RESEARCH REVIEW 2022

Carnegie Mellon University Software Engineering Institute

Chain Games: Powering Autonomous Threat Hunting

NOVEMBER 14–16, 2022

Phil Groce
Senior Network Defense Analyst

Document Markings

This material is based upon work funded and supported by the Department of Defense under Contract No. FA8702-15-D-0002 with Carnegie Mellon University for the operation of the Software Engineering Institute, a federally funded research and development center.

DM22-0881

Introduction

Threat hunting is a critical part of cyber defense, but the amount of data available to threat hunters is overwhelming.

To develop effective autonomous threat hunting techniques, we are developing Chain Games, a set of games in which threat hunting strategies can be evaluated and refined.

Chain Games: Powering Autonomous Threat Hunting

Motivation and Approach

What is Threat Hunting?

Carnegie Mellon University Software Engineering Institute

Intrusion Detection/Prevention

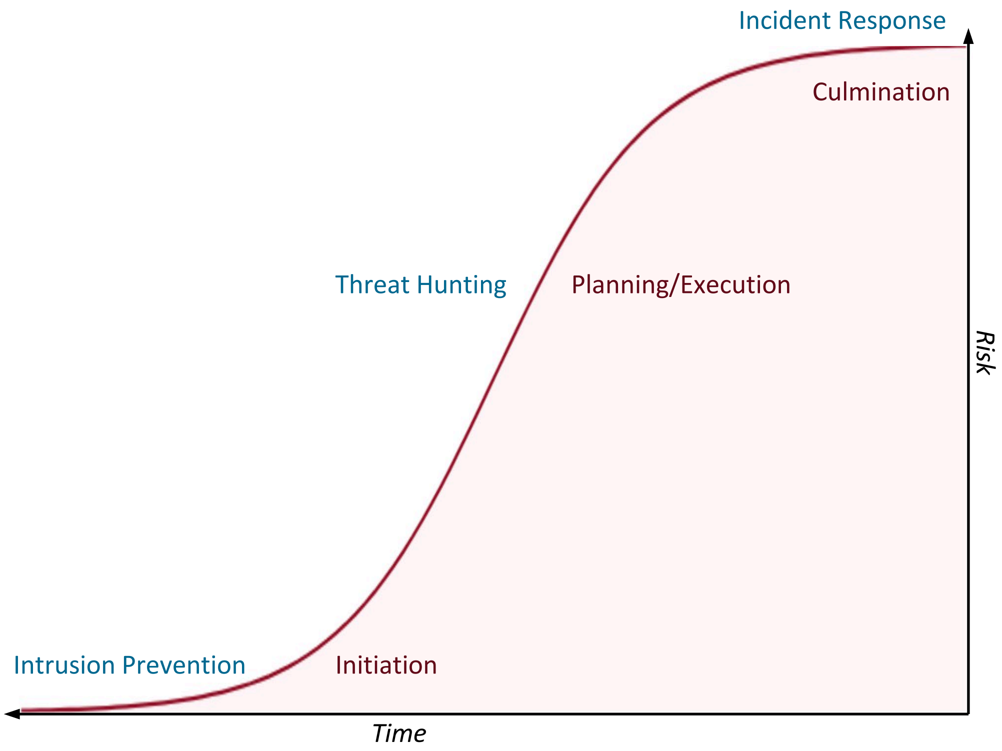
- How do we keep the attackers out?

Incident Response

- How do we mitigate what the attackers did?

Threat Hunting

- How do we find/remove the attackers who got in?



