National Institute of Standards and Technology
Technology Administration, U.S. Department of Commerce

Measuring and Improving the Effectiveness of Defense-in-Depth Postures

PETER MELL, JAMES SHOOK - NATIONAL INSTITUTE OF STANDARDS AND TECHNOLOGY

RICHARD HARANG – ARMY RESEARCH LABORATORY

ACSAC SECOND ANNUAL INDUSTRIAL CONTROL SYSTEM SECURITY WORKSHOP
LOS ANGELES, USA, DECEMBER 2016

Outline / Summary of Findings

- There are 3 important components to an idea defense in depth (DiD) architecture: depth, width, and strength
- We can express and measure these DiD components using layered colored attack graphs
  - Depth can be measured using shortest color path (SCP)
  - Width can be measured using minimum color cut (MCC)
  - SCP and MCC measurements are NP-Hard
- There exist computationally efficient and effective approximation algorithms for determining SCP and MCC.
- Empirical experiments conducted on both random graphs and generated industrial control system graphs (ICS)

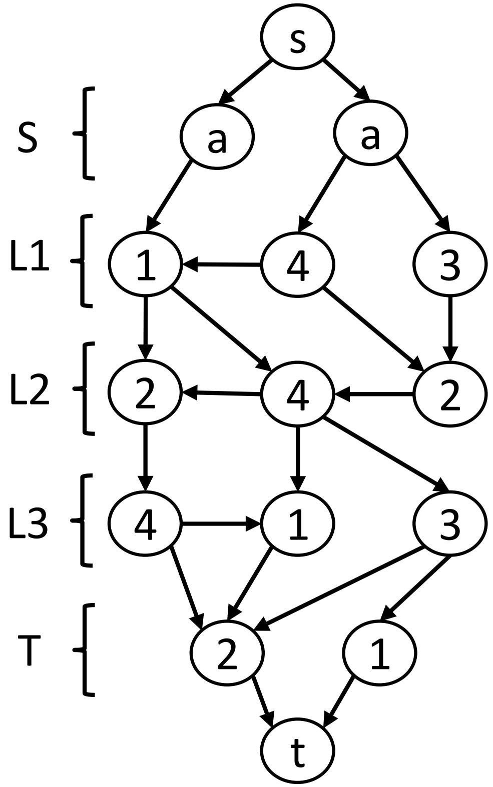
3 Important Components to an Ideal Defense in Depth Architecture

- It must be 'deep', meaning that there are many *independent* layers of security.
  - Independence refers to each layer as representing a distinct challenge to the attacker
- It must be 'narrow', meaning that the number of *independent* attack paths is minimized (equivalent to finding the smallest set of distinct vulnerabilities whose mitigation could cut all attack paths).
  - Independence refers to each step of a particular attack path representing a distinct challenge to the attacker
- We will see that this necessary concept of independence takes a simple algorithmic problem and makes it NP-Hard.
- And it must be 'strong', meaning that each layer must provide the greatest possible deterrent to an attacker.

In this work, we focus on measuring only depth and width because the strength of a layer is difficult to accurately assign.

Colored Layered Attack Graphs

- The a nodes represent locations where attacks are expected to originate
- L1-L*n* are sets of hosts at differing distances from S (standard shortest path distance)
- Set T is the set of hosts with critical information or functionality (the crown jewels)
- The numbers in a node represent the 'colors' which are distinct vulnerabilities types
- s and t are placeholder 'start' and 'end' nodes (for algorithmic convenience)
- NOTE: Multiple nodes may map to a single host
- NOTE: Edges may represent attacks within a host or between hosts



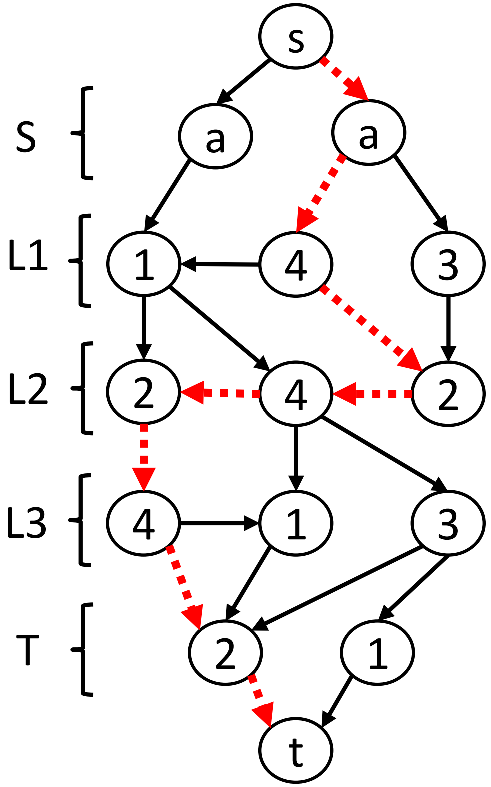
Depth Metric: Shortest Color Path (SCP)

- We must find a path from s to t that uses that fewest number of colors.

