A Protocol Independent Approach in Network Covert Channel Detection

Md. Ahsan Ayub, Steven Smith, Ambareen Siraj

Department of Computer Science

Tennessee Technological University

Cookeville, TN, USA

CERIAS
Cybersecurity Education, Research and Outreach Center

Tennessee Tech College of Engineering

Tennessee Tech Cybersecurity Education, Research & Outreach

Content

- Introduction
- Research Problem
- Related Work
- Methodology
- Results
- Conclusion

Network Covert Channel

- Hidden or unintended form of communication via existing network protocol
- Used to secretly leak sensitive data as well as hide military and secret service communication [1]
- Two types of network covert channels:
  - Storage covert channel
  - Timing covert channel

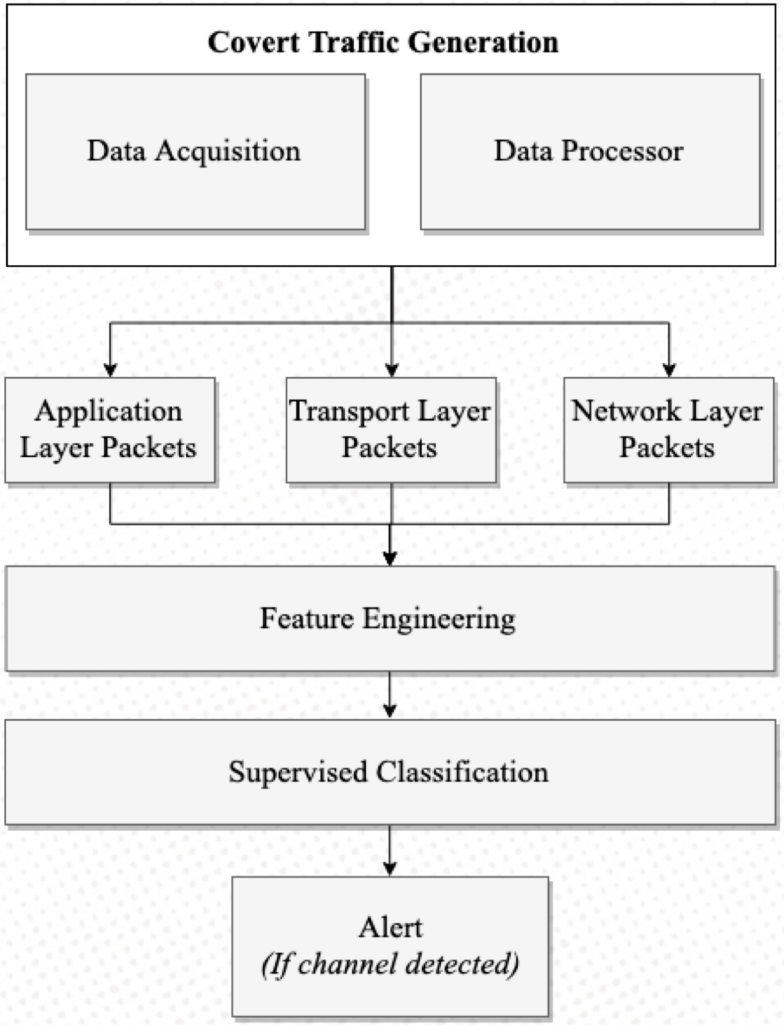
Research Problem

- Embedding hidden data in every OSI layer
- Majority of work done to propose effective detection methods in a *specific* network protocol [2 - 4]
  - Traffic normalization
  - Statistical method
  - Machine learning
- Could we develop a protocol independent machine learning approach to detect these covert channel across multiple protocols?

