

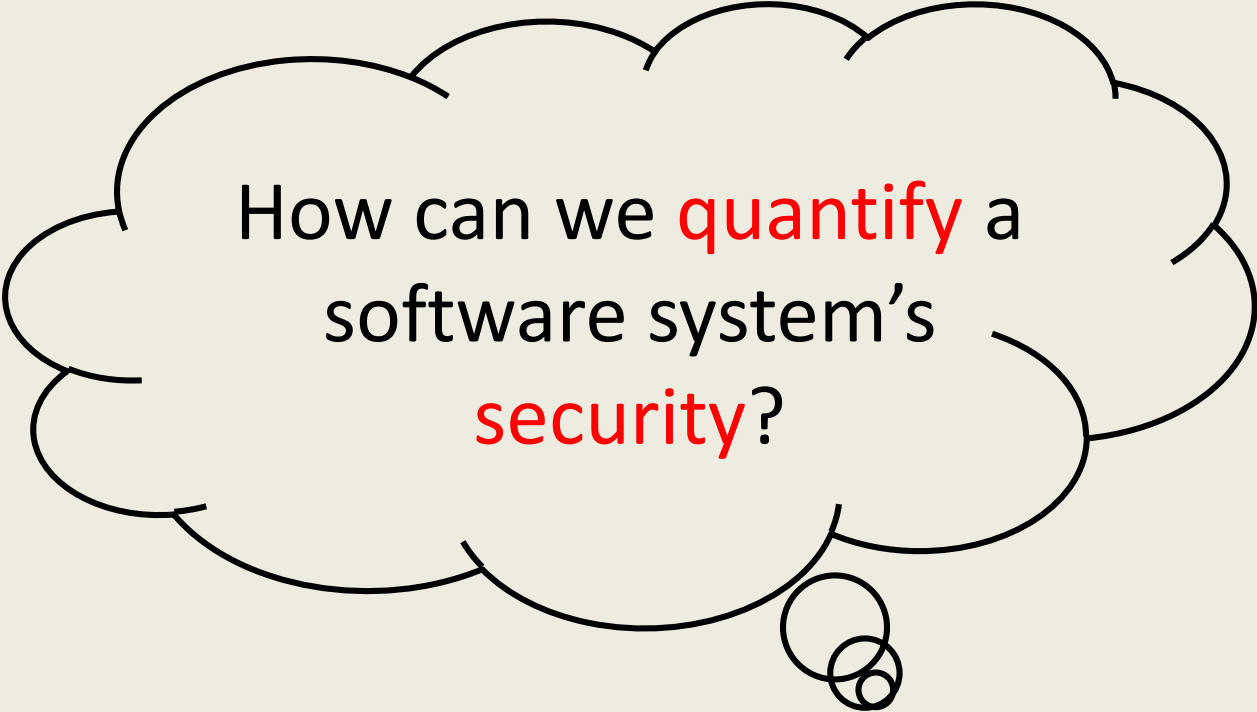
# Game Theoretic Approaches to Attack Surface Shifting and Reduction

Pratyusa K. Manadhata

HP Labs

[manadhata@hp.com](mailto:manadhata@hp.com)

# Context: Attack Surface Measurement (ASM)



How can we **quantify** a  
software system's  
**security**?

Measure the system's **attack surface** [MW10]

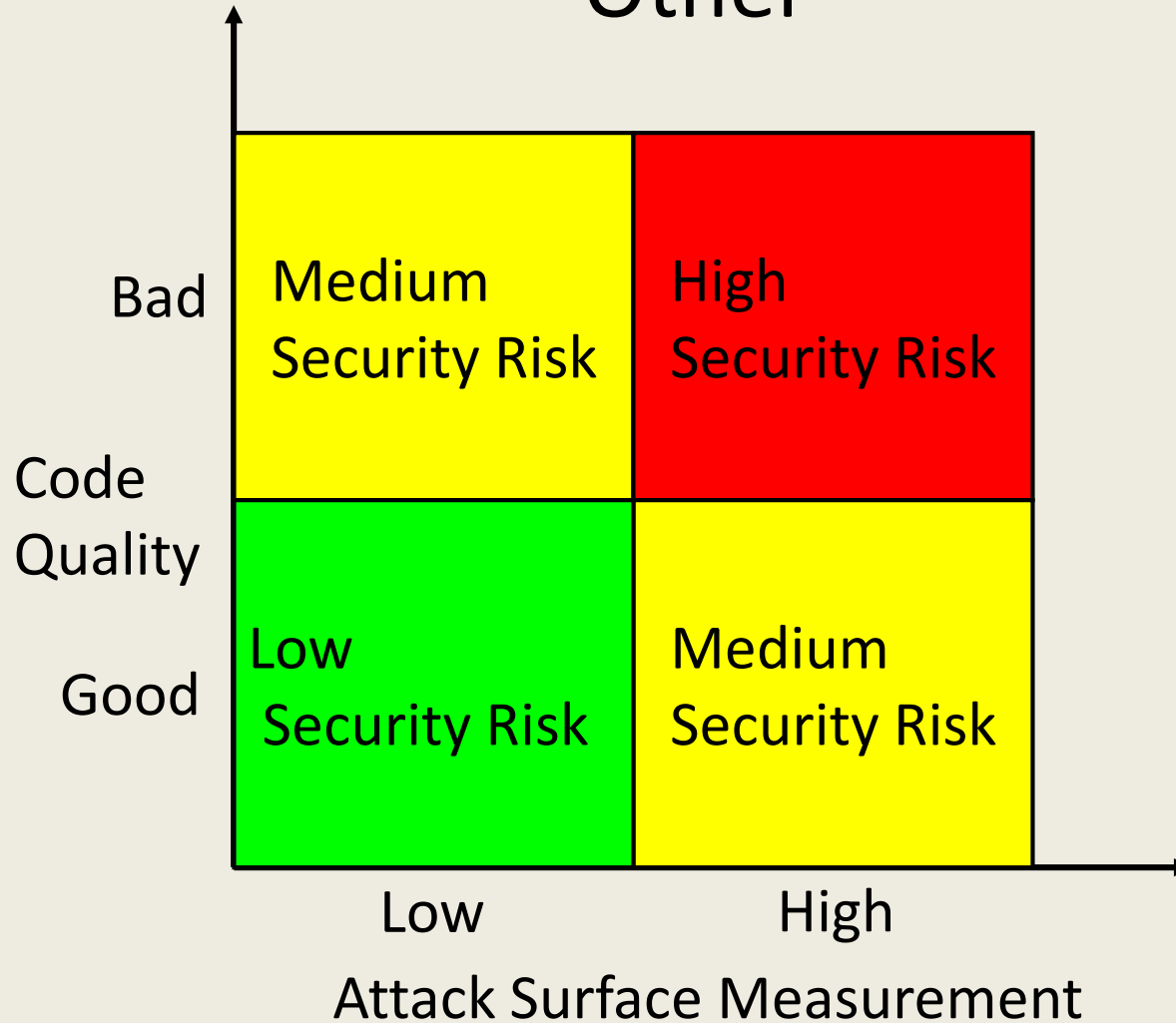
# Attack Surface Reduction (ASR) Mitigates Risk