- We count only distinct colors instead of nodes because we assume that if an attacker can exploit a vulnerability on a host *a* then the attacker can exploit the same vulnerability on some host *b* (provided logical connectivity is available).

- The requirement for logical connectivity is easy to take into account because it is modelled by the edges in G.

- We refer to the entire operation of measuring the depth as finding the shortest color path (SCP).

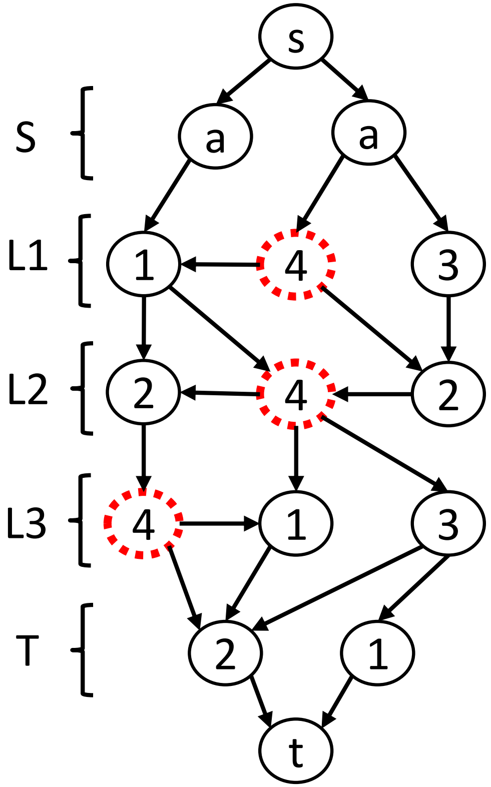


Width Metric: Minimum Color Cut (MCC)

- We must find the smallest set of colors such that at least one color in the set will occur on every possible attack path.

- Another way to view this is to find a cut set of nodes with the fewest colors that eliminates all paths from s to t. We thus refer measuring the width as finding the minimum color cut (MCC).

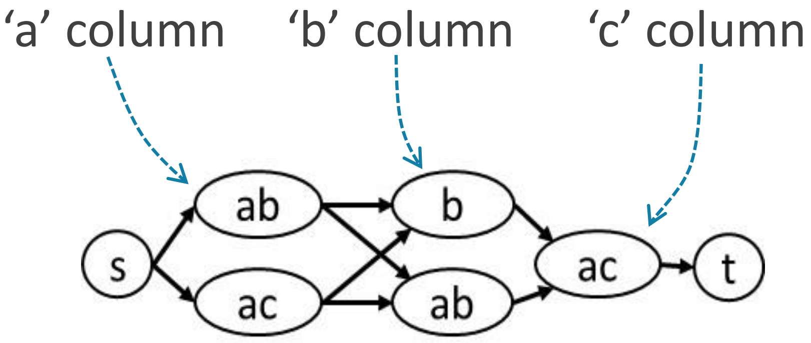
- Note that the nodes in S are not candidates for the MCC as they represent the attackers (they are not vulnerability options for the attackers to exploit). However, the nodes in T are eligible as they represent exploitation opportunities for the attackers on critical targets. Nodes s and t are not eligible as they were added for algorithmic convenience.



SCP and MCC Measurements are NP-Hard by Polynomial Time Set Cover Reductions

SCP: let U={a, b, c} and S={ac, ab, b}

Transform to the below colored attack graph in polynomial time:



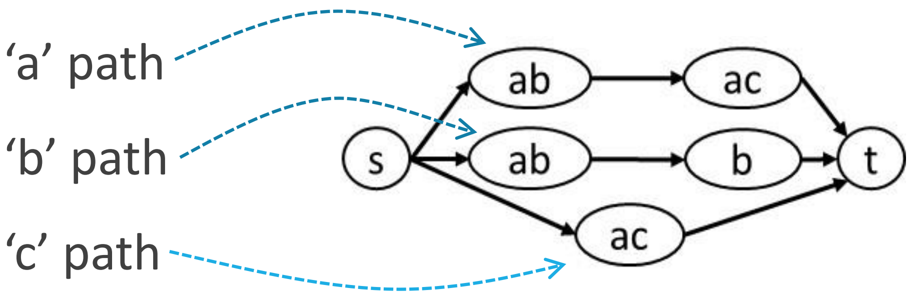
‘a’ column ‘b’ column ‘c’ column

We create one column of elements from S for each element of U.

A minimum SCP is the set of colors that correspond to the minimum set within S that covers U.

MCC: let U={a, b, c} and S={ac, ab, b}

Transform to the below colored attack graph in polynomial time:



‘a’ path

‘b’ path

‘c’ path

We create one s-t path of elements from S for each element of U.

A minimum MCC is the set of colors that correspond to the minimum set within S that covers U.

