# Determining Placement of Intrusion Detectors for a Distributed Application through Bayesian Network Modeling

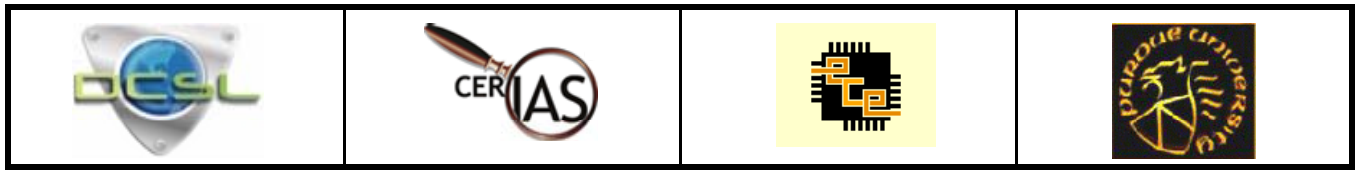

**Gaspar Modelo-Howard, Saurabh Bagchi, and Guy Lebanon**

Dependable Computing Systems Lab (DCSL) &

Center for Education and Research in Information Assurance and Security (CERIAS)

School of Electrical and Computer Engineering

Purdue University

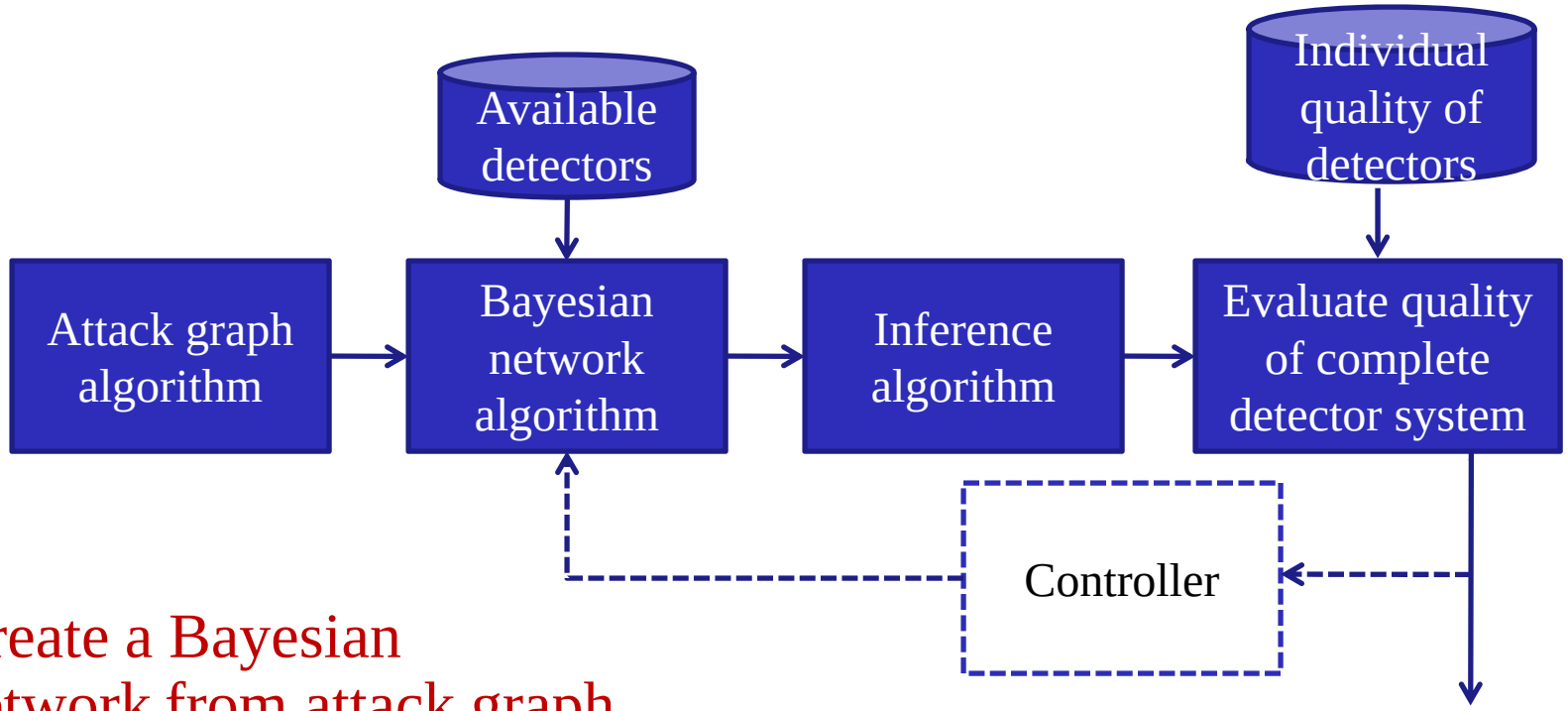
# Motivation

- Placement and choice of detectors are more an art than a science, relying on expert knowledge
- Impact of choice is significant on the accuracy and precision of overall detection function
- More (detectors) is not always better
  - Economic cost
  - Administration cost
  - Performance cost

# Contribution

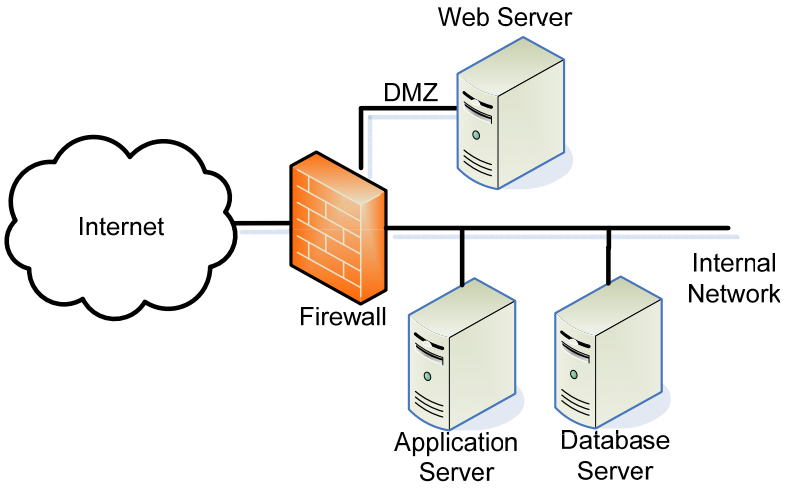
- Method to evaluate the effect of detector configuration on system security
- Provide a greedy algorithm for determining detector settings in a distributed system

