Volatile Memory

Behavioral Game Theory in Defensive Security

Kelly Shortridge

Y’all. It’s time to dismantle some game theory.

Do you believe bug-free software is a reasonable assumption?

Do you believe wetware is more complex than software?

Traditional Game Theory relies on the assumption of bug-free wetware

Introducing: The Notorious B.G.T.
(behavioral game theory)

“Think how hard physics would be if particles could think”

—Murray Gell-Mann

“Amateurs study cryptography, professionals study economics”

—Geer quoting Allan Schiffman

Spoiler alert

Your defensive essentials:

- Belief prompting
- Decision tree modeling
- Fluctuating your infrastructure
- Tracing attacker utility models
- Falsifying information

Stop using:

- Static strategies & playbooks
- Game Theory 101
- Block & tackle
- Vuln-first approach

Here's my story why

What game is infosec?

Types of games

- Tic tac toe, poker, chess, options contracts (zero-sum)
- Stock market, Prisoner's Dilemma (non-zero-sum)
- Global nuclear warfare (negative sum)
- Trade agreements, kitten cuddling contest (positive sum)

- Complete information vs incomplete information
- Perfect information vs imperfect information
- Information symmetry vs information asymmetry

Defender Attacker Defender (DAD) games

- Sequential games in which sets of players are attackers and defenders
- Typically assumes people are risk-neutral & attackers want to be maximally harmful (inconsistent with all attacker types as well as IRL data)
- First move = defenders choosing a defensive investment plan
- Attackers observe defensive preparations & chooses attack plan

The infosec game

- A type of a DAD game (continuous defense continuous attack)
- Non-zero-sum
- Incomplete information
- Imperfect information
- Information asymmetry abounds
- Sequential
- Dynamic environment

This is a (uniquely?) tricky game.

Limitations of Game Theory

- Assumes people are rational (they aren't)
- Deviations from Nash Equilibrium are common
- Static environments vs. dynamic environments
- Impossible to ever be "one step ahead" of your competition

“I feel, personally, that the study of experimental games is the proper route of travel for finding ‘the ultimate truth’ in relation to games as played by human players”