Empirical Design: Random Layered Graphs

Generation of random layered graphs:

- Chose number of layers, nodes per layer, attack sources, and high value targets
- Chose probabilities for intra-layer edges and inter-layer edges
- The number of layers chosen effects (but does not decide) the depth of the graph
- The number of nodes per layer effects (but does not decide) the width of the graph
- By varying the number of layers and nodes per layer, we can scale the generated graphs

Graph Characteristics:

- Variable number of distinct vulnerabilities
- 10 layers and 100 nodes per layer
- 100 attack sources and 100 high value targets
- Varying the number of layers and the number of nodes per layer resulted in no change to the relative effectiveness of the algorithms

Experiments conducted on an Ubuntu virtual machine with 10GB RAM on a commodity laptop

Emperical Design: Industrial Control System

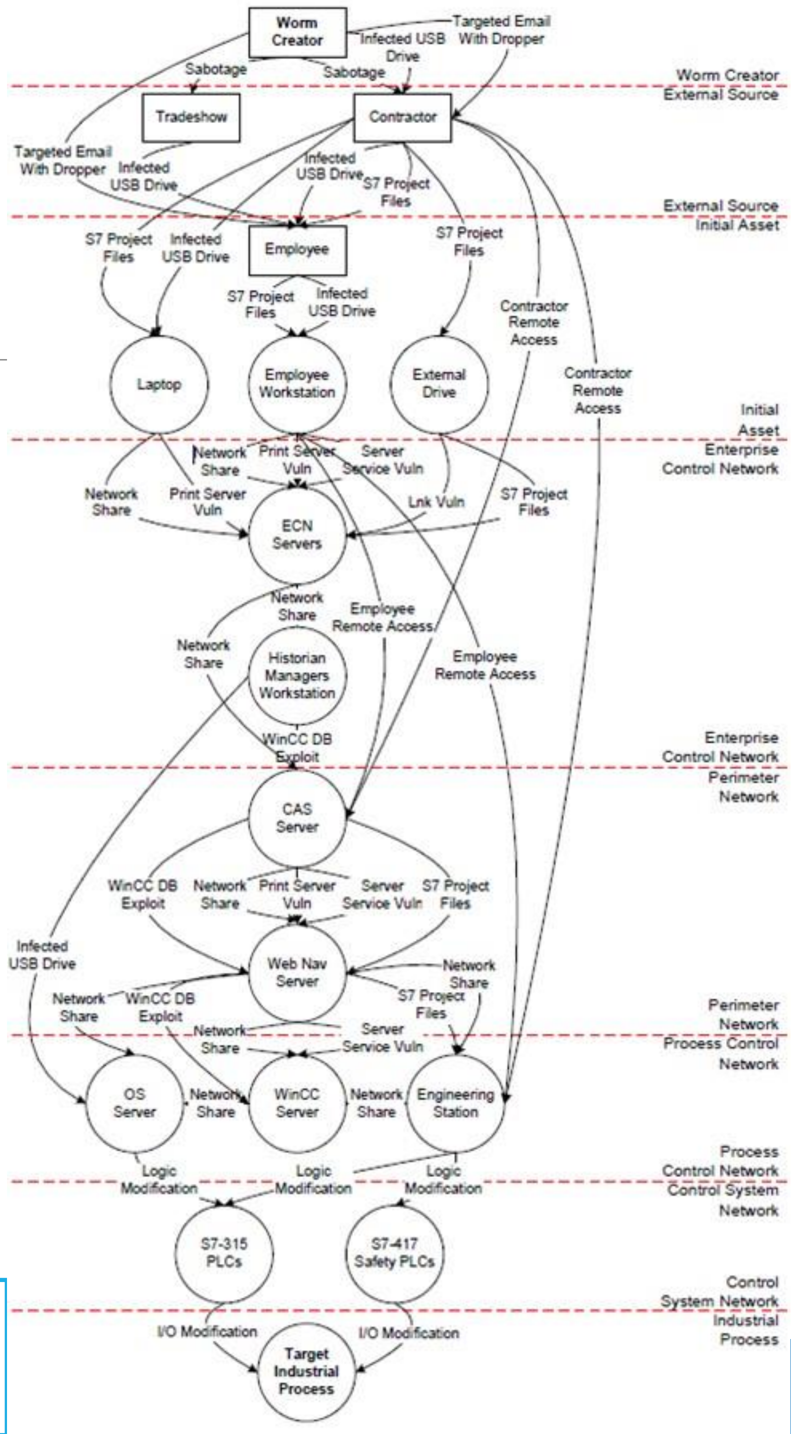
Attack template shows ICS system secured with ‘best practice’ methods.

- Nodes represent actors that we instantiate into actual instances
- Edges represent possible attack vectors

We generate colored layered attack graphs conformant with the attack template but that randomly instantiate actor types into actors instances and randomly enable available attack edges from the template.

