Distributed Security Monitoring and Federated Learning of BGP Announcements

Abie Safdie

Problem, Motivation, and Related Work

- Problem: Security threats of BGP announcements
- Motivation: Develop a distributed, federated learning system that monitors BGP announcements in real-time to analyze security threats of BGP announcements
- Related Work: No system that provides real-time monitoring and detection BGP announcements. There are methods to *prevent* BGP prefix hijacking, but none for detection (besides laborious manual system admin monitoring)

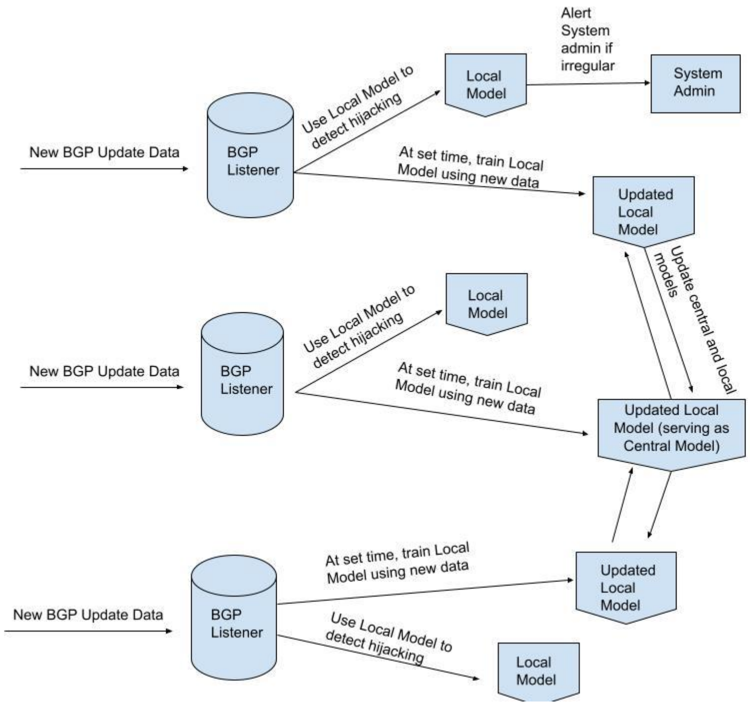
Terminology and System Overview

- Internet Routing and BGP (Border Gateway Protocol)
- BGP Prefix Hijacking
- Federated Learning
- RIPE Database
- Distributed Security Monitoring

Detecting BGP Prefix Hijacking as a Distributed System with regards to Federated Learning

- **Sharing Resources:** Sharing trained models across BGP listeners
- **Transparency:** Local Models are unaware they are using distributed resources (BGP data)
- **Openness:** Integration into other procedures/systems (longest prefix matching)
- **Dependability:** One listener detects, others can depend on their judgement
- **Efficiency and Scalability:** Data spread across multiple listeners more efficiently train models. Easily scalable, just add more listeners.

Federated Learning Architecture



Multithreading

- BGP Listeners run in their own threads to simulate real-world environment
  - Creates need for synchronization
  - One listener serves as leader
    - Election by bullying
  - Leader controls mutual exclusion
  - Leader model serves as “central” model

Federated Learning Procedure and Trust Algorithm

- Models for AS path length, AS path, and BGP community (Announcements only)
  - Scale this up in real system
- Every announcement received, use models to detect hijacking and update trust of BGP Peer
- $\Delta\text{Trust} = \sum_{i=0}^{n-1} \left(1 - e^{-k|1.75 \cdot \text{bgp\_attribute}_i - 1|}\right) \cdot \text{sgn}(1.75 \cdot \text{bgp\_attribute}_i - 1) \cdot d$
  - Where n is the number of bgp_attributes
  - k and d are scaling constants
- Trust falls below a certain threshold alert system admin