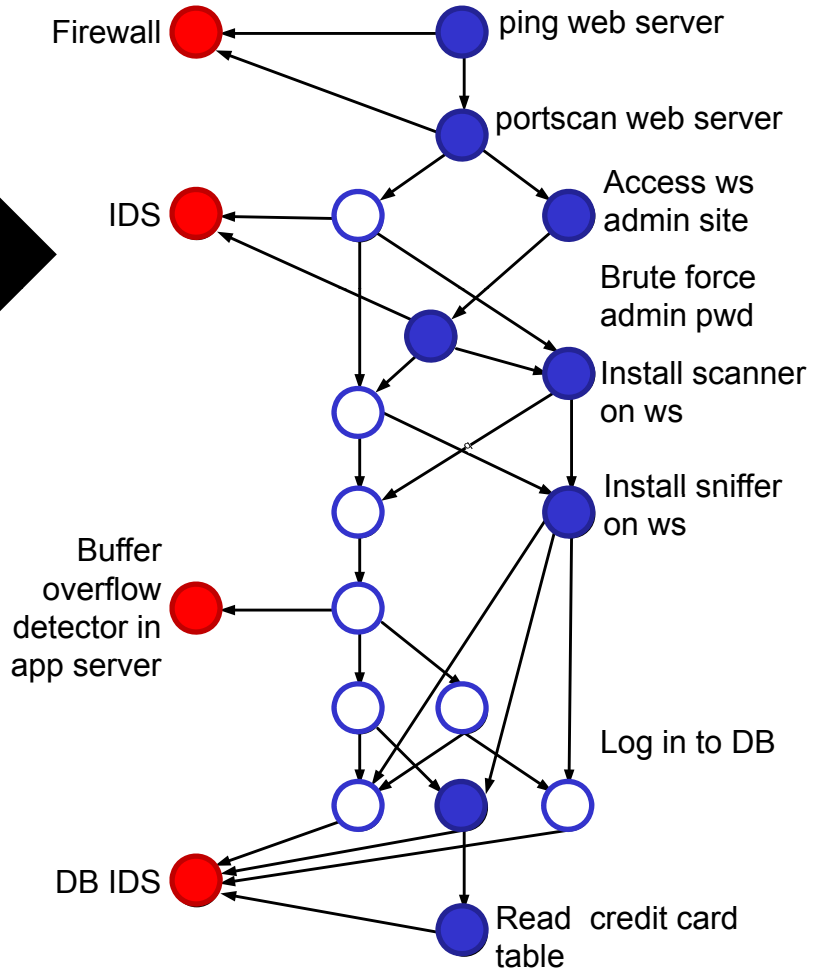
# Overall Framework



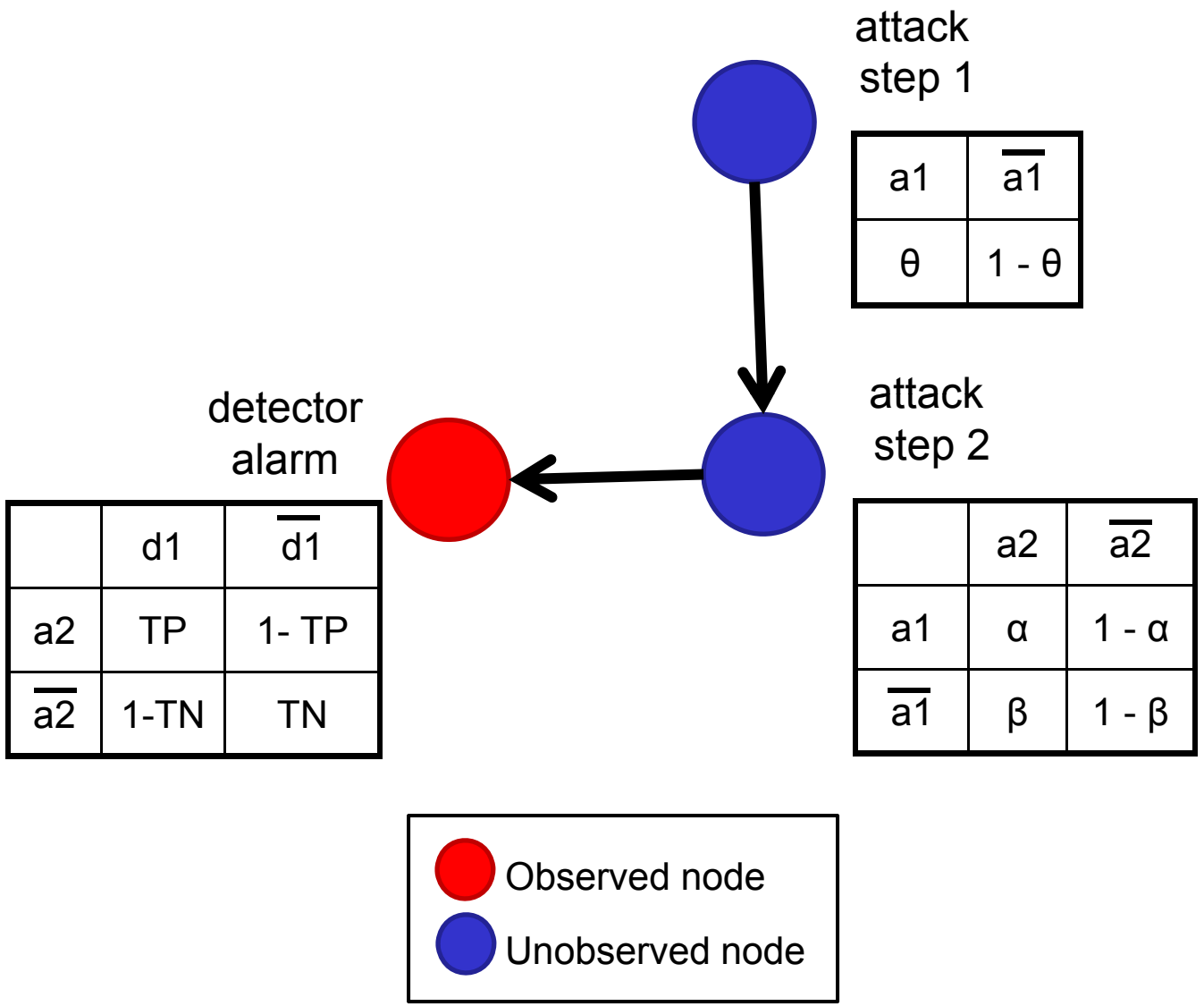

- Create a Bayesian network from attack graph, adding detectors and CPTs
- Run inference algorithm to compute possible combination of detectors, given an ultimate goal (from attacker perspective)