Eight Layers

- Work Creator
- External Source
- Initial Asset
- Enterprise Control Network
- Perimeter Network
- Process Control Network
- Control System Network
- Industrial Process

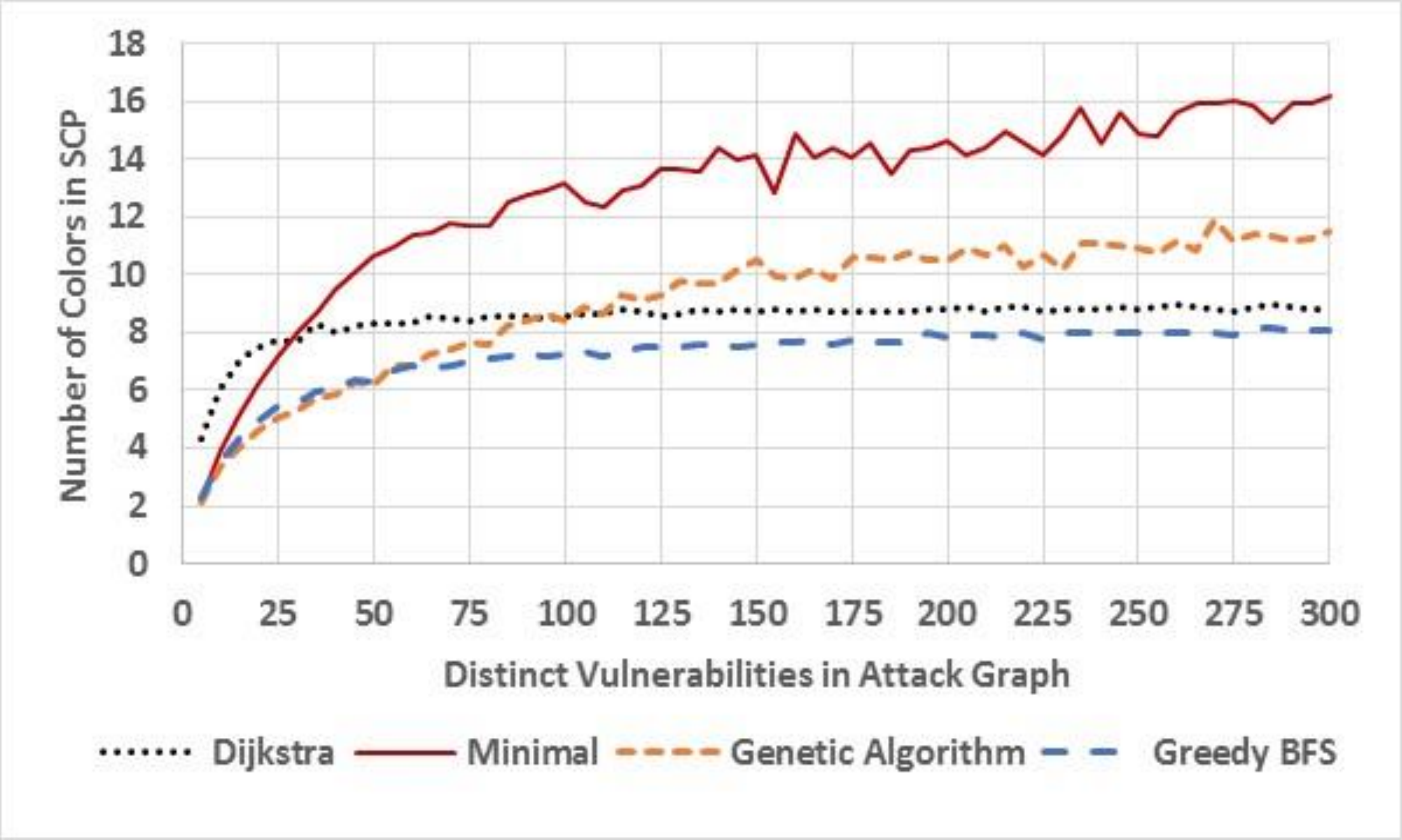


Source: E. Byres, A. Ginter and J. Langill, "How Stuxnet Spreads - A Study of Infection Paths in Best Practice Systems," 2011.

Shortest Color Path Algorithms (Depth Measurement)

