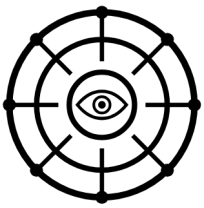


The [redacted] Collective

DEF CON CTF Proposal, 2025

Theoretical Analysis

Website https://overseers.ctf.ing
Email watchtower@overseers.ctf.ing
Social @_overseers

Contents

1 Abstract

Competitive Capture-the-Flag (CTF) formats face an inherent tension between introducing strategic complexity and preserving skill-based outcomes. This analysis presents a quantitative framework of a novel map-based CTF format that introduces spatial navigation, resource allocation, and real-time strategic decision-making alongside traditional technical challenges. We develop a mathematical framework modeling team performance through latent skill parameters, stochastic challenge difficulty, and state-transition dynamics calibrated to realistic competition conditions. Using Monte Carlo simulation across 500 independent competitions with 12 teams each, we evaluate whether the strategic layer undermines the primacy of technical skill in determining outcomes. Our results demonstrate a Pearson correlation coefficient of $r = 0.8915$ ($p < 0.0001$) between team skill ranking and final competition ranking, indicating that technical ability explains 79.5% of outcome variance. Sensitivity analyses confirm this correlation remains robust ($r > 0.87$) across substantial variations in model parameters, including challenge difficulty variance and skill-impact coefficients. These findings establish that the proposed format preserves skill as the dominant determinant of success while introducing tactical depth, providing theoretical assurance that strategic elements enhance rather than compromise competitive integrity.

2 Mathematical Framework

For our model, we use 1 minute as the base unit. All values mentioned below are in minutes unless otherwise specified.

For any challenge c, we use d_c to denote its distance from the center of the board.

Let D be the edge length of the board (240 for Days 1 & 2 and 120 for Day 3).

2.1 Skill Modeling

Each team i has a single latent **skill parameter** z_i drawn from a standard normal distribution:

$$z_i \sim \mathcal{N}(0, 1)$$

which we call the z-score of team i.

This distribution captures natural variation in team abilities, with $z = 0$ representing average skill, positive values indicating above-average teams, and negative values indicating below-average teams.

For each game simulation, we draw 12 teams' z-scores independently from this distribution. The rankings based on these z-scores represent the ground-truth skill rankings, and will be used to correlate against final competition rankings.

2.2 State-Transition Modeling

For each challenge c, we define an **expected state transition duration** T_c that represents the average time required for an average-skilled team to achieve a meaningful progression (completing a meaningful unit of work), such as when a new exploit is finished, a new bot is developed, or a bonus challenge is solved.

Intuitively T_c can be understood as the *difficulty* of the challenge c: higher T_c means the challenge is more difficult, and vice versa.

2.2.1 A/D Services & KotH Challenges

For challenge c with multiple interaction opportunities throughout their lifecycle (Attack/Defense services and King-of-the-Hill challenges), we model:

$$T_c = \frac{D - d_c + 60}{\kappa}$$

Where κ is a hyperparameter.

The numerator $D - d_c + 60$ represents the effective lifetime of a challenge, accounting for:

- The fastest time a team can reach a service c is $D - d_c$.
- The service c gets retired at time $2(D - d_c) + 60$. That is, when the shrinking boundary reaches that service.

Therefore, the service's lifecycle is roughly $[D - d_c, 2(D - d_c) + 60]$, or a total lifetime of $D - d_c + 60$.

The denominator κ can be intuitively understood as the number of expected interactions with the challenge throughout the lifetime of the challenge. For our simulation, we set $\kappa = 5$.

For example, with this formula for a challenge c at $d_c = 225$, we model that an average team would be able to have the first state transition at 15^{th} minute, and another state transition at 30^{th} minute, so on and so forth.

2.2.2 Bonus & 1v1 Speedruns

For single-solve challenges, we expect them to be of similar difficulties. Therefore, we use fixed expected state-transition times:

- Bonus Challenges: $T_c = 30$
- 1v1 Speedruns: $T_c = 15$

As indicated in our proposal.

Note that state-transition for them only happens once, which is when they are solved.

2.2.3 Realistic Deviations

To model realistic behavior of challenges, we introduce some noise to the expected state-transition times:

First, each challenge c carries an intrinsic difficulty multiplier $\epsilon_c \sim \mathcal{N}(1, 0.1)$, representing natural variation in challenge construction.

