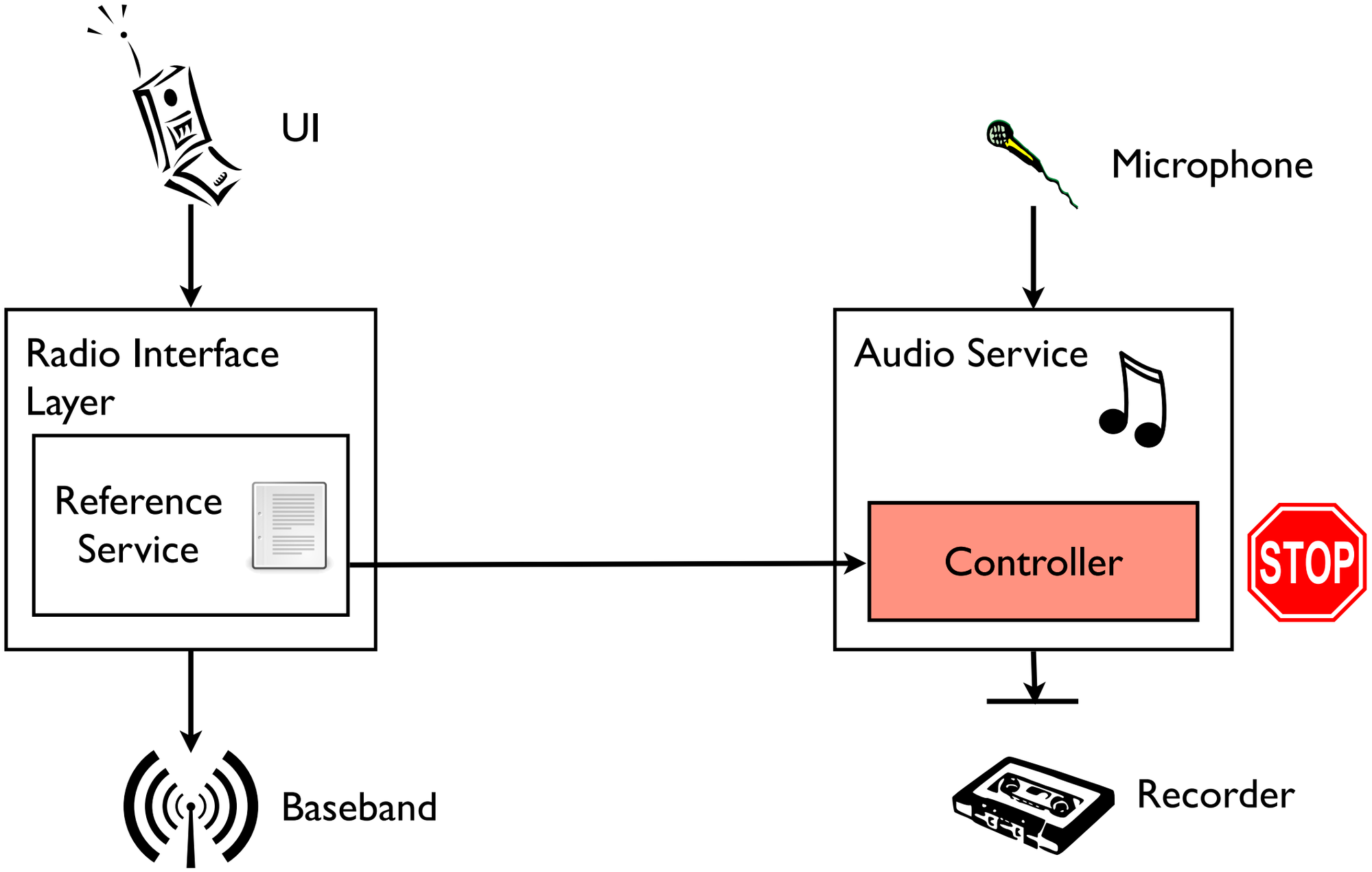
Defense: disable recording when a sensitive number is called



Defense: disable recording when a sensitive number is called



Defense: disable recording when a sensitive number is called



Research Directions

- **Currently exploring**
  - Situational awareness
  - Is the sensor in a trustworthy environment?
  - Overhear familiar people
- **Future?** Distributed mining
  - Distributed video processing to locate people?
- **Future?** Defensive architectures
  - Context aware sensor use