Related Work

- Wendzel *et al.* [2015] presented 11 patterns derived from 109 different techniques among which following 4 patterns represented 69.7% of existing covert channels. [5]
  - Reserved / unused pattern
  - Add redundancy pattern
  - Value modulation pattern
  - Random value pattern
- Public covert network channel dataset unavailable

Methodology



Covert Traffic Generation

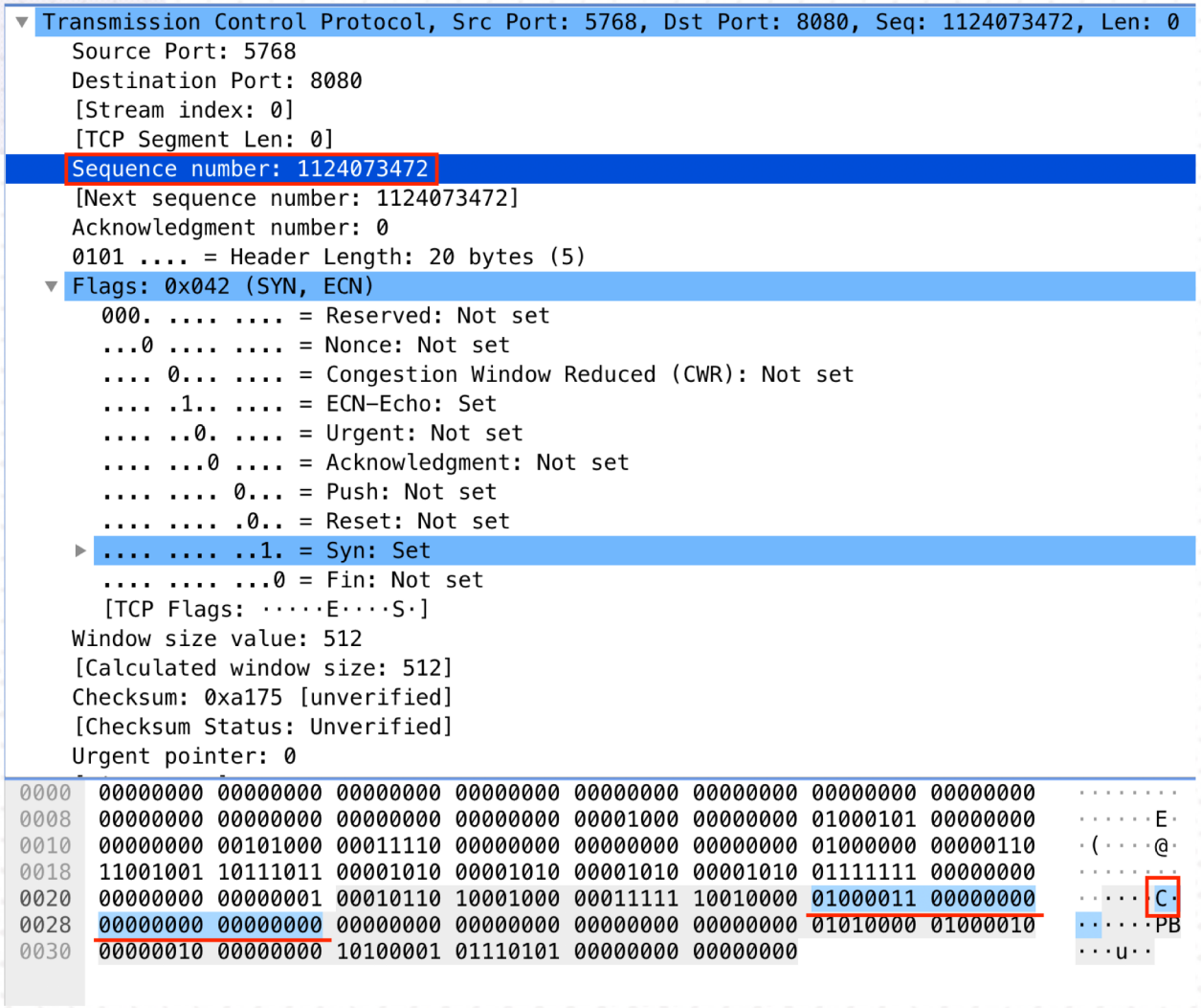
- Network layer (IPv4) and Transport layer (TCP) covert datasets were generated using Craig H Rowland's tool. [6]
  - A program written in C language
  - Covert data is hidden in *IP ID* field for IPv4 datagram.
  - Covert data is embedded in *TCP Seq* field for TCP segment.
- Application layer (DNS) covert dataset was generated using DNS2TCP application.
  - Covert data is encoded in DNS Query and DNS Response.

