GAME THEORY AND SECURITY
PART 5: GAME THEORY FOR CYBER SECURITY

Yevgeniy Vorobeychik
Assistant Professor, CS & Biomedical Informatics
Vanderbilt University

OUTLINE

- Games with attack graphs and plans
- Games on networks

GAMES WITH ATTACK GRAPHS AND PLANS

THREAT/ADVERSARY MODELING

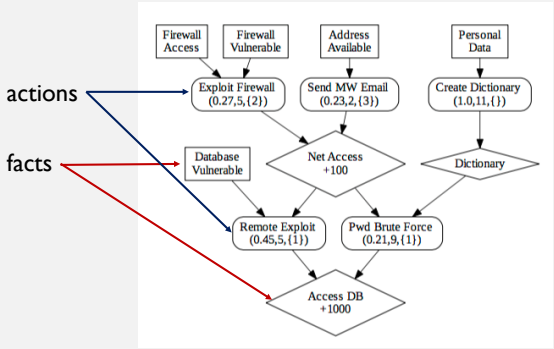
- Basic picture in threat modeling and mitigation:
  - *Attacker*:
    - has a set of capabilities
    - can choose from a collection of "actions" (activities, exploits, attacks)
      - actions can have costs (time, likelihood of detection/capture, etc)
    - has a collection of goals
      - goals may have different importance to the attacker/defender
      - may be content to achieve a subset of goals
    - constructs a plan (a sequence of actions) to achieve goals starting with capabilities
  - *Defender*:
    - system of interest has initial state (vulnerabilities, NW configuration, etc)
    - *mitigations*: can block attack actions, initial capabilities, patch vulnerabilities, deploy honeypots/IDS
      - mitigations can be costly

4

ATTACK GRAPHS

- Attack graphs are a common way to model threats in cyber security
- There are several notions of attack graphs which are used
  - State attack graphs: nodes are states; edges are possible state transitions; attacker chooses a path from initial state to goal state
  - Action attack graphs: nodes are actions; edges correspond to actions satisfying preconditions of other actions
  - Dependency attack graphs, or attack trees (AND nodes, corresponding to actions, and OR nodes, corresponding to state variables or "facts")

EXAMPLE OF DEPENDENCY ATTACK GRAPH



GAMES ON ATTACK GRAPHS

- Example: Durkota et al., IJCAI 2015; using dependency attack graphs
- Defender: hardens the network; for example, adding honeypots
  - Formally: can add x_t honeypots of type (configuration) t
  - Effect: increases likelihood that an attack on type t hits a honeypot (and attacker is thereby caught)
- Attacker: chooses a policy of interaction with hosts of a chosen type
  - A policy is a full contingent plan (actions can fail or succeed)
  - Attacker chooses an optimal policy after observing the defender's strategy (Stackelberg game)
  - Assume attack graph is monotonic (once a fact becomes true, it cannot be undone)
  - Solve as an MDP

AI + PLANNING

- One of the main branches of AI is (formal/logic-based) planning
  - Model the planning problem using formal logic
  - Numerous heuristic planning tools and some well-accepted planning languages (STRIPS, PDDL)
- (*Deterministic*/*Classical*) Planning problem:
  - The world = a set of logical variables = *state* (of the world)
  - Start: initial state of the world (variables that are **true** at time 0)
  - Goal(s): variables (or Boolean expressions) that the planner wants to satisfy
  - Actions: actions/steps a planner can take towards the goal
    - preconditions: variables that must be true for the action to apply
    - effects: variables that become true/false as a result of the action
  - A plan: a partial order of actions that achieve goals starting from the initial state

8

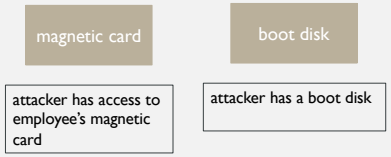
ATTACKER AS A PLANNER

- Initial state = attacker capabilities (expressed as logical variables), system vulnerabilities, initial NW config, etc
- Goals = attacker's goals (expressed as logical variables)
  - Data X exfiltrated
  - Root access to machine Y
  - Different goal variables may vary in importance
- Actions = attack actions

