Studying Personality and Attacker Behavior in a Deceptive Multi-Stage Capture-the-Flag Environment

Khalid Alasiri
School of Computing and Augmented Intelligence
Arizona State University
Tempe, Arizona, USA
kalasir1@asu.edu

Rakibul Hasan
School of Computing and Augmented Intelligence
Arizona State University
Tempe, Arizona, USA
Rakibul.Hasan@asu.edu

***Abstract*—Current approaches to prevent attacks by advanced persistent threat actors rely on detecting attacks and reactively respond to them. This has shown to generate an unsustainable number of false alarms. Recently, researchers have been increasingly advocating for a proactive defense paradigm, and deception-based defense has emerged as the most promising approach. This approach, however, is fundamentally human-centric, which requires profiling and understanding the attackers, predicting their next step, and proactively preparing for that.**

Unfortunately, studies investigating how well attack behaviors can be profiled has been almost non-existent, as they require mimicking real world cost-benefit structures within the experimental setup and the participation of skilled cyber professionals who are hard to recruit. To fill this gap, we present a realistic experimental testbed, and results from a pilot (N=5) and a main (N=20) study. The experimental setup involves a multi-stage Linux Capture-the-Flag environment with embedded deception. It includes several real and fake vulnerabilities including password cracking, SQL injection, and exploitable process and cron jobs. During the study, we log all activities from the participants and collect screen recordings.

With automated and manual analyses of these data, we find that deceptive signaling a promising defensive tactic that can influence the attackers' strategy. We also identify several promising correlations between behaviors and traits that can guide future studies, but many expected correlations were not observed, possibly due to several confounding factors and experiment design limitations. Based on these observations, We provide several design guidelines for future studies. Thus, this exploratory study lays the foundation of future empirical research in deception-based cyber defense.

***Index Terms*—attacker profiling, cyber deception, Capture the Flag, behavioral analytics, personality traits, human factors security**

I. Introduction

Cyber defense has traditionally been reactive, primarily focusing on patching vulnerabilities and investigating incidents after they have occurred. Recently, however, researchers have increasingly focused on proactive defense strategies to anticipate and mitigate threats before they materialize [1], [2]. However, proactive defense strategies are fundamentally human-centric; it relies on understanding the attackers, identifying patterns in their decision-making and attack strategies, predicting their next moves and proactively preparing for them at runtime [1], [2].

One of the most popular proactive techniques is the use of deception, which can misled attackers and provide the defenders an opportunity to observe the attacker's behaviors and time to prepare for the attack [2]–[4]. However, almost all of the studies that design or evaluate deception-based defense techniques, such as deceptive signaling, have been simulation-based (e.g., [2], [4]). Realistic experiments with human attackers have been scarce (except for [2]). This is because participants need to be highly trained for a study that mimics the open and complex system akin to the real world attacks. Recruiting skilled cyber professionals is notoriously difficult; even when they are accessible, such experiments are prohibitively expensive. A second major barrier is creating an experimental setup that mimics the real world incentive structure. In addition to the skill levels and availability of tools, attackers' behaviors depend on how motivated they are, the expected reward from a successful attack, and the loss they will incur if it fails—which ranges from simply being blocked to being designated as a terrorist if identity is leaked. Replicating such a cost-benefit structure in an academic study is challenging. Consequently, there remains a large gap in the literature for empirical studies involving human attackers to investigate the extend their behavioral patterns can be determined and leveraged in deception-based defense.

To fill this gap, we report findings from an exploratory study involving human attackers to understand the extent to which behavioral trends can be linked to personality traits. We focus on traits as they have been established as a reliable predictor of human behaviors, particularly in decision-making under uncertainty [5]. Traits such as resilience and risk tolerance play a substantial role in how attackers select a target, respond to failure, and persist through obstacles [6], [7]. While some prior studies (e.g., [1], [2]) have suggested the cognitive and decision-making under uncertainty aspects of cyber attackers, a systematic attempt to link personality traits to behaviors, and using this link for proactive cyber defense is missing.

Concretely, we conduct an experiment where participants

(N=20) with varying level of cybersecurity expertise were invited to attack a simulated environment that mimicked a real world corporate network. This study was preceded by a pilot study (N=5) that included expert cyber security professionals. During the two hours long experiment, all activities of the participants were logged, along with their screen recordings. We analyzed this data to identify trends in their behaviors and correlate them with personality traits.