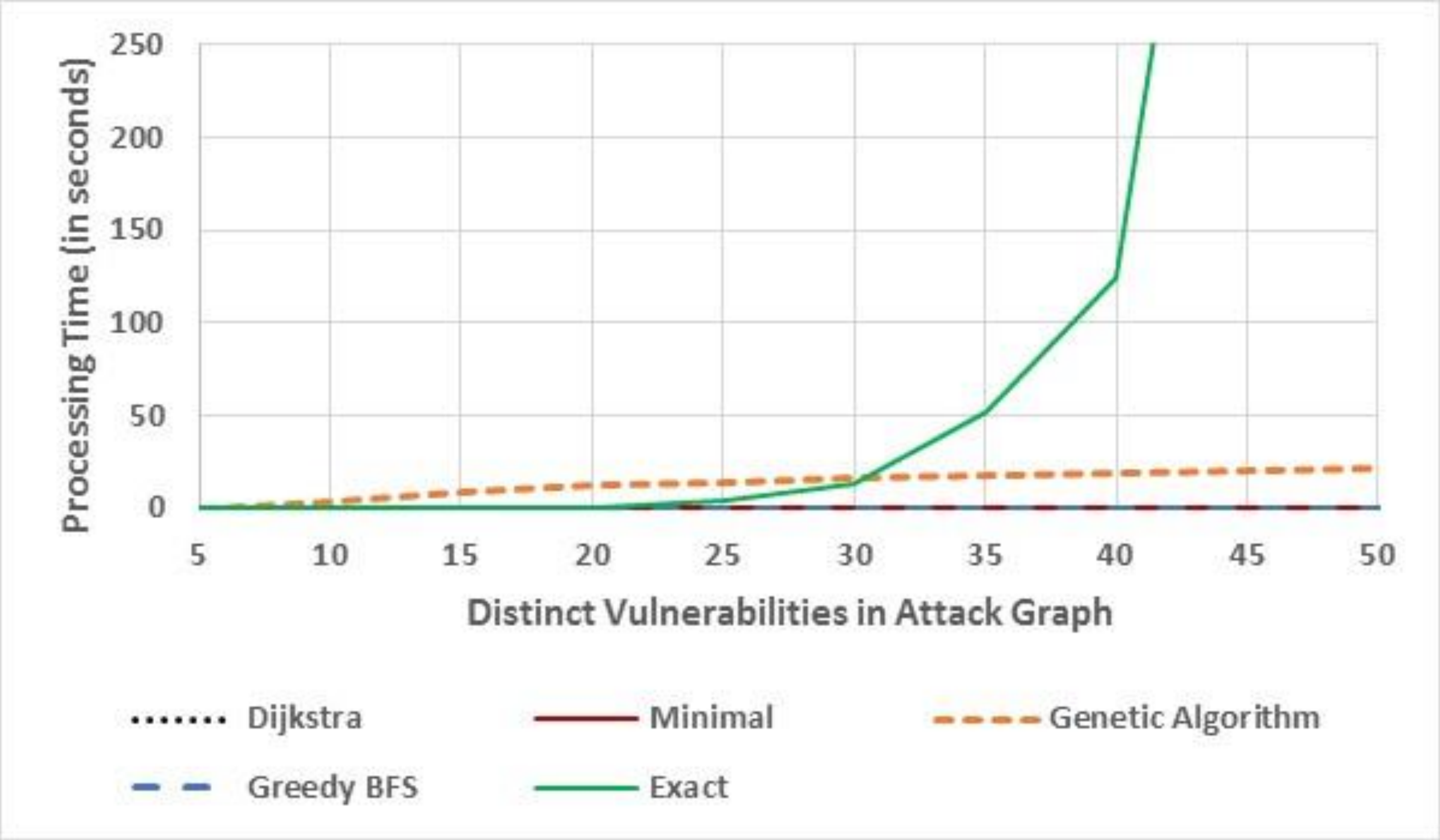
- ✓ Greedy breadth first search
- Genetic algorithm
- Linear minimization (i.e., find ‘necessary’ colors)
  - Very fast and used to optimize all other algorithm results
- Dijkstra shortest path (naïve approach)
- Exact algorithm

Depth Measurements for Random Layered Graphs



Note that as the number of distinct vulnerabilities approaches the number of nodes in the attack graph attack graph, the Dijkstra and Greedy BFS approaches converge.