**Traditional** industry approach: code quality improvement

Software will ship with **known** and **future** vulnerabilities

**Reduce** attack surface to increase the **difficulty** and decrease the **impact** of future exploitation

# Code Quality and ASR Complement Each Other

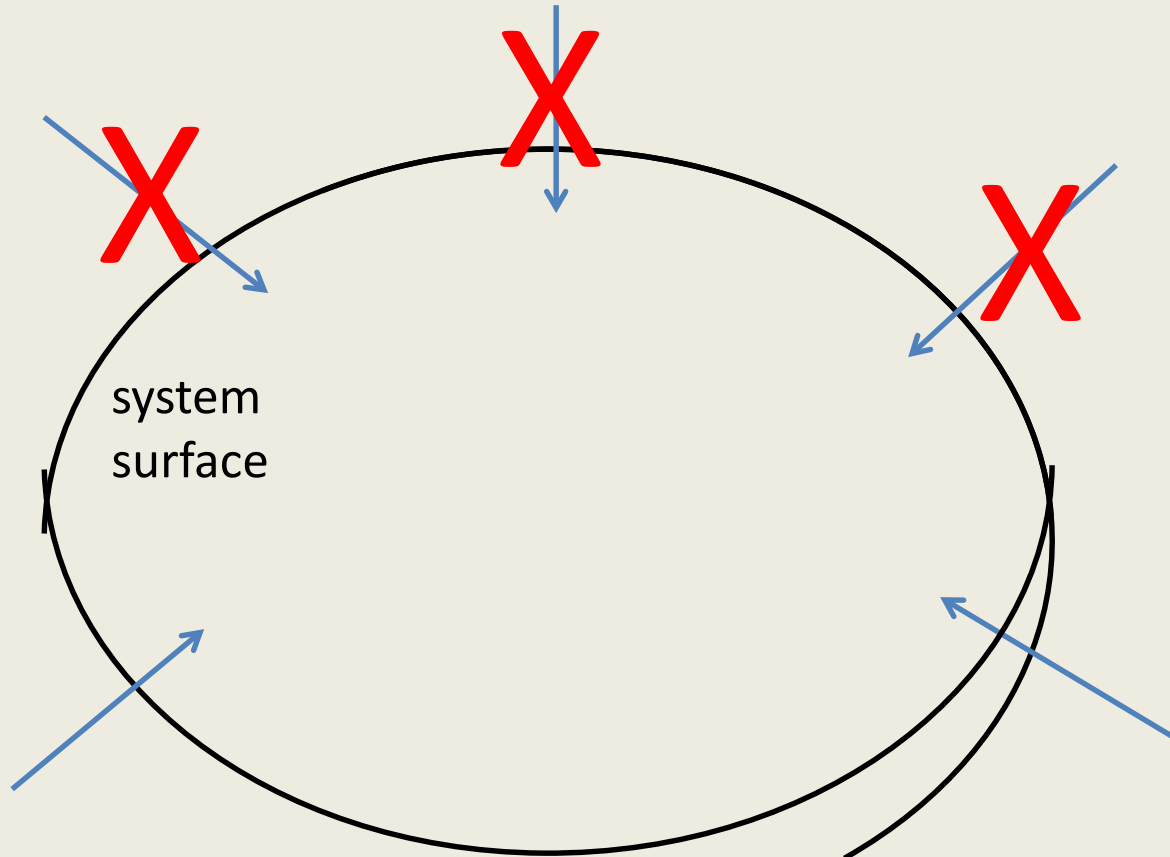


# ASR in the Industry

- Microsoft
- SAP
- MuSecurity
- OpenSSH
- Firefox
- ...