Carnegie Mellon University Software Engineering Institute

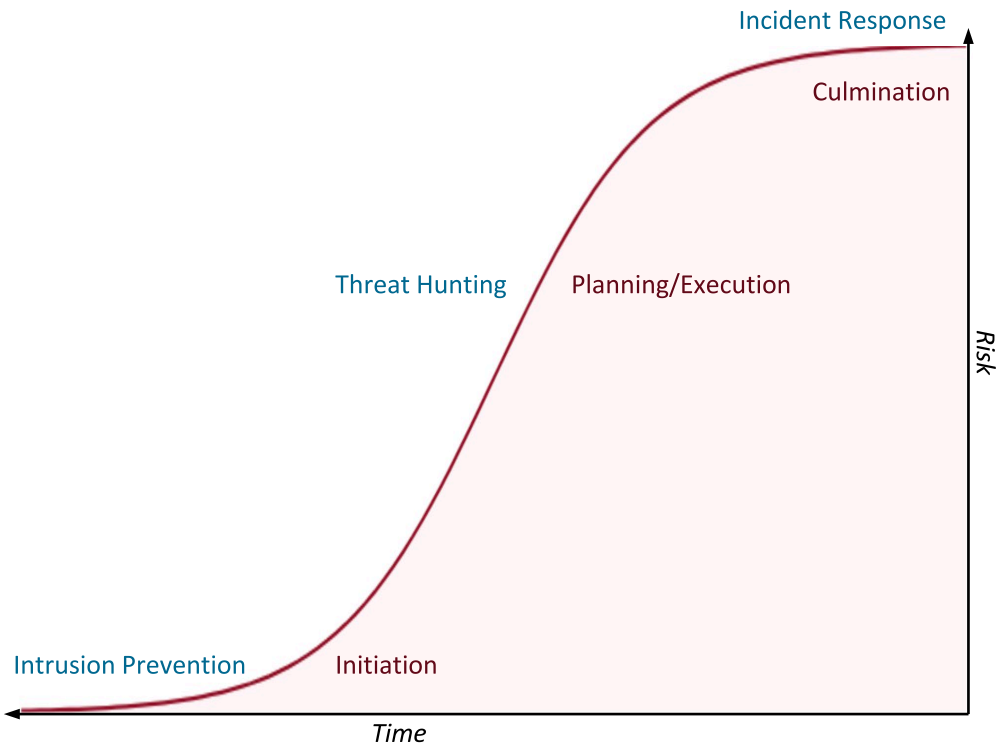
Benefits of Autonomous Threat Hunting

Threat hunting takes time and skill.

Inexpensive, faster hunting could:

- Investigate more data sources
- Coordinate for coverage
- Help triage human threat hunts

The key to faster, less expensive threat hunting is autonomy.



Cyborg Security. *The Threat Hunter's Hypothesis.* https://www.cyborgsecurity.com/library/guides/the-threat-hunters-hypothesis-2/

Approaches to Autonomy

Long-term goal: autonomy

- Predication
- Investigation
- Conclusion

Short-term goal: modeling

- Quantitatively evaluating and developing strategies
- Rapid strategic development
- Capturing the adversarial quality of threat hunting activity

Cyber Deception Games (CDG) and Cyber Camouflage Games (CCG)

Carnegie Mellon University Software Engineering Institute

2018: Cyber Deception Games [1]

- Situates work in the Cyber Kill Chain
  - Focuses on reconnaissance
- Is a zero-sum game
- Defender is deceiver

2019: Cyber Camouflage [2]

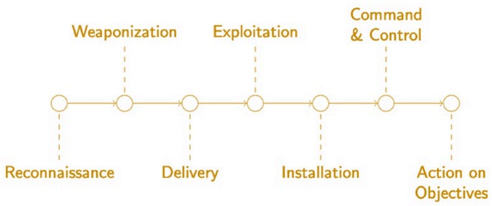
- Is extended to general-sum games
- Defender is still deceiver

Schlenker A, Thakoor O, Xu H, Fang F, Tambe M, Tran-Thanh L, Vayanos P, Vorobeychik Y, “Deceiving cyber adversaries: A game theoretic approach,” in Proceedings of the 17th International Conference on Autonomous Agents and MultiAgent Systems, International Foundation for Autonomous Agents and Multiagent Systems, 2018, pp. 892-900.

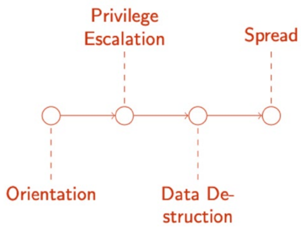
Thakoor O, Tambe M, Vayanos P, Xu H, Kiekintveld C, Fang F. “Cyber Camouflage Games for Strategic Deception," in Decision and Game Theory for Security, Springer International Publishing, 2019, pp. 525-541.

Kill/Attack Chains

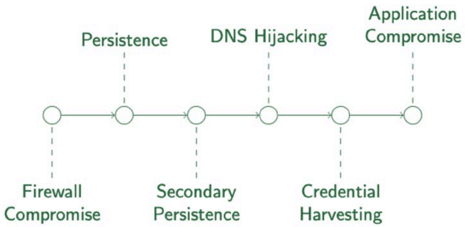
Carnegie Mellon University Software Engineering Institute



The Cyber Kill Chain



Ransomware (NotPetya)



APT Campaign (DriftingCloud)

Attack behavior is often conceptualized as chains.

- Decomposes attacks
- Categorizes attack behaviors

ISACA Now Blog. *Ransomware Analysis – Executions Flow and Kill Chain.* https://www.isaca.org/resources/news-and-trends/isaca-now-blog/2017/ransomware-analysis-executions-flow-and-kill-chain

Lockheed-Martin. *The Cyber Kill Chain*. https://www.lockheedmartin.com/en-us/capabilities/cyber/cyber-kill-chain.html

Volexity. *DriftingCloud: Zero-Day Sophos Firewall Exploitation and an Insidious Breach.* https://www.volexity.com/blog/2022/06/15/driftingcloud-zero-day-sophos-firewall-exploitation-and-an-insidious-breach/

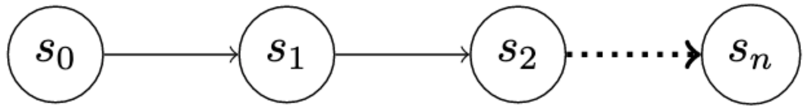
Chain Games: Powering Autonomous Threat Hunting

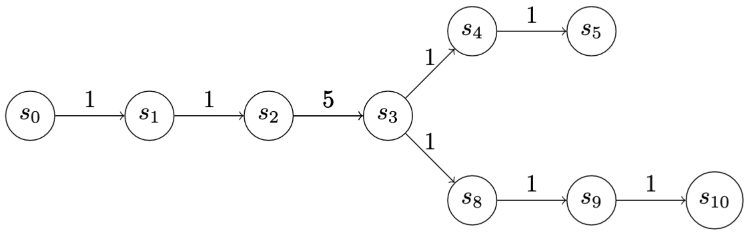
Simple Chain Games

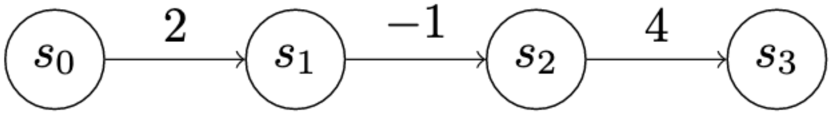
Chain Games –1

Chain Games are played on state chains.