Execution Time: Depth Measurements



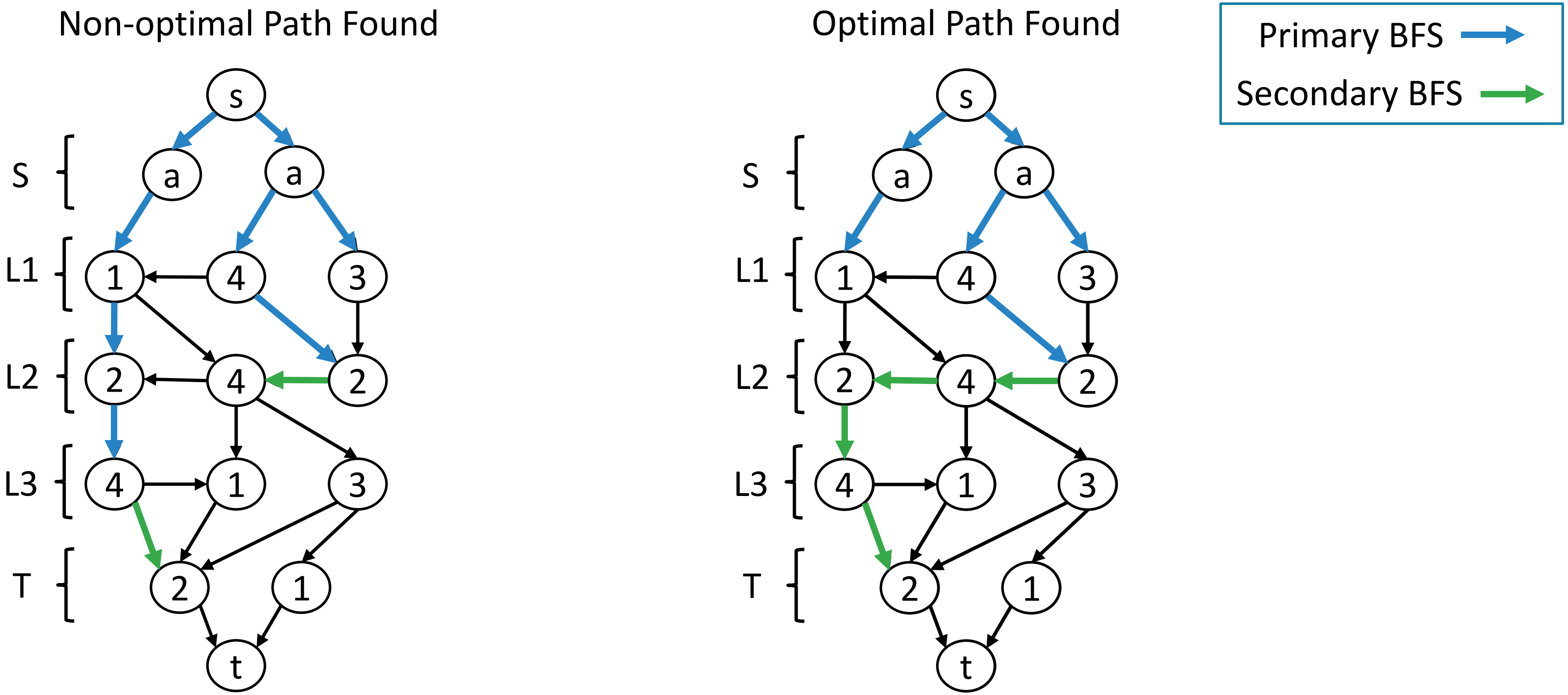
At 300 distinct vulnerabilities, the Greedy BFS took only 0.3 seconds

Greedy Breadth First Search Algorithm

- Primary Breadth First Search (BFS)
  - Normal BFS labels each node with its predecessor in the search so the shortest path can be read out once the terminal node is reached.
  - We modify the normal BFS to be color aware by labelling each visited node with the set of ALL colors on the identified shortest path.
  - The BFS is also not allowed to visit any node that has already been visited by either the primary or secondary BFS
  - We then perform a secondary BFS from each visited node, *x* (see below)

- Secondary BFS from some node *x*
  - Perform a BFS from *x* limited to visiting the following nodes:
    - Nodes not visited by any primary or secondary BFS
    - Nodes whose color is an element of x's color set label
  - Label all visited nodes with the set of ALL colors on the identified shortest path from the start node of the primary BFS.