# Moving Target Defense [GPS09, JGSWW11]

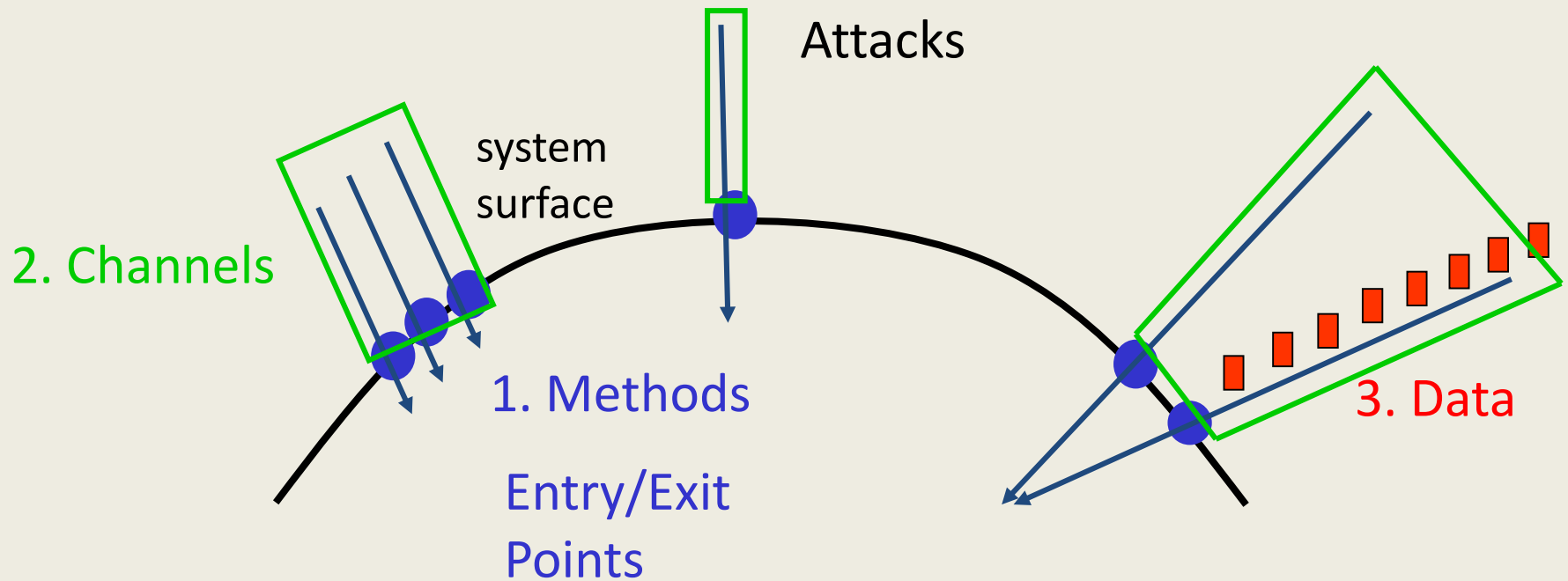
- **Shift** the attack surface
  - “Attacks only work once if at all”



# Outline

- Introduce the notion of attack surface reduction
- **Formalize** the notion of **attack surface shifting**
- Explore **game theoretic approaches** to shift and reduce the attack surface

# Intuition Behind Attack Surfaces

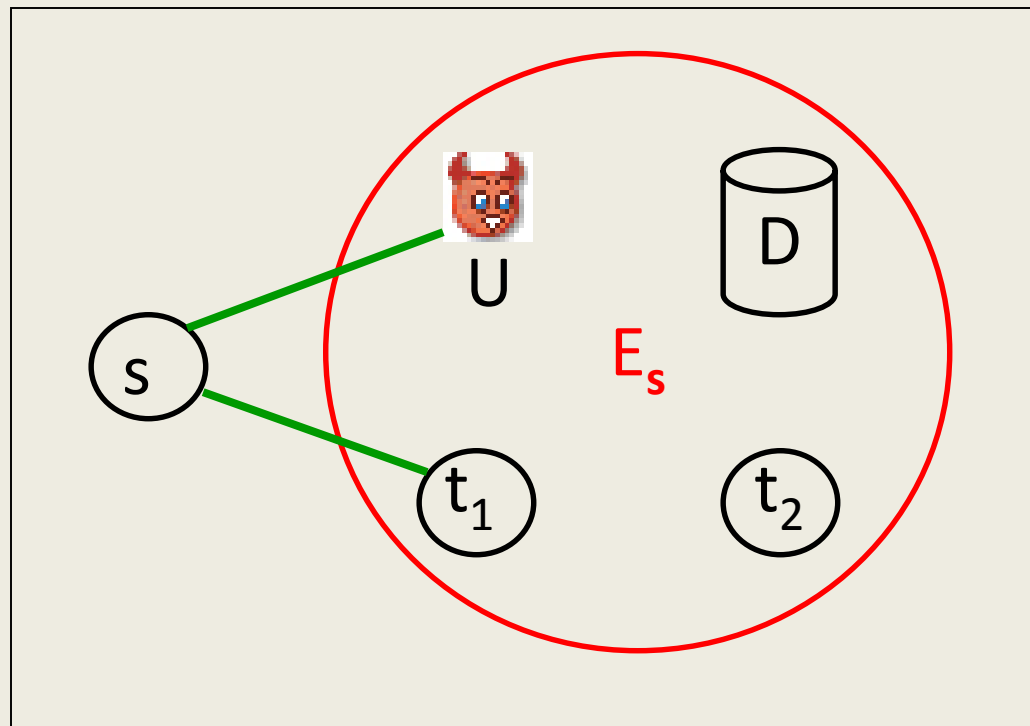


Hence we define a system's attack surface in terms of the system's **resources** (i.e., methods, channels, and data items).