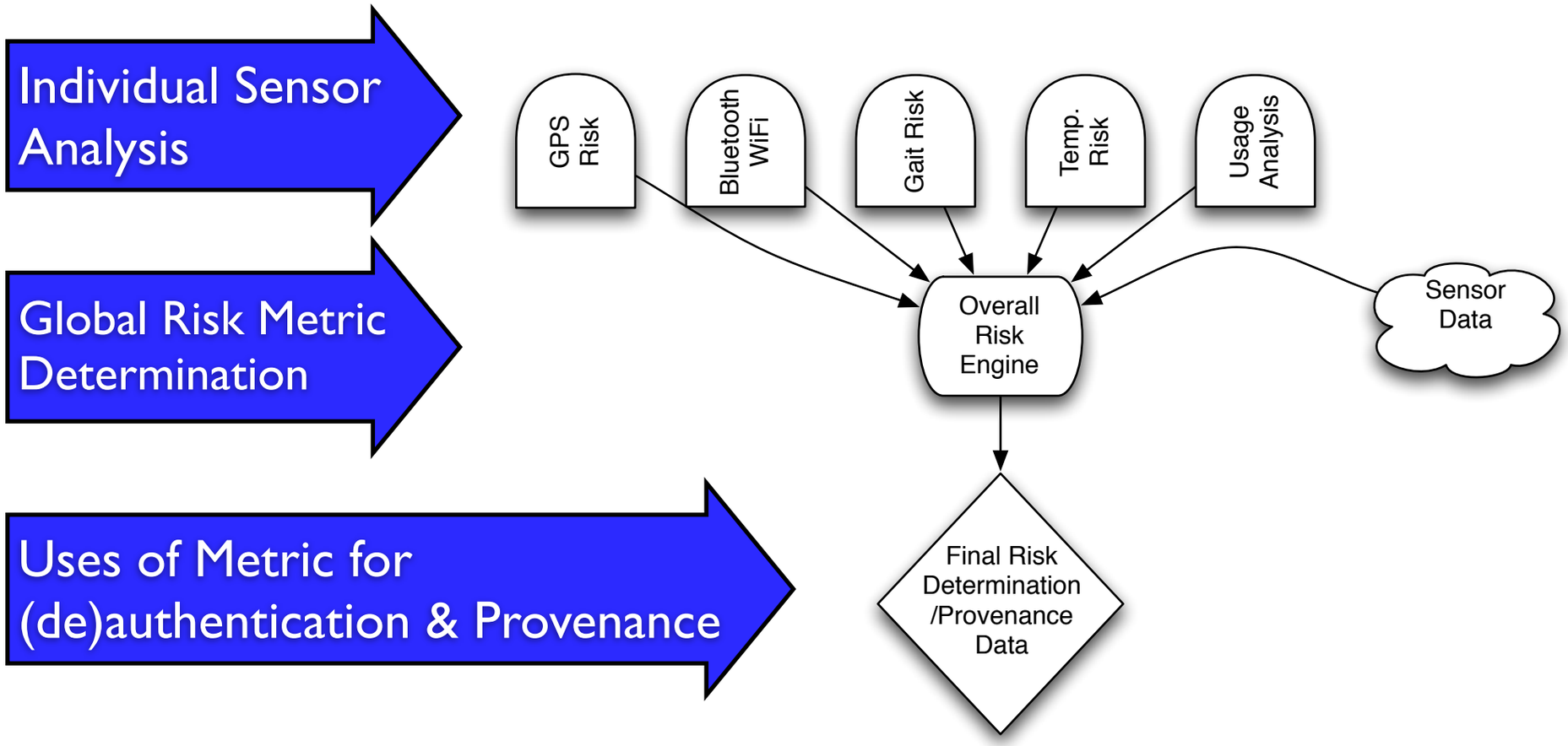
Threats to sensor environment

Sensor based (de)authenticatioion and sensor data provenance

Shaun Deaton, Mehool Intwala, Ali Khalfan, Nathaniel Husted

Steve Myers, Apu Kapadia

Risk Measurement Architecture



- If final risk is low sensor data reported as is, possibly with Provenance Data.
- If risk is high, force authentication of phone before reporting data or mark with high-risk provenance data.

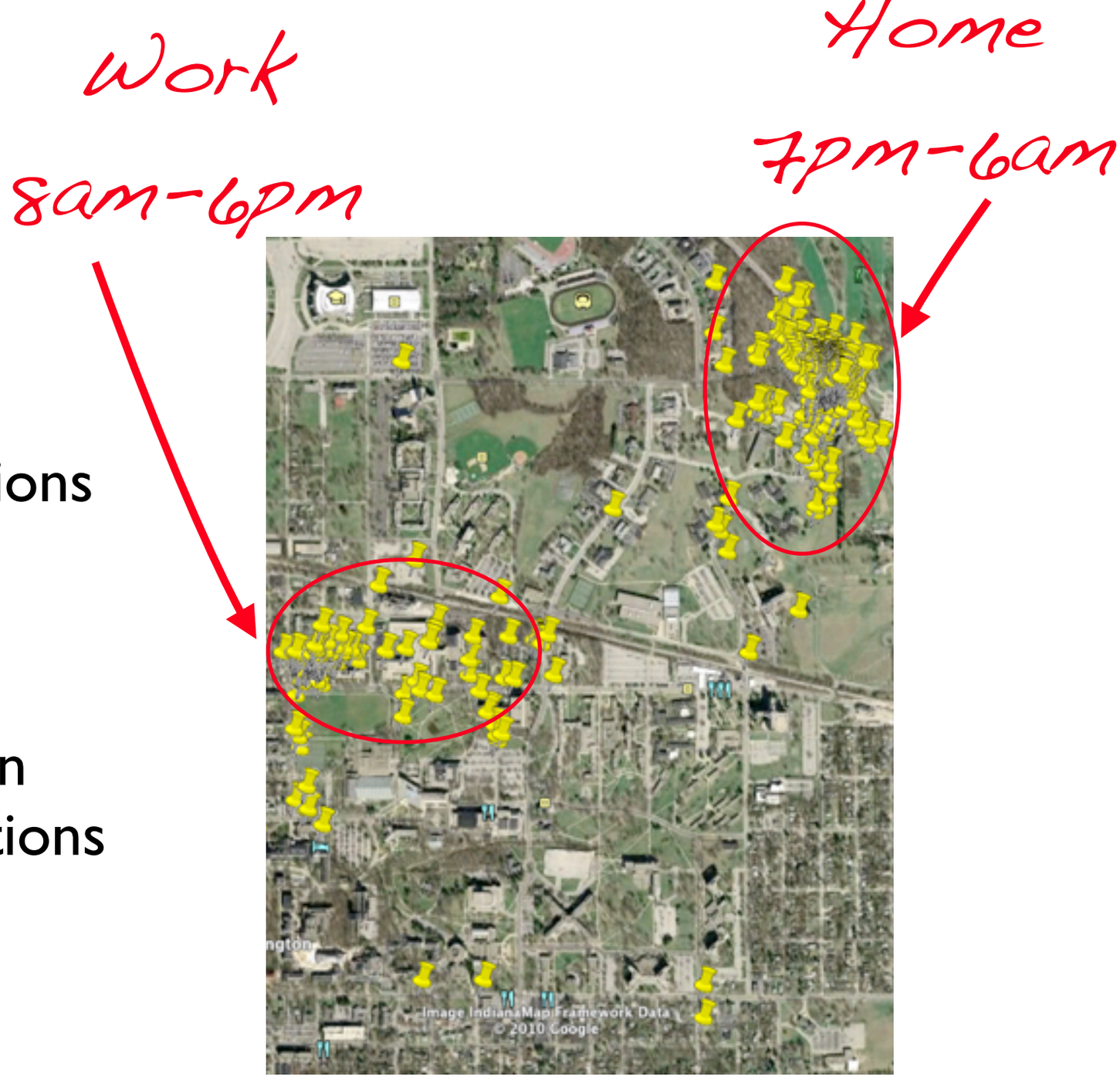
INDIANA UNIVERSITY

Measuring Performance

- Acquired Reality Mining Dataset from MIT Reality Mining Labs [Eagle and Pentland 2006].
- Study that collected real-world cell phone usage data
- 9 month collection period
- 100 Human Users
  - 75 Faculty and Students in Media Lab
  - 25 Students in Business School.
- Cell Phone Tower IDs
- Proximal Bluetooth Devices
- Application Usage
- Communication Patterns