—John Nash

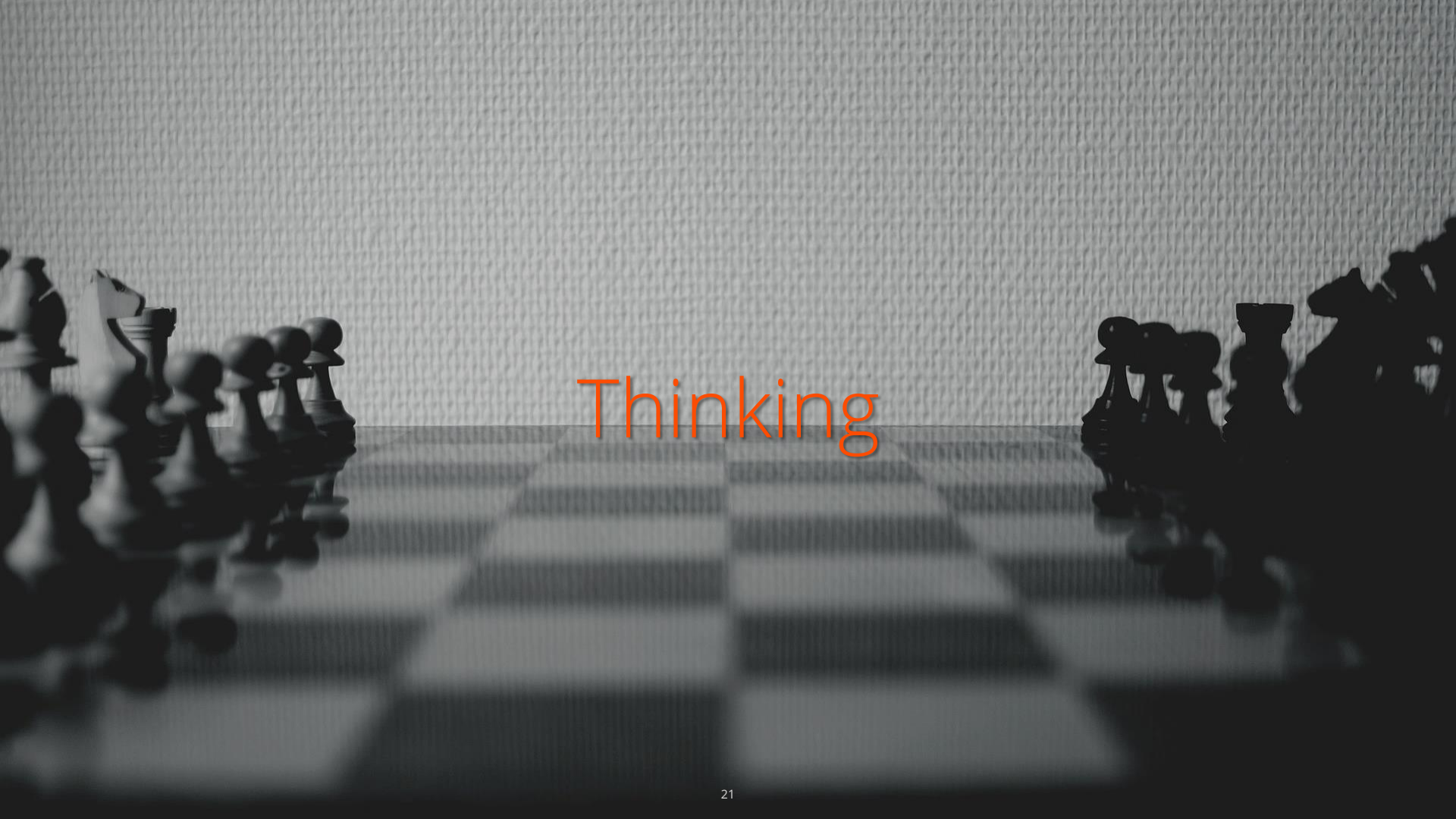
Behavioral game theory

- Experimental – examining how people actually behave in games
- People predict their opponent's moves by either "thinking" or "learning"
- Lots of models – QLK, EWA, Poisson CH, etc.
- Determining model parameters are the key challenges
- Brings in neuroscience & physiology (eyetracking, fMRI, etc)

Model parameters can guide what's important to consider

BGT models (cognitive models)

- Quantal level-k (QLK)
  - Iterated strategic reasoning & cost-proportional errors
- Subjective Utility Quantal Response (SUQR)
  - Linear combination of key features at each decision-stage
- Experienced Weighted Attraction (EWA)
  - Reinforcement learning & belief learning models
- Cognitive Hierarchy (CH)
  - Different abilities in modeling others' actions (a bit of Dunning-Kruger)


Thinking

Thinking = modeling how opponents are likely to respond

Man, like machine

- Working memory is a hard constraint for human thinking
  - e.g. Quantal level-k model incorporates “iterated strategic reasoning” – limit on levels of high-order beliefs maintained
- Enumerating steps beyond the next round is hard
- Humans aren’t that great at recursion

Thinking strategy: Belief Prompting

- “Prompt” the player to consider who their opponents are and how their opponents will act / react
- Model assumptions around capital, time, tools, risk aversion
- Goal is to ask, “if I do X, how will that change my opponent’s strategy?”