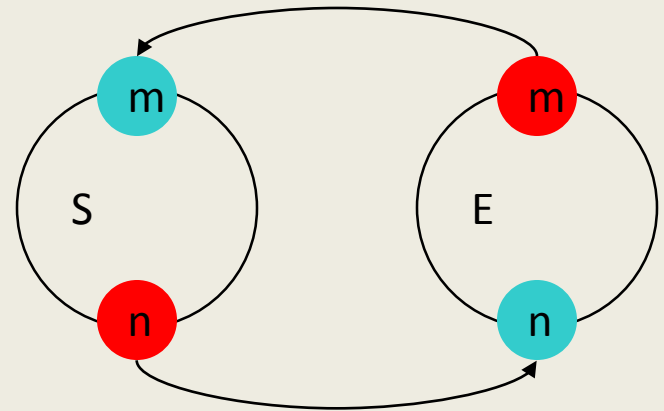
# Model of a System and its Environment

A system,  $s$ , and its environment,  $E_s = \langle U, D, T = \{t_1, t_2\} \rangle$ .



# I/O Automata [LT89]

- Action Signature
  - Input, Output, Internal actions
  - Pre and Post conditions  $m.pre$  and  $m.post$



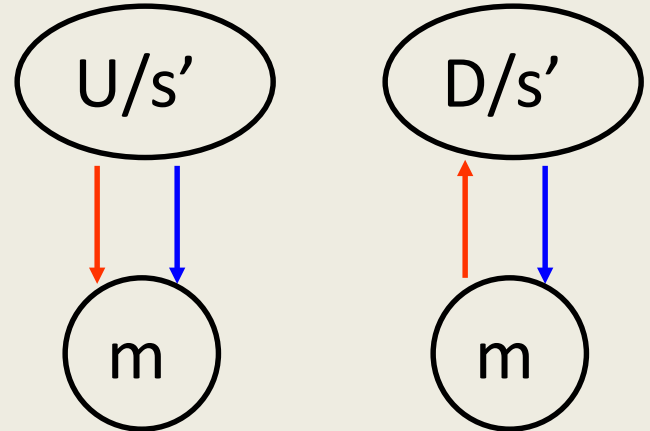
- Composition
  - $E_s = (U_{io} \parallel D_{io} \parallel ( \parallel t_{io} ))$
  - $P = s_{io} \parallel E_s \quad t_{io} \sqcap T_{io}$

# Not All Resources Are Part of the Attack Surface

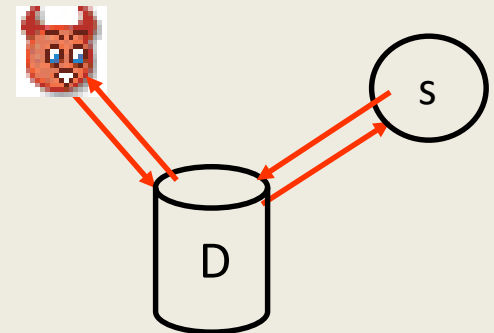
- Only those resources that the attacker can use to **send data into or receive data from** the system are relevant.
- We introduce the formal **entry point and exit point framework** to identify the relevant resources.

# Entry Point and Exit Point Framework

- Entry Points/Exit Points
  - **Direct** (input/output action)
  - **Indirect** (internal action)



- Channels (e.g., sockets and pipes)
  - $c \in \text{Res}(m.\text{pre})$
- Untrusted Data Items (e.g., files)
  - $d \in \text{Res}(m.\text{post}), d \in \text{Res}(m.\text{pre})$



# Attack Surface Definition

- Definition
  - **M**: set of entry points and exit points
  - **C**: set of channels
  - **I**: set of untrusted data items.

attack surface =  $\langle M, C, I \rangle$

# Larger Attack Surface Leads to More Attacks

**Attacks** ( $s$ ) = The set of **executions** of  $(s \parallel E_s)$  that contain either an **input action** or **output action** of  $s$ .

Theorem: Given an environment,  $E$ , if  $AS(A) \geq AS(B)$ , then  $Attacks(A \parallel E) \supseteq Attacks(B \parallel E)$ .

# Not All Resources Contribute Equally to the Attack Surface

- Contribution  $\propto$  **Damage Potential**

$$\text{Contribution} \propto (\text{Attacker Effort})^{-1}$$

- Contribution = 
$$\frac{\text{Damage Potential}}{\text{Attacker Effort}}$$

# Attack Surface Measurement (ASM)

- $ASM(A) \geq ASM(B)$  if there exists a nonempty set,  $R$ , of resources s.t.  
 $\forall r \in R. \text{contribution}(r, A) \geq \text{contribution}(r, B).$

Theorem: Given an environment,  $E$ , if  $ASM(A) \geq ASM(B)$ , then  $\text{Attacks}(A || E) \supseteq \text{Attacks}(B || E)$ .



# Quantitative Attack Surface Measurement

- Assume **der**: method  $\rightarrow$  Q.
  - Similarly, for channel and data.

$$\text{ASM} = \left\langle \sum_{m \in M} \text{der}(m), \sum_{c \in C} \text{der}(c), \sum_{d \in I} \text{der}(d) \right\rangle$$

# Numeric Damage Potential-Effort Ratio

Resource	Damage Potential	Attacker Effort
Method	Privilege	Access Rights
Channel	Protocol	Access Rights
Data Items	Type	Access Rights

Impose a **total ordering** among the values of the attributes and assign numeric values accordingly, e.g.,  
root = 5 and auth = 3.

