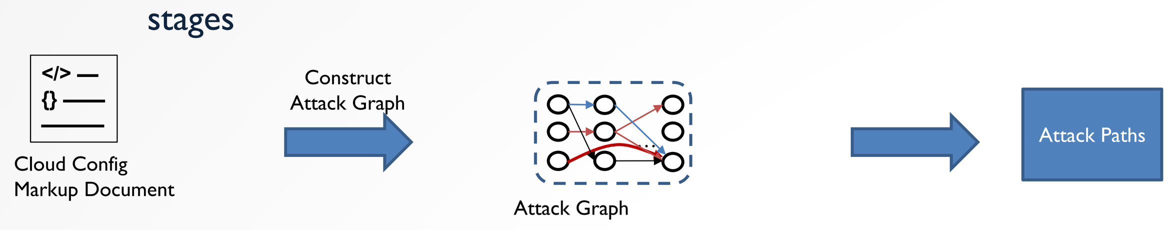
Learning Factor Graphs for Preempting Multi-Stage Attacks in Cloud Infrastructure

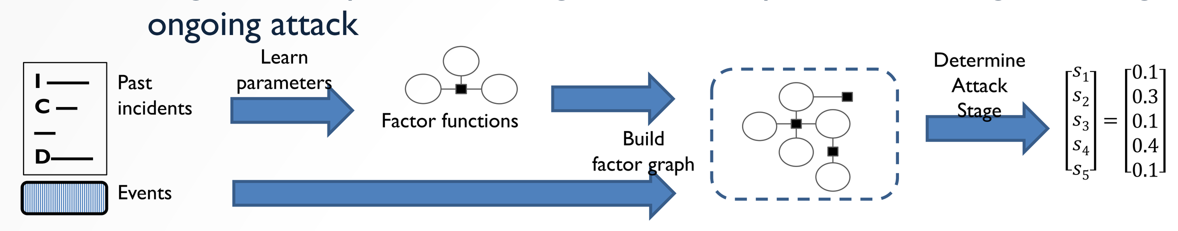
Phuong Cao, Alexander Withers, Zbigniew Kalbarczyk, Ravishankar Iyer
University of Illinois at Urbana-Champaign

Goals

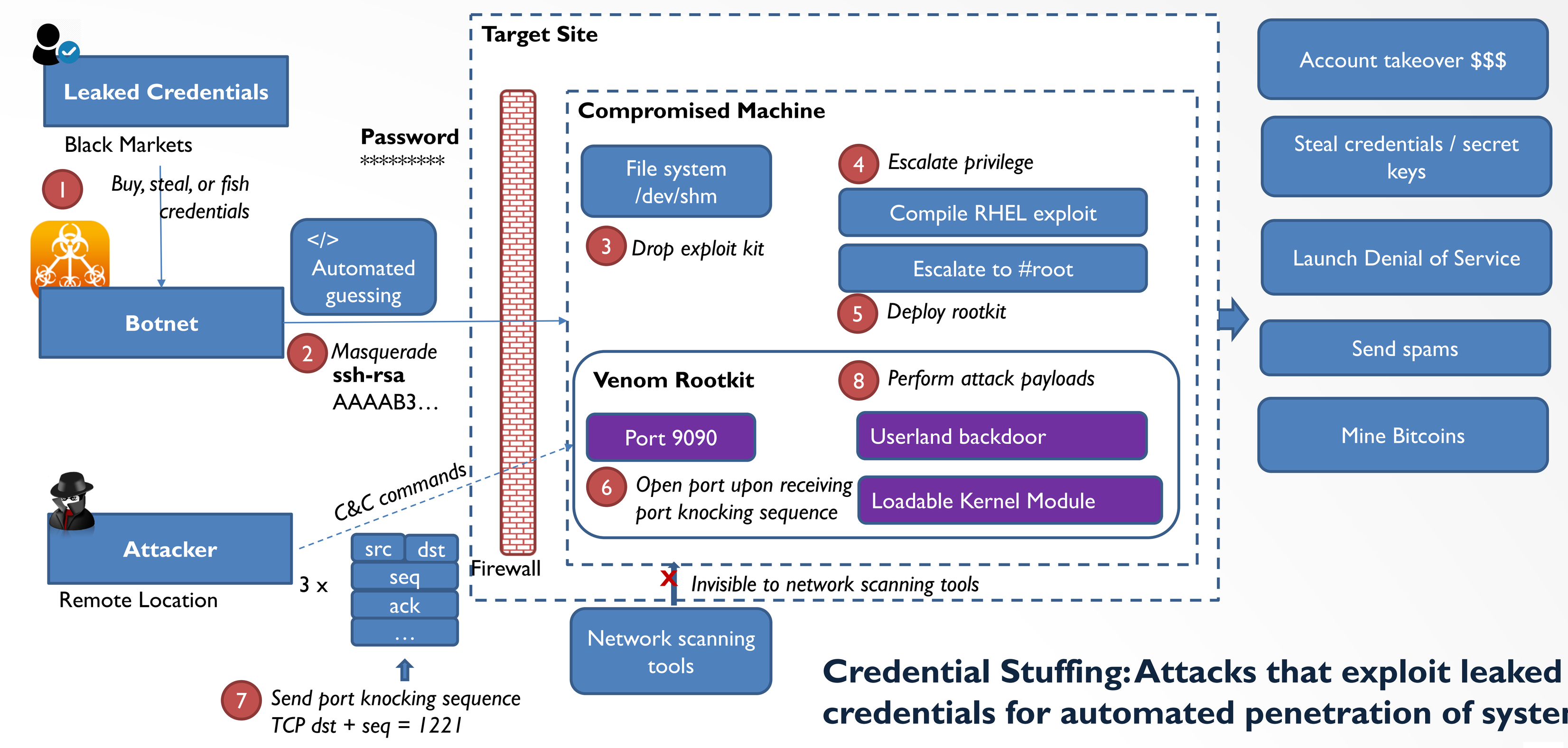
- Detect multi-stage attacks that make use of stolen credentials in large enterprise networks, e.g., cloud infrastructure
- Employ factor graphs, a probabilistic graphical model, to capture attacker behavior and detect malicious activities.
  - Learning graph structure that represents dependencies among observed events and attack stages



