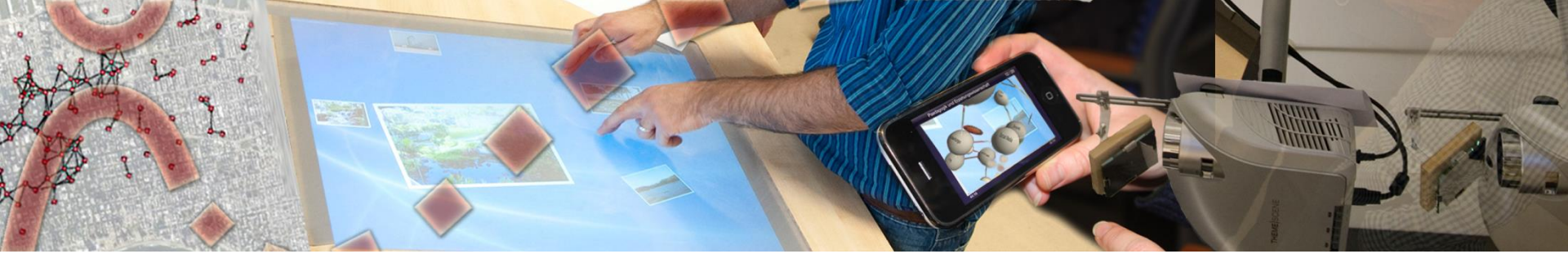


Analyzing Flow-based Anomaly Intrusion Detection using Replicator Neural Networks

Carlos García Cordero
Sascha Hauke
Max Mühlhäuser
Mathias Fischer

Telecooperation Lab

CYSEC

CRISP
Center for Research in Security and Privacy

TECHNISCHE UNIVERSITÄT DARMSTADT

The Beautiful World of IoT

The Beautiful World of IoT

TECHNISCHE UNIVERSITÄT DARMSTADT

Goals

- Performance
- Usability
- Transparency
- Accessibility
- Ubiquity
- **Security?????**

The Beautiful World of IoT?

- **Latest IoT news:**
  - **09-2014:** IoT – Major players agree on goals, but little else
  - **12-2014:** New Hacker-Friendly search engine that lists vulnerable IoT devices (censys.io)
  - **11-2015:** Millions of IoT Devices using the same hard-coded crytpo-keys
  - **09-2016:** World's largest 1Tbps DDoS Attack launched from 152,000 hacked smart devices
  - **10-2016:** Largest botnets consists of mostly IoT devices
  - **10-2016:** 12-Year-Old SSH bug exposes more than 2 million IoT devices
  - **Recently:** Mirai botnet, WannaCry, etc

Outline

TECHNISCHE UNIVERSITÄT DARMSTADT

~~1. Motivation~~

2. Introduction

- What is all this about?

3. Background

- Intrusion Detection is Large Networks

4. Intrusion Detection using Replicator Neural Networks

- Detecting Anomalies in Network Flows
- Replicator Neural Networks

5. Evaluation

- Experiments and Results

6. Conclusion and Future Work

Introduction:
What is all this about?

TECHNISCHE UNIVERSITÄT DARMSTADT

- **Understanding the Title:**
  - Analyzing Flow-based Anomaly Intrusion Detection using replicator Neural Networks

- **Analyzing Flow-based**
  - **Purpose:** analyze network data encoded as flows
  - **Enables:** affordable analysis of massive traffic

- **Anomaly Intrusion Detection**
  - **Purpose:** automatically detect deviations from normality
  - **Enables:** identification of new attacks without user intervention

- **using Replicator Neural Networks**
  - **Purpose: classify**
  - **Enables:** unsupervised (and robust) identification of anomalous traffic

