# Audit Games

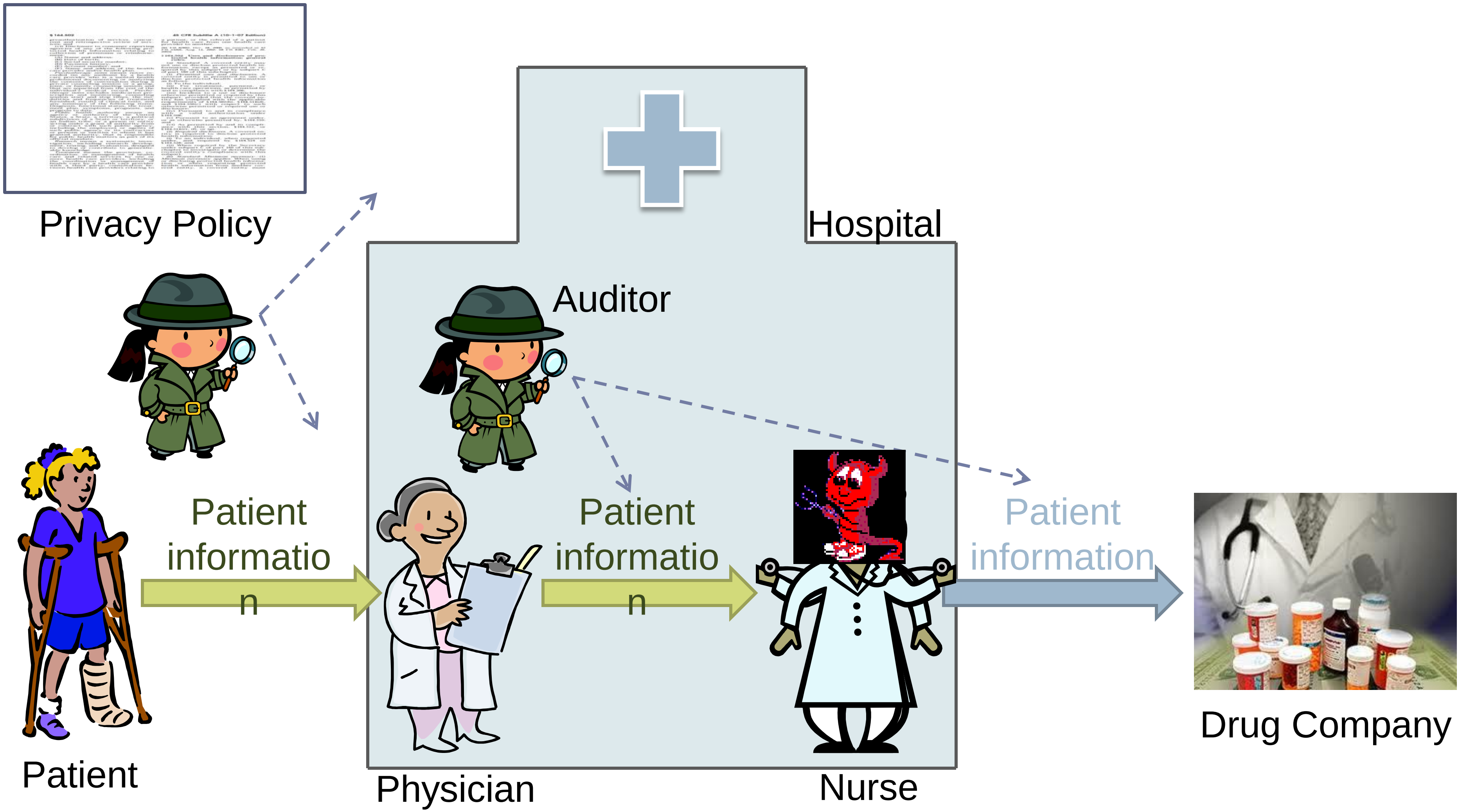
Anupam Datta

Carnegie Mellon University

February 2013

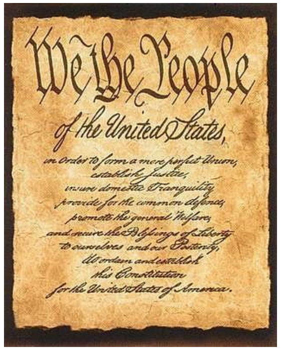
# Repositories of Personal Information


# Healthcare Privacy



# A Research Area

- Formalize Privacy Policies
  - Precise definitions of privacy concepts (restrictions on information flow)
    - Information used *only for* a *purpose*
    - All disclosure clauses in HIPAA & GLBA