Result show that, deceptive signaling, a powerful defense strategy that has been studied in simulated attack contexts [1], also influence the decision-making of human attackers. Additionally, having fake vulnerabilities can help deter attackers by wasting their time, cognitive efforts, and other resources. Correlations between behavioral trends and personality traits were mixed, and sometime unexpected. We largely attribute these results to confounding factors and design limitations. Based on these findings, we discuss their implication for deception-based proactive cyber defense, and provide design guidelines for future studies.

II. Related Work

Attacker profiling has been informed by personality and motivation research. Canudo et al. [8] applied self-control theory and personality models, including the Big Five and Dark Triad traits (Machiavellianism, narcissism, and psychopathy), to differentiate hacker types. They found that black hats exhibit lower self-control and higher openness, along with stronger sensation-seeking motivations, while white hats tend toward prosocial motivations. Grey hats displayed a blend of these traits, suggesting fluid identities. Relatedly, Hani et al. [5] employed machine learning to classify hackers based on Big Five traits, achieving high predictive accuracy and revealing intra-group distinctions. Gaia et al. [9] examined how Dark Triad traits, opposition to authority, and thrill-seeking drive hacking propensities, offering a multifaceted view of hacker motivations.

Resilience and persistence have been recognized as important traits in cybersecurity contexts. Joinson et al. [6] introduced the Human Cyber-Resilience Scale, measuring an individual's capacity to resist, recover from, and learn following cyber incidents. The PERC (Persistence, Effort, Resilience, and Challenge-Seeking) task developed by Porter et al. [7] provides a performance-based, language-independent method to quantify persistence, effort, resilience, and challenge-seeking. These constructs are relevant to CTF scenarios where attackers may need to reattempt failed paths, adapt to deceptive obstacles, and sustain engagement over multiple stages.

In parallel, researchers have explored structured behavioral datasets to study adversarial operations. Tovarnák et al. [10] released a comprehensive dataset from a two-day cyber defense exercise on the KYPO Cyber Range Platform, including synchronized network flows and system logs from enterprise-like environments. Such datasets enable fine-grained analysis of attacker behaviors, though they often lack the integrated psychological dimensions that the present study incorporates.

Other work has focused on integrating psychological profiling with operational cybersecurity scenarios. Padilla et al. [11] proposed a platform-agnostic experimental methodology to capture real-time human decisions and actions during cybersecurity exercises, bridging the gap between abstract trait profiling and observed attacker behavior. Tshimula et al. [12] explored psycholinguistic analysis with large language models to derive attacker psychological profiles from textual outputs, complementing behavioral profiling with linguistic signals.

Deceptive techniques have been a core strategy in cybersecurity for decades, used to mislead adversaries, delay attacks, and gather intelligence. Early foundational work such as Spitzner's *Honeypots: Tracking Hackers* [13] established the role of honeypots as a means to attract and observe malicious actors in controlled environments. Rowe and Rrushi's *Introduction to Cyberdeception* [14] further expanded this field by formalizing concepts, taxonomies, and implementation strategies for deception in cyber operations, framing it as an interdisciplinary domain involving technical, cognitive, and strategic considerations.

Building on these foundations, Cranford et al. [1] developed a cognitive theory of cyber deception within the ACT-R (Adaptive Control of Thought–Rational) cognitive architecture, using instance-based learning to model attacker decision-making under deceptive signaling. Their behavioral experiments with the Insider Attack Game demonstrated that deception can influence attacker choices by exploiting biases such as confirmation bias, underscoring the importance of considering bounded rationality in cyber defense. This cognitive modeling perspective is directly relevant to environments like multi-stage CTFs, where deception can be systematically embedded to elicit measurable behavioral patterns.

Ferguson-Walter et al. [2] advanced empirical research on deception with the Tularosa Study, a large-scale controlled experiment involving over 130 professional red teamers. By integrating technical deception methods with psychological profiling, cognitive testing, and telemetry collection, they quantified how deception affects attacker strategies, persistence, and decision-making. The combination of behavioral and psychological measures in that work provides a methodological precedent for profiling attackers in CTF environments enriched with deceptive cues.

In summary, prior work has laid foundations in cyber deception, personality-based attacker profiling, and cyber-range data collection; our contribution is to integrate these strands in a trait-informed CTF environment.