To model real-world scenarios where challenges prove unexpectedly easier or harder than designed due to having unintended solutions or being unintuitive/"nasty", we introduce a deviation hyperparameter χ, where $0 \leq \chi \leq n$ and n is the number of challenges on the game board.

For each simulation day, χ challenges are randomly selected and assigned a deviation multiplier $\delta \in \{\frac{1}{2}, 2\}$ with equal probability. For our simulation, we set $\chi = 1$.

Therefore, the *true* state-transition time for challenge c becomes:

$$\hat{T}_c = T_c \cdot \epsilon_c \cdot \delta_c$$

Where δ_c is 1 if challenge c is not selected for deviation.

2.2.4 Skill Adjustment

Finally, to model how long a team i would take to achieve a state-transition on challenge c, we apply an exponential adjustment factor based on the team's z-score z_i:

$$\hat{T}_{c,i} = \hat{T}_c \cdot e^{-\beta z_i}$$

Where β is a hyperparameter controlling how much skill impacts challenge completion time.

For our simulation, we set $\beta = \dfrac{\ln 2}{2}$. This parameterization ensures:

- A team at average skill ($z = 0$) completes transitions in average time.
- A team at $z = +2$ (95^{th} percentile) completes transitions in half the average time.
- A team at $z = -2$ (5^{th} percentile) requires double the average time.

Note that this formulation ensures that transition times follow a **log-normal distribution** across the team population.

3 Simulation Methodology

Unless otherwise specified, we use the same values as those in our proposal.

3.1 Environment Setup

The simulation implements a discrete-time system with 60 ticks per minute (1-second granularity) as indicated in our proposal. The environment maintains:

- **Challenge state**: Position, type, difficulty modifiers, retirement status
- **Team state**: Position, score, challenge progress, active speedruns
- **Global state**: Elapsed time, 1v1 cooldowns

At each tick, each team can choose one of the following actions:

- Move towards a direction (with $\frac{1}{60}$ unit of distance)
- Initiate an 1v1 challenge towards another team within vision range, if possible
- Stay put

And the code updates the environment state according to the rules outlined in our proposal.

3.1.1 Challenge Placement

For full-size maps (240 minutes), A/D Services and KotH Challenges are placed at the following positions:

Table 1: Challenge Distribution

Distance to Center	Angular Offset
225	0°
	10°
	20°
210	5°
	15°
	25°
180	0°
	10°
	20°
150	7.5°
	22.5°
120	15°

Continued on next page

Table 1: Challenge Distribution (Continued)

Distance to Center	Angular Offset
90	7.5° 22.5°
60	15°
15	15°

While for half-size maps (120 minutes), only the challenges within 120 distance (exclusive) to center are placed (4 challenges only).

For our simulation, these positions are assigned with 70% probability for A/D services.

To model for randomization on challenge placement, each challenge is moved by a random angular offset drawn from $\mathcal{U}_{[0,\pi]}$ (uniform distribution of range $[0°, 180°]$) and a random distance of $\mathcal{N}(0, 2)$ (normal distribution with mean 0 and standard deviation 2).

3.1.2 Team Placement

As outlined in our proposal, teams start at the edge of the board (at distance D) evenly spaced.

To model randomization of exploration path, all teams are offset by a random angular shift drawn from $\mathcal{U}_{[0°,2.5°]}$.

3.2 Team Strategy

Teams employ a greedy evaluation function that considers:

- **Immediate point value**: Expected points per minute from current position
- **Travel opportunity cost**: Points forfeited during movement
- **Challenge lifecycle**: Remaining time before retirement
- **Current state**: Number of state transitions happened and pending submissions

The strategy prioritizes:

- High-value challenges with long remaining lifetimes
- Challenges close to current position (minimizing travel time)
- Unvisited A/D services or KotH Challenges
- Bonus challenges with recent spawns

The evaluation function considers each action (move towards a challenge, initiate 1v1 (if possible), or stay put), assigns a utility score to each of them based on heuristics about current environment, and selects the action with the highest score.

Critically, **the strategy cannot access hidden parameters** such as team's true skill levels (z_i), exact difficulty multipliers (ϵ_c, δ_c), or opponents' internal states. Teams must use heuristics based on observable information only.

Another important point is that all teams share the same strategy evaluation function. Any systematic ranking differences therefore come from the latent skill parameters and not from over-fitting the strategy to particular teams.

For more details, please refer to our simulation code (Appendix A), specifically the `Team::evaluate` method.