- Enforce Privacy Policies
  - Audit and Accountability
    - Detect violations of policy
    - Identify agents to blame for policy violations
    - Resource allocation for inspections and punishments (economic considerations)

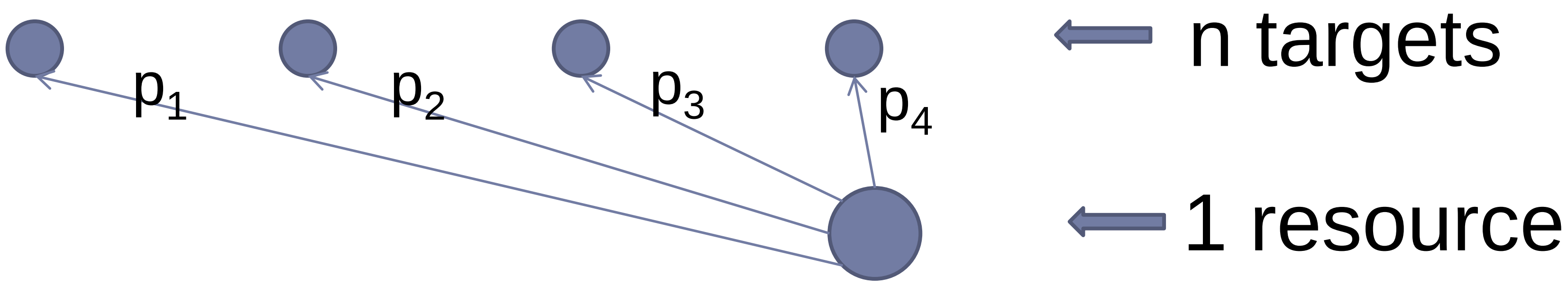

Project page: Privacy, Audit and Accountability

# Play in Three Acts

1. Rational Adversary Setting
2. Byzantine Adversary Setting
3. Research Directions

[Blocki, Christin, Datta, Procaccia, Sinha; 2013]

# Audit Game Model [BCDPS'13]




- If a violation is found, adversary is fined
- Utility when target $t_i$ is attacked
  - Defender: $p_i\, U_{a,D}(t_i) + (1 - p_i)U_{u,D}(t_i) - ax$
  - Adversary: $p_i\, (\, U_{a,A}(t_i) – x\, ) + (1 - p_i)U_{u,A}(t_i)$

Price of punishment [Becker'68]

# Stackelberg Equilibrium Concept

- Defender commits to a randomized resource allocation strategy
- Adversary plays best response to that strategy

- Appropriate equilibrium concept
  - Known defender strategy avoids security by obscurity
  - Predictable adversary response

- Goal
  - Compute optimal defender strategy

# Related Work

- Security resource allocation games [Tambe et al. 2007-]
  - Computes Stackelberg equilibrium
  - Deployed systems for resource allocation for patrols at LAX airport, federal air marshals service; under evaluation by TSA, US coast guard
- Audit games generalize security resource allocation games with the punishment parameter
  - Computing Stackelberg equilibrium becomes more challenging
  - Applicable to similar problems

# Computing Optimal Defender Strategy

Solve optimization problems $P_i$ for all $i \in \{1,..,n\}$ and pick the best solution

$$\max\ p_i\, U_{a,D}(t_i) + (1 - p_i)U_{u,D}(t_i) - ax$$

subject to

$$p_j(\ U_{a,A}(t_j) - x\ ) + (1 - p_j)U_{u,A}(t_j) \leq p_i\ (\ U_{a,A}(t_i) - x\ ) + (1 - p_i)U_{u,A}(t_i)$$

$\forall j \in \{1,..,n\}$

$p_i$'s lie on the probability simplex

$0 \leq x \leq 1$

Adversary's best response is attacking target $t_i$

# Algorithmic Challenges

1. Quadratic constraints
   - $p_i x$ terms
2. Non-convex optimization problem
   - Constraints representable as $x^T A x + Bx + c \leq 0$
   - A is not positive semi-definite