# Attack Surface Measurement Method

1. **Identify** a set, **M**, of entry points and exit points, a set, **C**, of channels, and a set, **I**, of untrusted data items.
2. **Estimate** each relevant resource's damage potential-effort ratio, **der**.
3. **Compute** Attack Surface Measurement =  
$$\left\langle \sum_{m \in M} \text{der}(m), \sum_{c \in C} \text{der}(c), \sum_{d \in I} \text{der}(d) \right\rangle .$$

# Shifting the Attack Surface

- Scenario: A system's defender is trying to protect the system from an attacker.
- Goal: Shift the attack surface such that **old attacks** don't work any more
  - may introduce **new attacks**

# Not All Changes Shift the Attack Surface

- Changing the attack surface by changing **features**
  - Add/remove resources
  - Change existing resource's contribution
- Shifting the attack surface
  - **Remove** at least one existing resource
  - **Reduce** an existing resource's contribution

# Definition of Shifting

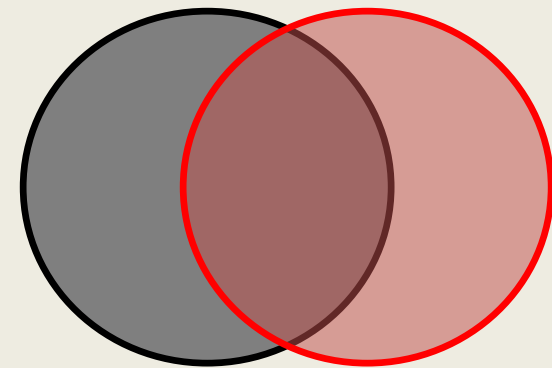
- $R_o$ : **old** attack surface
- $R_n$ : **new** attack surface
- $r_o$ : a resource,  $r$ 's, **contribution** to  $R_o$
- $r_n$ :  $r$ 's contribution to  $R_n$

$$\Delta AS = |R_o \setminus R_n| + |\{r: (r \in R_o \cap R_n) \wedge (r_o > r_n)\}|$$

# Shifting Prevents Old Attacks

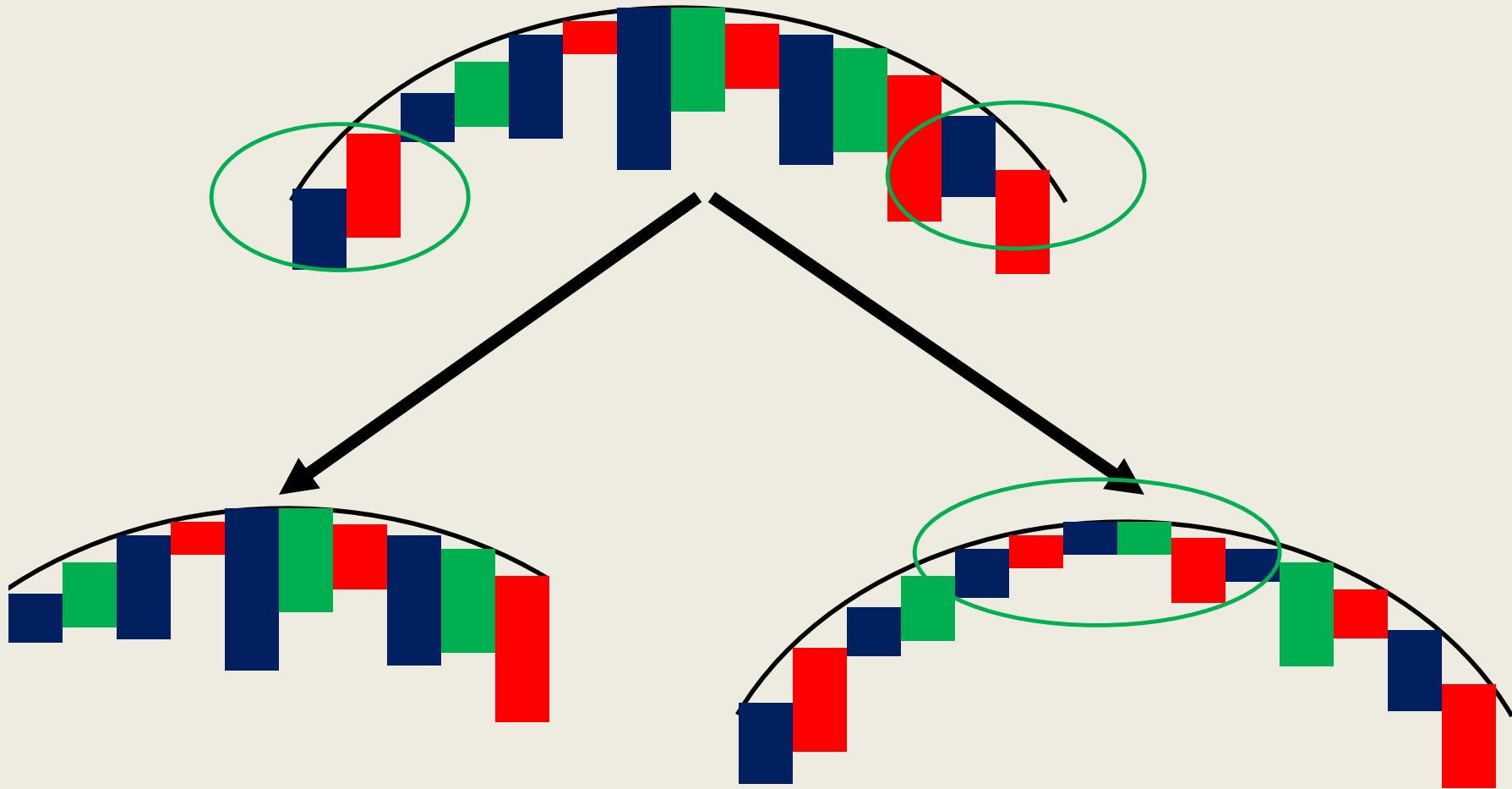
- Given a system,  $S$ , an environment,  $E$ , and  $S$ 's attack surface,  $R$ , the **set of attacks** on  $S$  is  **$\text{Attacks}(S_R || E)$** .