Covert Traffic Generation: Network Layer

```
▼ Internet Protocol Version 4, Src: 10.10.10.10, Dst: 127.0.0.1
    0100 .... = Version: 4
    .... 0101 = Header Length: 20 bytes (5)
  ▶ Differentiated Services Field: 0x00 (DSCP: CS0, ECN: Not-ECT)
    Total Length: 40
    Identification: 0x4300 (17152)
  ▼ Flags: 0x0000
      0... .... .... .... = Reserved bit: Not set
      .0.. .... .... .... = Don't fragment: Not set
      ..0. .... .... .... = More fragments: Not set
      ...0 0000 0000 0000 = Fragment offset: 0
    Time to live: 64
    Protocol: TCP (6)
    Header checksum: 0xa4bb [validation disabled]
    [Header checksum status: Unverified]
    Source: 10.10.10.10
    Destination: 127.0.0.1
```

```
0000  00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000   ........
0008  00000000 00000000 00000000 00000000 00001000 00000000 01000101 00000000   ......E·
0010  00000000 00101000 01000011 00000000 00000000 00000000 01000000 00000110   ·(C·····@·
0018  10100100 10111011 00001010 00001010 00001010 00001010 01111111 00000000   ........
0020  00000000 00000001 00010110 10001000 00011111 10010000 11110010 00010000   ........
0028  00000000 00000000 00000000 00000000 00000000 00000000 01010000 00000010   ......P·
0030  00000010 00000000 11110010 10100100 00000000 00000000                     ......
```

Covert Traffic Generation: Transport Layer



Covert Traffic Generation: Application Layer

```
▶ User Datagram Protocol, Src Port: 53, Dst Port: 42899
▼ Domain Name System (response)
    Transaction ID: 0x410e
  ▶ Flags: 0x8580 Standard query response, No error
    Questions: 1
    Answer RRs: 1
    Authority RRs: 0
    Additional RRs: 0
  ▼ Queries
    ▼ IWUV2BXqBA.iohost.theworldofnet.online: type TXT, class IN
        Name: IWUV2BXqBA.iohost.theworldofnet.online
        [Name Length: 38]
        [Label Count: 4]
        Type: TXT (Text strings) (16)
        Class: IN (0x0001)
  ▼ Answers
    ▼ IWUV2BXqBA.iohost.theworldofnet.online: type TXT, class IN
        Name: IWUV2BXqBA.iohost.theworldofnet.online
        Type: TXT (Text strings) (16)
        Class: IN (0x0001)
        Time to live: 3
        Data length: 13
        TXT Length: 11
        TXT: AIWUAABXqEA
        TXT Length: 0
        TXT:
    [Request In: 4]
    [Time: 0.500719255 seconds]
```