III. Methodology

A. Overview

This experiment investigates how individual psychological traits shape attacker behaviors during a controlled multi-stage Capture-the-Flag (CTF) style privilege-escalation challenge. We collect pre-challenge personality measures and detailed behavioral traces (command logs, task choices, and timing information) as participants interact with a mixture of genuine

and deceptive attack paths. The overarching goal is to examine whether specific, observable behaviors, such as repeated retries, strategic pivots after failure, or early engagement with high-risk opportunities—can serve as reliable indicators of underlying personality traits relevant to cyber offense.

B. Selecting Personality Traits

In total, we measure five personality traits: Conscientiousness and Negative Emotionality (Neuroticism) from the five Big Five dimensions using the BFI-2-S [20], domain-general risk propensity (GRiPS) [19], persistence [16], and resilience [18]. The big five traits have been linked to decision-making in cyber security and other contexts by numerous studies [21]–[23]; they also correlate to characteristics that are relevant to cyber attacks: fluid intelligence, problem-solving ability, and susceptibility to deception [24]–[27]. Likewise, persistence and resilience are essential qualities to succeed in cyberattacks [2], [28]. Finally, we include risk propensity that determine engagement in risky behaviors, cyber attack being a prominent one, and can facilitate developing deception-based defense mechanisms. The task design includes behavioral probes that reflect these traits, such as repeated attempts on deceptive paths, pivots after failure, early engagement with high-value targets, and exploration across tools and users. Table I summarizes how these traits are measured and how we expect it to manifest in the logged behaviors.

C. Experimental Infrastructure

Our study setup has been implemented as an isolated research environment that supports repeatable, fine-grained behavioral logging. Participants accessed the challenge through a browser-based workspace that provided an interactive shell connected to a dedicated Linux container preloaded with common security tools. Each container is ephemeral and network-isolated from real systems. All shell commands, file accesses, and relevant system events are collected via a logging stack and stored with timestamps for later analysis. Screen and audio recordings from the remote session are captured in parallel to provide context for interpreting behavioral traces (e.g., verbal reasoning, visible search activity). This infrastructure allows us to align psychometric measures, in-game actions, and think-aloud protocols on a shared timeline.

D. Challenge Tasks

1) Environment Overview: Our experimental setup simulates an internal corporate environment with multiple user accounts at different privilege levels (e.g., entry-level employee, IT staff, financial manager, system administrator), realistic business assets (e.g., reports, credentials, configuration files), and several potential escalation paths. Participants start from a low-privilege account and are instructed to "break out of the box" by discovering assets, escalating privileges, and collecting as many flags as possible. Each flag contributes to a cumulative score, while certain actions are treated as being "seen" by a monitoring system (e.g., triggering a logging alert). The task instructions frame some paths as higher-value, higher-effort, or more likely to attract attention, but the analysis does not rely on an in-game point-penalty system. Compared to typical CTF challenges, the environment resembles an internal assessment setting with realistic misconfigurations and overlapping avenues for attack rather than a series of isolated puzzles.

Within this environment we embed both true vulnerabilities that could be exploited to access assets and escalate the privilege, as well as carefully designed decoy tasks intended to elicit differences in persistence, resilience, risk-taking, and openness to experience.

2) Design Principles: To meet the above stated goals, the environment was constructed around three principles:

- **Realism.** Tasks are framed as plausible corporate misconfigurations (e.g., backup archives, internal web apps, scheduled jobs) rather than abstract puzzles, to encourage naturalistic strategies.
- **Multiple viable options.** At any point, participants can choose among several potential avenues (password cracking, web exploitation, privilege escalation, etc.), enabling us to observe differences in task selection, switching, and abandonment.
- **Embedded deception.** Some ostensibly promising vectors are intentionally difficult or impossible to complete within the allotted time. These "trap" tasks are designed as behavioral probes for our focal traits (e.g., persistence in the face of repeated failure, willingness to pursue high-risk opportunities, or openness to exploring alternative explanations).

The following task descriptions therefore emphasize the behavioral probes and hypothesized indicators rather than technical exploit details.

3) Task-Level behavioral Probes: As stated above, our experimental setup allows multiple paths for the participants to explore. Each of them have various 'Tasks' they need to complete, each task presents participants with a plausible privilege-escalation opportunity accompanied by cues (e.g., through names of the file) that are realistic in the context of penetration testing of a corporate network. The underlying technical configuration is tuned so that the tasks differ in difficulty and in how rewarding or deceptive they are, allowing us to observe how participants react to progress, failure, and uncertainty. The instructions set we have prepared for participants explicitly mentions the risk-reward tradeoffs: successfully accessing potentially higher valued assets (such as credential files or higher privileged user accounts) will provide higher rewards (more points), but those assets are more likely to be monitored for unauthorized accesses, increasing the chance of getting caught.