Theorem: Given an environment,  $E$ , an old attack surface,  $R_o$ , a new attack surface,  $R_n$ , if  $\Delta AS > 0$ , then  $\text{Attacks}(S_{R_o} || E) \setminus \text{Attacks}(S_{R_n} || E) \neq \phi$ .



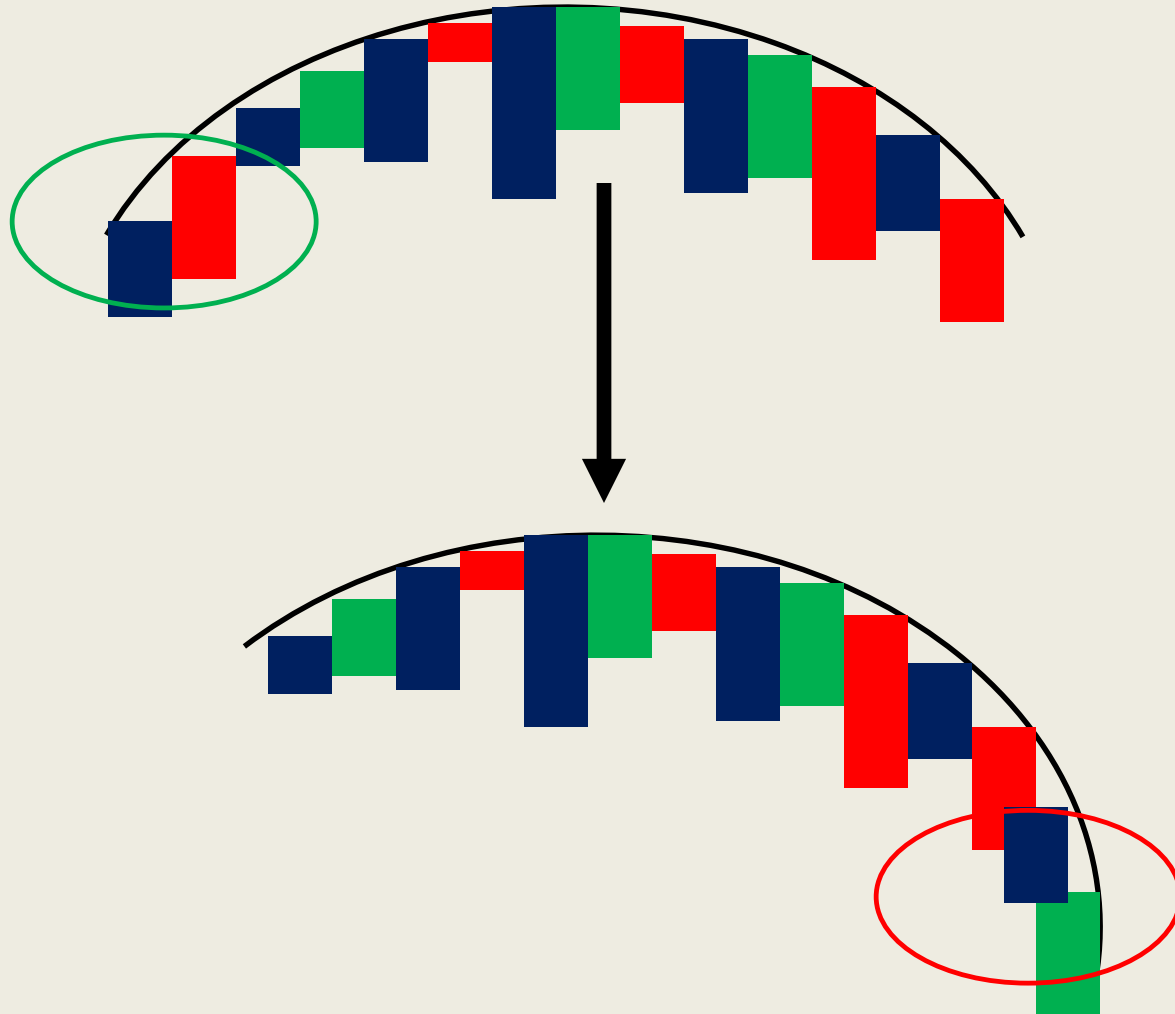
Old attacks    New attacks

# Disable Features: AS Shift and ASM Reduction

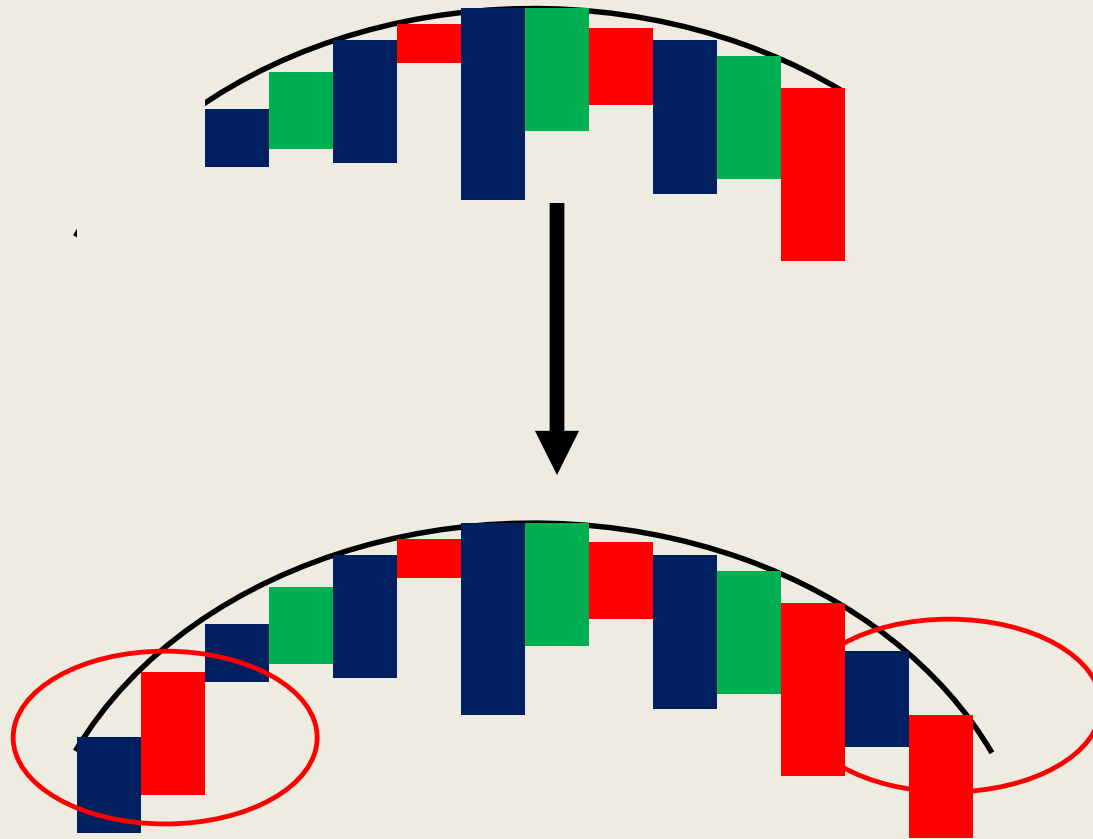




# Enable and Disable Features: AS Shift and ASM Reduction/Addition/Identical



# Enable Features: No AS Shift and ASM Addition



# Summary of Scenarios

<b>Scenario</b>	<b>Feature</b>	<b>AS Shift</b>	<b>ASM</b>
A	Disabled	Yes	Reduction
B	Enabled and Disabled	Yes	Reduction