Source: http://reality.media.mit.edu

Positional Sensor Data



- Learn habitual locations and times in 4hr windows (HMM)
- Predict risk based on preceding 4hr positions

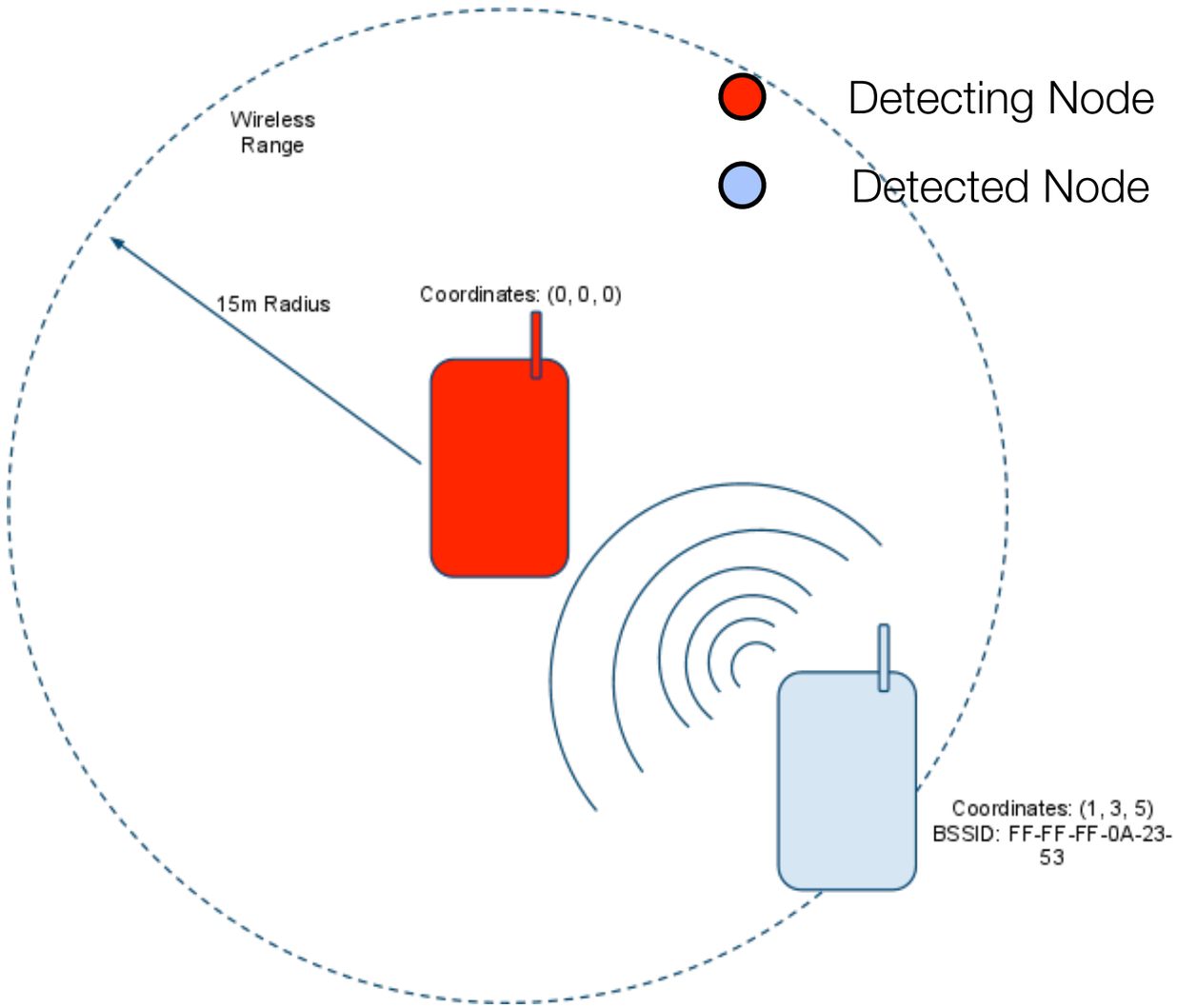
INDIANA UNIVERSITY

Detecting Friends, Foes and Strangers Through Bluetooth

- Proximity of certain devices suggest low risk (Wife's phone, my bluetooth earpiece, laptop, PS3, etc....)

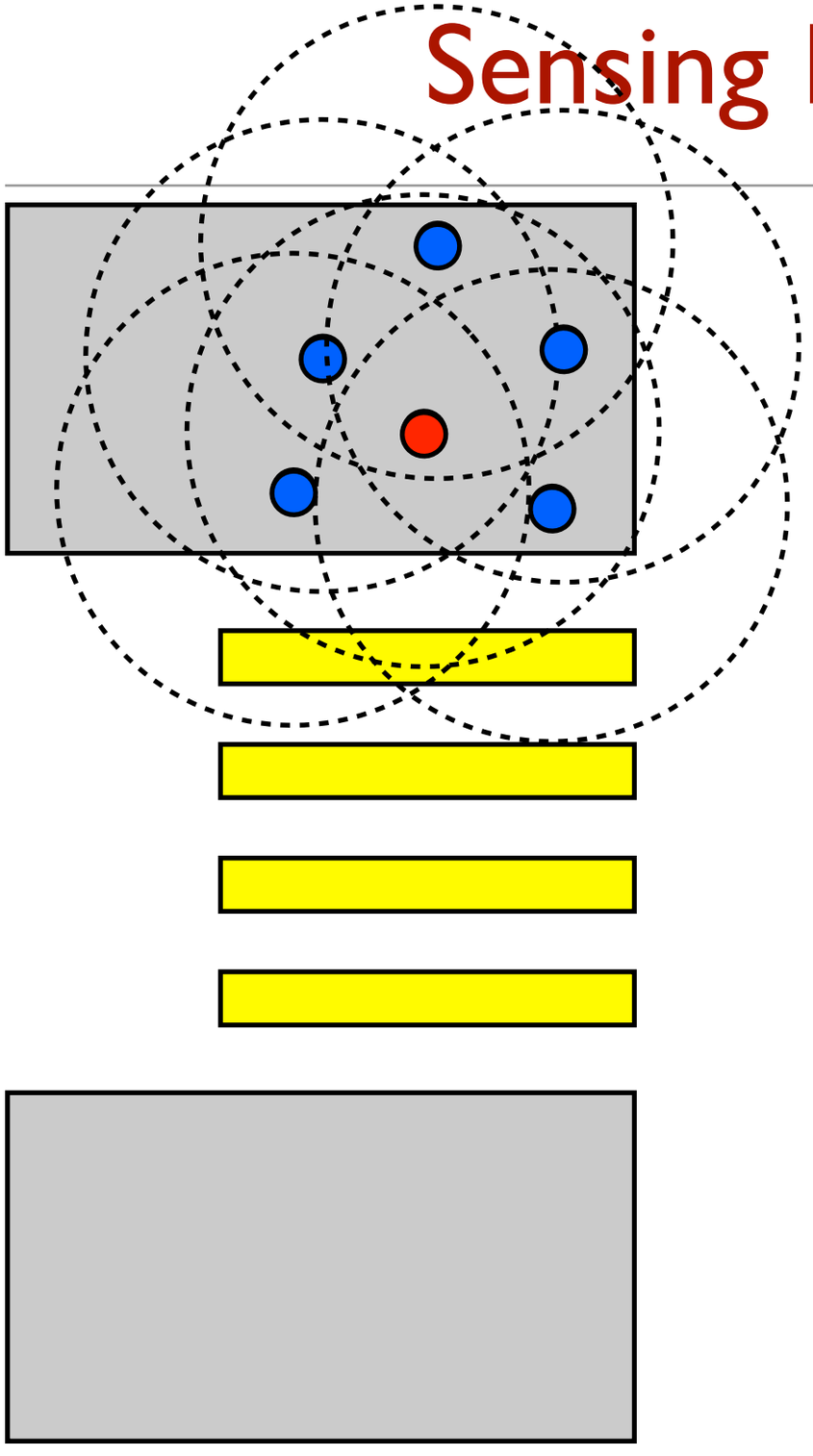
- Proximity of certain devices suggest high risk (Enemy's phone, competitor's phone, device which has only questionable purposes)