Generic belief prompting guide

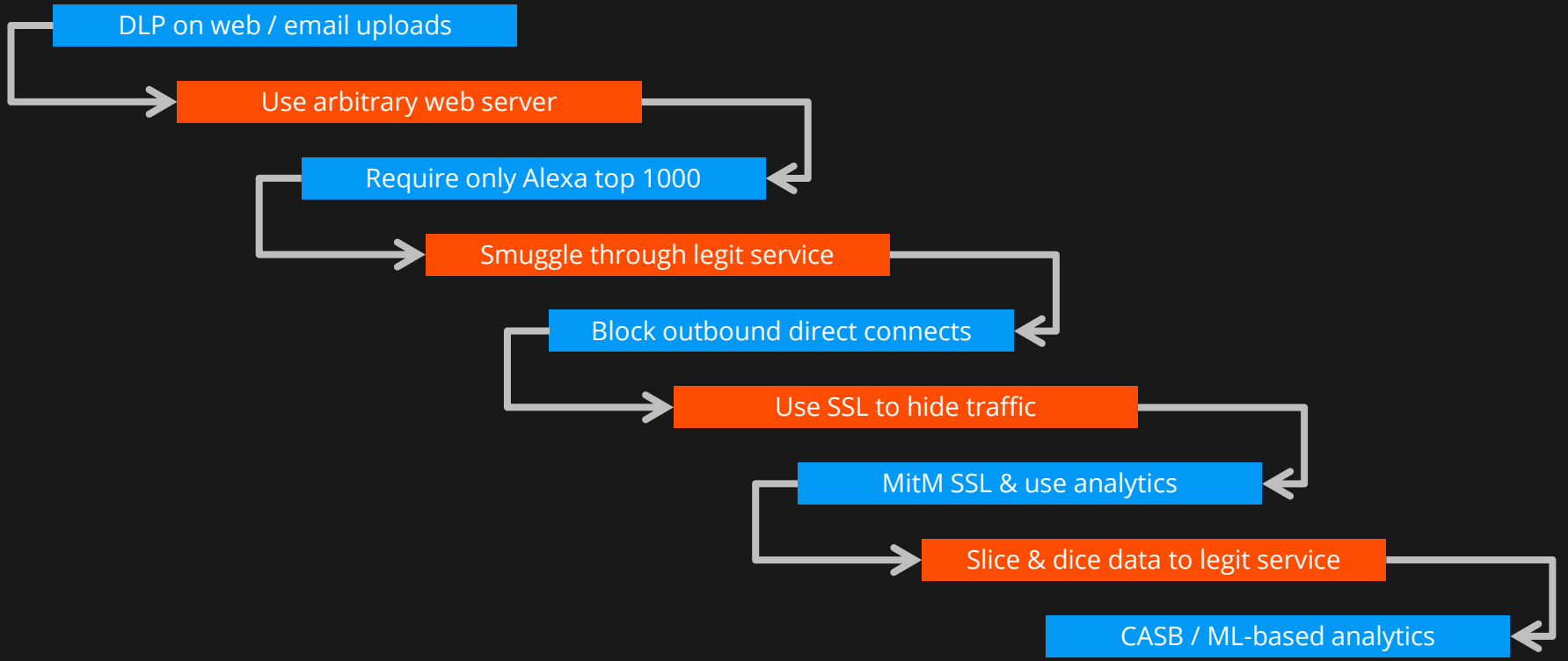
- For each move, map out:
  - How would attackers pre-emptively bypass the defensive move?
  - What will the opponent do next in response to the defensive move?
  - Costs / resources required for the opponent's offensive move?
  - Probability the opponent will conduct the offensive move?

Examples of belief prompting

- Should we use anti-virus or whitelisting?
  - Requires modifying malware to evade AV signatures
  - Adds recon step of figuring out which apps are on whitelist
  - Former is easier to implement, so attackers will prob choose it

- Skiddie lands on one of our servers, what do they do next?
  - Perform local recon, escalate to whatever privs they can get
  - Counter: priv separation, don't hardcode creds
  - Leads to: attacker must exploit server, risk = server crashes