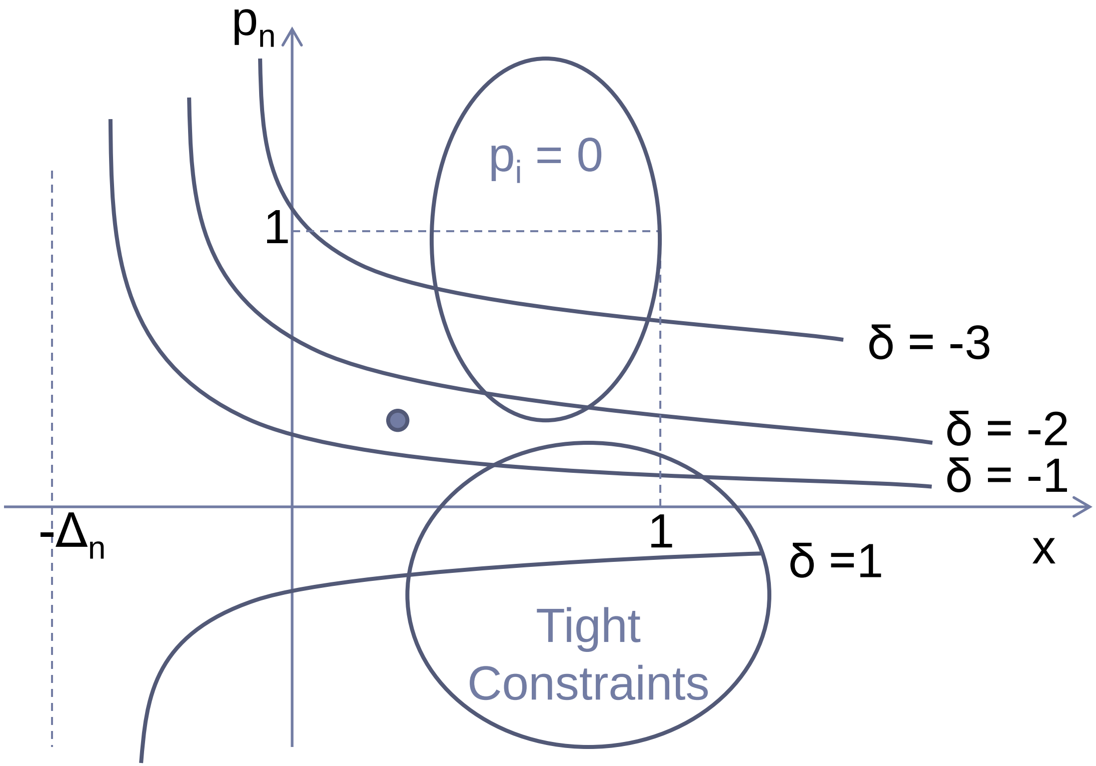
# Properties of Optimal Point

- Rewriting quadratic constraints

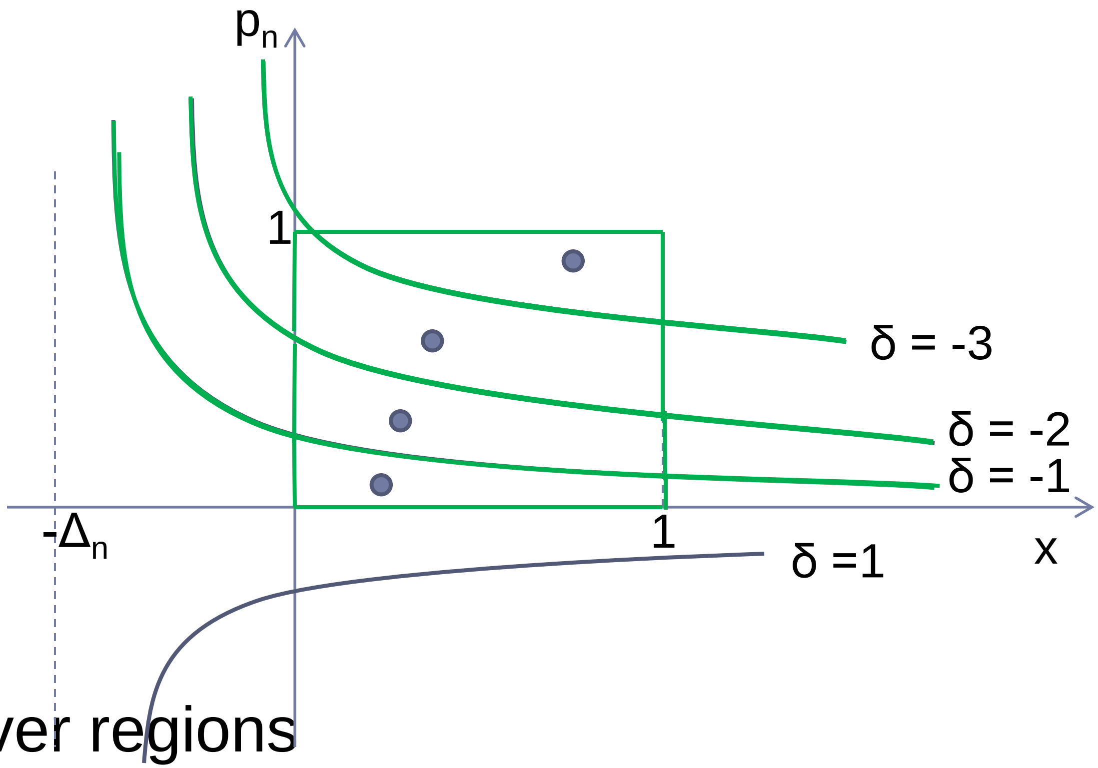
  $p_j(-x - \Delta_j) + p_n(x + \Delta_n) + \delta_{j,n} \leq 0$