- Controller to adjust detector setting when network changes (not currently implemented)

# Sample Network and Corresponding Bayesian Network



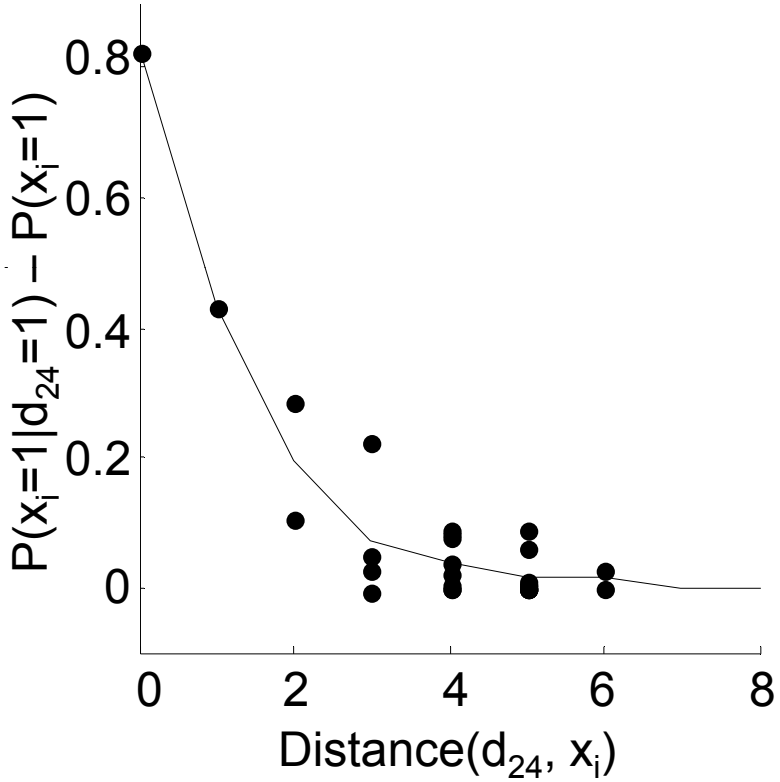



# Probability Values on a Bayesian Network



# Exp. 1: Impact of Distance between Detectors and Attack Steps

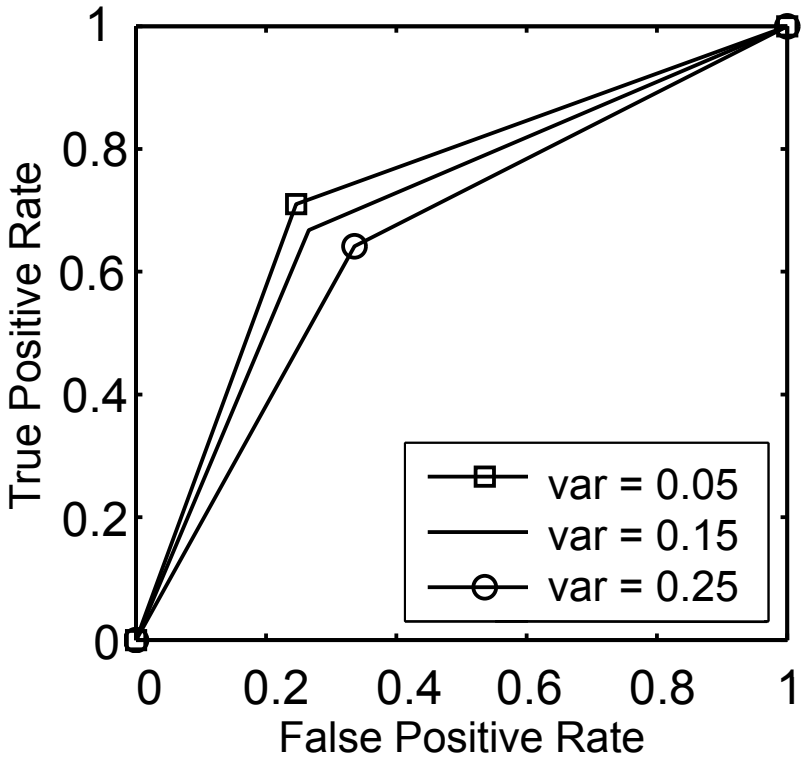
- Objective: quantify gain in placing detector close to service where attack might occur
- Results show that a detector can affect nodes inside a “radius” of up to three edges