Below, we summarise the main tasks in terms of their narrative framing and the behaviors they are intended to elicit. For each task, we define a small set of log-based indicators (e.g., number of retries, switches to alternative strategies, or timing of high-risk actions) that form the basis of our hypotheses about persistence, resilience, risk-taking, and openness.

TABLE I
FOCAL PERSONALITY TRAITS, MEASUREMENT TOOLS, AND EXPECTED BEHAVIORAL INDICATORS IN THE EXPERIMENTAL SETUP.

Trait	Definition	Measurement	Expected behavioral Indicators
Persistence	Sustained effort toward goals despite difficulties [15], [16]	Motivational Persistence Scale (MPS-16) [16]	High counts of repeated attempts on the same task or exploit path; longer time spent before abandoning a deceptive task.
Resilience	Capacity to adapt positively and recover from setbacks [17], [18]	Brief Resilience Scale [18]	Switching to alternative strategies shortly after failures; higher ratio of "fail → new strategy" vs. "fail → stop" behaviors.
Risk-Taking	Willingness to engage in uncertain or high-stakes actions [19]	General Risk Propensity Scale (GRiPS) [19]	Earlier and more frequent initiation of risky actions (e.g., privilege escalation attempts, aggressive scans, or use of high-value accounts).
Conscientiousness	Tendency toward organization, deliberation, and goal-directed behavior [20]	Big Five Inventory–2 Short Form (BFI-2-S) (Conscientiousness domain) [20]	More structured task progression; fewer unsupported pivots; more consistent follow-through after selecting a path.
Negative Emotionality (Neuroticism)	Tendency toward anxiety, frustration, and emotional reactivity under stress [20]	Big Five Inventory–2 Short Form (BFI-2-S) (Negative Emotionality domain) [20]	Longer pauses after failure; earlier abandonment of difficult paths; more frequent stopping or switching under uncertainty.

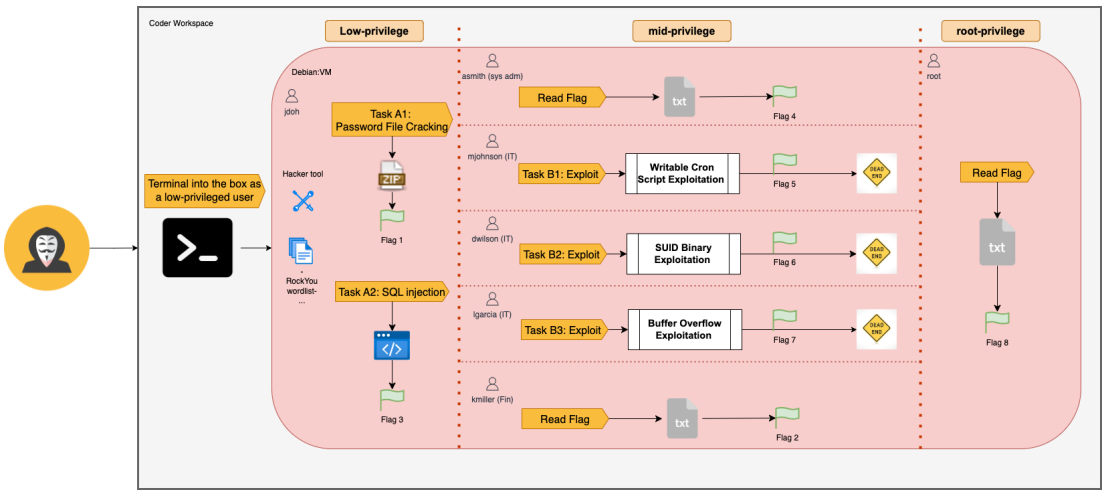

Fig. 1. The network structure, assets, and (real and fake) vulnerabilities in the system that attackers can exploit. Dead-end icons mark decoy exploit paths that reveal a flag but do not provide additional privileges or access to new assets.

a) Task A1: Password Cracking.: Participants discover an encrypted archive labeled as a corporate credentials backup in a low-privilege user's home directory. Supporting artefacts (onboarding notes, internal documentation) suggest that it may contain valuable information, and that standard password-cracking tools could be used to access it. In reality, the archive is protected by a long, cryptographically random password that does not appear in any provided or common wordlists, making practical cracking within the session time infeasible and turning it into a deceptive "time sink".