Background:
Intrusion Detection in Large Networks

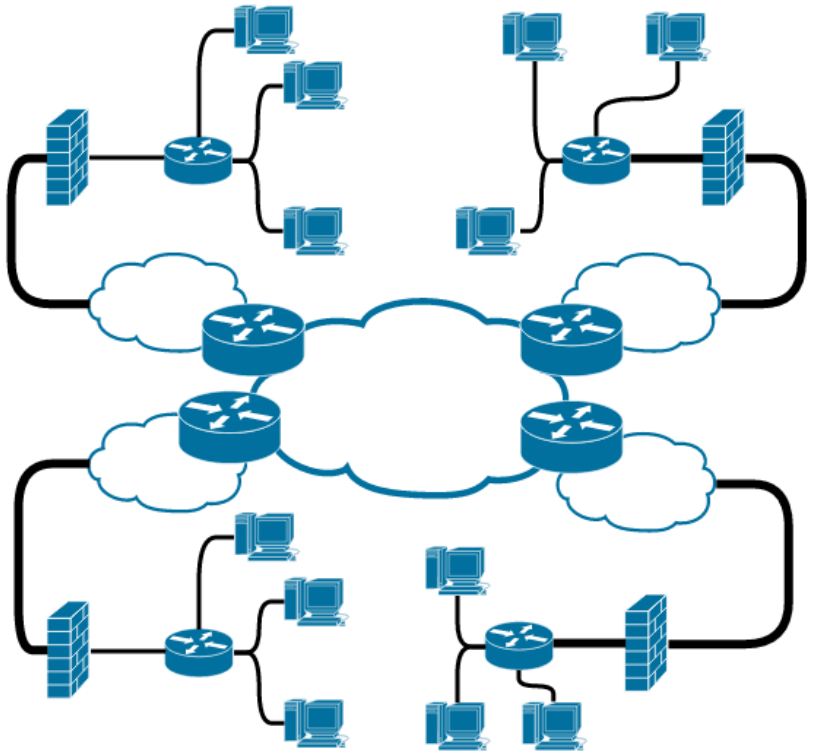
TECHNISCHE UNIVERSITÄT DARMSTADT

- Large networks are the new battle ground
  - Botnets
  - Coordinated Crowds
  - Denial of Service attacks
  - … and all other coordinated attacks
- Network attacks affect the victim and the Internet infrastructure
  - Disruption to the targets and the service providers
  - Defensive mechanisms need to be **further from the edge**
- Large Networks are difficult to monitor
  - Thousands of heterogeneous devices
  - Terabytes of data (every hour)
  - Attacks are easily overlooked

Background:
Intrusion Detection in Large Networks

TECHNISCHE UNIVERSITÄT DARMSTADT

- Detecting anomalies in large networks

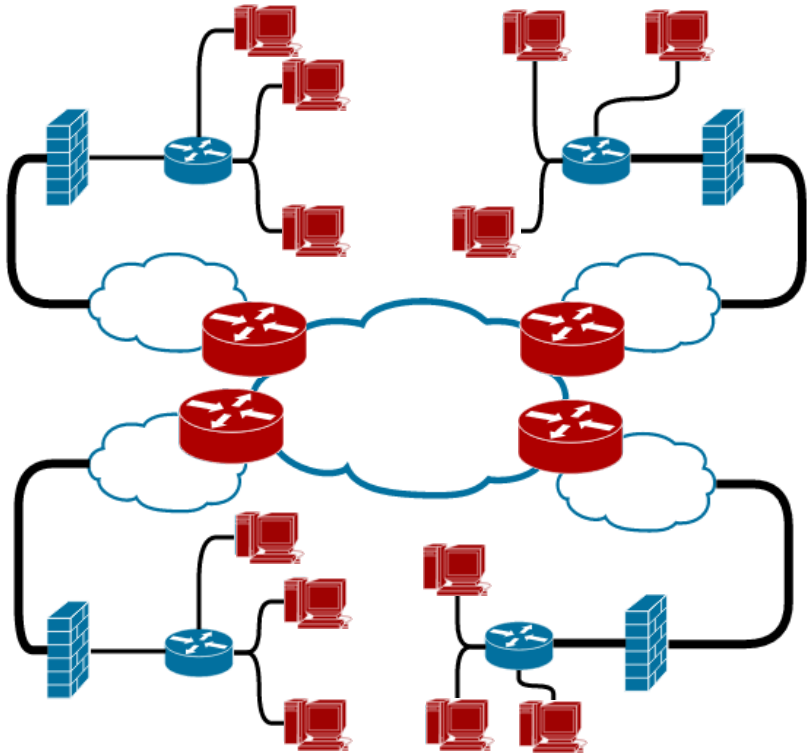


- ML Requirements
  1. Scalable detection
  2. High accuracy (with low false positives)
  3. Work without supervision (unsupervised)
  4. Resilience to model poisoning

