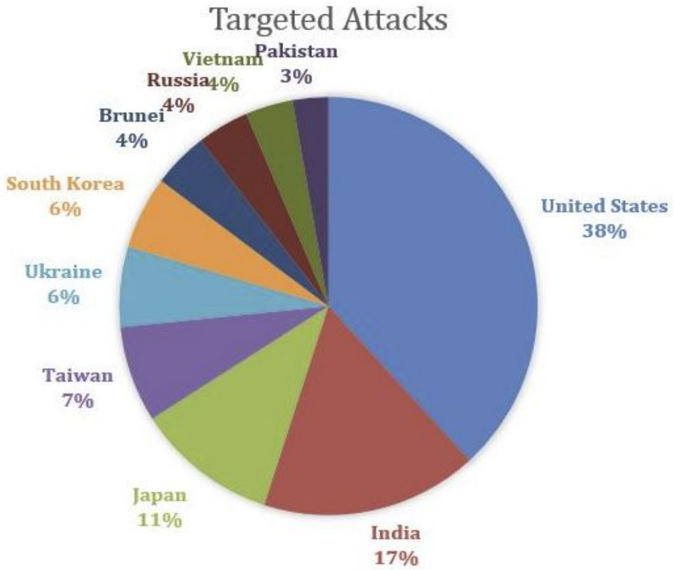
Graph Techniques for Cybersecurity

Benjamin Bowman
George Washington University
bbowman410@gwu.edu

Thank you ARCS Foundation!

Why Cybersecurity?

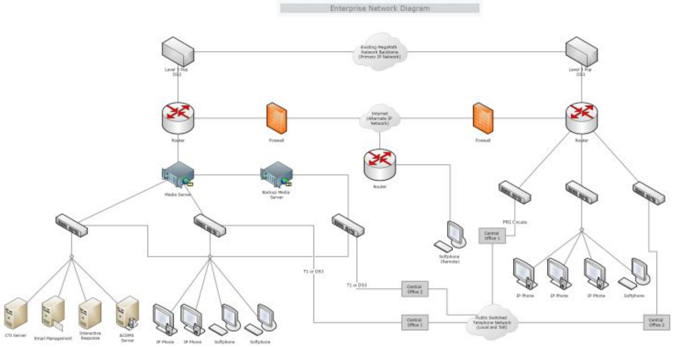
- Ransomware damage costs > **$4 B** in 2017
- Damage related to cybercrime projected to hit **$6 T** annually by 2021
- **20.4 B** IoT devices by 2020
- **90%** of Automobiles will be Internet Connected by 2020



Source: Symantec

Why Graphs?

- Cybersecurity relevant data highly amenable to graph representation
  - Network communication graph, program control flow graph, code property graphs



- Robust data structure capable of representing complex multimodal relationships

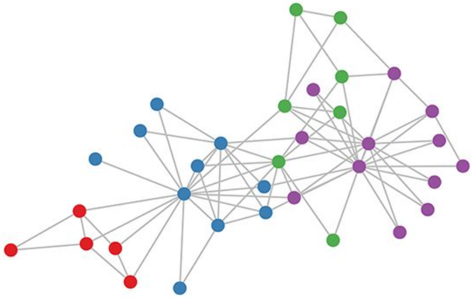
- Many analysis techniques ranging from traditional graph theory to more modern graph learning

Research Vision

Cyber-Relevant Data

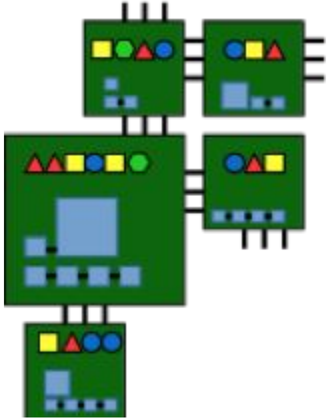
- DNS logs
- Sys logs
- Netflow
- Source Code
- OSI
- etc.

Graph Representation



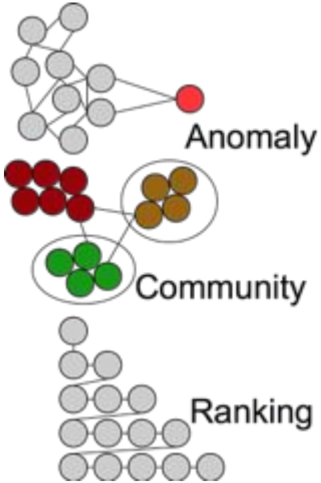
Graph Algorithms

Graph System (hardware, software)



GPU, NVM, Supercomputers

Actionable Cyber Knowledge or Insight



Anomaly Detection via Network Log Analysis

Network Security Today

- **365 day average time to detection -** monitoring and detection techniques insufficient
- Largely signature based, reactionary, requires human expert input
- Algorithmic methods often focused on singular data types

Source: Lockheed Martin

Existing techniques fail to utilize the complex relationships between disparate yet related cyber data points