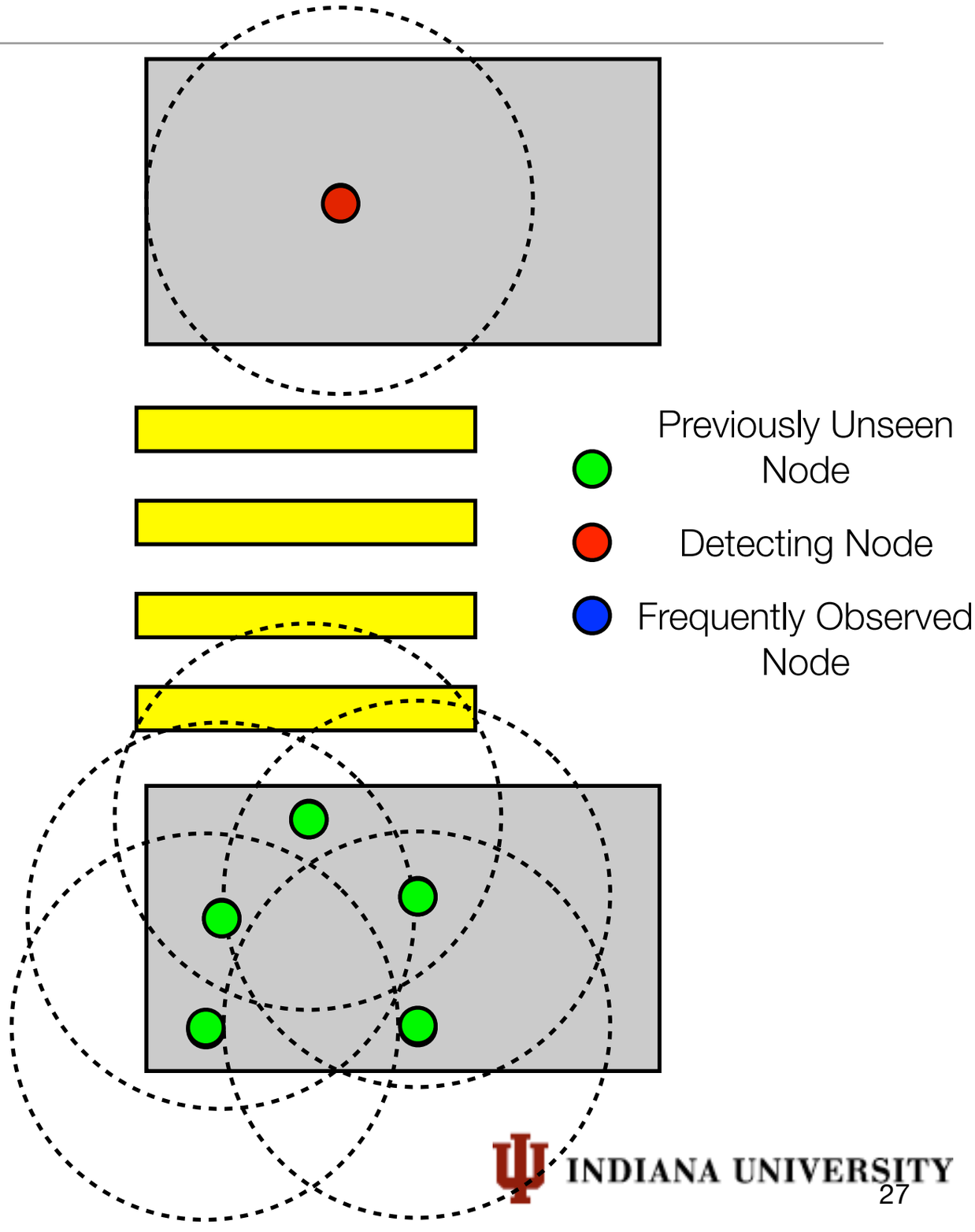
- Many strangers (unknown phones) increases risk