Background:
Intrusion Detection in Large Networks

TECHNISCHE UNIVERSITÄT DARMSTADT

- Detecting anomalies in large networks



- Data Requirements
  1. Detect away from the edge
     - Detecting at the edge

     - Detecting away from the edge

  2. Reduce dimensionality of network data

Flows

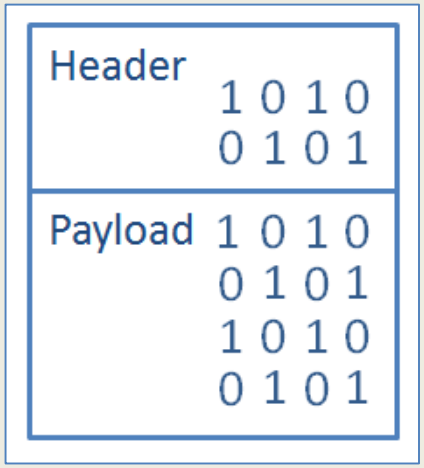
Intrusion Detection in Large Networks

- Working with network data: Network Flows

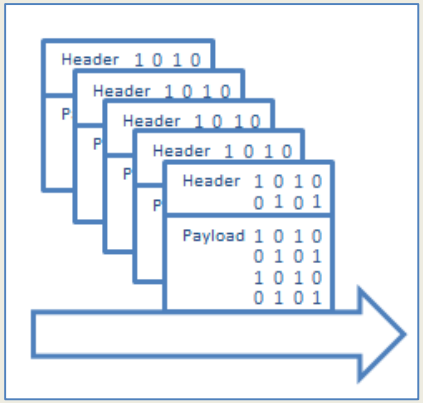
Bytes

1 0 1 0 1 0 1 0
0 1 0 1 0 1 0 1
1 0 1 0 1 0 1 0
0 1 0 1 0 1 0 1

Packets



Flows



Background:
Intrusion Detection in Large Networks

TECHNISCHE UNIVERSITÄT DARMSTADT

- Network Flows

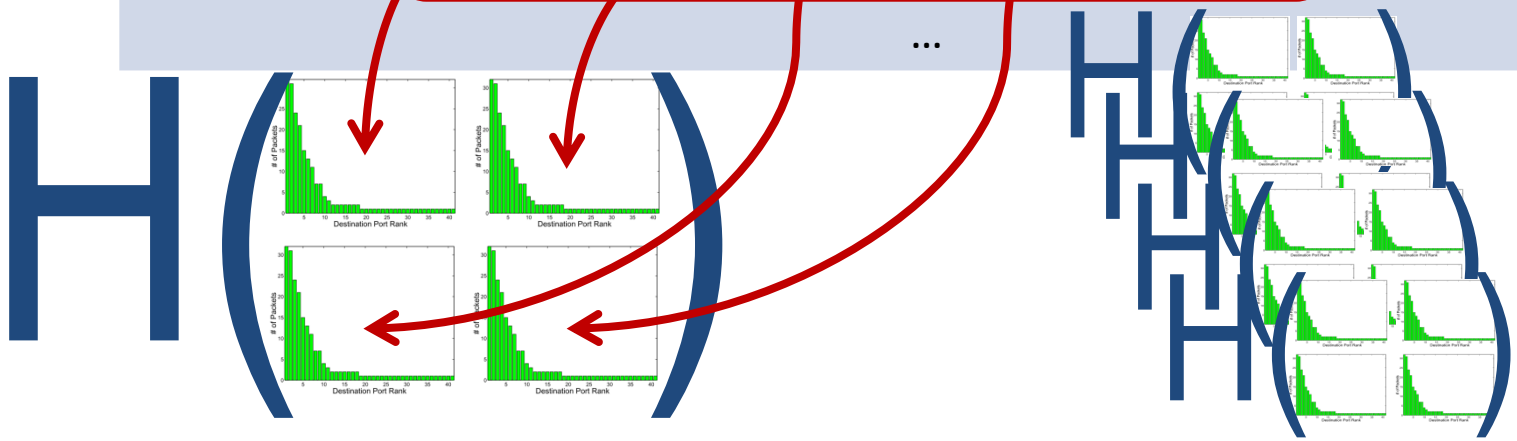
#	Time	Src IP	Dst IP	Src Port	Dst Port	Bytes	Packets
1	X	21.65.71.124	110.45.78.12	51478	80	512	245
2	X	64.73.26.110	21.65.71.124	49652	80	56	1523
3	Y	110.45.78.12	64.73.26.110	58471	21	10,548	235
4	Y	21.65.71.124	110.45.78.12	49652	23	45,687	672
...							