Network Behavior Modeling via Unsupervised Graph Learning

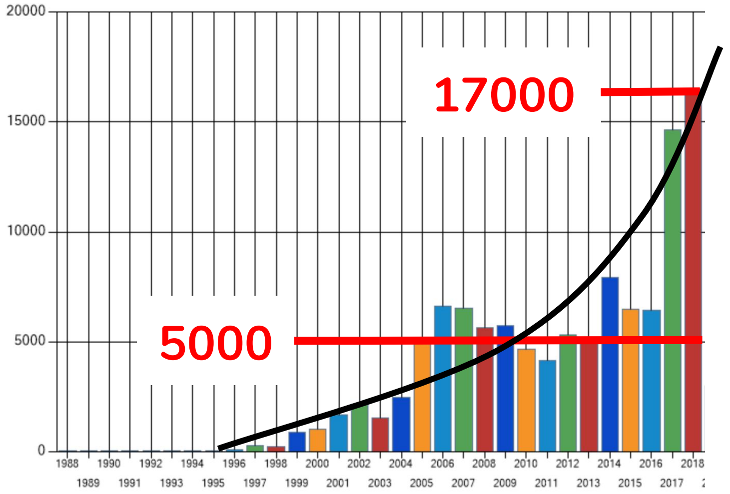
- **Goal:** Utilize unsupervised graph learning techniques on streaming graphs containing many different cyber-relevant logs to learn behavior and pattern-of-life of network entities

- **Key Idea:** Identify malicious behavior as series of behavior changes

Vulnerability Detection in Software Source Code

Vulnerability Detection Today

Vulnerabilities by Year



Source: NVD

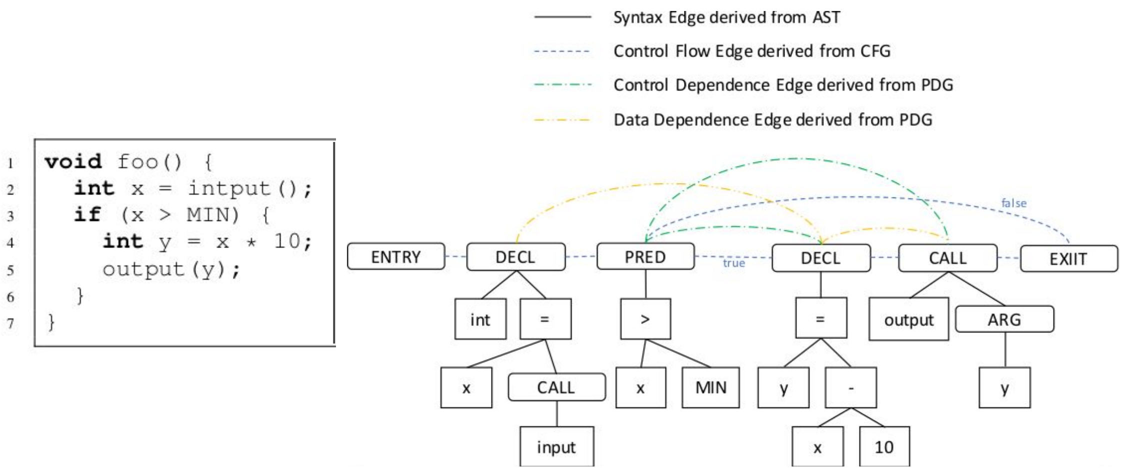
- **5000** vulnerabilities in 2013 vs **17000** in 2018
- Techniques fall into two main camps: **pattern detection** and **similarity detection**
- Patterns **manually** generated by human experts - not scalable
- Similarity detection typically based on **exact** matching - not flexible

Existing techniques are either good at finding one vulnerability in many programs, or finding many vulnerabilities in one program

vGraph: A Generalized Graph Representation for Vulnerability Detection and Discovery

- **Goal:** Be able to find many different vulnerabilities, in many different programs

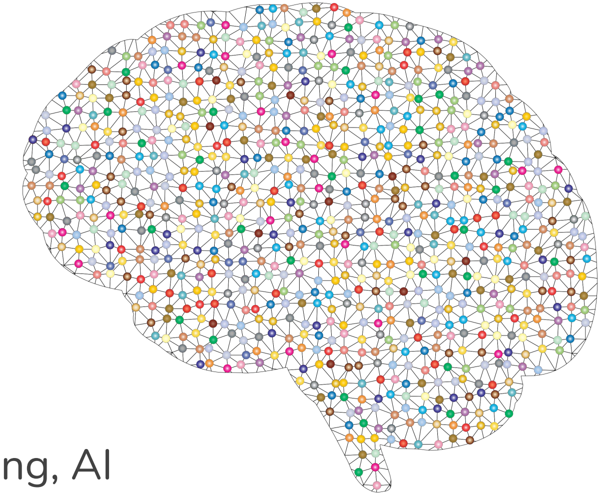
System	Precision (%)	Recall (%)	F1 (%)
VUDDY	85	91	88
VulPecker	90	60	70
VulDeePecker	79	83	81
vGraph	92	89	90



- **Key Idea:** Use a complex graph structure to represent **only the structure highly related to the vulnerability**

- Apply approximate subgraph matching techniques

Conclusion



- Graphs are powerful representations of data and have an active research community

- Recent advancements in Machine Learning, Deep Learning, AI on graphs are proving very effective

- Cybersecurity tasks and data are highly amenable to a graph representation and analysis (e.g., vulnerability detection, network behavior modeling, etc)

- Advancements in cybersecurity will be critical to the future of our increasingly digital society

Thanks for Listening!

Thank you ARCS for your support!

Benjamin Bowman
George Washington University
bbowman410@gwu.edu