- States represent positions in the network conveying advantage (or disadvantage) to the attacker.
- The utility and cost of occupying a state can be quantified.
- Progress through the state chain motivates the attacker; stopping progress motivates the defender.





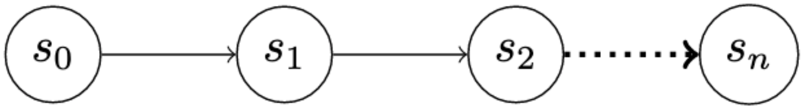


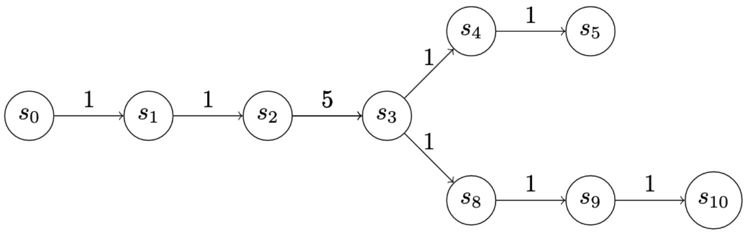
Chain Games –2

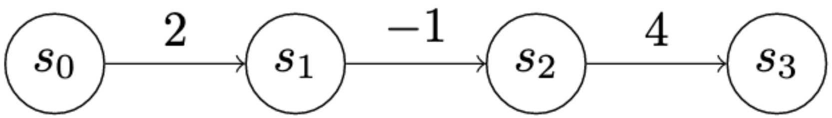
Carnegie Mellon University Software Engineering Institute

Rules

- Two players (Attacker and Defender)
- Fixed number of turns
- General-sum (with zero-sum components)
- Simultaneous action







Chain Game Version 0: Actions and Payoffs

Attacker Actions

- Advance *A* (Cost: 1)
  - Advances to next state in chain

Defender Actions

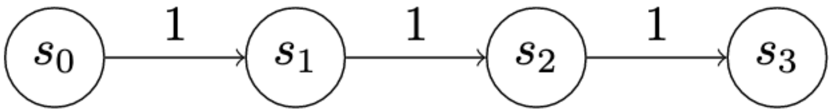
- Defend *D* (Cost: 1)
  - Negates attacker *A* action

Common Action

- Wait *W* (Cost: 0)
  - No additional effect

Payoffs

- Attacker gets positional payoff for each advance
- Defender gets negated positional payoff for each advance



Uniform-Value Chain

Chain Game Version 0: Dominant Strategies

Attacker

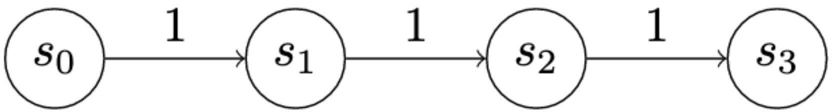
- Always A

Defender

- Never D (i.e., always W)

Takeaways

- The full value of a strategy is its utility across **all opponent strategies**
- Changes in costs/payoffs lead to different analytic outcomes



	WW	AW	WA	AA
WW	$(0,0)$	$(-2,1)$	$(-2,1)$	$(-4,2)$
WD	$(-1,0)$	$(-3,1)$	$(-1,-1)$	$(0,-3)$
DW	$(-1,0)$	$(-1,-1)$	$(-3,1)$	$(-3,0)$
DD	$(-2,0)$	$(-2,-1)$	$(-2,-1)$	$(-2,-2)$

Payout Matrix Over Two Turns, Uniform-Value Chain

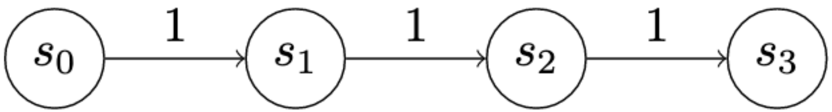
Introducing Camouflage

Attacker Actions

- Noisy Advance N
- Camouflaged Advance C
- C more costly than N

Defender Actions

- Weak Detect L(ow), Strong Detect H(igh)
- L only detects N
- H more costly than L