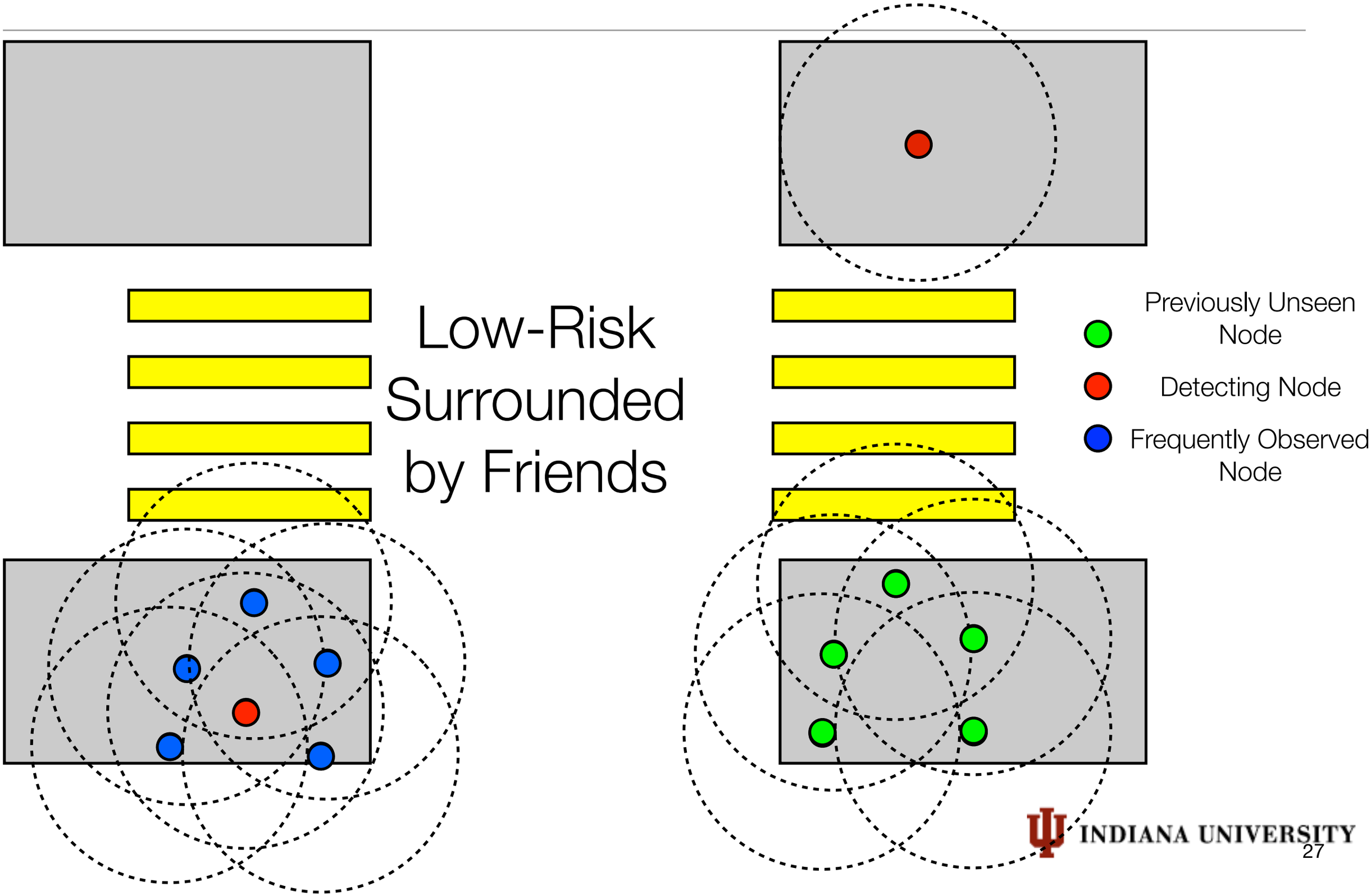
INDIANA UNIVERSITY

Sensing Friends and Strangers

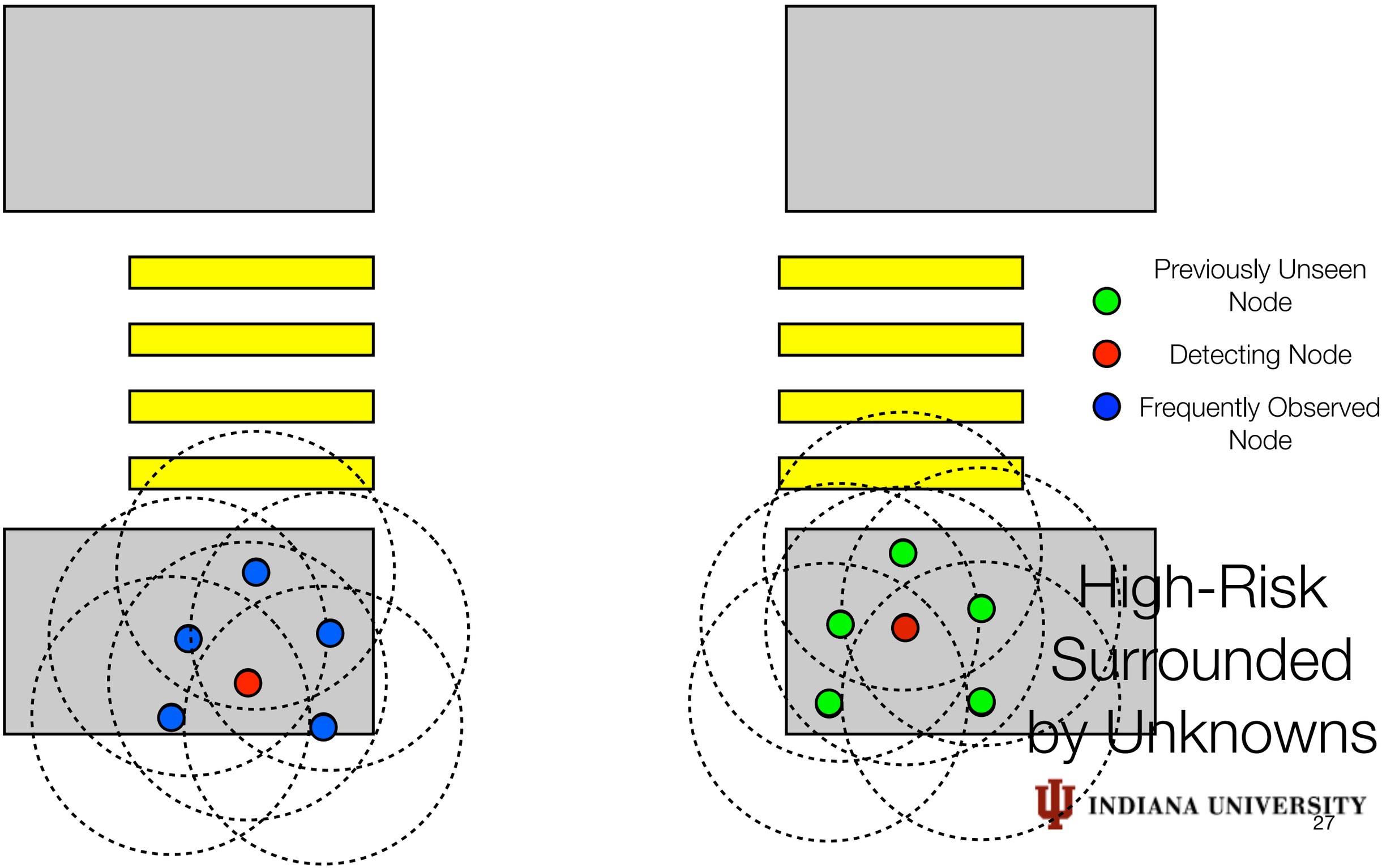




Sensing Friends and Strangers



Sensing Friends and Strangers



White-List, Grey-List, Black-List

- White-list contains IDs whose presence ensures safety
- Black-list contains IDs whose presence ensures danger
- Grey-list: Determine frequently present IDs, and base risk assessment on this.
- Use history of observed IDs to determine how likely they are to be seen.

$$Pr(id_i) = \frac{\text{Observations of } i}{\text{Total IDs Observed}}$$

- Compute Entropy of Distribution