  - Learning graph parameters, i.e., factor functions among observed events and hidden attack stages, that represents strength of their dependencies using a factor graph correspond to ongoing attack



Motivating Example: A Credential Stuffing Attack



Credential Stuffing: Attacks that exploit leaked credentials for automated penetration of systems.

Approach

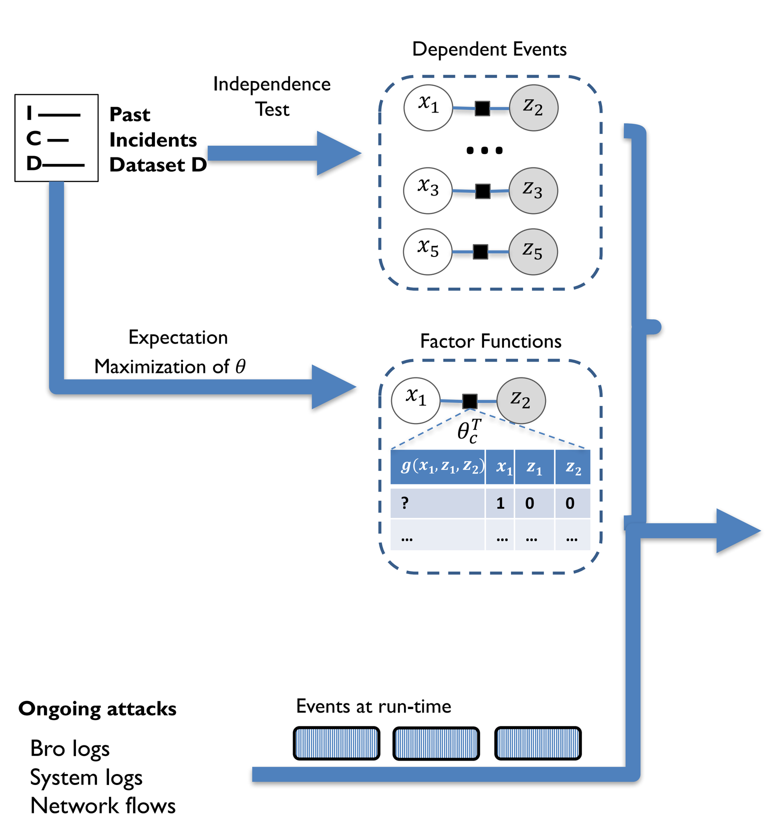
Learning graph structure (offline and runtime)