9

EXAMPLE: INITIAL CAPABILITIES

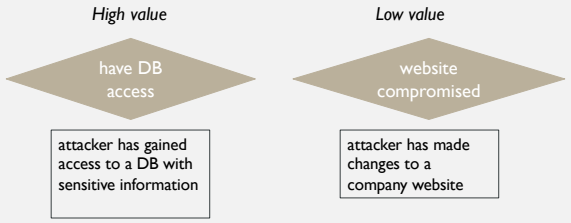
initial attacker capabilities



10

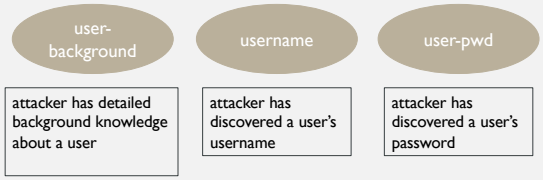
EXAMPLE: ATTACKER GOALS

attacker goals



11

EXAMPLE: STATE VARIABLES



12

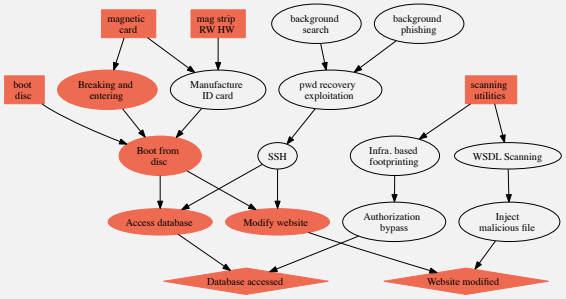
EXAMPLE: ATTACK ACTION

preconditions *actions* *add effects*

Common vulnerabilities & exploits: http://capec.mitre.org

13

("ACTION") ATTACK GRAPH, AND AN OPTIMAL PLAN FOR AN EXAMPLE SCENARIO



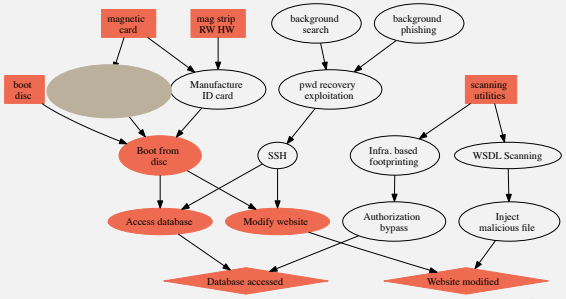
14

PLAN INTERDICTION

- Defender can block some of the attack actions
  - by patching vulnerabilities, introducing IDS, improving physical security, etc

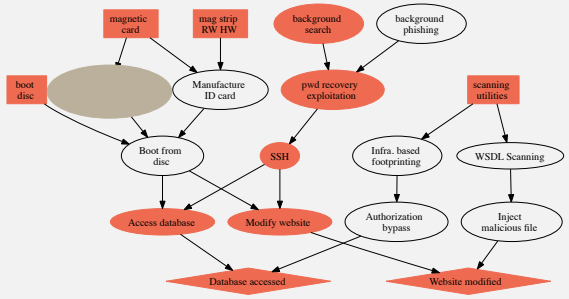
Letchford & V, AAMAS 2013

PLAN INTERDICTION AND CIRCUMVENTION



16

PLAN INTERDICTION AND CIRCUMVENTION



17

DETERMINISTIC PLAN INTERDICTION PROBLEM

DPIP: defender chooses an optimal subset of mitigations, accounting for attacker's best response plans

(optimal: maximizing defender's utility)

(utility = Value of goals – cost of mitigations)

18

DETERMINISTIC PLAN INTERDICTION PROBLEM (DPIP)