$$H(D) = \sum_i Pr(id_i) \cdot \log(1/Pr(id_i)).$$

- Compare spot-entropy of observed data to Entropy and put in a logistic sigmoid

$$Risk = \frac{1}{1 - e^{-\sum_{\text{obs.}_i}(-\log(Pr(id_i)) - H(D))}}$$

Research directions

- Safe places, safe faces: how to determine?
- Metric design, improving accuracy of automated detection techniques
- Statistical and logical combination of risk measurements from different sensors

Threats to communications

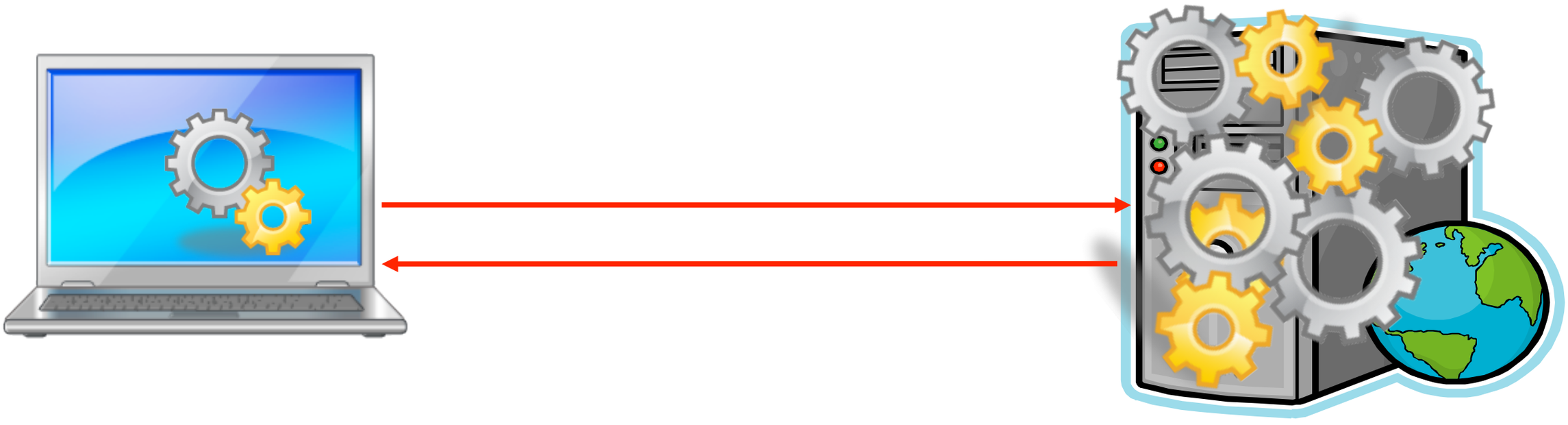
Sidebuster: Automated Detection and Quantification of Side-Channel Leaks in Web Application Development