The goal of learning graph structure, i.e., factor graph, is to automatically establish dependencies among observed events and hidden attack stages by using an X^2 independence test on training data D. The dependencies are used to construct a set of model candidates $m_i \in M$, e.g., simple model using only strongest dependencies or complex model using all dependencies.

$$score_{\{MAP\}}(m_i|D) = max_\theta \log P(x,z,m_i,D,\theta) + log(\theta P(\theta|m_i)) - dim(m_i)ln|D|$$

A model candidate m_i is scored based on three terms in respective order:

- Goodness of fit with training data ($max_\theta \log P(x,z,m_i,D,\theta)$)
- Entropy of θ to avoid overfitting and favor model stability ($log(\theta P(\theta|m_i))$)
- Complexity of model and availability of training data ($dim(m_i)ln|D|$)



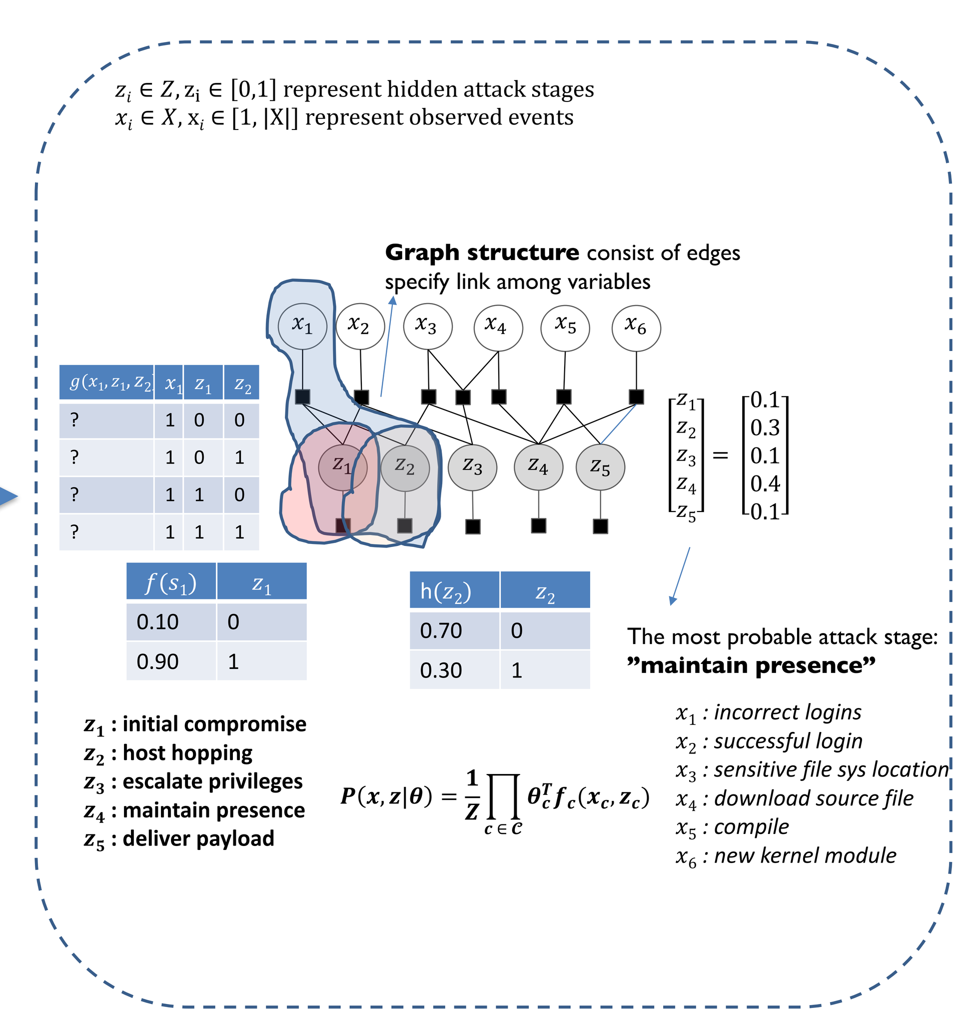


Illustration of the proposed approach in the context of a credential stuffing attack.

Learning graph parameters (offline)

The goal of learning graph parameters is to automatically define parameters θ_c^T of factor functions in tabular forms.