$d_{24}$ = detector
$x_i$ = attack step

# Exp. 2b: Impact of Imperfect Knowledge

- Objective: mimic different levels of expertise of system administrator
  - Expert, intermediate, naïve
- As variance increases, performance suffers
  - BN shows inherent resilience
  - Depends on location of attack steps and detectors



# Algorithm

- Input:
  - Bayesian network $BN = (V, CPT(V))$
  - Set of detectors $D = (d_i, V(d_i), CPT[i][j])$
  - Cost
- Output:
  - Set of selected detectors and corresponding locations
- Greedy approach to select those detectors that provide greatest coverage (benefit) given a threshold (cost)

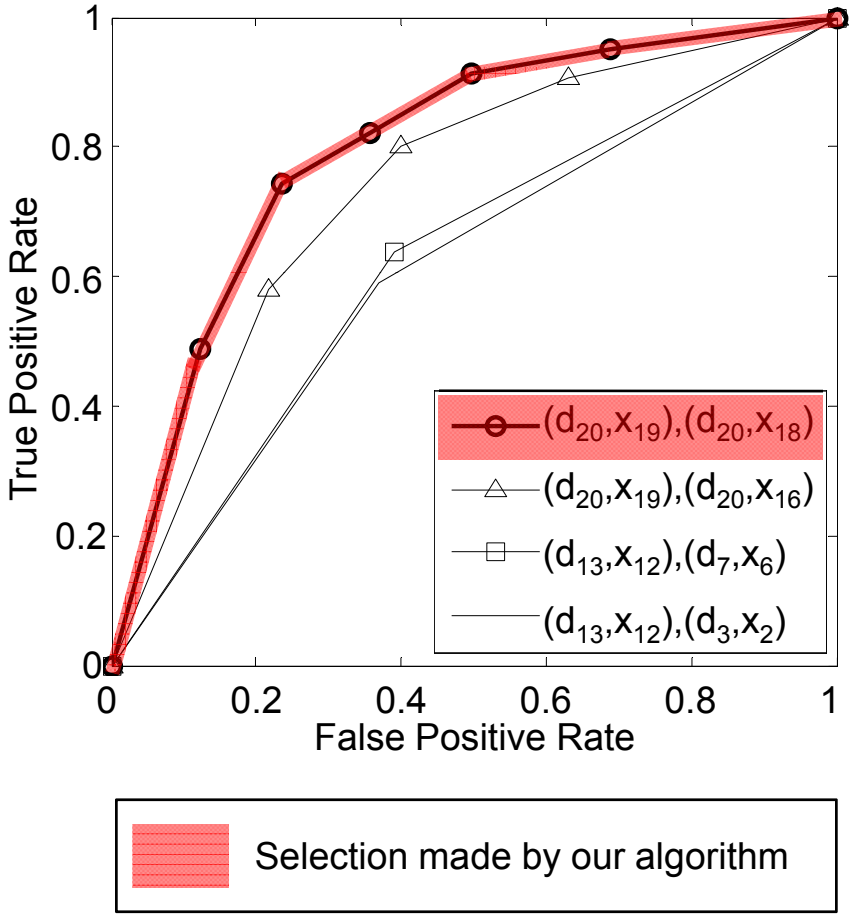
$$System\text{-}Benefit = \sum_{i=1}^{M} [Benefit_{f_i}(TN) \bullet Precision(f_i,d,a) + Benefit_{f_i}(TP) \bullet Recall(f_i,d,a)]$$

$$Precision = TP / (TP+FP) \qquad Recall = TP / (TP+FN)$$

- Algorithm is not guaranteed to provide optimal solution since the exact problem is NP-hard