	WWW	WWN	WWC	WNW	WNN	WNC	WCW	WCN	WCC
WWW	(0,0)	(-1,0)	(-2,0)	(-1,0)	(-2,0)	(-3,0)	(-2,0)	(-3,0)	(-4,0)
WWL	(-3,2)	(-1,-1)	(-2,-1)	(-4,2)	(-2,-1)	(-3,-1)	(-5,2)	(-3,-1)	(-4,-1)
WWH	(-3,1)	(-4,1)	(-2,-2)	(-4,1)	(-5,1)	(-3,-2)	(-5,1)	(-6,1)	(-4,-2)
WLW	(-3,2)	(-4,2)	(-5,2)	(-1,-1)	(-2,-1)	(-3,-1)	(-2,-1)	(-3,-1)	(-4,-1)
WLL	(-6,4)	(-4,1)	(-5,1)	(-4,1)	(-2,-2)	(-3,-2)	(-5,1)	(-3,-2)	(-4,-2)
WLH	(-6,3)	(-7,3)	(-5,0)	(-4,0)	(-5,0)	(-3,-3)	(-5,0)	(-6,0)	(-4,-3)
WHW	(-3,1)	(-4,1)	(-5,1)	(-4,1)	(-5,1)	(-6,1)	(-2,-2)	(-3,-2)	(-4,-2)
WHL	(-6,3)	(-4,0)	(-5,0)	(-7,3)	(-5,0)	(-6,0)	(-5,0)	(-3,-3)	(-4,-3)
WHH	(-6,2)	(-7,2)	(-5,-1)	(-7,2)	(-8,2)	(-6,-1)	(-5,-1)	(-6,-1)	(-4,-4)
LWW	(-3,2)	(-4,2)	(-5,2)	(-4,2)	(-5,2)	(-6,2)	(-5,2)	(-6,2)	(-7,2)
LWL	(-6,4)	(-4,1)	(-5,1)	(-7,4)	(-5,1)	(-6,1)	(-8,4)	(-6,1)	(-7,1)
LWH	(-6,3)	(-7,3)	(-5,0)	(-7,3)	(-8,3)	(-6,0)	(-8,3)	(-9,3)	(-7,0)
LLW	(-6,4)	(-7,4)	(-8,4)	(-4,1)	(-5,1)	(-6,1)	(-5,1)	(-6,1)	(-7,1)
LLL	(-9,6)	(-7,3)	(-8,3)	(-7,3)	(-5,0)	(-6,0)	(-8,3)	(-6,0)	(-7,0)
LLH	(-9,5)	(-10,5)	(-8,2)	(-7,2)	(-8,2)	(-6,-1)	(-8,2)	(-9,2)	(-7,-1)
LHW	(-6,3)	(-7,3)	(-8,3)	(-7,3)	(-8,3)	(-9,3)	(-5,0)	(-6,0)	(-7,0)
LHL	(-9,5)	(-7,2)	(-8,2)	(-10,5)	(-8,2)	(-9,2)	(-8,2)	(-6,-1)	(-7,-1)
LHH	(-9,4)	(-10,4)	(-8,1)	(-10,4)	(-11,4)	(-9,1)	(-8,1)	(-9,1)	(-7,-2)
HWW	(-3,1)	(-4,1)	(-5,1)	(-4,1)	(-5,1)	(-6,1)	(-5,1)	(-6,1)	(-7,1)
HWL	(-6,3)	(-4,0)	(-5,0)	(-7,3)	(-5,0)	(-6,0)	(-8,3)	(-6,0)	(-7,0)
HWH	(-6,2)	(-7,2)	(-5,-1)	(-7,2)	(-8,2)	(-6,-1)	(-8,2)	(-9,2)	(-7,-1)
HLW	(-6,3)	(-7,3)	(-8,3)	(-4,0)	(-5,0)	(-6,0)	(-5,0)	(-6,0)	(-7,0)
HLL	(-9,5)	(-7,2)	(-8,2)	(-7,2)	(-5,-1)	(-6,-1)	(-8,2)	(-6,-1)	(-7,-1)
HLH	(-9,4)	(-10,4)	(-8,1)	(-7,1)	(-8,1)	(-6,-2)	(-8,1)	(-9,1)	(-7,-2)
HHW	(-6,2)	(-7,2)	(-8,2)	(-7,2)	(-8,2)	(-9,2)	(-5,-1)	(-6,-1)	(-7,-1)
HHL	(-9,4)	(-7,1)	(-8,1)	(-10,4)	(-8,1)	(-9,1)	(-8,1)	(-6,-2)	(-7,-2)
HHH	(-9,3)	(-10,3)	(-8,0)	(-10,3)	(-11,3)	(-9,0)	(-8,0)	(-9,0)	(-7,-3)