Kehuan Zhang, Zhou Li,
Rui Wang, XiaoFeng wang
Indiana University

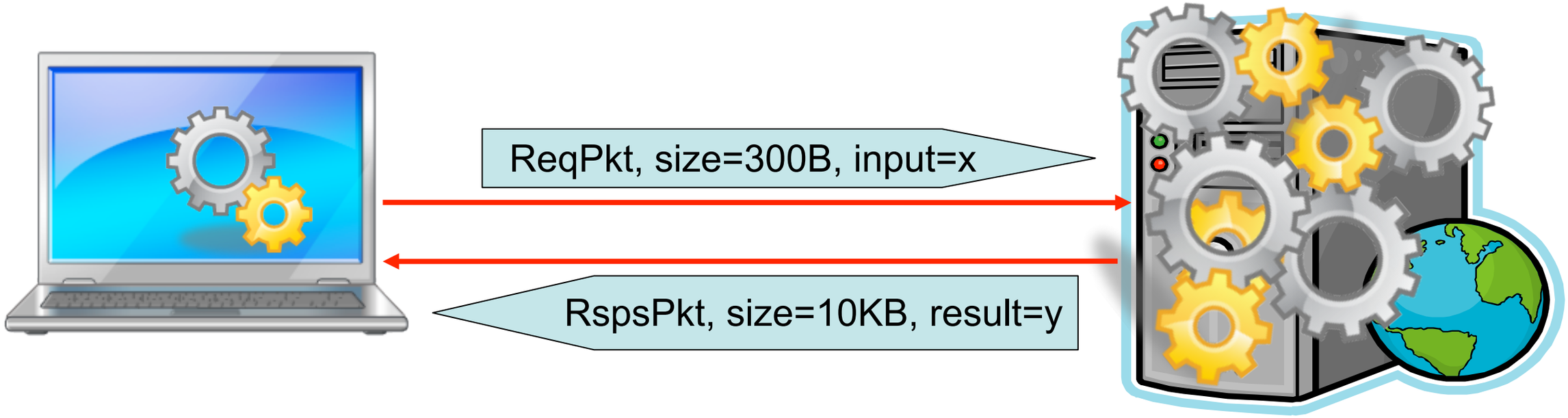
Shuo Chen
Microsoft Research

Oakland, CCS 2011

Side-channel Leaks in Web Applications



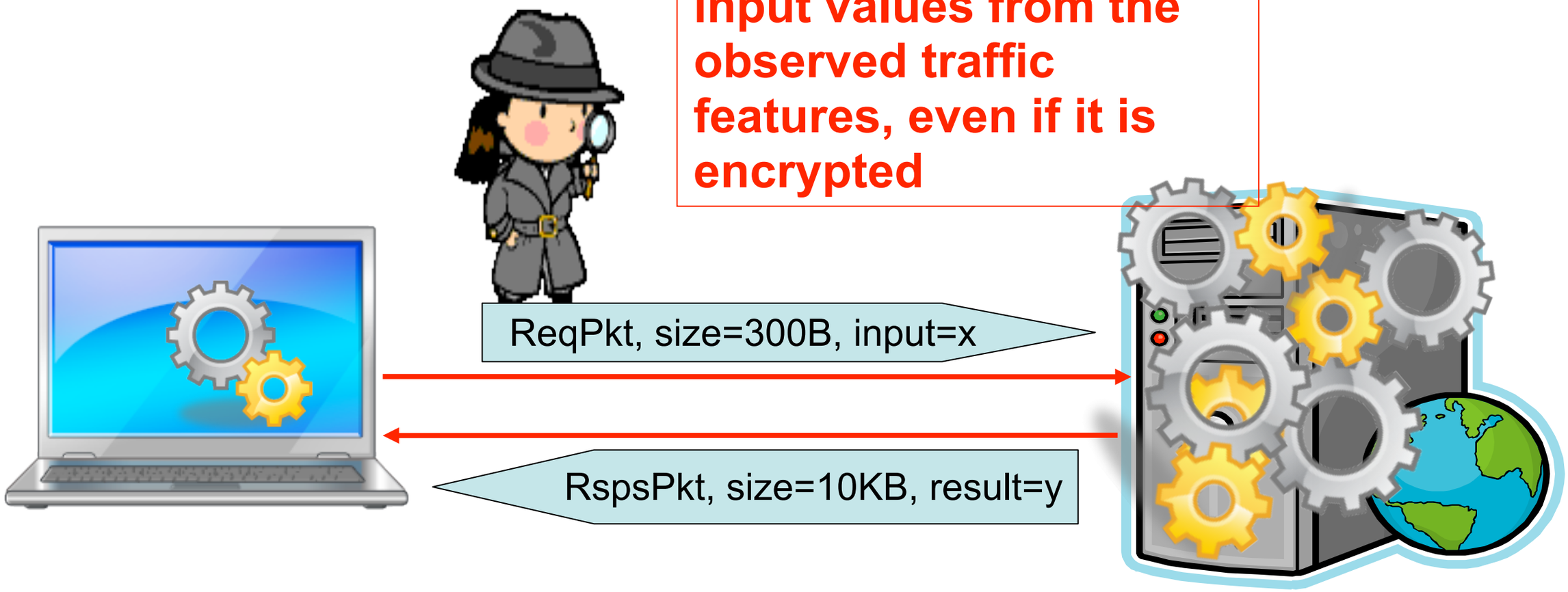
Side-channel Leaks in Web Applications



INDIANA UNIVERSITY

Side-channel Leaks in Web Applications

Attackers can infer the input values from the observed traffic features, even if it is encrypted



INDIANA UNIVERSITY

How to Mitigate the Side-channel Threat?

- First step: detect the problem from web applications
  - Where does information leak happen?
  - How serious is the problem?

Ideas

- Information flow analysis to locate where leaks happen
  1. Sensitive data "taints" network traffic
  2. Content of the data associated with different traffic features

- Quantify the information being leaked

An example – online health profile management

User Type

- patient
- doctor

Profile type

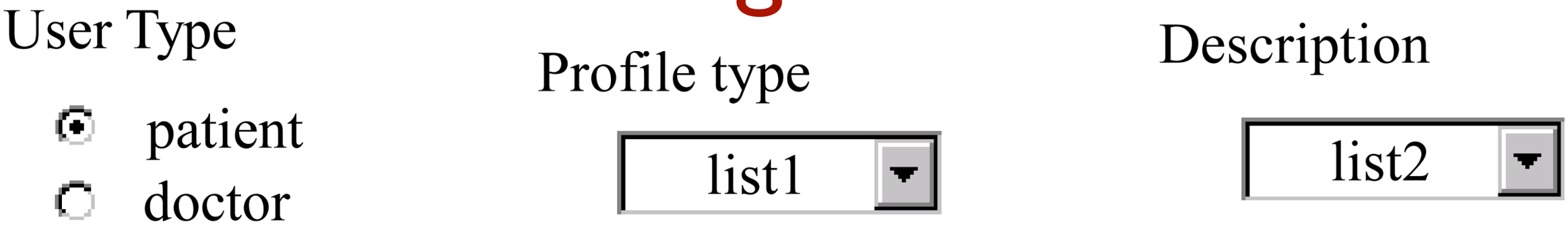
list1

Description

list2

INDIANA UNIVERSITY

An example – online health profile management



Step 1: trace the propagation of sensitive data

```
onChange() {
    temp = list1.getSelected();
    value1 = temp;
    rpc1(value1, callback1);
}
 callback1(result) {
     list2.addItem(result);
 }
```

```
rpc1(value1) {
   return
     getDecriptFromDB(value1);
}
```


INDIANA UNIVERSITY

An example – online health profile management

User Type

- patient
- doctor

Profile type

list1

Description

list2

Step 1: trace the propagation of sensitive data

```
onChange() {
    temp = list1.getSelected();
    value1 = temp;
    rpc1(value1, callback1);
}
 callback1(result) {
     list2.addItem(result);
 }
```

```
rpc1(value1) {
   return
     getDecriptFromDB(value1);
}
```

Sensitive data will be sent to network, so information could be leaked here!