Expectation Maximization algorithm is used for learning parameters of each factor function because it can handle missing or incomplete training data, which is the case for most multi-stage attacks.

Input. Training dataset D of past attacks
Init. Start with a random initialization of θ^0
Repeat each iteration until converge:

E-step. Calculate expected likelihood of log likelihood function

$$Q(\theta^{t+1}|\theta^t) = E_{\{z|x,\theta^t\}}[\log(P(x,z,\theta^t)]$$

M-step. Maximize parameters of $\theta^{\{t+1\}}$

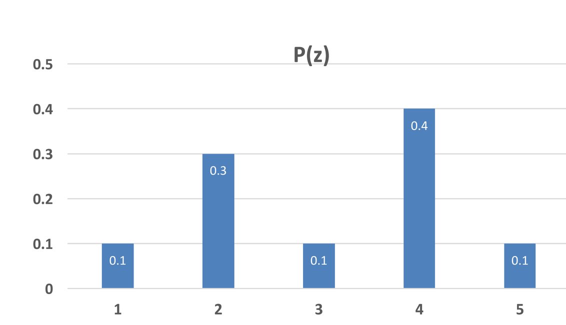
$$\theta^{\{t+1\}} = argmax_{\{\theta\}}\, Q(\theta^{t+1}|\theta^t)$$

Inference of ongoing attack stages (runtime)

Given a factor graph of an ongoing attack at runtime, inference is to determine the most likely unknown attack stage and output a confidence level for each stage.

$$z^* = argmax_{\{z\}}\, P(x,z,\theta)$$

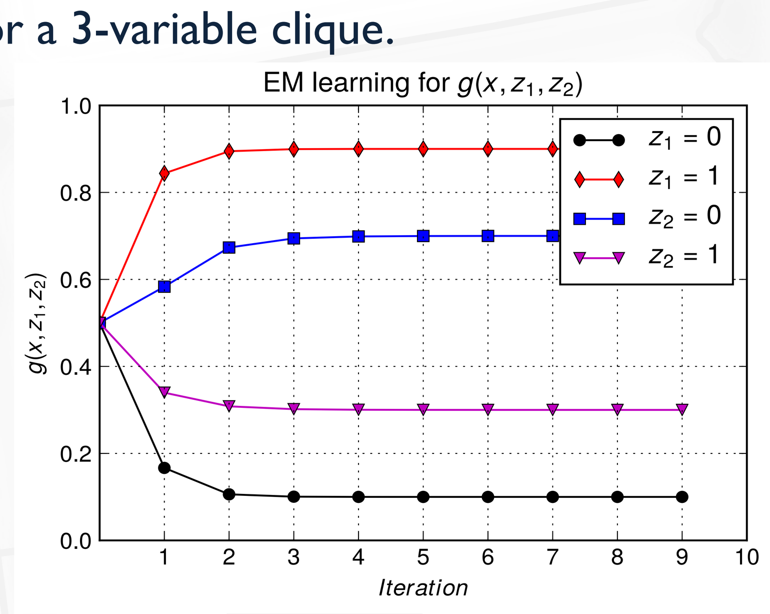
At this stage, off-the-shelf inference techniques such as Belief Propagation, Monte Carlo Markov Chain, or Variational Inference can be employed.



Determine response to the identified attack stage, e.g., z_i = **maintain presence** is the most probable attack stage in this example.

Result on learning parameters

Expectation-Minimization algorithm shows a fast convergence rate, i.e., 3 iterations, of marginal probability of z_i for a 3-variable clique.



Future Work

1. Automatically build graphs for evaluating security of pre-deployed cloud applications
2. Evaluate of learned graphs in terms of detection accuracy or model complexity
3. Build models to automatically respond to on-going attacks

References

[1] Koller, Daphne, and Nir Friedman. Probabilistic graphical models: principles and techniques. MIT press, 2009.

[2] Cao, Phuong, et al. "Preemptive intrusion detection." Proceedings of the 2014 Symposium and Bootcamp on the Science of Security. ACM, 2014.

Acknowledgement

This material is based upon work supported by the Maryland Procurement Office under Contract No. H98230-14-C-0141

National Center for Supercomputing Applications staffs for insightful discussions on incident reports of multi-stage attacks

***HoTSoS** Symposium and Bootcamp*
HOT TOPICS *in the* **SCIENCE OF SECURITY**
APRIL 4-5, 2017 | HANOVER, MARYLAND