  where $\Delta_j \geq 0$

$\Delta_j = U_{u,A}(t_j) - U_{a,A}(t_j)$



# Overview of Algorithm




- Iterate over regions
- Solve sub-problems $EQ_j$
  - Set probabilities to zero for curves that lie above & make other constraints tight
- Pick best solution of all $EQ_j$

# Solving Sub-problem $EQ_j$

1. $p_j(-x - \Delta_j) + p_n(x + \Delta_n) + \delta_{j,n} = 0$
   - Eliminate $p_j$ to get an equation in $p_n$ and x only
2. Express $p_n$ as a function f(x)
   - Objective becomes a polynomial function of x only
3. Compute x where derivative of objective is zero & constraints are satisfied
   - Local maxima
4. Compute x values on the boundary
   - Found by finding intersection of $p_n = f(x)$ with the boundaries
   - Other potential points of maxima
5. Take the maximum over all x values output by Steps 3,4

Steps 3 & 4 require computing roots of polynomials

# Computing Roots of Polynomials

- Using existing algorithms
  - Splitting circle method [Schonage 1982] can approx. irrational roots to precision $K$ in time polynomial in $K$
    - *Steps 3 and 4 take imprecision into account*
  - LLL [Lenstra et al. 1982] can find rational roots exactly

# Main Theorem

- *The problem can be approximated to an additive $\epsilon$ factor in time $O(n^5 K + n^4 \log(1/\epsilon))$ using only the splitting circle method, where K is the bit precision of inputs.*

- Using LLL the time is still polynomial $O(\max\{n^{13}K^3, n^5 K + n^4 \log(1/\epsilon)\})$, and if the solution is rational the exact solution is found.

# Play in Three Acts

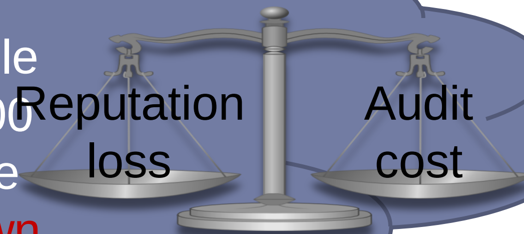
1. Rational Adversary Setting
2. Byzantine Adversary Setting
3. Research Directions

[Blocki, Christin, Datta, Sinha; CSF 2011]

# Audit Model

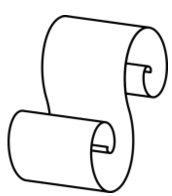
Auditor

Auditing budget: $3000/ cycle
Cost for one inspection: $100
Only 30 inspections per cycle
Employee incentives unknown

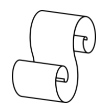



Access divided into 2 types

Reputation Loss from 1 violation (internal, external)