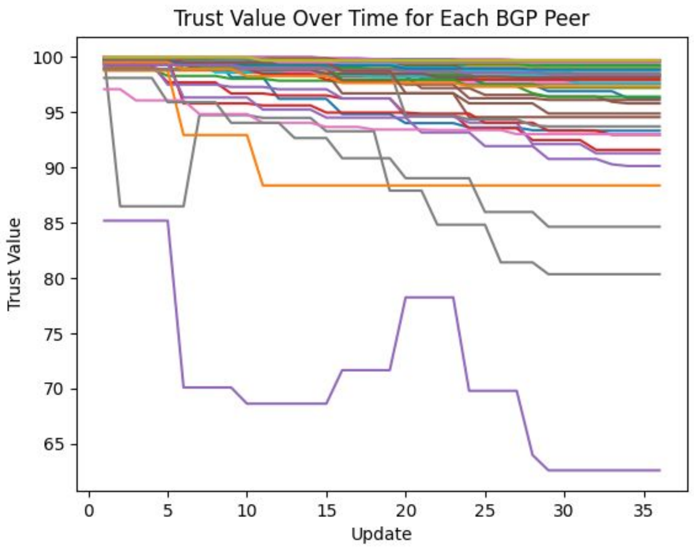
Distributed Security Monitoring

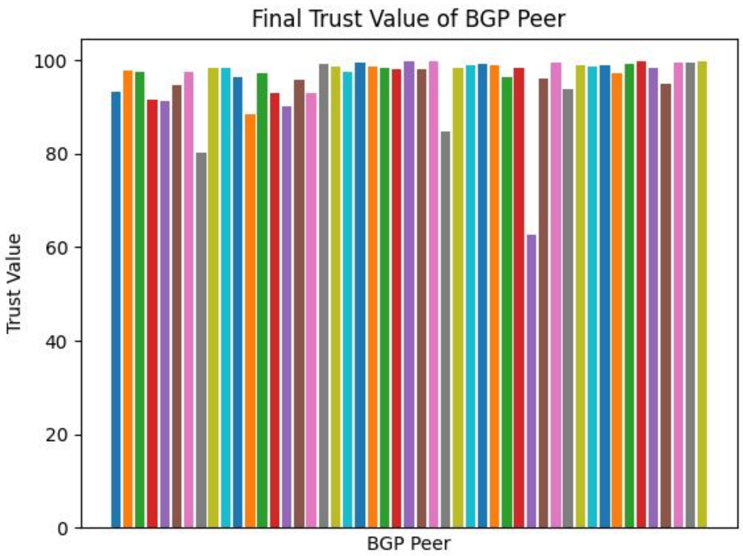
- Listeners contain a shared list of all BGP Peers they have sent announcements
- Any Listener can access and update any BGP Peers trust value
- Monitor trust values in a distributed manner
  - Achieves sharing resources, transparency, openness, dependability, and scalability

Evaluation Methodology

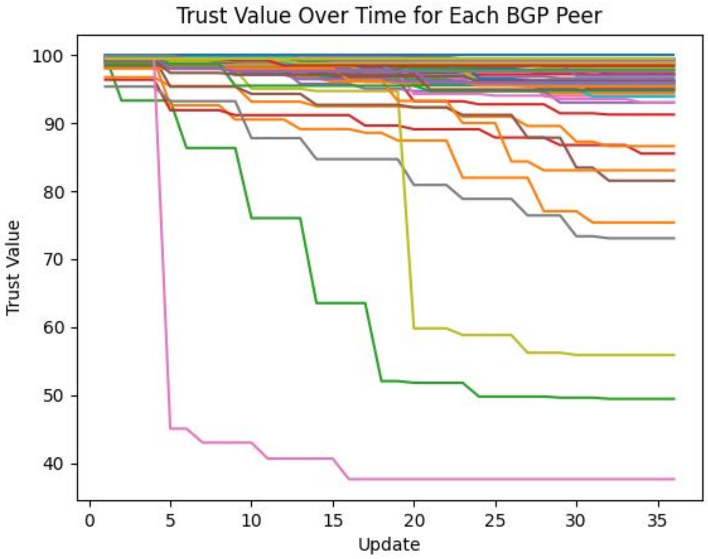
- Manually examined some BGP updates to check expected behavior
- After “conclusion” of BGP data from RIPE, “final” data is printed using matplotlib
  - All BGP Peers who have sent an update have their trust saved for analysis
  - Generate graphs for trust over time and final trust

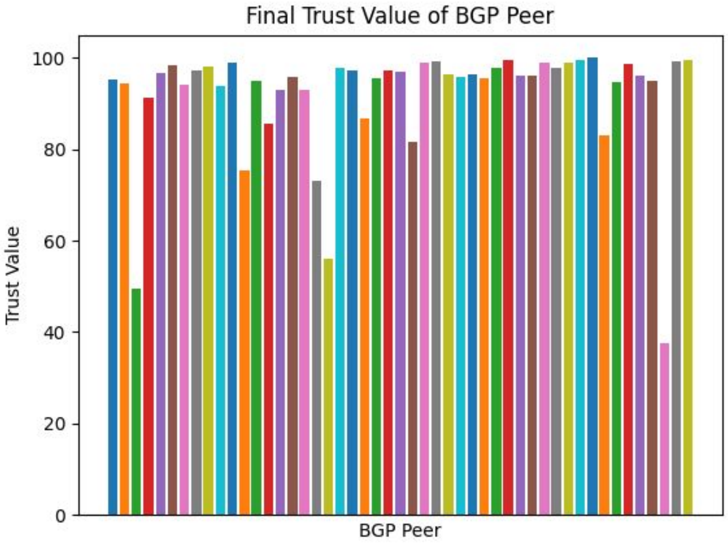
Results for Youtube (4 months of BGP data)





Results for UOregon (4 months of BGP data)





Demo?

Limitations

- Machine Learning
  - Use sophisticated machine learning models
- BGP attributes
  - Time Update Advertised, Number of updates in given time frame, Withdraws
- Resources
  - More data, more destinations

Discussion

- Detecting BGP Prefix Hijacking
- RIPE Database
- Distributed Security Monitoring
- Federated Learning
- Trust

Thank You