Example progression: Exfiltration



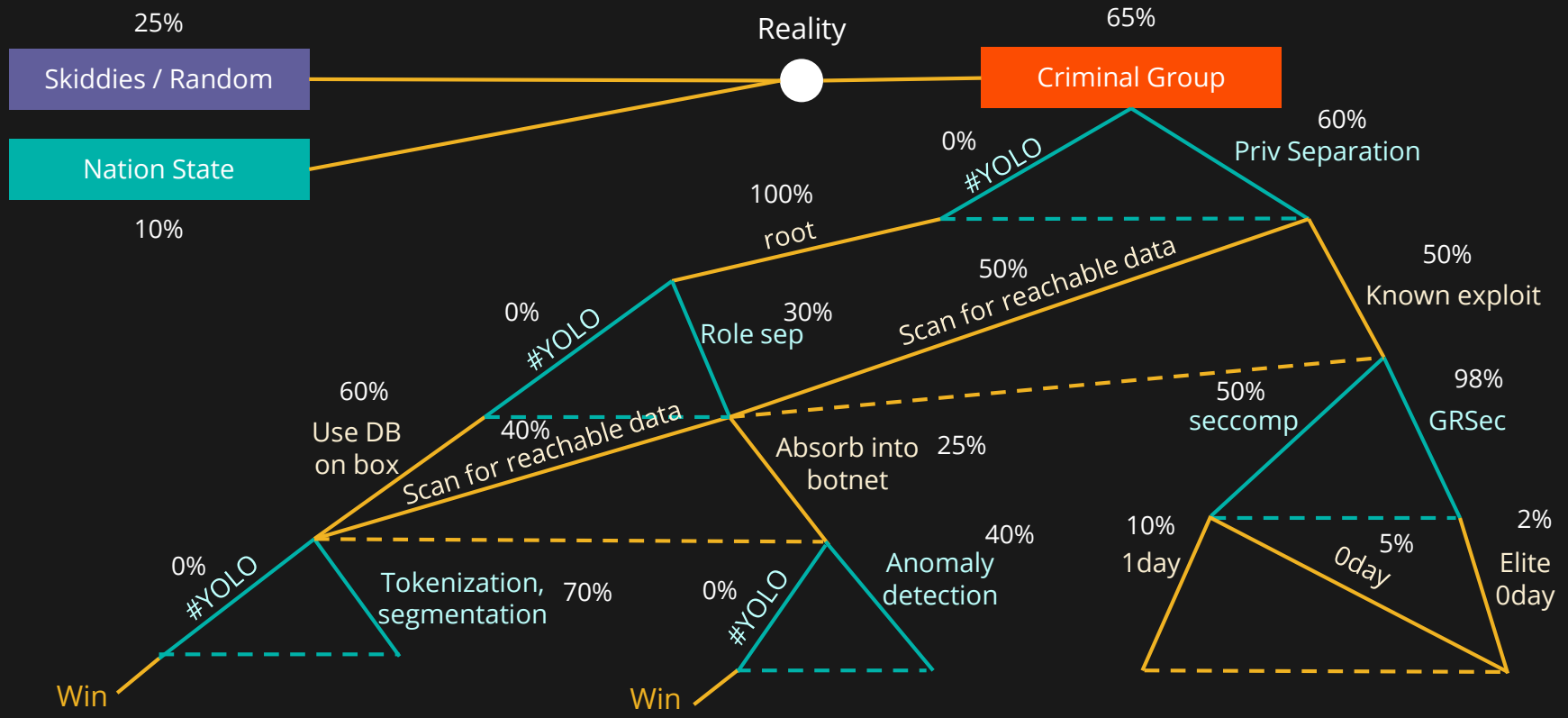
Decision tree modeling

- Model decision trees both for offense and defense
  - Use kill chain as guide for offense's process
- Theorize probabilities of each branch's outcome
  - Phishing is far more likely the delivery method than Stuxnet-style
  - Creates tangible metrics to deter self-justification

"Attackers will take the least cost path through an attack graph from their start node to their goal node"

– Dino Dai Zovi, "Attacker Math"

Example DAD tree (for illustrative purposes)



Feedback loop

- Decision trees help for auditing after an incident & easy updating
- Historical record to refine decision-making process
- Mitigates “doubling down” effect by showing where strategy failed

Decision prioritization

- Defender's advantage = they know the home turf
- Visualize the hardest path for attackers – determine your strategy around how to force them to that path
  - Remember attackers are risk averse!
- Commonalities on trees = which products / strategies mitigate the most risk across various attacks


Learning

Learning = predicting how players will act based on prior games / rounds

Human brains

- Error-driven reinforcement learning (“trial & error”)
- People have “learning rates,” how much experiences factor into decision making
- Dopamine neurons encode reward prediction errors (feedback expected minus feedback received)