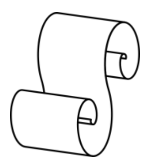


100 accesses



30 accesses


$500, $1000

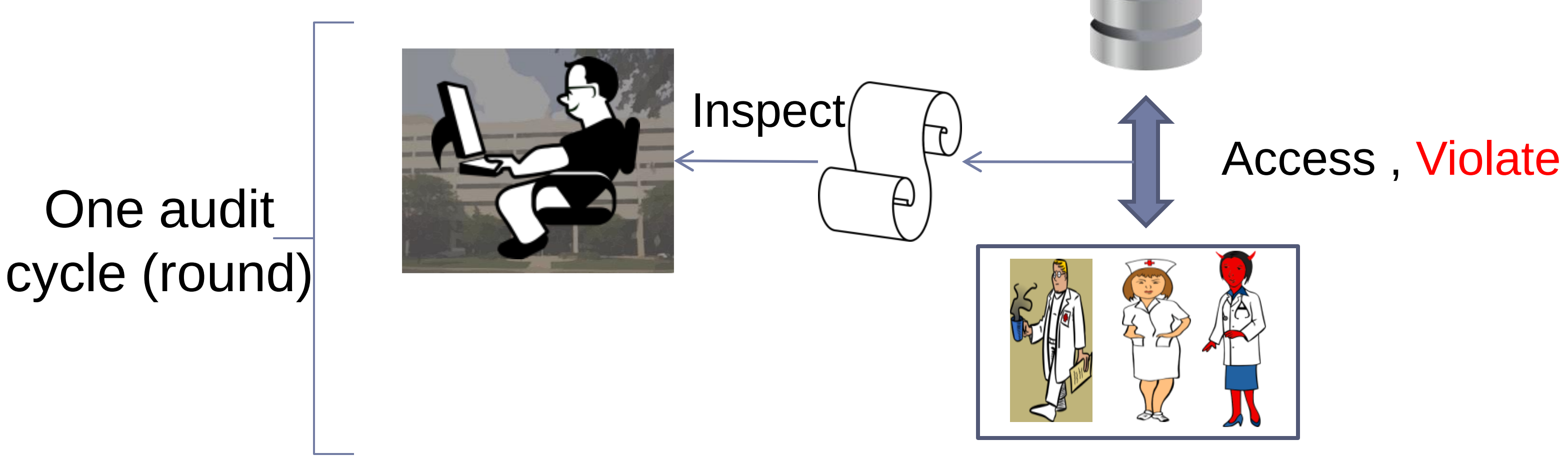


70 accesses

$250, $500

# Repeated Game Model for Audit

- Game model

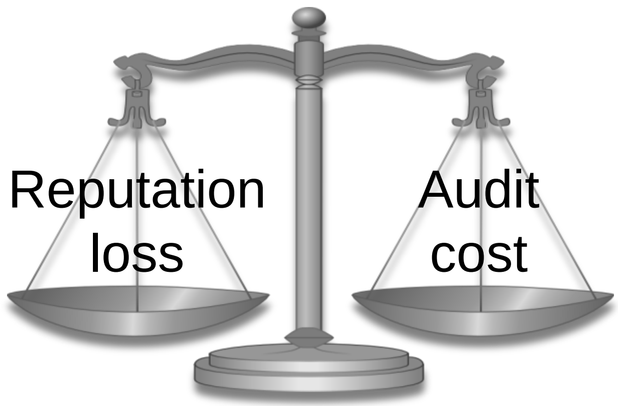



- Typical actions in one round
  - Emp action: (access, violate) = ([30,70], [2,4])
  - Org action: inspection = ([10,20])

Imperfectio
n

# Game Payoffs

- Organization's payoff




  - Audit cost depends on the number of inspections
  - Reputation loss depends on the number of violations caught

- Employee's payoff unknown

# Audit Algorithm Choices

Only 30 inspections

Consider 4 possible allocations of the available 30 inspections

Sandra Bullock

| | | | | |
|---|---|---|---|---|
| Sandra Bullock | 0 | 10 | 20 | 30 |
| | 30 | 20 | 10 | 0 |
| Weights | 1.0 | 1.0 | 1.0 | 1.0 |

Choose allocation probabilistically based on weights

# Audit Algorithm Run

| No. of Access | Actual Violatio |
|---|---|
| 30 | 2 |
| 70 | 4 |


| | | | |
|---|---|---|---|
| 0 | 10 | 20 | 30 |
| 30 | 20 | 10 | 0 |

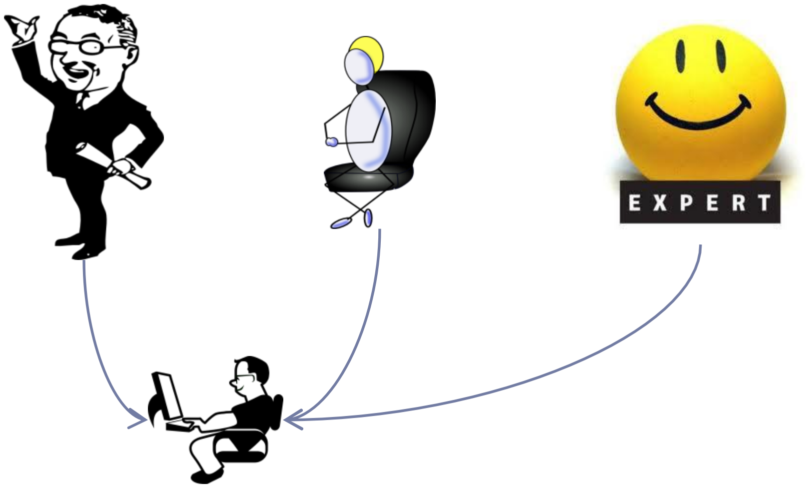
Loss

| Int. Caught | Ext. Caught |
|---|---|
| 1 | 1 |
| 2 | 1 |


$1500

$1250

# Property of Effective Audit Mechanism




- Audit mechanism should be <u>comparable</u> to <u>best expert in hindsight</u>
- Audit:  Experts recommend resource allocations