3.3 Assumptions

To simplify our simulation, we make the following assumptions:

- We use a single probability value $p_i \sim \mathcal{U}_{[0,1]}$ to model team i's *tendency* to initiate/accept an 1v1 challenge. This value is drawn once at the start of the day and remains constant throughout. This value is used internally by the team's strategy.
- Two teams will engage in 1v1 only if both teams initiate an 1v1 challenge towards each other at the same tick.
- When the team reaches the vision range of a service/challenge, they will immediately commit the pending state transitions.
  - That means for example, when a team gets the solution for a Bonus challenge (a state transition), they will submit it as soon as they are within vision range of the challenge.
- No drawing or forfeit scenarios in 1v1 challenges.
- No running out of 1v1 challenges (i.e., infinite 1v1 challenges available).
- No SLA penalties for A/D services.
- No ties in KotH challenges.
- When a team i's number of committed state transitions on an A/D service is greater than another team j's, then team i can successfully attack team j on that service.
- When a team i's number of committed state transitions on an A/D service is not less than another team j's, then team i can successfully defend team j on that service.
- When a team i's number of committed state transitions on a KotH challenge is greater than another team j's, then team i's bot should rank higher than team j's bot on that challenge.
- When a team i's number of committed state transitions on a KotH challenge is equal to another team j's, then whichever team that has a greater z-score should rank higher.

We argue that these assumptions are reasonable simplifications that do not materially impact the validity of our simulation results.

3.4 Scoring

For each day, teams accumulate points as per the scoring rules outlined in our proposal. More specifically for the purpose of our simulation:

3.4.1 A/D Services

At the beginning of each round (every minute), the code checks the committed number of state-transitions $s_{c,i}$ for every team i for each A/D service c, where

- For all other teams $j \neq i$, if $s_{c,i} > s_{c,j}$, team i earns 1 Point for successful attack.
- For all other teams $j \neq i$, if $s_{c,i} \geq s_{c,j}$, team i earns 1 Point for successful defense.

3.4.2 KotH Challenges

At the beginning of each round (every minute), the code ranks all teams based on their committed number of state-transitions $s_{c,i}$ for each KotH challenge c. For teams with equal $s_{c,i}$, the team with higher z-score ranks higher as per the assumption. However, if a team has $s_{c,i} = 0$, they are tied at the bottom regardless of skill.

Points are awarded based on ranking as per the proposal: last place ($s_{c,i} = 0$) earns 0 Point, second last earns 1 Points, ...

3.4.3 Bonus Challenges

When a team i submits a state transition for a Bonus challenge c, they immediately earn 600 Points. The challenge is then considered solved and removed from the board.

3.4.4 1v1 Speedruns

When two teams i and j are engaged in a 1v1 challenge on challenge c, the team with less state transition time for that challenge will win and earn 300 Points. The losing team earns 0 Points.

At the end of Day 3, team scores are aggregated with Day 3 scores doubled to determine final rankings.

4 Statistical Analysis

We run 500 independent simulations of the full 3-day competition as described above. For each simulation, we record each team i's final score at the end of Day 3 and its z-score z_i.

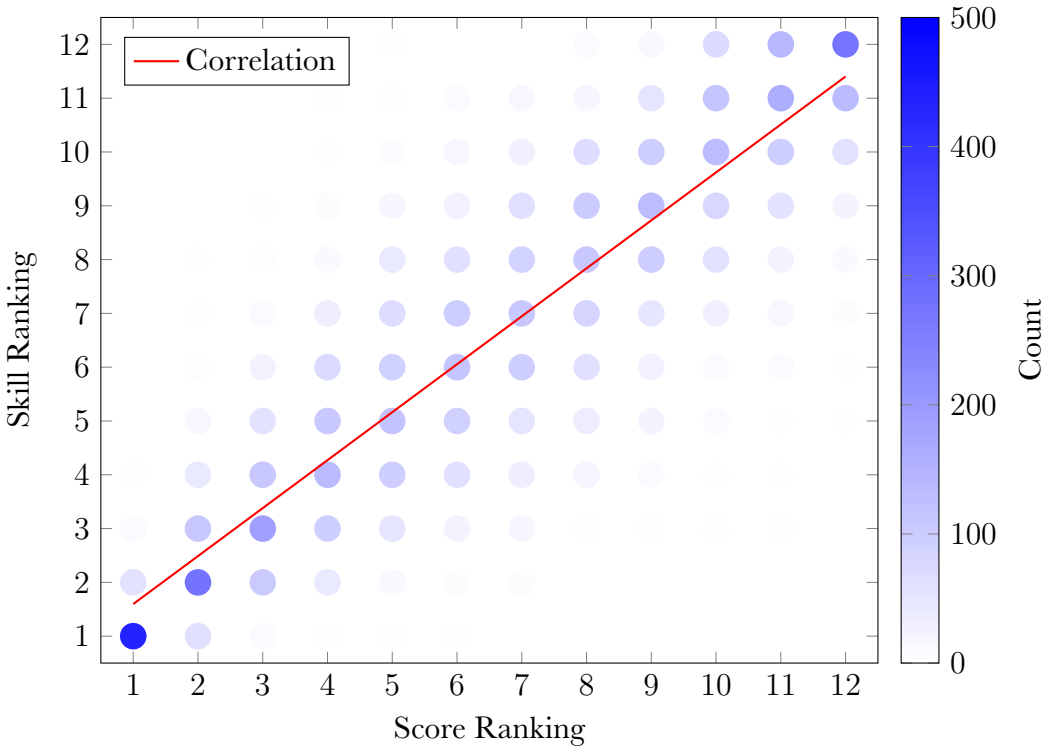
We then generate a score ranking of teams for each simulation based on final scores, and another skill ranking based on teams' z-scores. These rankings are integer values ranging from 1 (best) to 12 (worst).