- DPIP_DP (deterministic plan interdiction decision problem): can the defender achieve target utility?
- Theorem: DPIP_DP is PSPACE-Complete
- Proof: by reduction from partial satisfaction planning
- Formulation: based on an Integer Program for computing an optimal partial satisfaction plan (Vossen et al., 1999; Briel et al., 2004)
  - Maximize defender utility
  - subject to:
    - a plan is feasible/legal
    - a plan is the best plan for the attacker (*by comparison to all feasible plans*)

19

DPIP FORMULATION

$$\max_{D_a, D_m, y_{a,t}, \delta_p} \sum_{l \in L} V_l^D s_l - \sum_{m \in M} D_m C_m^D$$

s.t. :

$\forall_a \quad D_a \leq \sum_m D_m A_{m,a}$

$\forall_{m,a} \quad D_a \geq D_m A_{m,a}$

compute which actions are interdicted

$\forall_{a,t} \quad y_{a,t} \leq (1 - D_a)$ ← attacker cannot choose interdicted actions

$\forall_{p,a} \quad \delta_p \geq D_a$

$\forall_p \quad \delta_p \leq \sum_{a \in p} D_a$

identify non-interdicted plans

$\forall_p \quad \sum_{l \in L} V_l^A s_l - \sum_{a,t} C_a^A y_{a,t} \geq U^A(p) - Z\delta_p$ ← choose the best non-interdicted attack plan

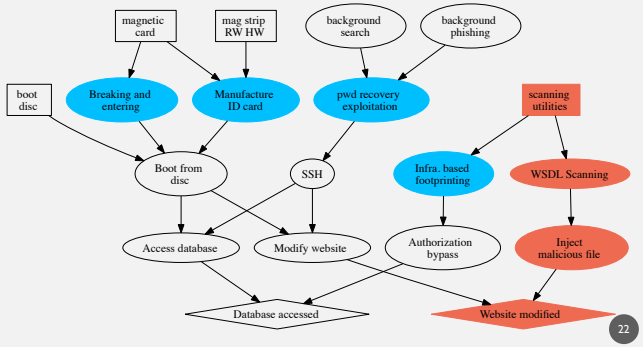
plan feasibility constraints

20

DPIP: CONSTRAINT GENERATION

- Use constraint generation to compute optimal interdiction
  - Can solve an IP to compute an optimal plan
  - Leverage best heuristic planning tools from AI research community
    - SGPLAN5: state-of-the-art heuristic partial satisfaction planner; winner of IPC

OPTIMAL PLAN INTERDICTION EXAMPLE



22

WHAT ABOUT UNCERTAINTY?

- DPIP assumes everything is certain/deterministic
- Usually uncertain about:
  - Attacker capabilities
  - Existence of vulnerabilities (of specific classes)
  - Attacker goals
  - Execution and effects of actions
  - Attacker's observation of the current state of the system
- Questions:
  - Can we extend DPIP to accommodate all/some of this uncertainty?
  - If not, how should we model it?

23

UNCERTAINTY ABOUT CAPABILITIES/GOALS

- Can naturally model within the DPIP framework
  - Attacker has multiple types ("personalities"), each type corresponding to a set of capabilities/goals (*Bayesian plan interdiction problem; BPIP*)
  - Each type computes an optimal plan in response to mitigations
  - Objective now to maximize expected defender utility wrt distribution over attacker types

$$\sum_{\theta} \sum_{l \in L} p_{\theta} V^{D}_{\theta,l} s_{\theta,l} - \sum_{m \in M} D_m C^{D}_{m}$$

24

EXECUTION UNCERTAINTY

- *Special case*: only uncertain about whether an action succeeds or not, success is observable, and action can be repeated (just as in Durkota et al., IJCAI 2015)
  - Can be done using DPIP framework by computing expected number of tries for each action
- *Generalizing*: state is observable, but action may have non-deterministic effects
  - Example: "port scan" action (info you get is uncertain)
  - Attack planning problem is an MDP
  - MPD interdiction (MDPIP)
  - Leverage (dual) LP for computing optimal value in an MDP
  - State space explosion: must now explicitly represent all states in the system