- Condense representation of network data
- Communication summary of two endpoints

Detecting Anomalies in Network Flows

- Extracting features from network flows

#	Time	Src IP	Dst IP	Src Port	Dst Port	Bytes	Packets
1	X	21.65.71.124	110.45.78.12	51478	80	512	245
2	X	64.73.26.110	21.65.71.124	49652	80	56	1523
3	Y	110.45.78.12	64.73.26.110	58471	21	10,548	235
4	Y	21.65.71.124	110.45.78.12	49652	23	45,687	672
			...				



$$H^i = \{h_1^i, h_2^i, \dots, h_D^i\}$$

$$X = \{H^1, H^2, \dots, H^N\}$$

Detecting Anomalies in Network Flows

$$H^i = \{h_1^i, h_2^i, \dots, h_D^i\} \qquad X = \{H^1, H^2, \dots, H^N\}$$

- Anomaly Detection using the **Subspace Method**
  - Principal Component Analysis (PCA) on the X set
  - Compute all Principal Components
  - Select components that define the:
    - Normal subspace – Components capturing the most variance of X
    - Abnormal subspace – Components not used for the normal subspace

	PC1	PC2	PC3	PC4
Eigenvalue	1.0563	0.2062	0.0511	0.0213
Variance	79.11%	15.44%	03.83%	01.60%
Accumulated Variance	79.11%	94.56%	98.39%	100%

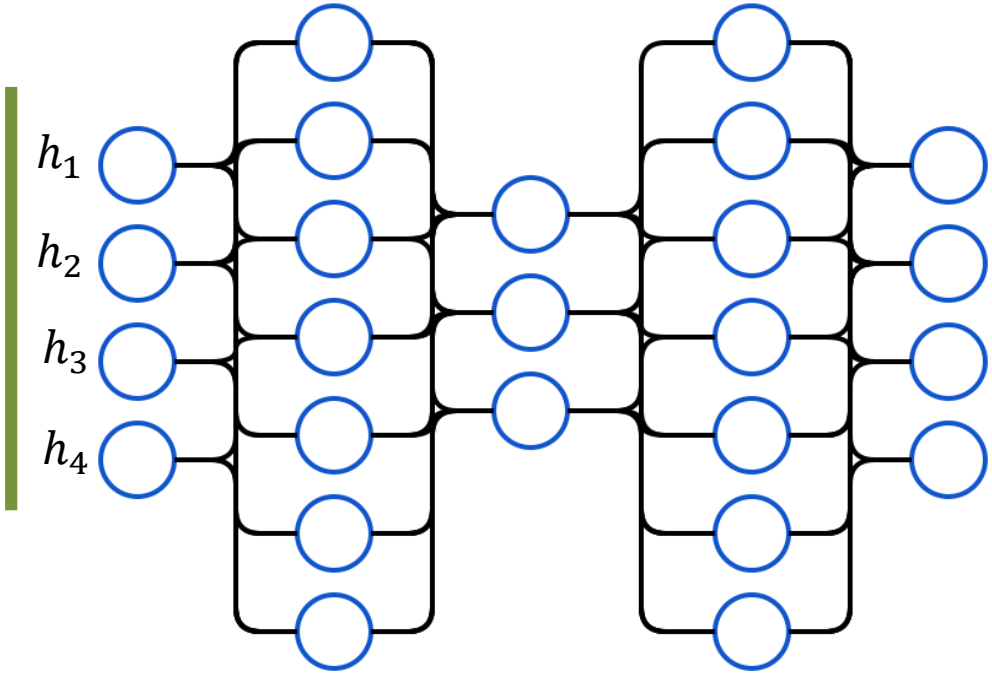
  - Disadvantages
    - PCA is inefficient with many dimensions
    - Abnormal space prone to contamination