C	Enabled and Disabled	Yes	Identical
D	Enabled and Disabled	Yes	Addition
E	Enabled	No	Addition

# Scenario choice is a Security-Usability Trade-off

- While shifting the attack surface, which features to disable and which features to enable?
  - More features => more usable system
  - More features => larger attack surface

# A Game Theoretic Approach to Moving Target Defense

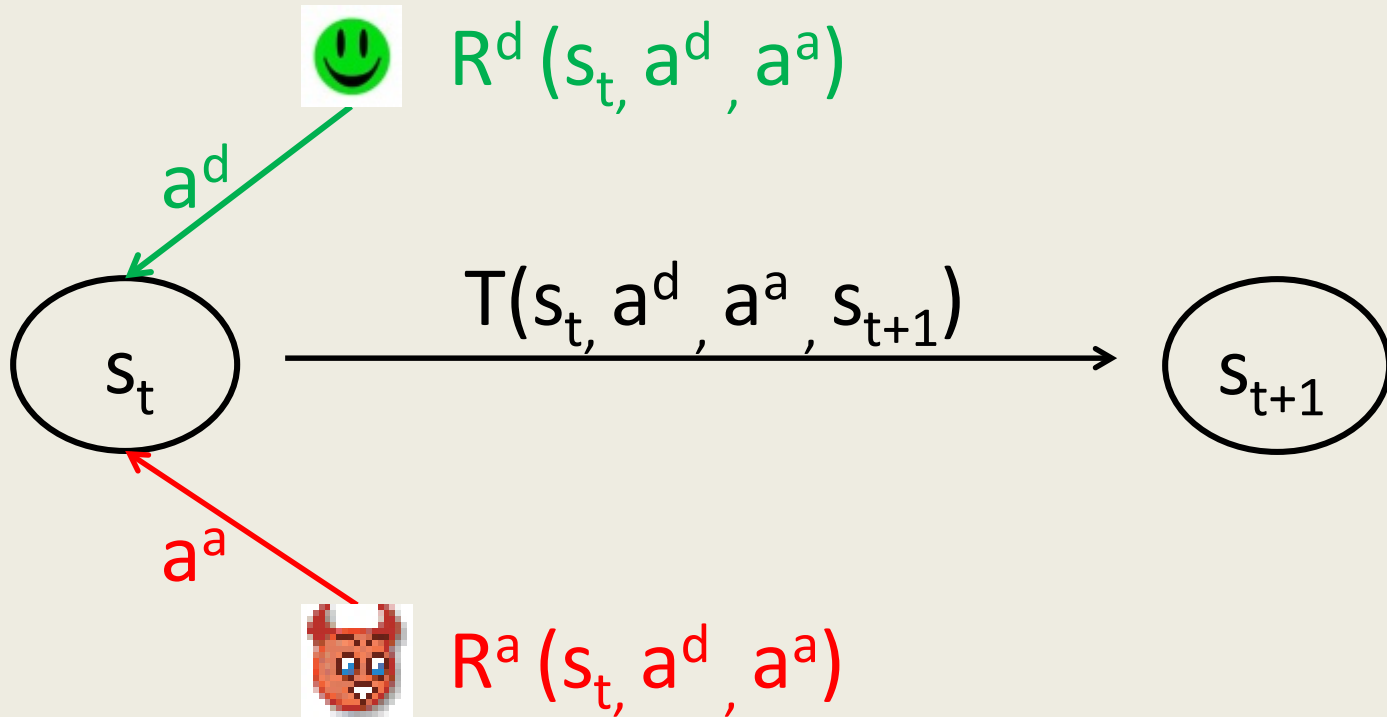
- Prior work: **static** software development process
  - No assumptions about the attacker
- Moving target defense is a **dynamic** scenario
  - **Interaction** between a defender and an attacker is a game
  - **Explicit** attacker model