# Low Regret

- Low regret of *s* w.r.t. *s1* means *s* performs as well as *s1*

- Desirable property of an audit mechanism
  - Low regret w.r.t all strategies in a given set of strategies

$$regret \rightarrow 0 \text{ as } T \rightarrow \infty$$

- Audit setting
  - Audit mechanism recommended resource allocation performs as well as best fixed resource allocation in hindsight

# Challenges in Audit Setting

- Sleeping experts
  - Not all experts available in each audit round (e.g., [300,10] in Figure 1)
- Imperfect information
  - In each round, only one expert's advice is followed and associated loss observed
  - Requires loss estimation for outcome for all other experts

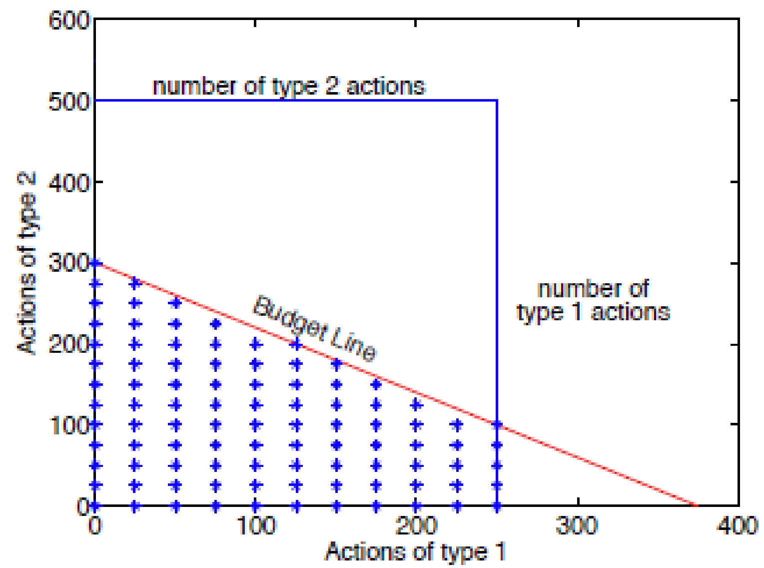
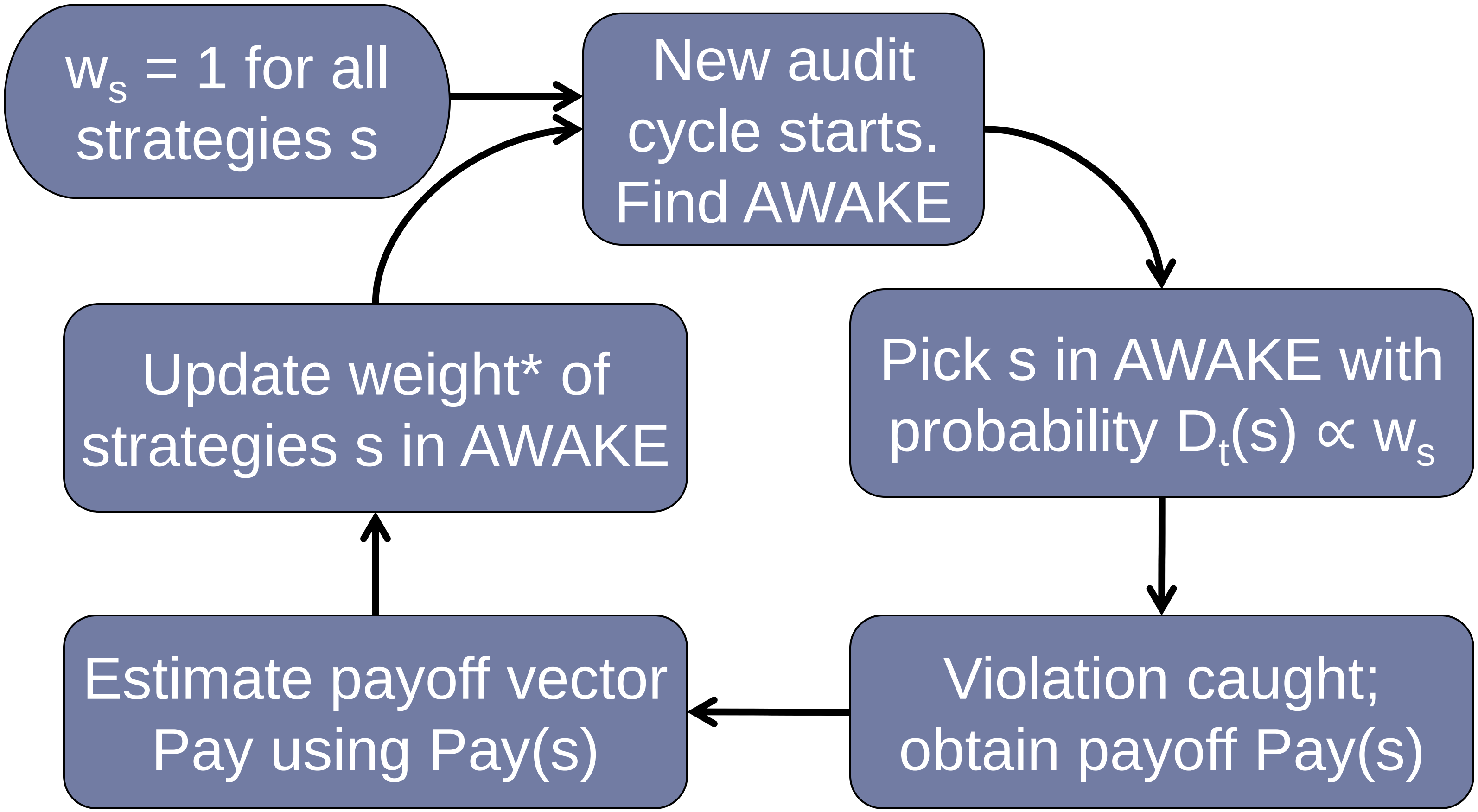