These rankings are then aggregated across all 500 simulations to produce a distribution of (score ranking, skill ranking) pairs.

4.1 Primary Results

The following figure visualizes this distribution as a scatter plot, where the color intensity of each point indicates the frequency of that (score ranking, skill ranking) pair occurring across all simulations.

Figure 1: Score-Skill Distribution



4.2 Interpretation

The correlation between Score Ranking and Skill Ranking across all simulations yields:

$$r = 0.8915 \quad (p < 0.0001)$$

This Pearson correlation coefficient indicates that 79.5% of the variance in final rankings is explained by underlying team skill ($r^2 = 0.795$).

Notably, the highest-skilled team (Skill Rank 1) achieved the highest final score (Score Rank 1) in 429 out of 500 simulations, or 85.8% of the time. This provides strong evidence that the format rewards technical excellence: the best team wins the vast majority of competitions.

The observed correlation strength ($r = 0.8915$) places our format firmly in the "very strong correlation" category by conventional standards.

Also, because the team strategy is simple and identical for all teams, it likely *underestimates* the advantage that genuinely stronger strategic play would yield; the correlation results should be read as a lower bound on how much skill can shine through.

The remaining 20.5% of unexplained variance arises from stochastic elements inherent to the simulation: intrinsic challenge difficulty variation (ϵ_c), unexpected difficulty deviations (δ_c), and 1v1 encounter probabilities (p_i). These effects are amplified when teams have similar z-scores, as small performance differences become magnified in rank comparisons due to low signal-to-noise ratio, causing occasional rank inversions.

This can be observed in the scatter plot, where the top-right and bottom-left corners are densely populated (indicating stable rankings for elite and weak teams), while the center region shows more spread among similarly-skilled competitors.

As a comparison, here are existing researches on how well pre-competition team rankings line up with competition outcomes in popular sports:

- Soccer, FIFA ranking vs World Cup results: Pearson $r = 0.425$ (Mens), $r = 0.683$ (Womens).[1]
- Soccer, UEFA ranking vs Euro 2020 final placement: Pearson $r = 0.48$.[2]
- NCAA Basketball, Elo rating rank vs Committee seed: Spearman $\rho = 0.89$–0.94.[3]

We also considered doing correlation analysis on past DEF CON CTF results using CTFtime.org rankings (as it is the only publicly available ranking source with historical data). Unfortunately, due to how CTFtime.org rankings are calculated, more active teams tend to have better rankings. Due to the "team merging for DEF CON CTF" phenomenon, many of the finalist teams have only participated in DEF CON CTF, leading to a lack of data points and bias in rankings. Aggregating rankings of their underlying sub-teams (teams before merging) might be possible, but the exact composition of merger teams is unknown, and the aggregation method is also unclear. Therefore, we decided not to pursue this direction further.

4.3 Robustness Checks

To ensure the robustness of our findings, we conducted sensitivity analyses by varying key hyperparameters in our model.

[1] Brandon JOLY, Tom STOJSAVLJEVİC, and Mehmet DİK. "FIFA/Coca-Cola World Rankings on the Predictability of the Men's and Women's FIFA World Cup: A Comparative Analysis". In: *Proceedings of International Mathematical Sciences* 4 (Aug. 2022). DOI: 10.47086/pims.1153373.