25

FACTORED MDP INTERDICTION

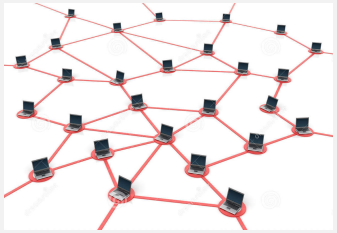
- Can leverage factored state representation in MDPs, and associated techniques
  - Dual LP formulation for factored MDPs used in the bi-level interdiction problem
- Requires value function approximation
  - If x is a state vector,
  - $V(x) = \sum_i \alpha_i \varphi_i(x)$, where $\varphi_i(x)$ are the basis functions
  - We can use Fourier representation of Boolean functions, where the basis functions are parity functions (exact!)
    - A parity function on a subset of variables S, $\chi_S(x) = \prod_{i \in S} x_i$
  - Key challenge: choosing the "right" subsets to get a small number of basis functions

Panda and V, UAI 2017, to appear

GAMES ON NETWORKS

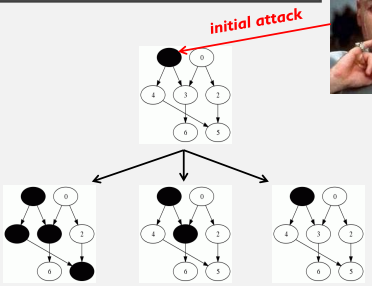
SECURITY OF NETWORKED SYSTEMS

- A central issue in cyber security is that systems that need to be protected are interconnected and interdependent
  - Viruses spread among connected computing devices
  - Computing systems depend on availability and integrity of one another



NETWORK CAN MEDIATE CONSEQUENCES OF ATTACKS

- For example, if a node on a network is successfully attacked (e.g., infected with malware), it can spread it to others, and so on
- This process of spread can be modeled as a stochastic diffusion process
  - Independent cascades: neighbors are affected independently with a predefined probability
- Attack model: attacker chooses a subset of nodes on a network to attack (starting a diffusion process)



V & Letchford, JAAMAS, 2015

DEFENDING AGAINST ATTACKS

- Defender chooses which nodes to protect
- Protection can prevent, or reduce consequences of, direct attacks
  - More generally, chooses security configuration for all nodes, with attack consequences a function of this
- Restriction #1: defense only impacts direct attacks (a secured node can still be a part of a cascade if not directly attacked); only 1 target attacked
  - Example: organization nodes are secured only against external intrusion

RESTRICTION #1: OPTIMAL DEFENSE VIA. SIMULATIONS + LP

- *Cascade probabilities independent of decisions*
  - *Simulate cascades from each node (possible attack target)*
  - *Use resulting utilities to construct a compact game matrix (utility depends only on whether a target is attacked, and how it is defended)*
  - *Solve LPs for optimal security configuration*

$$\max_{q} \quad \sum_{o} U_{o,\hat{t}} q_{o,\hat{t}} - \sum_{t} \sum_{o} c_{o,t} q_{o,t}$$

Subject to:

$$q_{o,t} \geq 0$$

$$\sum_{o} q_{o,t} = 1$$

$$\sum_{o} V_{o,\hat{t}} q_{o,\hat{t}} \geq \sum_{o} V_{o,t} q_{o,t}$$

for each node:

31

MORE GENERALLY

- Use stochastic local search (e.g., coordinate greedy, where one target configuration is optimized at a time)
- Attacker's decision: submodular (influence maximization)

MULTI-DEFENDER GENERALIZATION

- A network is partitioned among N defenders
- Each defender chooses security configuration only for nodes allocated to them, and their utility is a function only of which nodes they own are affected by attacks
  - Interdependent security games, but with complex decisions for each player
- Attacker: still chooses a subset of nodes to attack on the entire network

Smith, Lou, & V, IEEE Intelligent Systems, 2017

STRONG STACKELBERG EQUILIBRIUM?