Trait measures. Risk-taking (willingness to invest in a high-reward but uncertain path) and Persistence (time and number of retries spent on an ultimately uncrackable archive).

b) Task A2: SQL Injection and Database Exploration (Risk-Taking and Persistence).: A web application exposed within the environment contains a login form that can be bypassed using SQL injection, granting access to an internal employee database. Once inside, participants can enumerate accounts, extract password hashes, and choose which user accounts to target.

Trait measures. Risk-taking (pursuing high-value but risky accounts, attempting risky actions such as modifying records), Persistence (iterative refinement of payloads and cracking strategies).

c) Task B1: An (apparently) exploitable Cron Script (Persistence vs. Exploratory Resilience).: Participants will find a root-owned scheduled backup script that appears to be a classic privilege-escalation vector: the script is writable and executed periodically by a privileged user. However, we have another process, hidden from the attacker, that monitors any edit to the script and automatically restores it to its original state, thus, preventing the expected escalation but leaving a trace in the logs.

Trait measures. Persistence and Resilience(repeated overwriting of the script despite the lack of success), Openness to experience / exploratory behavior (stepping back to investigate

the hidden mechanism, e.g., inspecting running processes or related files; knowing when to abandon the path).

d) Task B2: SUID Binary Honeypot.: A SUID binary named *vim.basic*, which is owned by a high-privileged user, presents a potential opportunity for privilege escalation. Its name and basic functionality, which can be explored by executing and interacting with it, suggest that it might be exploitable. However, any attempts to misuse the binary are detected through input validation and checks for injection attempts. Instead of allowing actual privilege escalation, these attempts are logged.

Trait measures. Risk-taking: risk-seeking participants execute the binary quickly and experiment with potentially dangerous arguments, whereas others spend more time on reconnaissance (e.g., inspecting permissions or documentation) before executing it, or defer it until later.

e) Task B3: An Apparent Buffer Overflow Vulnerability.: Participants encounter a privileged program that behaves like a simple note-editing tool (e.g., opening and appending text), and basic inspection reveals cues such as references to *strcpy()* that typically indicate unsafe memory handling. These signals make the binary appear vulnerable to a classic buffer overflow. However, the program includes a hidden length check that terminates execution before any unsafe function is reached, meaning the apparent vulnerability cannot actually be exploited despite the convincing surface indicators.

Trait measures. This task probes how participants respond when a textbook exploit path fails, contrasting high persistence (sustained debugging and multiple exploit attempts) with openness to experience (willingness to reconsider assumptions, check compilation flags, and pivot to alternative explanations or attack vectors).

Progression scope: The environment contains two broad progression levels. Level One contains the initial A-tasks, where participants choose between the archive/password path and the SQL-injection/database path. Level Two contains the later B-task probes that become visible after lateral movement into another user account. We intentionally did not add additional privilege levels beyond these stages because the study was designed for a bounded human-subjects session with a two-hour minimum, whereas real-world intrusions may unfold over days or weeks. The B-tasks were therefore designed as interaction probes rather than complete escalation chains: each appeared to offer a plausible route toward higher privilege, but none provided confirmed root access. This design allowed participants to inspect, attempt, abandon, and switch among later-stage tasks, producing behavioral traces without requiring full compromise during the session.

E. Data Sources and Planned Analysis

Each session yields three data streams: (1) system logs with time-stamped shell commands, file accesses, and relevant system events; (2) self-report measures from the pre-challenge personality inventories (MPS-16, BRS, GRiPS, BFI-2-S) and a progression survey on strategies, reactions to deception, and persistence / pivot decisions; and (3) think-aloud screen-and-audio recordings that provide qualitative context for interpreting log-derived behaviors.

From the logs we derive behavioral metrics aligned with our focal traits, such as number and duration of retries on deceptive tasks (persistence), frequency of "fail $\rightarrow$ new strategy" transitions (resilience), time to first high-risk action (risk-taking), and diversity of tools and paths explored (openness). These metrics are synchronised with personality scores and post-challenge responses to examine correlations between traits and behaviors, complemented by exploratory mixed-methods inspection of notable outlier cases using the think-aloud data.