[2] Adam Metelski et al. "How the value of football players influences a team's chances of victory -a Euro 2020 example". In: *Journal of Physical Education and Sport* 22 (Jan. 2022), pp. 167–173. DOI: 10.7752/jpes.2022.01021.

[3] Neil Paine. *Where are all the obvious NCAA upsets?* Mar. 2019. URL: https://fivethirtyeight.com/features/where-are-all-the-obvious-ncaa-upsets.

For each hyperparameter variation, we keep all other parameters fixed to their baseline values and rerun the simulations, recalculating the Pearson correlation coefficient between Score Ranking and Skill Ranking.

4.3.1 Sensitivity to Interaction Constant

We tested $\kappa \in \{3, 5, 7, 10\}$:

Table 2: Sensitivity to κ

κ	Correlation (r)	p-value
3	0.9008	< 0.0001
5	0.8915	< 0.0001
7	0.9092	< 0.0001
10	0.9102	< 0.0001

We observed that variation in κ does not significantly affect the correlation. This is expected, as κ uniformly scales challenge lifetimes without altering relative difficulties.

4.3.2 Sensitivity to Skill Impact

We tested $\beta \in \{0.2, \frac{\ln 2}{2}, 0.5\}$:

Table 3: Sensitivity to β

β	T_c Multiplier at $z_i = 2$	Correlation (r)	p-value
0.2	$\times 0.67$	0.8741	< 0.0001
$\frac{\ln 2}{2}$	$\times 0.5$	0.8915	< 0.0001
0.5	$\times 0.37$	0.9020	< 0.0001

As expected, stronger skill impact increases outcome determinism. Our chosen $\beta = \frac{\ln 2}{2}$ represents a moderate assumption where elite teams $z = +2$ work at double efficiency.

4.3.3 Sensitivity to Unexpected Deviations

We tested $\chi \in \{0, 1, 3, 5\}$:

Table 4: Sensitivity to χ

χ	% of Deviated Challenges	Correlation (r)	p-value
0	0%	0.8934	< 0.0001

Continued on next page

Table 4: Sensitivity to χ (Continued)

χ	**% of Deviated Challenges**	**Correlation (r)**	***p*-value**
1	8.3%	0.8915	< 0.0001
3	25%	0.8868	< 0.0001
5	38.9%	0.8820	< 0.0001

As expected, increasing the proportion of unexpected challenge difficulty deviations slightly decreases the correlation between skill and final rankings. However, even with nearly 40% of challenges deviating, the correlation remains very strong. This suggests that while randomness affects individual matches, it does not undermine skill as the primary success factor.

5 Conclusion

Our theoretical analysis establishes three key findings:

1. **Skill Dominance**: With $r = 0.8915$, final competition rankings are overwhelmingly determined by team technical ability. The strategic layer introduces tactical depth without displacing skill as the primary performance determinant.
2. **Bounded Randomness**: The natural variance in outcomes for similarly skilled teams introduces healthy competitive uncertainty. Similar to other CTF formats, this variance is enough to maintain excitement and reward adaptability, but insufficient to allow weak teams to outperform strong ones through luck alone.
3. **Format Robustness**: Sensitivity analysis demonstrates that the skill-ranking correlation remains strong across a wide range of parameter choices, including scenarios with substantial challenge difficulty variance.

The mathematical framework and empirical results presented here provide quantitative assurance that our proposed format preserves the fundamental principle that the best teams win while introducing novel strategic and spectator-engagement dimensions to the competition.

Appendix A Code & Data

The full simulation code and analysis data is available at:
`https://gist.github.com/superfashi/2cb6fa2e454557267d4082da5cb79503`

The repository contains the following files:

- `data.py`: Main simulator code. Contains implementations for the game environment, teams, challenges, and the core simulation loop.
- `data_ext.sh`: Shell script to run batch simulations (default 500 runs) and export results into `output.jsonl`.
- `data_vis.py`: Collect result from `output.jsonl`, analyze the data, and generate visualizations such as the score-skill distribution plot.
- `output.jsonl`: Output file containing simulation results from 500 runs used in this analysis.

The simulation comes with a UI to visualize individual game runs for verification purposes. It can also be run in headless mode for batch simulations as used in this analysis. To run the headless mode, use `python data.py --headless`.

To run the simulation, we require Python packages `numpy` for numerical computations and `arcade` for UI rendering.

To run the analysis, we require Python packages `scipy` for statistical calculations and `matplotlib` for visualization.