Figure 1. Feasible audit space, represented by the shaded area.

# Regret Minimizing Audits (RMA)




$$^{*}\, w_s \leftarrow w_s . \gamma^{-Pay(s)+\gamma . \Sigma_{s'} D_t(s') Pay(s')}$$

# Audit Algorithm Run

| No. of Access | Actual Violatio |
|---|---|
| 30 | 2 |
| 70 | 4 |


| | | | |
|---|---|---|---|
| 0 | 10 | 20 | 30 |
| 30 | 20 | 10 | 0 |

Observed Loss

Estimated Loss

| Int. Caught | Ext. Caught |
|---|---|
| 1 | 1 |
| 2 | 1 |


| | | | |
|---|---|---|---|
| $2000 | $1500 | $1000 | $1000 |
| $750 | $1250 | $1250 | $1500 |

| Updated weights | 0.5 | 0.5 | 2.0 | 1.5 |
|---|---|---|---|---|

Learn from experience: weights updated using observed and estimated loss

# Guarantees of RMA

- With probability $1 - \epsilon$ RMA achieves the regret bound

$$2\sqrt{\frac{2\ ln\,N}{T}} + \frac{2\ ln\,N}{T} + 2\sqrt{\frac{2\ ln\left(\frac{4N}{\epsilon}\right)}{T}}$$

  - N is the set of strategies
  - T is the number of rounds
  - All payoffs scaled to lie in [0,1]

# Related Work

- Weighted Majority Algorithm [LW89]:
  - Average Regret: $O((\log N)/T)^{1/2}$
  - Defender cannot run this algorithm unless he observes the adversaries moves (perfect information setting)
- Imperfect Information Setting [ACFS02]:
  - Average Regret: $O(((N \log N)/T)^{1/2})$
  - Regret bound converges to 0 much slower
- Our regret bounds are of the same order as the perfect information setting assuming loss estimation function is *accurate* and *independent*

# Play in Three Acts

1. Rational Adversary Setting
2. Byzantine Adversary Setting
3. Research Directions

# Research Directions

- Augmenting model and algorithm
  - Repeated interaction
  - Multiple defender resources constrained by audit budget
  - Multiple heterogeneous targets attacked by adversary
  - Information flow violations
  - Combining rational and byzantine adversary model

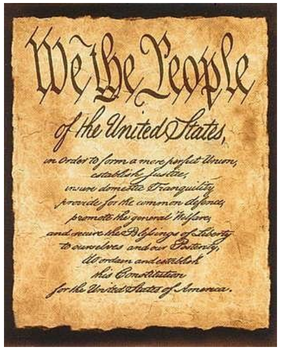
- Acquiring parameters of model
  - Ponemon studies, Verizon data breach reports

Initial results in [BCDS'12]

- From risk management to privacy protection
  - Why should organizations invest in audits to protect privacy?
  - What public policy interventions are most effective in encouraging thorough audits (e.g., HHS audits, data breach notification law)?