# Two-Player Stochastic Game Model

## [LW02]

- Game =  $\langle S, A^d, A^a, T, R^d, R^a, \beta \rangle$
- $S$ : set of states
- $A^*$ : action sets
- $T: S \times A^d \times A^a \times S \rightarrow [0,1]$ : transition function
- $R^*: S \times A^d \times A^a \rightarrow \mathbb{R}$ : reward functions
- $\beta$ : discount factor

# Game Play



Goal: maximize discounted reward.

# States, Actions, and Transitions

- State: **Feature** → **Configuration**
- Action: Feature → **FeatureAction**
- Transition: **Specific** to a system and its environment



# Reward Functions

- $\Delta F$ : change in features
- $\Delta AS$ : shift in the AS
- $\Delta ASM$ : change in the ASM

$$R^d: B_1^d (\Delta F) + B_2^d (\Delta AS) - C^d (\Delta ASM)$$

$$R^a: B^a (\Delta ASM) - C^a (\Delta AS)$$

# Optimal Defense Strategies

- Model the interaction as an **extensive** game
  - Complete and perfect information
  - General sum game
- Solution: Equilibrium

# Stationary and Dynamic Strategies

- Stationary strategy
  - **Independent** of history
  - **Nash** equilibrium
  - Non-linear program for stochastic games [FV96]
- Dynamic strategy
  - Optimal action after **every game history**
  - **Subgame perfect** Nash equilibrium
  - Dynamic programming approach [MG07]

# Future Work: Instantiate the Model

Challenges in applying the model to real-world systems

- State space explosion
  - Focus on an important set of features
- Transition probabilities
- Reward functions
  - Cost and Benefit functions

# Future Work: Model Efficacy

- How much **effort** is necessary to **instantiate** the model?
  - Is the model's **benefit** worth the effort?
- How does one **compare** alternative game models?
- **Alternative approaches** to achieve moving target defense?



# A Game Theoretic Approach

- Consider a feature's “reward” value
  - $B_1^d(\Delta F) + B_2^d(\Delta AS) - C^d(\Delta ASM)$
  - Add features in decreasing order of reward
  - Remove features in increasing order of reward

The simplistic approach ignores **feature interaction**.

# Shapley Value [S53]

- **Coalitional game**  $(N, v)$ 
  - $N$ : a set of players
  - $v: 2^N \rightarrow \mathbb{R}$  : **characteristic** function

$$\Phi_i(v) = \sum_{C \subseteq N \setminus i} \frac{|C|!(|N| - |C| - 1)!}{|N|!} \{v(C \cup \{i\}) - v(C)\}$$



# Choose Features According to their Shapley Value

- **Features** are **players** in a coalitional game
- Characteristic function: **Reward function**
- Shapley value: A **feature's contribution** to security and usability

# Related Work

- Moving target defense
  - A. Ghosh et al.: National cyber leap year summit 2009 co-chairs report, 2009.
  - S. Jajodia et al.: Moving Target Defense: Creating Asymmetric Uncertainty for Cyber Threats, Springer, 2011.
- Game theory and security
  - S. Roy et al.: A survey of game theory as applied to network security, HICSS 2010.
  - M. Manshaei et al.: Game Theory Meets Network Security and Privacy, ACM Trans. On Computational Logic, 2010.

# Summary

- **Formalized** the notion of shifting the attack surface
  - Introduced **game theoretic approaches** to shift and reduce the attack surface
- A **first step** in moving target defense
  - Understanding over time will lead to better approaches

# Backup

- [MW10], P. K. Manadhata and J. Wing, An Attack Surface Metric, IEEE Trans. on Software Engg., 2010.
- [GPS09] A. Ghosh et al.: National cyber leap year summit 2009 co-chairs report, 2009.
- [JGSWW11] S. Jajodia, A. Ghosh, V. Swarup, C. Wang, and X.S. Wang, Moving Target Defense: Creating Asymmetric Uncertainty for Cyber Threats, Springer, 2011.
- [LT89] N. Lynch and M. Tuttle, An introduction to input/output automata, CWI-Quarterly, 1989.
- [LW02] K. Lye and J. Wing, Game strategies in network security, International Journal of Information Security, 2005.
- [FV96] J. Filar and K. Vrieze, Competitive Markov decision processes, Springer, 1997.
- [MG07] C. Murray and G. Gordon, Finding correlated equilibria in general sum stochastic games, Tech. Rep. CMU-ML-07-113, Carnegie Mellon University, 2007.
- [S53] L. Shapley, A Value for  $n$ -person Games, In *Contributions to the Theory of Games*, volume II, 1953.