Replicator Neural Networks

$$H^i = \{h_1^i, h_2^i, \dots, h_D^i\}$$

$$X = \{H^1, H^2, \dots, H^N\}$$

- Replicator Neural Networks



- Same number of inputs and outputs
- Middle layer smaller than input
- Between 3 to 5 layers
- Original purpose:
  - data compression
  - dimensionality reduction

Anomaly Score:

$$AS(x) = \frac{1}{D}\sum_{i=1}^{D}(\varepsilon_i)^2$$

Evaluation
Experiments and Results

TECHNISCHE UNIVERSITÄT DARMSTADT

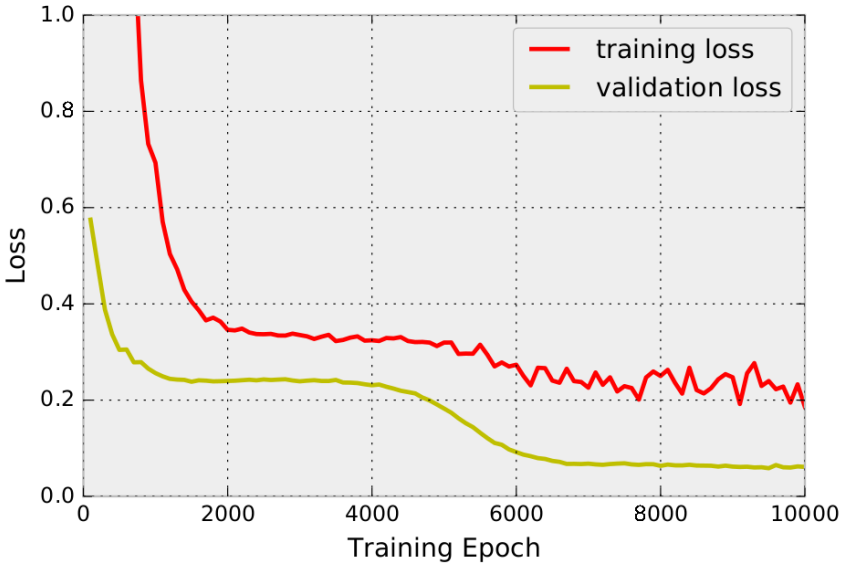
- 4 different sets of experiments
  1. Learning representations of network flows
  2. Detecting anomalies in the training data
  3. Detecting injected attacks
  4. ~~Comparison against the Subspace Method (PCA)~~

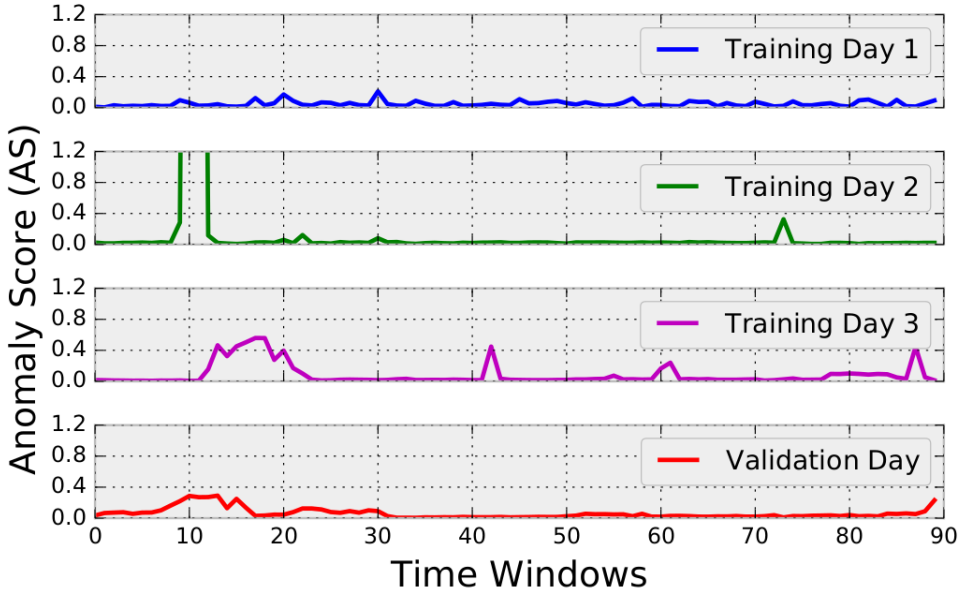
- Experimental setup
  - Dataset: MAWI lab
    - 3 training days
    - 1 validation day
  - Network flows exported every 10 seconds