	NWW	NWN	NWC	NNW	NNN	NNC	NCW	NCN	NCC	CWW
WWW	(-1,0)	(-2,0)	(-3,0)	(-2,0)	(-3,0)	(-4,0)	(-3,0)	(-4,0)	(-5,0)	(-2,0)
WWL	(-4,2)	(-2,-1)	(-3,-1)	(-5,2)	(-3,-1)	(-4,-1)	(-6,2)	(-4,-1)	(-5,-1)	(-5,2)
WWH	(-4,1)	(-5,1)	(-3,-2)	(-5,1)	(-6,1)	(-4,-2)	(-6,1)	(-7,1)	(-5,-2)	(-5,1)
WLW	(-4,2)	(-5,2)	(-6,2)	(-2,-1)	(-3,-1)	(-4,-1)	(-3,-1)	(-4,-1)	(-5,-1)	(-5,2)
WLL	(-7,4)	(-5,1)	(-6,1)	(-5,1)	(-3,-2)	(-4,-2)	(-6,1)	(-4,-2)	(-5,-2)	(-8,4)
WLH	(-7,3)	(-8,3)	(-6,0)	(-5,0)	(-6,0)	(-4,-3)	(-6,0)	(-7,0)	(-5,-3)	(-8,3)
WHW	(-4,1)	(-5,1)	(-6,1)	(-5,1)	(-6,1)	(-7,1)	(-3,-2)	(-4,-2)	(-5,-2)	(-5,1)
WHL	(-7,3)	(-5,0)	(-6,0)	(-8,3)	(-6,0)	(-7,0)	(-6,0)	(-4,-3)	(-5,-3)	(-8,3)
WHH	(-7,2)	(-8,2)	(-6,-1)	(-8,2)	(-9,2)	(-7,-1)	(-6,-1)	(-7,-1)	(-5,-4)	(-8,2)
LWW	(-1,-1)	(-2,-1)	(-3,-1)	(-2,-1)	(-3,-1)	(-4,-1)	(-3,-1)	(-4,-1)	(-5,-1)	(-2,-1)
LWL	(-4,1)	(-2,-2)	(-3,-2)	(-5,1)	(-3,-2)	(-4,-2)	(-6,1)	(-4,-2)	(-5,-2)	(-5,1)
LWH	(-4,0)	(-5,0)	(-3,-3)	(-5,0)	(-6,0)	(-4,-3)	(-6,0)	(-7,0)	(-5,-3)	(-5,0)
LLW	(-4,1)	(-5,1)	(-6,1)	(-2,-2)	(-3,-2)	(-4,-2)	(-3,-2)	(-4,-2)	(-5,-2)	(-5,1)
LLL	(-7,3)	(-5,0)	(-6,0)	(-5,0)	(-3,-3)	(-4,-3)	(-6,0)	(-4,-3)	(-5,-3)	(-8,3)
LLH	(-7,2)	(-8,2)	(-6,-1)	(-5,-1)	(-6,-1)	(-4,-4)	(-6,-1)	(-7,-1)	(-5,-4)	(-8,2)
LHW	(-4,0)	(-5,0)	(-6,0)	(-5,0)	(-6,0)	(-7,0)	(-3,-3)	(-4,-3)	(-5,-3)	(-5,0)
LHL	(-7,2)	(-5,-1)	(-6,-1)	(-8,2)	(-6,-1)	(-7,-1)	(-6,-1)	(-4,-4)	(-5,-4)	(-8,2)
LHH	(-7,1)	(-8,1)	(-6,-2)	(-8,1)	(-9,1)	(-7,-2)	(-6,-2)	(-7,-2)	(-5,-5)	(-8,1)
HWW	(-4,1)	(-5,1)	(-6,1)	(-5,1)	(-6,1)	(-7,1)	(-6,1)	(-7,1)	(-8,1)	(-2,-2)
HWL	(-7,3)	(-5,0)	(-6,0)	(-8,3)	(-6,0)	(-7,0)	(-9,3)	(-7,0)	(-8,0)	(-5,0)
HWH	(-7,2)	(-8,2)	(-6,-1)	(-8,2)	(-9,2)	(-7,-1)	(-9,2)	(-10,2)	(-8,-1)	(-5,-1)
HLW	(-7,3)	(-8,3)	(-9,3)	(-5,0)	(-6,0)	(-7,0)	(-6,0)	(-7,0)	(-8,0)	(-5,0)
HLL	(-10,5)	(-8,2)	(-9,2)	(-8,2)	(-6,-1)	(-7,-1)	(-9,2)	(-7,-1)	(-8,-1)	(-8,2)
HLH	(-10,4)	(-11,4)	(-9,1)	(-8,1)	(-9,1)	(-7,-2)	(-9,1)	(-10,1)	(-8,-2)	(-8,1)
HHW	(-7,2)	(-8,2)	(-9,2)	(-8,2)	(-9,2)	(-10,2)	(-6,-1)	(-7,-1)	(-8,-1)	(-5,-1)
HHL	(-10,4)	(-8,1)	(-9,1)	(-11,4)	(-9,1)	(-10,1)	(-9,1)	(-7,-2)	(-8,-2)	(-8,1)
HHH	(-10,3)	(-11,3)	(-9,0)	(-11,3)	(-12,3)	(-10,0)	(-9,0)	(-10,0)	(-8,-3)	(-8,0)