Case study: Veksler & Buchler

- Fixed strategy = prevent 10% - 25% of attacks
- Game Theory strategy = prevent 50% of attacks
- Random strategy = prevent 49.6% of attacks
- Cognitive Modeling strategy = prevent between 61% - 77%

Don’t be replaced by a random SecurityStrategy™ algorithm

Account for attacker cognition

- Preferences change based on experience (learning)
- Models should incorporate the “post-decision-state”
- Higher the attacker’s learning rate, easier to predict their decisions
- Allows opportunity to manipulate adversary learning
- Fine to take a “blank slate” approach

Exploit the fact that
you understand the local environment
better than attackers

New types of exploitation

Information Asymmetry Exploitation

- Disrupt attacker learning process by sending false information
- Honeytokens (e.g. Canaries), fake directories, false infrastructure data

Learning Rate Exploitation

- Predict their decisions & cut off that attack path
- Or, fluctuating infrastructure for unreliable prior experiences

Info asymmetry exploitation: falsifying data

- Information asymmetry = defenders have info adversaries need to intercept
- Dynamic environments = frequently in learning phase
- Either hide or falsify information on the legitimate system side
- Goal is to disrupt the learning process

Learning rate exploitation: model tracing

- Change in the expected utility of offensive action = learning-rate (success/failure – action-utility)
- Keep track of utility values for each attacker action
- For detected / blocked actions, attacker action & outcome are known variables (so utility is calculable)
- Highest “U” = action attacker will pursue

Model tracing example

- $\Delta U_A = \alpha (R - U_A)$
  - α = learning rate
  - R = feedback (success / failure)
- If $\alpha = 0.2$, R = 1 for win and -1 for loss, then for a "blank slate":
  - $\Delta U_A = 0.2(1-0) = 0.2$ → 20% more likely to conduct this again
  - Adjust learning rate based on data you see

Practical tips

- Leverage decision trees again to incorporate attacker utility models (doesn't have to be exact!)
- Honeytokens / canaries are easiest to implement
- Ensure you're detecting / collecting data on attacker activity across the entire kill chain
- Fluctuating infrastructure is a bit harder (but the future?)
- Recognize that strategies will change


Conclusion

It is no longer time for some Game Theory

(It may be time for an extended Biggie metaphor)

It’s like the more money infosec comes across,
the more problems we see

Empirically, people see B.G.T. be flossin’

B-G-T D-E-I-C-A[1], no info for the PLA

1. Abbreviation of (most of) the kill chain

tl;dr

Your defensive essentials:

- Belief prompting
- Decision tree modeling
- Fluctuating your infrastructure
- Tracing attacker utility models
- Falsifying information

Stop using:

- Static strategies & playbooks
- Game Theory 101
- Block & tackle
- Vuln-first approach

“Good enough is good enough. Good enough always beats perfect.”

—Dan Geer

Further reading

- David Laibson's Behavioral Game Theory lectures @ Harvard
- Vincent Crawford's "Introduction to Behavioral Game Theory" lectures @ USCD
- "Advances in Understanding Strategic Behavior," Camerer, Ho, Chong
- "Deterrence and Risk Preferences in Sequential Attacker–Defender Games with Continuous Efforts," Payappalli, Zhuang, Jose
- "Know Your Enemy: Applying Cognitive Modeling in the Security Domain," Veksler, Buchler
- "Improving Learning and Adaptation in Security Games by Exploiting Information Asymmetry," He, Dai, Ning
- "Human Adversaries in Opportunistic Crime Security Games: Evaluating Competing Bounded Rationality Models," Abbasi, Short, Sinha, Sintov, Zhang, Tambe
- "Solving Defender-Attacker-Defender Models for Infrastructure Defense" by Alderson, Brown, Carlyle, Wood
- "Behavioral theories and the neurophysiology of reward," Schultz
- "Measuring Security," Dan Geer
- "Mo' Money Mo' Problems" by the Notorious B.I.G.

Questions?

- Twitter: @swagitda_
- LinkedIn: /kellyshortridge
- Email: kelly@greywire.net