Experiments and Results

1. Learning representations of network flows

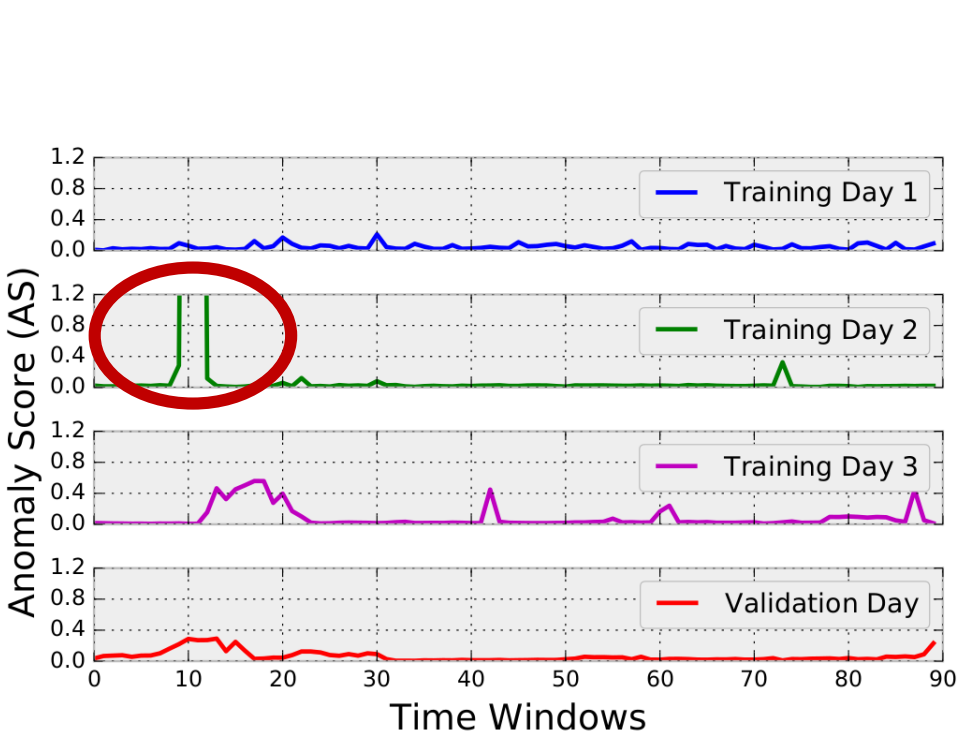
- Standard Replicator Neural Network with 5 layers
- 50% dropout in layers 2 and 4