INDIANA UNIVERSITY

An example – online health profile management

User Type

- patient
- doctor

Profile type

list1

Description

list2

Step 2: quantify the information leaks

INDIANA UNIVERSITY

An example – online health profile management

User Type

- patient
- doctor

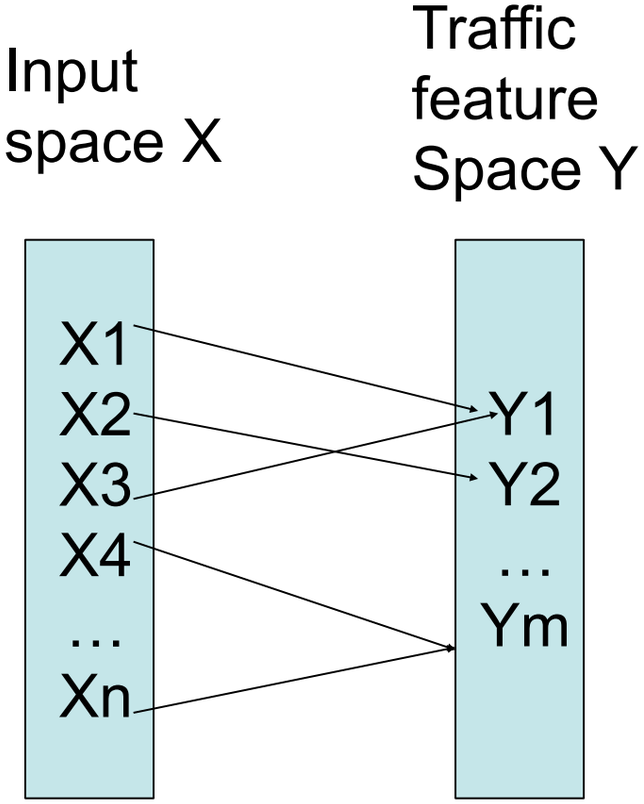
Profile type

list1

Description

list2

Step 2: quantify the information leaks



INDIANA UNIVERSITY

An example – online health profile management

User Type

- patient
- doctor

Profile type

list1

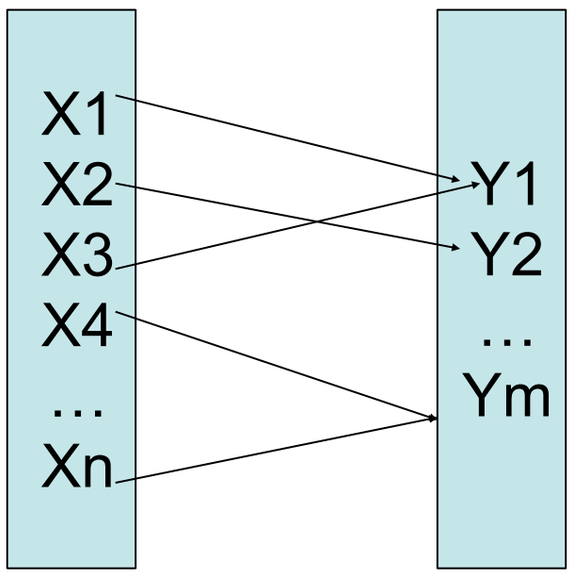
Description

list2

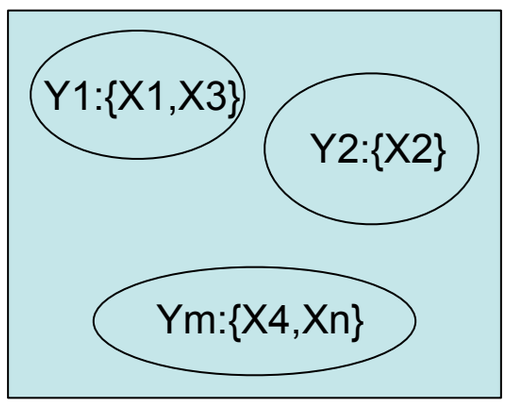
Step 2: quantify the information leaks

Input space X

Traffic feature Space Y



Partition the input space Using traffic features



INDIANA UNIVERSITY

An example – online health profile management

User Type

- patient
- doctor

Profile type

list1

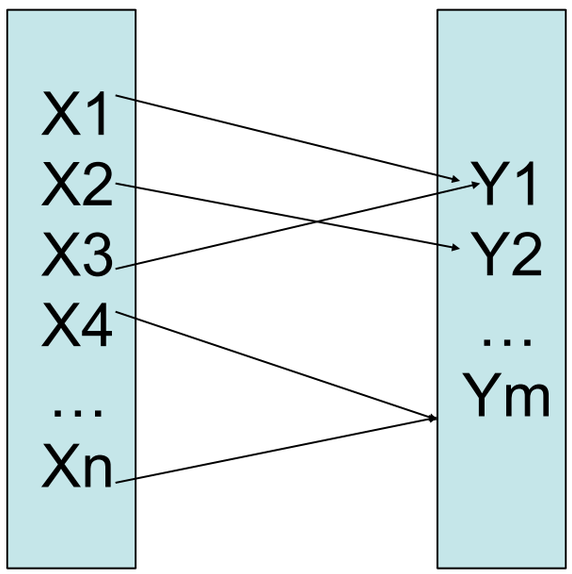
Description

list2

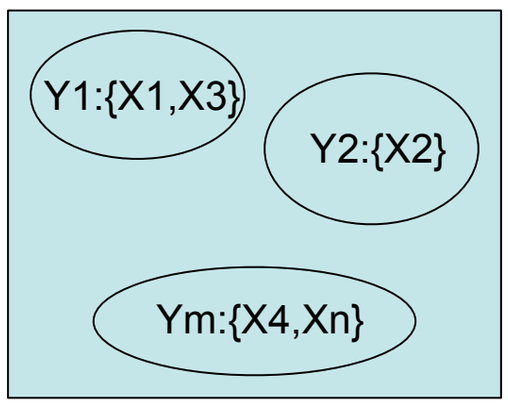
Step 2: quantify the information leaks

Input space X

Traffic feature Space Y



Partition the input space Using traffic features



Calculating *Conditional entropy*

Original information:
H(X)
Information remained:
H(X|Y)
Information leaked:
H(X) – H(X|Y)

INDIANA UNIVERSITY

Progress so far

- Automatic detection of side-channel leaks in web applications
- Techniques for quantifying information leaks
- Implementation and evaluation
  - Implement a prototype for GWT (Google Web Toolkit) programs
  - Evaluated our prototype over 6 real-world or synthesized applications

Research directions

- Quantify leaks through control flow
- Examine next generation sensor-cloud applications (e.g., humans in the loop)
  - Combinations of all possible user inputs to GUI
- Efficient padding schemes to reduce leaks

Conclusions

- The landscape of "sensor" based computing has changed
  - Smartphone class devices, tied to a human (owner)
- Several new challenges have emerged
  - Sensory malware targeting the owner of the sensor
  - Theft of device and environment tampering
  - Information leakage through communication traffic patterns
- Research directions
  - Context aware sensor use
  - Using sensors to assess the trustworthiness of other sensors
  - Quantifying and eliminating side channel leaks
- *Thank you to NSF for partially funding this research*