- SSE: break ties in defender's favor
  - Which defender? → Undefined!
  - In fact, can lead to under-investment in security
  - Suppose two defenders, two identical targets (one for each defender), and neither defender protects their target
    - According to defender 1, attacker will attack target 2 (breaks ties in his favor)
    - According to defender 2, attacker will attack target 1 (same logic)
    - Neither protects their target – but the attacker will attack one of these!
- Alternative: Average-Case Equilibrium (ACE); attacker breaks ties uniformly at random – but it doesn't always exist (although approximate versions typically do)

MULTI-DEFENDER GAMES

- Even with independent nodes: price of anarchy can be unbounded!
- Weakly dependent nodes: defender usually over-invests in security!
  - Arms race: I want to protect my assets slightly more than the other guys
- With sufficiently interdependent nodes, defender under-invests

STEALTHY DIFFUSION

- Often, attackers would not wish to maximize the spread of malware
  - They may have specific targets in mind
  - Networks are monitored, and malware may be discovered (and vulnerabilities patched) before reaching the target
- Stealthy diffusion model:
  - Defender (leader): chooses which nodes will be monitored
  - Attacker (follower): chooses a subset of starting nodes (from a set of feasible nodes that can be attacked, e.g., reachable externally)
    - Notice that the attacker's objective is not monotone: more nodes attacked may increase the chance of being caught!
    - If attacker is just a stochastic process (not deliberate), the defender's optimization problem is submodular
    - Attacker's problem is NP-Hard, and not submodular

Haghtalab et al., KAIS, 2017

GAME THEORY AND SECURITY

PART 6: GAME THEORY AND PRIVACY

OUTLINE

- Structured data sharing
- Sharing genomic data

SHARING STRUCTURED DATA

SHARING STRUCTURED DATA

Last Name	First Name	SSN	Race	Sex	Zip	Age	ICD-9 code
Doe	John	xxx-xx-xxxx	Caucasian	M	91902	15	520.1
Smith	Jane	xxx-xx-xxxx	African American	F	12033	85	466.11
Schwarz	Diane	xxx-xx-xxxx	Caucasian	F	33301	48	512.2
Rogers	Jared	xxx-xx-xxxx	Caucasian	M	85749	67	792.2

QUASI-IDENTIFIERS

Last Name	First Name	SSN	Race	Sex	Zip	Age	ICD-9 code
Doe	John	xxx-xx-xxxx	Caucasian	M	91902	15	520.1
Smith	Jane	xxx-xx-xxxx	African American	F	12033	85	466.11
Schwarz	Diane	xxx-xx-xxxx	Caucasian	F	33301	48	512.2
Rogers	Jared	xxx-xx-xxxx	Caucasian	M	85749	67	792.2

Quasi-identifiers: fields which, combined with other, *commonly available data*, can enable re-identification (figuring out who a particular anonymized record belongs to)

Example: voter registration data often includes name, race, sex, zip, DoB
can try to match voter registration records based on race, sex, zip, DoB with patient data; if only one such patient, we have their diagnostic code

QUASI-IDENTIFIERS

Last Name	First Name	SSN	Race	Sex	Zip	Age	ICD-9 code
[redacted]	[redacted]	[redacted]	Caucasian	M	91902	15	520.1
[redacted]	[redacted]	[redacted]	African American	F	12033	85	466.11
[redacted]	[redacted]	[redacted]	**Caucasian**	**F**	**33301**	**48**	**512.2**
[redacted]	[redacted]	[redacted]	Caucasian	M	85749	67	792.2

Example: suppose voter registration data has exactly one Diane Schwarz, caucasian, F, with zip 33301, and age 48

If there is only one person in shared data who is caucasian, F, zip 33301, age 48, we are reasonably certain it's Diane Schwarz, and we just found out her disease

GENERALIZATION OF ATTRIBUTES (COLUMNS)

Last Name	First Name	SSN	Race	Sex	Zip	Age	ICD-9 code
Doe	John	xxx-xx-xxxx	Caucasian	M	91902	15	520.1
Smith	Jane	xxx-xx-xxxx	African American	F	12033	85	466.11
Schwarz	Diane	xxx-xx-xxxx	Caucasian	F	33301	48	512.2
Rogers	Jared	xxx-xx-xxxx	Caucasian	M	85749	67	792.2