	CWN	CWC	CNW	CNN	CNC	CCW	CCN	CCC
WWW	(-3,0)	(-4,0)	(-3,0)	(-4,0)	(-5,0)	(-4,0)	(-5,0)	(-6,0)
WWL	(-3,-1)	(-4,-1)	(-6,2)	(-4,-1)	(-5,-1)	(-7,2)	(-5,-1)	(-6,-1)
WWH	(-6,1)	(-4,-2)	(-6,1)	(-7,1)	(-5,-2)	(-7,1)	(-8,1)	(-6,-2)
WLW	(-6,2)	(-7,2)	(-3,-1)	(-4,-1)	(-5,-1)	(-4,-1)	(-5,-1)	(-6,-1)
WLL	(-6,1)	(-7,1)	(-6,1)	(-4,-2)	(-5,-2)	(-7,1)	(-5,-2)	(-6,-2)
WLH	(-9,3)	(-7,0)	(-6,0)	(-7,0)	(-5,-3)	(-7,0)	(-8,0)	(-6,-3)
WHW	(-6,1)	(-7,1)	(-6,1)	(-7,1)	(-8,1)	(-4,-2)	(-5,-2)	(-6,-2)
WHL	(-6,0)	(-7,0)	(-9,3)	(-7,0)	(-8,0)	(-7,0)	(-5,-3)	(-6,-3)
WHH	(-9,2)	(-7,-1)	(-9,2)	(-10,2)	(-8,-1)	(-7,-1)	(-8,-1)	(-6,-4)
LWW	(-3,-1)	(-4,-1)	(-3,-1)	(-4,-1)	(-5,-1)	(-4,-1)	(-5,-1)	(-6,-1)
LWL	(-3,-2)	(-4,-2)	(-6,1)	(-4,-2)	(-5,-2)	(-7,1)	(-5,-2)	(-6,-2)
LWH	(-6,0)	(-4,-3)	(-6,0)	(-7,0)	(-5,-3)	(-7,0)	(-8,0)	(-6,-3)
LLW	(-6,1)	(-7,1)	(-3,-2)	(-4,-2)	(-5,-2)	(-4,-2)	(-5,-2)	(-6,-2)
LLL	(-6,0)	(-7,0)	(-6,0)	(-4,-3)	(-5,-3)	(-7,0)	(-5,-3)	(-6,-3)
LLH	(-9,2)	(-7,-1)	(-6,-1)	(-7,-1)	(-5,-4)	(-7,-1)	(-8,-1)	(-6,-4)
LHW	(-6,0)	(-7,0)	(-6,0)	(-7,0)	(-8,0)	(-4,-3)	(-5,-3)	(-6,-3)
LHL	(-6,-1)	(-7,-1)	(-9,2)	(-7,-1)	(-8,-1)	(-7,-1)	(-5,-4)	(-6,-4)
LHH	(-9,1)	(-7,-2)	(-9,1)	(-10,1)	(-8,-2)	(-7,-2)	(-8,-2)	(-6,-5)
HWW	(-3,-2)	(-4,-2)	(-3,-2)	(-4,-2)	(-5,-2)	(-4,-2)	(-5,-2)	(-6,-2)
HWL	(-3,-3)	(-4,-3)	(-6,0)	(-4,-3)	(-5,-3)	(-7,0)	(-5,-3)	(-6,-3)
HWH	(-6,-1)	(-4,-4)	(-6,-1)	(-7,-1)	(-5,-4)	(-7,-1)	(-8,-1)	(-6,-4)
HLW	(-6,0)	(-7,0)	(-3,-3)	(-4,-3)	(-5,-3)	(-4,-3)	(-5,-3)	(-6,-3)
HLL	(-6,-1)	(-7,-1)	(-6,-1)	(-4,-4)	(-5,-4)	(-7,-1)	(-5,-4)	(-6,-4)
HLH	(-9,1)	(-7,-2)	(-6,-2)	(-7,-2)	(-5,-5)	(-7,-2)	(-8,-2)	(-6,-5)
HHW	(-6,-1)	(-7,-1)	(-6,-1)	(-7,-1)	(-8,-1)	(-4,-4)	(-5,-4)	(-6,-4)
HHL	(-6,-2)	(-7,-2)	(-9,1)	(-7,-2)	(-8,-2)	(-7,-2)	(-5,-5)	(-6,-5)
HHH	(-9,0)	(-7,-3)	(-9,0)	(-10,0)	(-8,-3)	(-7,-3)	(-8,-3)	(-6,-6)

Payout Matrix Over Three Turns

Introducing Camouflage – Dominant Strategies

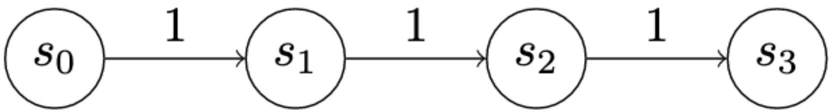
Carnegie Mellon University Software Engineering Institute