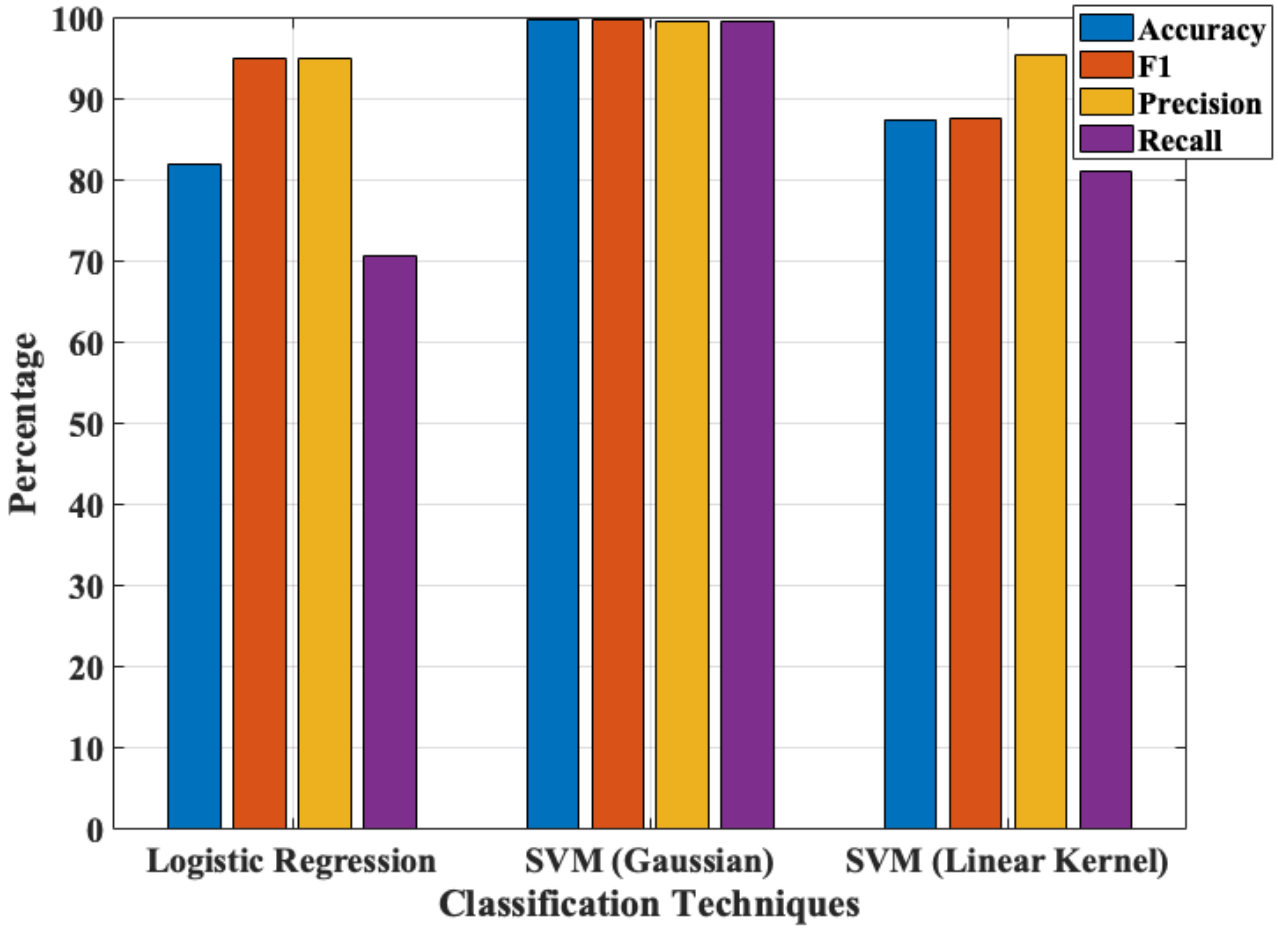
Methodology (Cont.)

- Developed a data processor engine to streamline the captured data and feed the machine learning model
- The engine performs the following activities:
  - Pruning undesired variables for each layer
  - Converting text into machine learning aware numeric values
  - Cross-validation of acquired dataset (both regular and covert)
  - Getting everything ready for supervised classification learning
- Supervised classification learning algorithms used:
  - Logistic regression, Support Vector Machine (Linear Kernel and Gaussian Kernel), K-nearest Neighbor, and Decision Tree