Generalization: instead of the specific value of attribute, specify a set of values it belongs to; this decreases chances someone can be uniquely matched

GENERALIZATION OF ATTRIBUTES (COLUMNS)

Last Name	First Name	SSN	Race	Sex	Zip	Age	ICD-9 code
Doe	John	xxx-xx-xxxx	Caucasian	M	91902	15	520.1
Smith	Jane	xxx-xx-xxxx	African American	F	12033	85	466.11
Schwarz	**Diane**	**xxx-xx-xxxx**	**Caucasian**	**F**	**33301**	**48**	**512.2**
Rogers	Jared	xxx-xx-xxxx	Caucasian	M	85749	67	792.2

Generalization: instead of the specific value of attribute, specify a set of values it belongs to; this decreases chances someone can be uniquely matched

GENERALIZATION OF ATTRIBUTES (COLUMNS)

Last Name	First Name	SSN	Race	Sex	Zip	Age	ICD-9 code
Doe	John	xxx-xx-xxxx	Caucasian	M	91902	15	520.1
Smith	Jane	xxx-xx-xxxx	African American	F	12033	85	466.11
Schwarz	**Diane**	**xxx-xx-xxxx**	**Caucasian**	**F**	**333***	**48**	**512.2**
Rogers	Jared	xxx-xx-xxxx	Caucasian	M	85749	67	792.2

Generalization: instead of the specific value of attribute, specify a set of values it belongs to; this decreases chances someone can be uniquely matched

GENERALIZATION HIERARCHY

Last Name	First Name	SSN	Race	Sex	Zip	Age	ICD-9 code
Doe	John	xxx-xx-xxxx	Caucasian	M	91902	15	520.1
Smith	Jane	xxx-xx-xxxx	African American	F	12033	85	466.11
Schwarz	**Diane**	**xxx-xx-xxxx**	**Caucasian**	**F**	**333***	**48**	**512.2**
Rogers	Jared	xxx-xx-xxxx	Caucasian	M	85749	67	792.2

Generalization Hierarchy: a hierarchy of increasingly specific sets of attribute values

*Zip: * -> 3**** -> 33***-> 333** -> 3330* -> 33301*

** : effectively we remove the column/attribute (generalized to all possible values)*

HIPAA

- Option 1 [Safe Harbor]: remove all direct identifiers (name, SSN, address, etc), "generalize" quasi-identifiers in a specific way (e.g., zip -> first three digits)
- Option 2 [Expert]: expert certification that data has "low risk" of being re-identified

HIPAA

- Option 1 [Safe Harbor]: remove all direct identifiers (name, SSN, address, etc), "generalize" quasi-identifiers in a specific way (e.g., zip -> first three digits)
- Option 2 [Expert]: expert certification that data has "low risk" of being re-identified by an ***anticipated recipient***

QUANTIFYING RISK

- This is not a reliability issue: there are attackers, who will respond to the specific data sharing decisions made by the data sharer
- Risk stems from the likelihood of identification **given that the attacker tries to do it**
- Model: attacker will attempt re-identification of a record iff they gain from this (ex ante)
  - Attacker is a data recipient, who is economically motivated to re-identify the data
  - This likely significantly over-estimates risk, since most data recipients would not attempt re-identification even if they can make money from it

Wan et al., PLoS One, 2015

DATA SHARER MODEL

- Share each record with each attribute *j* generalized to level *l* in the hierarchy
  - Call this *g*, g_{rjl} = *1* if *j* generalized to level *l* for record *r*
- Value (positive utility): higher value to data sharer if attributes are shared at greater specificity (less generalized); additive over records
- Risk (negative utility): a loss *L* for every record which is successfully re-identified