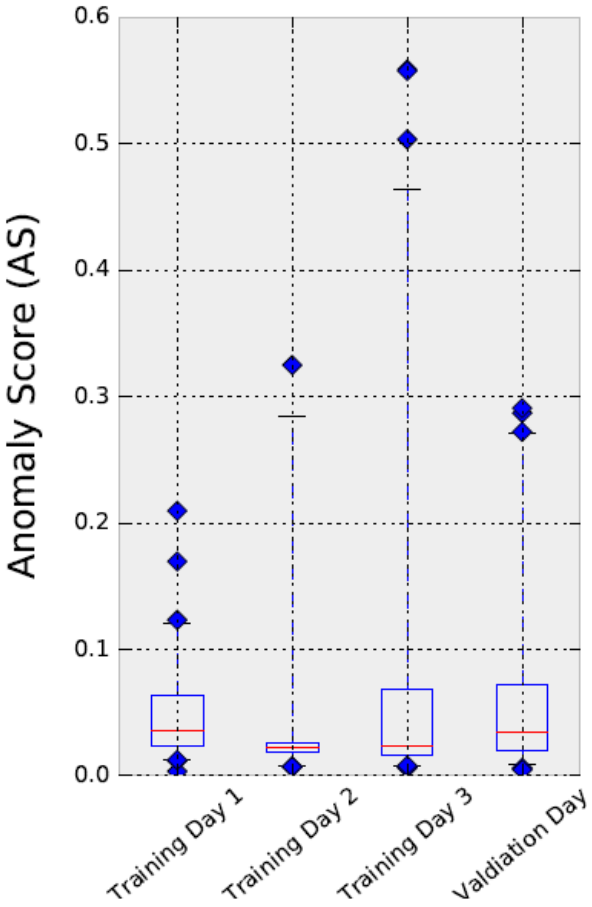




Experiments and Results

2. Detecting anomalies in the training data

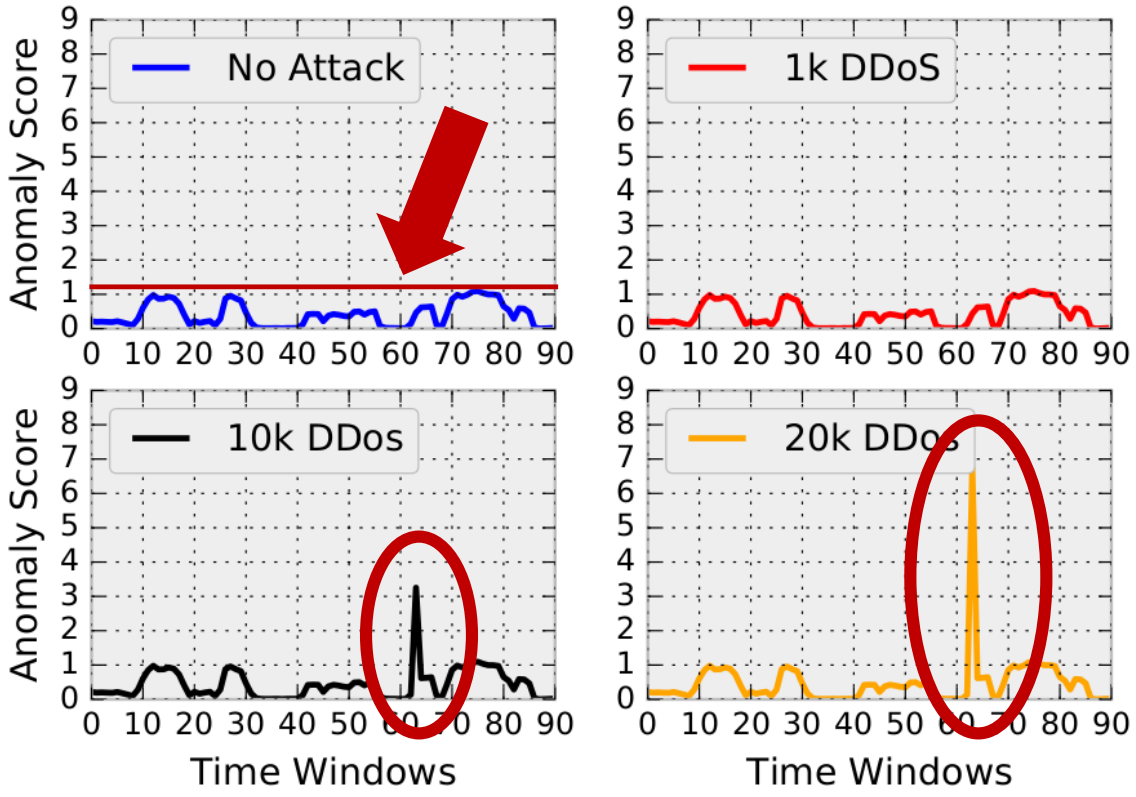




Experiments and Results

3. Detecting injected attacks

- DDoS attack with 4 intensities



Conclusion and Future Work

- Replicator Neural Networks (RNNs)
  - Effectively detect resource exhaustion attacks and network profiling techniques
  - Work in large and real-world backbone internet data
  - Detect attacks in the training data as well as new (unseen) data
  - No labeled data needed

- Future Work
  - Test different autoencoder configurations (beyond RNNs)
  - Add Gaussian noise to the inputs and other input transformation techniques
  - Use network flows and meta-flow features
  - Test other relevant deep learning techniques