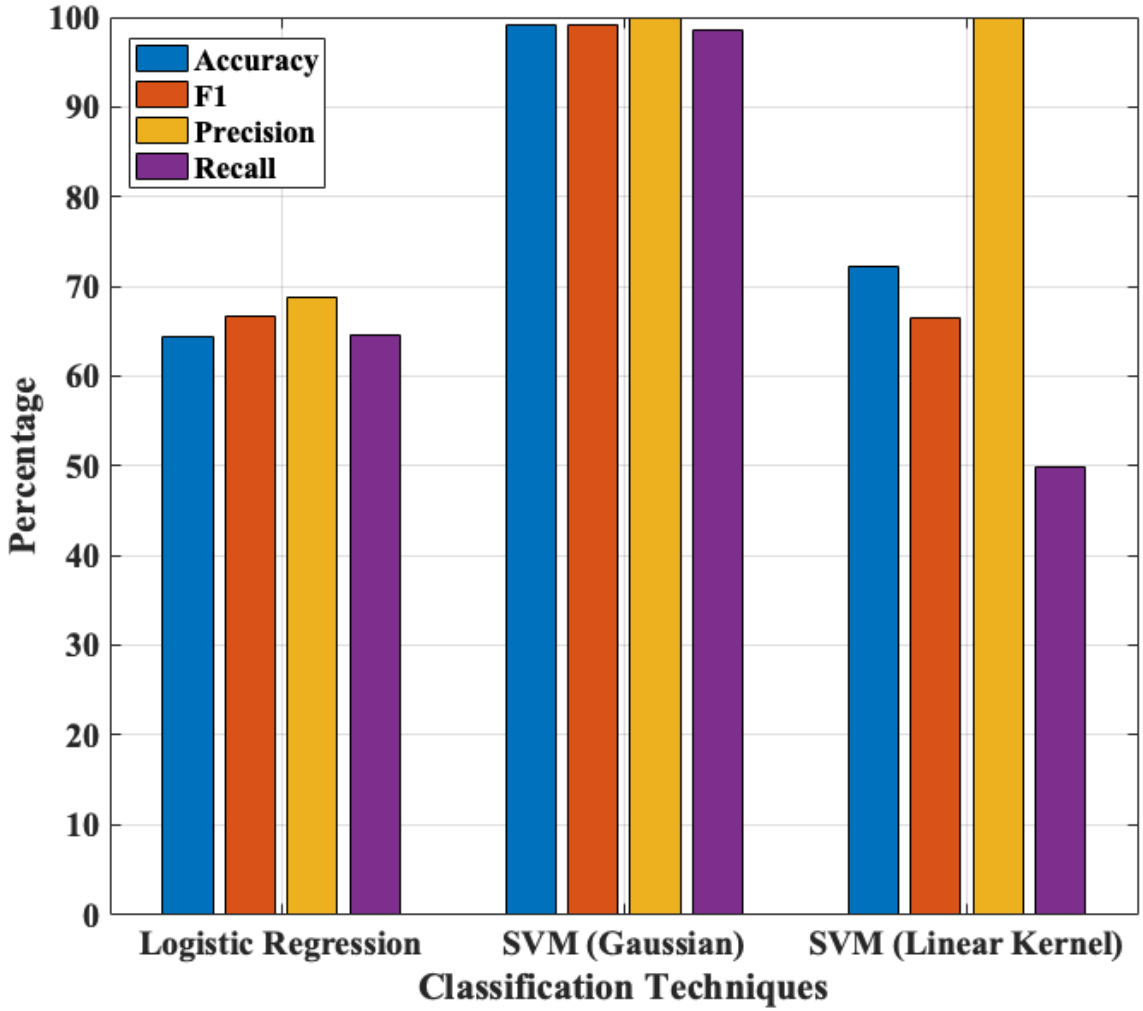
Results

- To be illustrated in three parts:
  - Network Layer
  - Transport Layer
  - Application Layer