F. Pilot Study

Before the main data collection, we ran five pilot sessions that were not included in the final 20-session analysis dataset. These sessions used a more open-ended version of the study: participants received the challenge briefing, worked in the environment, and completed the post-challenge questions, but they were not provided with the structured in-session progression survey used in the final protocol. For this step, we recruited participants who are at the senior years of their doctoral programs, focusing on cyber attack and defense research, and have already published papers at top security venues. However, results show that even this cohort struggled with properly navigate the system without much direction, identify vulnerabilities, and make progress to subsequent levels—especially given the limited study duration. Thus, we added a walkthrough video for the setup, clearer instructions for navigating the environment and sharing the screen, and links to relevant tutorials for some task types, and the progression survey. The the progression survey provided loose guidance for navigation and vulnerability identification and captured participants' task choices, switching decisions, and reasoning while they were working, so that later behavioral traces could be interpreted with more context.

G. Experimental Procedure

Sessions were conducted remotely. Participants were recruited from graduate cybersecurity courses whose syllabi include CTF-style tasks, with course extra credit offered for participation. Before connecting to the environment, each participant completed the pre-challenge survey containing the personality measures described above. During the session, participants were asked to spend a minimum of two hours and to think aloud while working, and they interacted with the environment through the web-based terminal. As they moved between tasks, they answered the in-session progression survey (the survey can be found here: https://zenodo.org/records/20508226), which provided initial instructions and some guidance on how to navigate the system and hints on potential vulnerabilities, and simultaneously captured participants' path choices, reasoning, reactions to obstacles, and decision-making. Sessions were screen- and audio-recorded for participants who consented. After completing the chal-

lenge, participants were debriefed about the deceptive elements.

Twenty-nine participants registered and completed the pre-challenge survey, 24 began the main session, and 20 produced valid sessions for analysis (Section V-A). The two-hour minimum was not strictly enforced, we accepted sessions longer than 1 hour and 30 minutes.

H. Ethical Considerations

The study protocol was approved by our institution's IRB, and all procedures comply with ethical guidelines for human-subject research. Participants provide informed consent before beginning the study and are explicitly told that the environment may contain deceptive elements similar to those used in real-world penetration testing (e.g., honeypots and decoy services). During the session they interact only with isolated research infrastructure, and no real organizations or third-party systems are affected. After the session, participants are debriefed about the specific deceptive mechanisms used and the goals of the study, and they are given the opportunity to withdraw their data. All collected data are stored securely, pseudonymized before analysis, and used only for research purposes.

IV. Data analysis process

We analyzed data from the 20 participants who completed all parts of the study. The data included the pre-challenge personality measure, timestamped activity logs from the challenge environment, and progression survey responses. Screen recordings were available for 18 sessions and were manually reviewed to add qualitative insights and explain behavioral patterns found in the log.

A. In-Session Logs

Inside each challenge VM, logging was injected through the shell environment and background monitors. Shell startup files loaded a logging profile, and a shell `DEBUG` trap recorded commands from participants. The logging layer also tagged common action types, including file reads, file operations, network commands, password-cracking tools, wordlist access, and user-switch attempts. Events other than commands were logged as well: a background process monitor recorded short-lived processes, a file monitor recorded file operations (e.g., to create or modify scripts), and the Flask web app produced request logs for the SQL-injection task. Logs for system initialization and other bookkeeping were recorded separately. These records were sent through `logger`/`rsyslog` with runtime `CTF[...]` tags and setup `CTF_SETUP[...]` tags, then exported from the container for analysis.

B. Log Normalization and Cleaning

Log files were parsed into an event table with columns corresponding to participant code, timestamp, event type, active user, working directory, command text, and normalized message. Participant behaviors were separated from system-generated activity by marking monitor events, setup records, process-polling rows, and shell-startup commands. Next, we integrated other events such as web activity and database interaction (Task A2), switching users, accessing files on user accounts that did not have any vulnerability, and so on.

Throughout this process, we manually compared logs with screen recordings to verify their correctness and fix anomalies. For example, one participant took a long break during the experiment, and a few others kept the session open after the experiment duration. Thus, we manually set the session duration and align in-between gaps in activities by manually reviewing log timings and screen recordings.