ATTACKER MODEL

- Choose whether to attack each record in isolation
  - Thus, we can just consider a separate game for each record
- Suppose the defender uses generalization strategy *g* (for record *r*)
- Binary decision: attack or do not attack ($a \in \{0,1\}$)
- *P(success|g,attack)*: probability an attack succeeds if data is shared according to *g*
  - *Attack success probability = 1 / {# equivalent records matched in external data}*
- If attack succeeds, attacker gains *L*
- Attack is costly: pay a cost *c* for each attack
- Attacker's utility: $U_A(g, attack) = L\,P(success|g, attack) - c$
- So: attack iff $P(success|g, attack) \geq c/L$

DATA SHARER

- Utility: $U_D(g, a) = v(g) - L\,P(S|g, a)$; $a = 1$ iff there is an attack
- So, the defender's optimization problem is:

$$\max_g v(g) - L\,P(S|g, a(g))$$

a(g): attacker's best response

The generalization hierarchy of a record is a lattice (a partial order from least to most specific)

Solution approach: lattice-based search

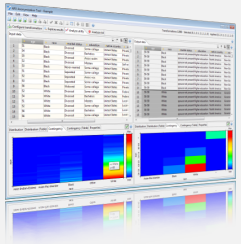
start at most specific and follow a sequence of local improvement steps

RECORD-LEVEL VS. DATABASE-LEVEL GENERALIZATION

- The approach described was record-level generalization
  - We treated the problem of sharing each record independently (no budget constraint on adversary; external dataset used to find equivalence groups)
- Often, we prefer database-level generalization:
  - The same generalization level for all records (simplify database schema / data analysis)
  - Can also use lattice-based search for this

OPEN-SOURCE SOFTWARE

- This approach is implemented in ARX, open-source anonymization (de-identification) software
- http://arx.deidentifier.org
- Implemented both record-level and data-level generalization
  - Using branch-and-bound search



Prasser et al., AMIA, 2017, to appear

SHARING GENOMIC DATA

METHODS FOR SHARING

- Summary statistics: for example, SNP minor-allele frequencies (MAFs)
  - MAFs would be shared for a (sub)set of SNPs over a pool of subjects in a given study (along with p-values for associations of SNPs with a phenotype)
  - dbGaP (database of genotypes and phenotypes): https://www.ncbi.nlm.nih.gov/gap
  - SPHINX database
- Existence queries: can ask about presence/absence of a specific allele in the dataset
  - Query model: one can send an arbitrary sequence of such queries
  - Beacon network: https://beacon-network.org/

SHARING SNP SUMMARY STATISTICS

- What could possibly go wrong?
  - After all, these are summary statistics, not actual genomes or SNPs!
- A series of attacks showed that in fact one can determine whether a particular individual is in the pool for which summary statistics are shared
  - Start with the genome of this individual (e.g., employee, co-worker, sibling, spouse, child)
    - House proposal would let employers demand workers' genetic test results (currently illegal under GINA): https://www.statnews.com/2017/03/10/workplace-wellness-genetic-testing/
  - Perform a statistical test (individual in pool vs. individual in reference dataset/population; etc)
  - If statistic is above some threshold, claim the individual is in the pool
  - Homer et al. PLoS Genetics, 2008; Sankararaman et al., Nature Genetics, 2009; Gymrek et al., Science 2013; etc.
- Concern: identifying that a specific individual is in the pool can reveal sensitive information (e.g., pool = a study of HIV drug effectiveness)

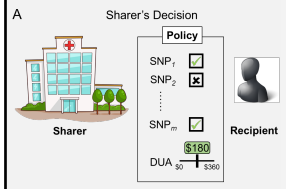
THE DATA SHARING GAME

- Consider the data sharing process as a game between the publisher (who shares SNP summary statistics) and a recipient (who may choose to re-identify individuals if that's beneficial to them)