# A Research Area

- Formalize Privacy Policies
  - Precise definitions of privacy concepts (restrictions on information flow)
    - Information used *only for* a *purpose*
    - All disclosure clauses in HIPAA & GLBA





- Enforce Privacy Policies
  - Audit and Accountability
    - Detect violations of policy
    - Identify agents to blame for policy violations
    - Resource allocation for inspections and punishments (economic considerations)

Project page: Privacy, Audit and Accountability

Thanks!
Questions?

# Proof of Property of Optimal Point

- Quadratic constraints

  $p_n( x + \Delta_n ) + \delta_{j,n} \leq p_j( x + \Delta_j )$ where $\Delta_j \geq 0$

- Fact 1: $p_j$ is 0 or the $j^{th}$ constraint is tight
- Fact 2a: if $p_n( x + \Delta_n ) + \delta_{j,n} \leq 0$ then $p_j$ is 0
  - $p_j( x + \Delta_j ) \geq 0$, thus the constraint cannot be tight, so $p_j$ is 0
- Fact 2b: if $p_n( x + \Delta_n ) + \delta_{j,n} > 0$ then tight constr
  - $p_j$ cannot be 0, so constraint has to be tight

# Problem $P_n$

Fortunately, the problem $P_n$ has another property that allows for efficient methods. Let us rewrite $P_n$ in a more compact form. Let $\Delta_{D,i} = U_D^a(t_i) - U_D^u(t_i)$, $\Delta_i = U_A^u(t_i) - U_A^a(t_i)$ and $\delta_{i,j} = U_A^u(t_i) - U_A^u(t_j)$. $\Delta_{D,i}$ and $\Delta_i$ are always positive, and $P_n$ reduces to:

$$
\begin{array}{ll}
\max\limits_{p_i, x} & p_n \Delta_{D,n} + U_D^u(t_n) - ax \,, \\
\text{subject to} & \forall i \neq n.\ p_i(-x - \Delta_i) + p_n(x + \Delta_n) + \delta_{i,n} \leq 0 \,, \\
& \forall i.\ 0 \leq p_i \leq 1 \,, \\
& \sum_i p_i = 1 \,, \\
& 0 \leq x \leq 1 \,.
\end{array}
$$

# Problem $Q_{n,i}$

$$
\begin{array}{ll}
\max\limits_{x, p_{(1)}, \ldots, p_{(i)}, p_n} & p_n \Delta_{D,n} - ax \; , \\
\text{subject to} & p_n(x + \Delta_n) + \delta_{(i),n} \geq 0 \; , \\
& \text{if } i \geq 2 \text{ then } p_n(x + \Delta_n) + \delta_{(i-1),n} < 0 \; , \\
& \forall j \geq i.\; p_n(x + \Delta_n) + \delta_{(j),n} = p_{(j)}(x + \Delta_j) \; , \\
& \forall j > i.\; 0 < p_{(j)} \leq 1 \; , \\
& 0 \leq p_{(i)} \leq 1 \; , \\
& \sum_{k=i}^{n-1} p_{(k)} = 1 - p_n \; , \\
& 0 \leq p_n < 1 \; , \\
& 0 < x \leq 1 \; .
\end{array}
$$

# Problem $R_{n,i}$

$$
\begin{aligned}
&\max_{x,p_n} \; p_n \Delta_{D,n} - ax\,, \\
&\text{subject to} \\
&p_n(x + \Delta_n) + \delta_{(i),n} \geq 0\,, \\
&\text{if } i \geq 2 \text{ then } p_n(x + \Delta_n) + \delta_{(i-1),n} < 0\,, \\
&p_n \left(1 + \sum_{j:i \leq j \leq n-1} \frac{x+\Delta_n}{x+\Delta_{(j)}}\right) = 1 - \sum_{j:i \leq j \leq n-1} \frac{\delta_{(j),n}}{x+\Delta_{(j)}}\,, \\
&0 \leq p_n < 1\,, \\
&0 < x \leq 1\,.
\end{aligned}
$$