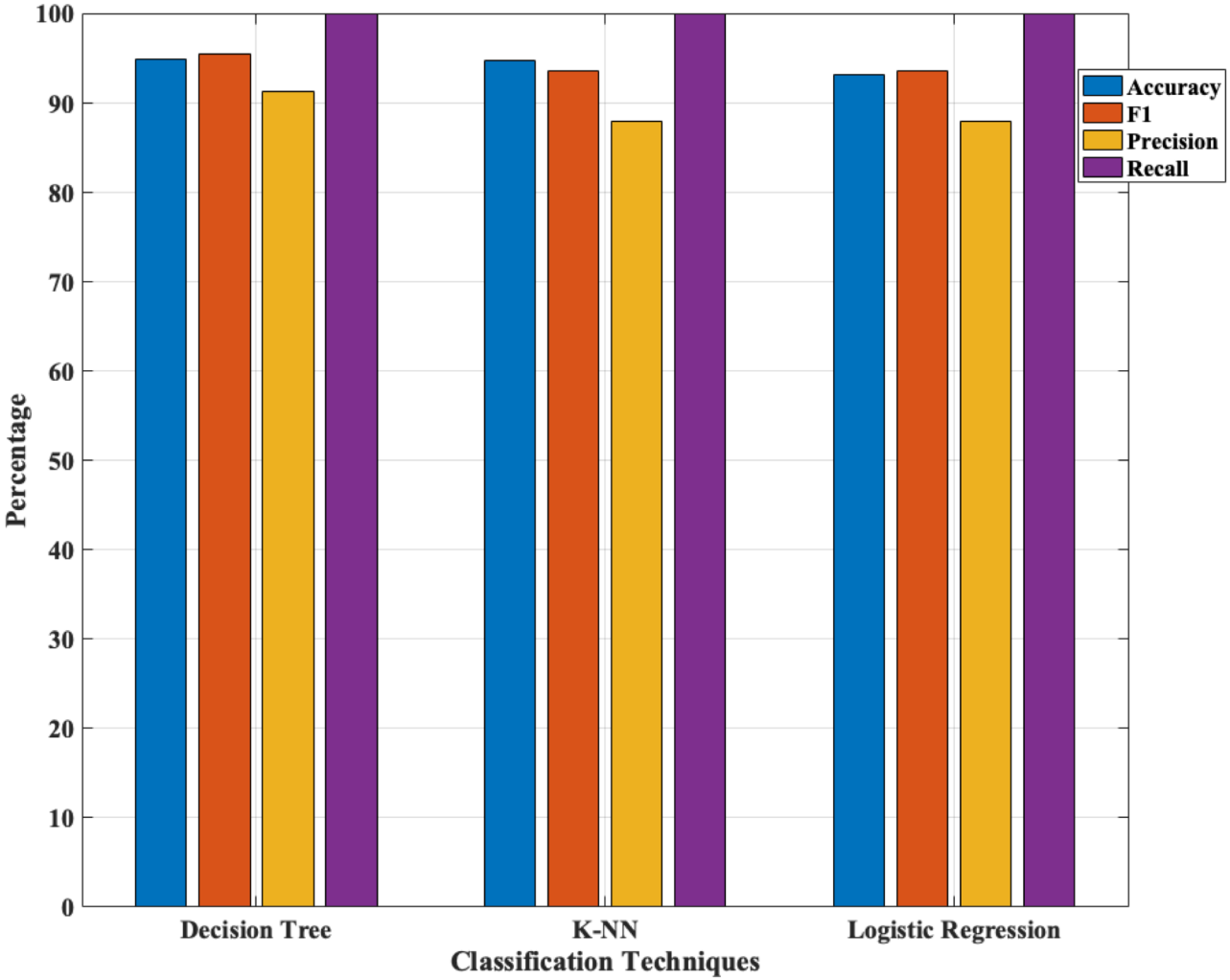
Results: Network Layer



Results: Transport Layer



Results: Application Layer



Conclusion

- We proposed a protocol independent approach.
- SVM with a gaussian kernel was effective for TCP/IP protocol suite, and decision tree classifier was found to be very effective for DNS dataset.
- Data link layer was not comprised in our study.
- Future work would be extending our work for other patterns and applying deep learning techniques.
- Generated datasets are available to share upon request.

References

1. David Llamas, C Allison, and A Miller. Covert channels in internet protocols: A survey. In Proceedings of the 6th Annual Postgraduate Symposium about the Convergence of Telecommunications, Networking and Broadcasting, PGNET, volume 2005, 2005.
2. Norka B Lucena, Grzegorz Lewandowski, and Steve J Chapin. Covert channels in ipv6. In International Workshop on Privacy Enhancing Technologies, pages 147–166. Springer, 2005.
3. Steven J Murdoch. Covert channel vulnerabilities in anonymity systems. Technical report, University of Cambridge, Computer Laboratory, 2007.
4. Eugene Tumoian and Maxim Anikeev. Detecting nushu covert channels using neural networks. Taganrog State University of Radio Engineering, 2005.
5. Steffen Wendzel, Sebastian Zander, Bernhard Fechner, and Christian Herdin. Pattern-based survey and categorization of network covert channel techniques. ACM Computing Surveys (CSUR), 47(3):50, 2015.
6. Craig H Rowland. Covert channels in the tcp/ip protocol suite. First Monday, 2(5), 1997.

Acknowledgement

- Anonymous reviewers
- Dr. Edward Zieglar, National Security Agency (NSA)
- Cybersecurity Education, Research and Outreach Center (CEROC) at Tennessee Tech University
- Department of Computer Science and College of Engineering (CoE) at Tennessee Tech University

THANK YOU!