Attacker

- Always W

Defender

- HLH

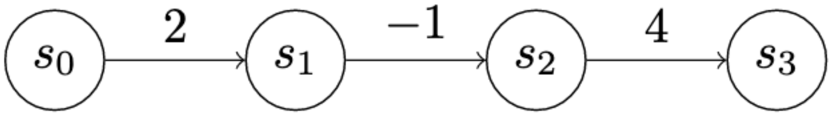


	WWW	WWN	WWC	...
WWW	(0, 0)	(−1, 0)	(−2, 0)	...
WWL	(−3, 2)	(−1, −1)	(−2, −1)	...
WWH	(−3, 1)	(−4, 1)	(−2, −2)	...
WLW	(−3, 2)	(−4, 2)	(−5, 2)	...
WLL	(−6, 4)	(−4, 1)	(−5, 1)	...
WLH	(−6, 3)	(−7, 3)	(−5, 0)	...
WHW	(−3, 1)	(−4, 1)	(−5, 1)	...
WHL	(−6, 3)	(−4, 0)	(−5, 0)	...
WHH	(−6, 2)	(−7, 2)	(−5, −1)	...
LWW	(−3, 2)	(−4, 2)	(−5, 2)	...
LWL	(−6, 4)	(−4, 1)	(−5, 1)	...
LWH	(−6, 3)	(−7, 3)	(−5, 0)	...
LLW	(−6, 4)	(−7, 4)	(−8, 4)	...
LLL	(−9, 6)	(−7, 3)	(−8, 3)	...
LLH	(−9, 5)	(−10, 5)	(−8, 2)	...
LHW	(−6, 3)	(−7, 3)	(−8, 3)	...
LHL	(−9, 5)	(−7, 2)	(−8, 2)	...
LHH	(−9, 4)	(−10, 4)	(−8, 1)	...
HWW	(−3, 1)	(−4, 1)	(−5, 1)	...
HWL	(−6, 3)	(−4, 0)	(−5, 0)	...
HWH	(−6, 2)	(−7, 2)	(−5, −1)	...
HLW	(−6, 3)	(−7, 3)	(−8, 3)	...
HLL	(−9, 5)	(−7, 2)	(−8, 2)	...
HLH	(−9, 4)	(−10, 4)	(−8, 1)	...
HHW	(−6, 2)	(−7, 2)	(−8, 2)	...
HHL	(−9, 4)	(−7, 1)	(−8, 1)	...
HHH	(−9, 3)	(−10, 3)	(−8, 0)	...

Payout Matrix Over Three Turns (detail)

More Complex Chains

- There is no dominant pure strategy for attacker or defender.
- Non-uniform chains represent more realistic attack conditions.
- Initial infection is valuable.
- Some positions of advantage may have value that justifies taking on intermediate risk.



2: WWW	2: WWA	2: WAW	2: WAA	2: AWW	2: AWA	2: AAW	2: AAA
0	$\frac{1}{3}$	$\frac{1}{3}$	0	$\frac{1}{3}$	0	0	0

1: WWW	1: WWD	1: WDW	1: WDD	1: DWW	1: DWD	1: DDW	1: DDD
$\frac{3}{11}$	$\frac{1}{33}$	$\frac{1}{33}$	0	$\frac{1}{33}$	0	0	$\frac{7}{11}$

Chain Games: Powering Autonomous Threat Hunting

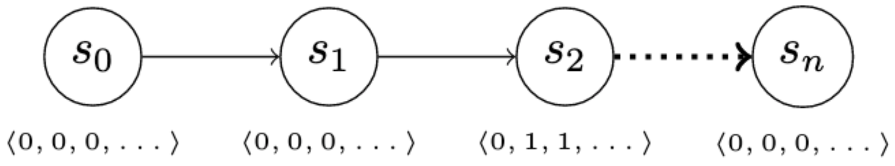
Future Work

Carnegie Mellon University Software Engineering Institute

Enriching the Game Space

Evidence

- The game is augmented with an **information vector** (IV)
  - Indicators of attacker activity
- Different kinds of attacker actions change different parts of the IV
- Defender actions collect evidence from IV
- New Defender *R*(emediate) actions stop attacker advances or evict the attacker



Simulation

Simulation is a way to model activities that are difficult to analyze exhaustively.

Simulation can model behavior that violates assumptions of rationality.

```
# Information common to all games of this type
_GAME_TYPE = pyspiel.GameType(
    short_name="chain_game_v0",
    long_name="chain game version 0",
    dynamics=pyspiel.GameType.Dynamics.SIMULTANEOUS,
    chance_mode=pyspiel.GameType.ChanceMode.DETERMINISTIC,
    information=pyspiel.GameType.Information.IMPERFECT_INFORMATION,
    utility=pyspiel.GameType.Utility.ZERO_SUM,
    # The other option here is REWARDS, which supports model-based
    # Markov decision processes. (See spiel.h)
    reward_model=pyspiel.GameType.RewardModel.TERMINAL,
    # Note again: num_players doesn't count Chance
    max_num_players=len(Players),
    min_num_players=len(Players),
    provides_information_state_string=False,
    provides_information_state_tensor=False,
    provides_observation_string=False,
    provides_observation_tensor=False,
    provides_factored_observation_string=False,
    # We can worry about parameters later
    parameter_specification={},
)
```

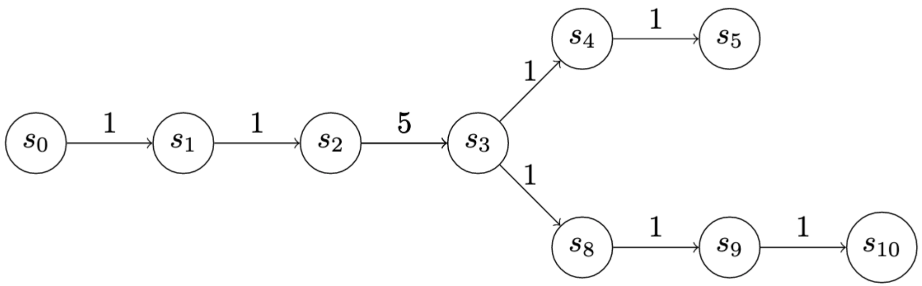
Game Specification with OpenSpiel [4]

[4] Deepmind. OpenSpiel: A Framework for Reinforcement Learning in Games. https://github.com/deepmind/open_spiel