Example Executions: Greedy BFS



Non-optimal Path Found

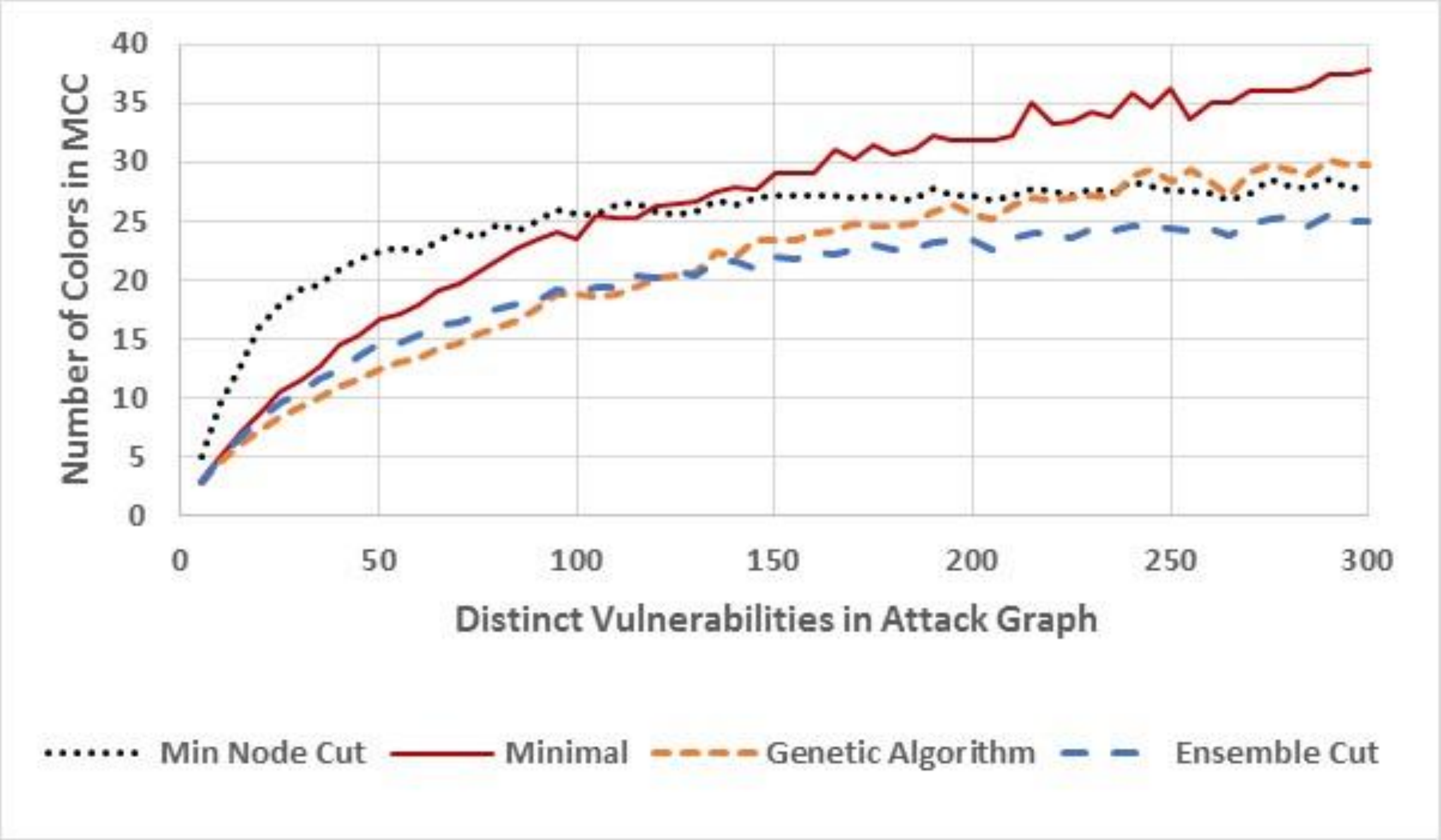
Optimal Path Found

Minimum Color Cut Algorithms (Width Measurement)

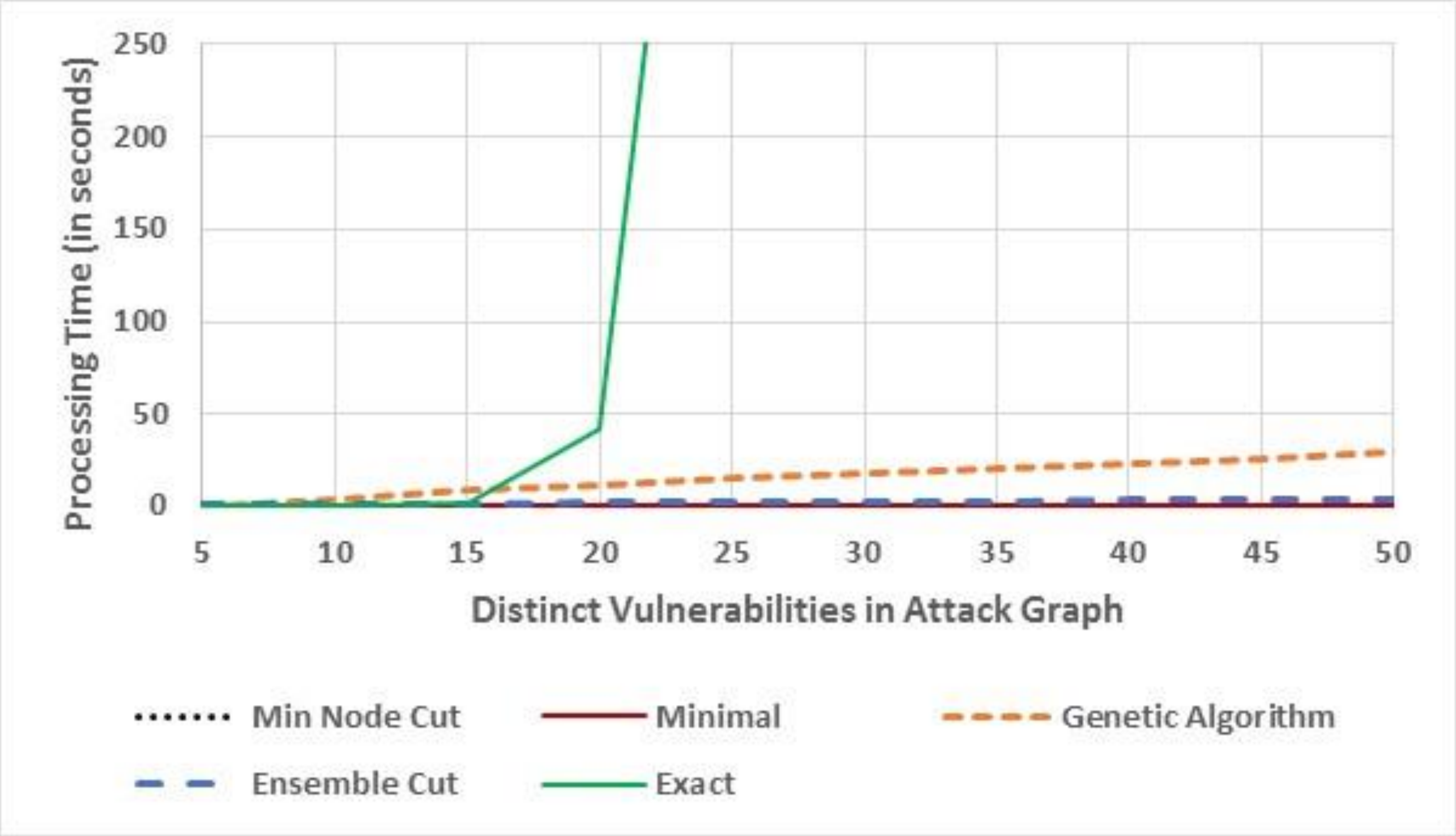
- ✓ Ensemble color aware node cut
- Genetic algorithm
- Linear minimization (i.e., find ‘necessary’ colors)
  - Very fast and used to optimize all other algorithm results