# Exp. 3: Impact on Choice and Placement of Detectors

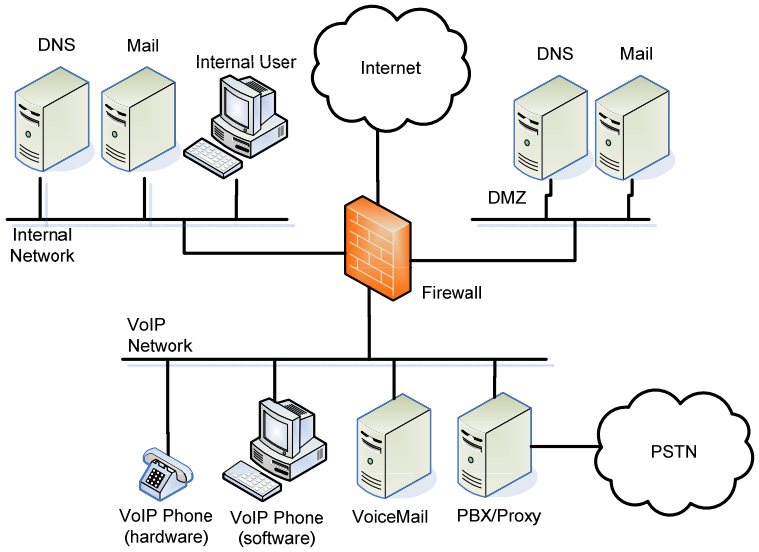
- Objective: determine impact of selecting detectors and corresponding locations
- Performance of detector pair (selected from algorithm) is compared against randomly selected pairs

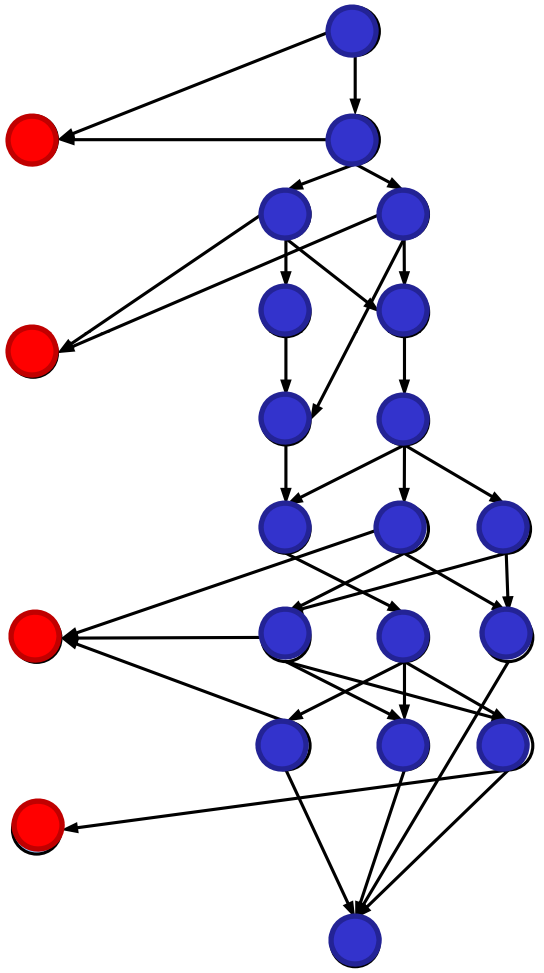


# Conclusions and Future Work

- Provide scientific basis for the choice and location of intrusion detectors
- Use of Bayesian networks allows to model relationships between attack steps and detectors in a distributed system
- Experiments validate that greedy choice of detectors gives good results when considering multiple detectors
- Future work:
  - Is the solution scalable to larger attack graphs and more detectors?
  - Is this the best polynomial time algorithm?
  - How can the algorithm be made incremental?

# Sample Network and Corresponding Bayesian Network



Detector
Attack step