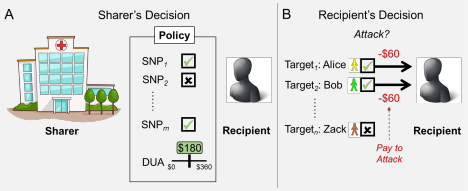
Wan et al., AJHG, 2017

THE DATA SHARING GAME

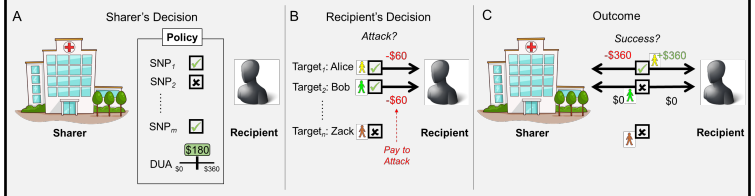
THE DATA SHARING GAME



THE DATA SHARING GAME



THE DATA SHARING GAME



MEASURING ATTACKER'S SUCCESS: *THE LIKELIHOOD RATIO TEST*

- Perhaps the most applicable statistical test is likelihood ratio test
  - Likelihood that target in pool / likelihood target not in pool (pool corresponds to a clinical study) [Sankararaman et al., Nature Genetics, 2009]
  - Attack when threshold on the test is exceeded
- More precisely, let's take a target i with genotype $a_i = (a_{i1},\ldots,a_{im})$, where $a_{ij} = 1$ if i has the minor allele in SNP j (simplifying things a bit; in reality, we need to consider diploid genes, but this conveys the idea)
- Model:
  - a finite pool of SNPs is drawn from a larger population
  - pool statistics are shared: for each SNP, $\hat{p}_j$ represents MAF in the pool; p_j is MAF in the population (determined from a large "representative" sample)

LIKELIHOOD RATIO TEST

- The likelihood ratio test for a target i is then

$$L_i = \prod_j \left(\frac{\hat{p}_j}{p_j}\right)^{a_{ij}} \left(\frac{1-\hat{p}_j}{1-p_j}\right)^{1-a_{ij}}$$

- *What is missing: how does this connect to attacker's probability of success (which will determine their expected utility)?*

BAYESIAN MODEL

- Wish to determine the probability that target i is in a dataset (pool) D, given their genomic information a_i

$$\Pr\{i \in D|a_i\} = \frac{\Pr\{a_i|i \in D\}\Pr\{i \in D\}}{\Pr\{a_i\}}$$

- Now, notice that $\Pr\{a_i|i \in D\} = \prod_j \hat{p}_j^{a_{ij}}(1-\hat{p}_j^{a_{ij}})$, while if we assume that a is sampled iid from the population with statistics p, $\Pr\{a_i\} = \prod_j p_j^{a_{ij}}(1-p_j^{a_{ij}})$
- Consequently, $\Pr\{i \in D|a_i\} = qL_i$, where q is the prior probability that i is in D.

ATTACKER'S UTILITY

- Let g be the gain from a successful attack and c the cost of the attack The expected gain from attacking a target i is:
  - $g\Pr\{i \in D|a_i\} - c = gqL_i - c$
  - Thus, i is attacked iff $L_i \geq \frac{c}{gq}$

PUBLISHER'S UTILITY

- Publisher's utility:
  - value of shared data (as a function of the subset of SNP statistics actually shared) – expected loss from successful attacks on individuals in the pool
- Important issue: we don't know attacker's target set!
  - We can assign each individual in the pool a probability that they are a part of the attacker's target set
  - Measure of the attacker's strength; congruent with a natural model where the target set of size k is sampled randomly from the population
- Goal: to choose a subset of SNPs to share so as to balance privacy and security
- Solution approach: genetic algorithm to choose a subset of SNPs to share (for each, calculate utility based on the attacker's decision which targets to attack)

SUMMARY

- Game theory has had a number of successful applications in cyber security and privacy
  - Attack graph and plan games
  - Security games on networks
  - Structured data release
  - Genomic data release
- Typically, single-defender and single-attacker games, but we have explored modeling possibilities where there are multiple defenders

MOVING FORWARD

- Many open questions remain in research and practice
  - What about multiple attackers?
  - What is the right solution concept for multi-defender games?
  - What about bounded-rational defenders and attackers?
  - What are the right models for privacy-preserving data sharing?
  - How can these models and solution approaches make an impact in practice?