- Minimum node cut (naïve approach)
- Exact algorithm

Width Measurements for Random Layered Graphs



Execution Time: Width Measurements



Ensemble Color Aware Node Cut Algorithm

- Iteratively calculate the minimum node cut that separates the nodes in set S from t in the colored attack graph being evaluated. After each iteration, remove all nodes of some chosen color. We repeat this until there does not exist a path from nodes in S to t.

- Methods for Choosing the Color to Remove in Each Iteration

  1. Choose the color that is most frequent within the discovered node cut.

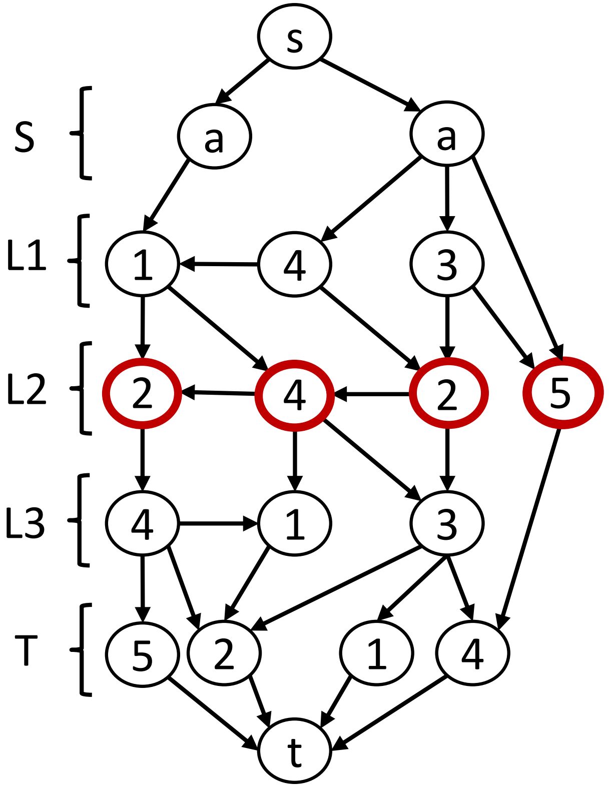
  2. Restricted to the colors found in the standard node cut, choose the color that occurs most frequently in the entire attack graph.

  3. Use method 1 above unless there is no color with an occurrence within the cut of greater than 1. In that case, use method 2.

- For our ensemble approach, we used all three methods.

Example Executions: Color Aware Node Cut Method 3

Expanded Example Graph



1. Minimum node cut finds a cut of 4 nodes with 3 distinct colors

2. Algorithm removes all nodes of color 2 because this color is the most popular within the identified cut

3. Next minimum node cut finds a cut of 2 nodes with 2 distinct colors (4 and 5)

4. Both colors 4 and 5 are equally popular within the discovered node cut, so we have a tie.

5. The algorithm removes all nodes of color 4 because this color is more popular than color 5 in the rest of the graph

Uses of SCP and MCC Measurements

- Understand networks DID posture with respect to expected sources of attacks (both internal and external). Most useful for networks with ‘crown jewels’.
- Can measure single network over time to determine trend in DID posture.
- Can be used to compare DID postures between multiple networks within the same enterprise (to be used to prioritize remediation funding).
- The ‘colors’ returned from our algorithms can be used to highlight vulnerability types that, if fixed, could significantly improve the DID posture.

Conclusion

- A networks defense in depth characteristics can be modeled using a layered colored attack graphs
- Depth can be measured using shortest color path (SCP)
- Width can be measured using minimum color cut (MCC) for width
- While NP-Hard, we have shown that computationally efficient and effective approximation algorithms exist for determining both the SCP and MCC.

NCCoE
NATIONAL CYBERSECURITY CENTER OF EXCELLENCE
NIST
NATIONAL INSTITUTE OF STANDARDS AND TECHNOLOGY
UNITED STATES DEPARTMENT OF COMMERCE
100 Bureau Drive