Carnegie Mellon University Software Engineering Institute

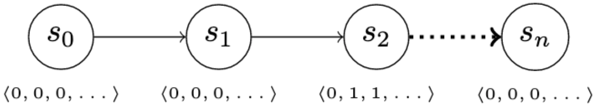
Mapping to the Problem Domain

- Reflect patterns of adversary behavior in chains
  - Distribution of positional payoffs
  - Introduce attack graphs and attacker choice



- Reflect relationships between network activities (Attacker advances) and evidence in IV

- Evaluate real-world threat hunting strategies in simulation



Chain Games: Powering Autonomous Threat Hunting

Extra Slides

Introducing Camouflage –2

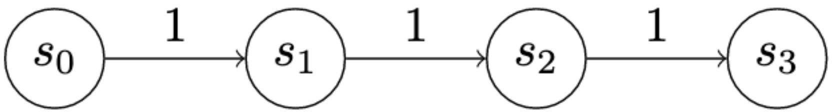
Carnegie Mellon University Software Engineering Institute

Attacker Actions

- Noisy Advance N
- Camouflaged Advance C
- C more costly than N

Defender Actions

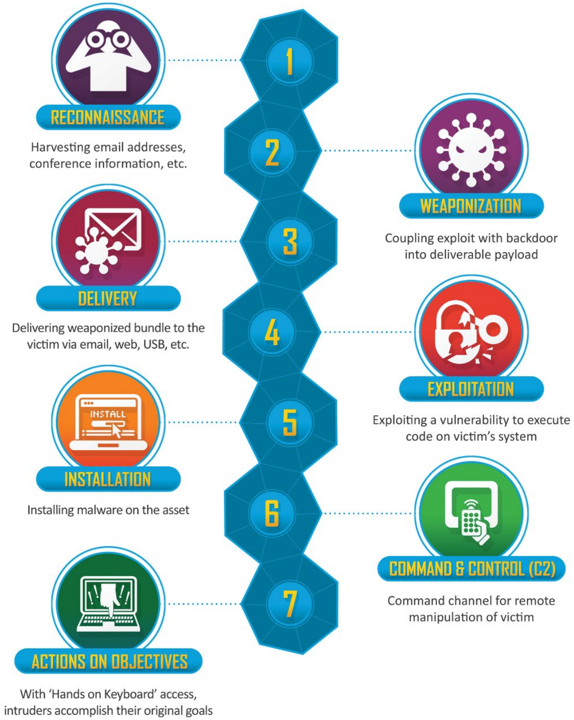
- Weak Detect L(ow), Strong Detect H(igh)
- L only detects N
- H more costly than L



	WWW	WWN	WWC	...
WWW	(0, 0)	(−1, 0)	(−2, 0)	...
WWL	(−3, 2)	(−1, −1)	(−2, −1)	...
WWH	(−3, 1)	(−4, 1)	(−2, −2)	...
WLW	(−3, 2)	(−4, 2)	(−5, 2)	...
WLL	(−6, 4)	(−4, 1)	(−5, 1)	...
WLH	(−6, 3)	(−7, 3)	(−5, 0)	...
WHW	(−3, 1)	(−4, 1)	(−5, 1)	...
WHL	(−6, 3)	(−4, 0)	(−5, 0)	...
WHH	(−6, 2)	(−7, 2)	(−5, −1)	...
LWW	(−3, 2)	(−4, 2)	(−5, 2)	...
LWL	(−6, 4)	(−4, 1)	(−5, 1)	...
LWH	(−6, 3)	(−7, 3)	(−5, 0)	...
LLW	(−6, 4)	(−7, 4)	(−8, 4)	...
LLL	(−9, 6)	(−7, 3)	(−8, 3)	...
LLH	(−9, 5)	(−10, 5)	(−8, 2)	...
LHW	(−6, 3)	(−7, 3)	(−8, 3)	...
LHL	(−9, 5)	(−7, 2)	(−8, 2)	...
LHH	(−9, 4)	(−10, 4)	(−8, 1)	...
HWW	(−3, 1)	(−4, 1)	(−5, 1)	...
HWL	(−6, 3)	(−4, 0)	(−5, 0)	...
HWH	(−6, 2)	(−7, 2)	(−5, −1)	...
HLW	(−6, 3)	(−7, 3)	(−8, 3)	...
HLL	(−9, 5)	(−7, 2)	(−8, 2)	...
HLH	(−9, 4)	(−10, 4)	(−8, 1)	...
HHW	(−6, 2)	(−7, 2)	(−8, 2)	...
HHL	(−9, 4)	(−7, 1)	(−8, 1)	...
HHH	(−9, 3)	(−10, 3)	(−8, 0)	...

Payout Matrix Over Three Turns (detail)

Kill/Attack Chains



The Cyber Kill Chain graphic is reused with permission from Lockheed Martin Corporation. [3]

Attack behavior is often conceptualized as chains.

- Decomposes attacks
- Categorizes attack behaviors

Attack graphs are a composition of attack chains.

[3] Lockheed-Martin. The Cyber Kill Chain. https://www.lockheedmartin.com/en-us/capabilities/cyber/cyber-kill-chain.html