C. Command Keywords and Command Types

For each participant-generated command, we extracted the command keyword, a normalized command template, and a contextual command type. The command keyword captures the main command or tool name, such as `ls`, `cat`, `curl`, `john`, `sudo`, or `fcrackzip`. This gives a tool-level view of what participants used. The normalized template replaces the command argument values— such as file paths, quoted strings, hashes, numbers, and host:port— with placeholders (`john --wordlist=rockyou.txt backup.zip` and `john --wordlist=common.txt creds.zip` fall under the same template). We introduced the template because treating every argument change as a new attempt would inflate behavioral signals. The keyword indicates which tool was used, while the template indicates the strategy. To make the distinction concrete, consider a participant working on the A1 archive who issues `fcrackzip -D -p rockyou.txt creds.zip`, then `fcrackzip -D -p common.txt creds.zip`, then `john --wordlist=rockyou.txt hashes.txt`. At the keyword level this is two tools, `fcrackzip` and `john`. At the template level the two `fcrackzip` commands collapse to one template (`fcrackzip -D -p <FILE> <FILE>`)—the same attack with a different wordlist—so they count as one repeated strategy rather than two new attempts, while the `john` command is a different template and counts as a new approach. All three share the same type, archive/hash cracking, and the same phase, exploitation.

Command type was assigned at the action level because the same keyword can serve different purposes depending on its arguments and task context. In A2, for example, `curl http://localhost:8000/` was labeled as web probing because it checked whether the application was reachable, while a `curl "http://localhost:8000/login?user=admin'--"` request containing login parameters and SQL-injection markers was labeled as SQL injection. Similarly, `echo` could be supporting activity or part of code creation, and `sudo` could indicate privilege escalation, user switching, or routine tool use depending on the arguments and surrounding context. We assigned these types using a rule-based classifier that used the command keyword, command arguments, working directory, active user, and logging-layer labels. The resulting command types are summarized in Table II. To check the classifier, we reviewed the unique normalized command

templates and adjusted the rules when a template was clearly mislabeled or ambiguous.

After assigning command types, we mapped each command action to one of three broader behavioral phases: exploration, exploitation, or supporting activity. Exploration included commands used to inspect the environment, such as navigation, file listing, file reading, web probing, binary analysis, identity/context checks, and system/network inspection. Exploitation included commands that directly attempted to gain access, recover credentials, switch users, or escalate privileges, such as archive/hash cracking, SQL injection payloads, user-switching commands, and privilege-escalation attempts. Supporting activity included commands that did not clearly fit either phase. We used these phase labels to compute exploration-to-exploitation ratios at the participant and progression-stage levels.

D. Task and Progression Labeling

We defined task participation as any action related to a task, including reconnaissance, such as reading or searching a task file. We did not require a task to be completed for a participant to count as having engaged in it, because our goal is in how participants allocated attention and effort across the challenge, not whether they succeeded.

Task labels were assigned using regex-based patterns applied to each action's command text, working directory, active user, and normalized message (the event-type tag from the VM logger, such as `FILE_ACCESS` or `USER_SWITCH`). These patterns served as a first-pass heuristic to identify likely engagement with A1, A2, B1, B2, B3, other user space, and root-related activity, as well as tool usage signals such as `john`, `fcrackzip`, `sqlmap`, `gdb`, `checksec`, and `pspy`. We used these signals to establish task participation and path order, not to determine whether a task succeeded, since a regex match confirms that relevant activity occurred, but not its outcome. Seeing `sqlmap` in a command means the participant attempted SQL injection, not that it worked. A1 represents the archive and password path. A2 represents the web application and database path. B1, B2, and B3 are later tasks counted only when activity occurs in the correct non-initial user space for that task. Actions in another non-root user account that did not map to B1, B2, or B3 were labeled as other user space. Commands and paths that referenced root-level targets were labeled separately as root-related activity; root access was counted only when the active user was actually root. These mappings were applied automatically through the same regex pipeline from the logs, then cross-checked a sample of labels against the activity logs and screen recordings to catch timing gaps, session breaks, and cases where the active user context was ambiguous.

Progression levels were defined from these task labels. Level One contains A-task activity on A1 and A2. Level Two contains B-task activity on B1, B2, and B3, counted only when it occurs in the correct user space. User switching is not a level of its own — it is the action that moves a participant from Level One to Level Two, and Level Two is defined by the B-task activity that follows it, not the switch itself.

E. Behavioral metrics

After pre-processing the logs, we computed metrics at the participant, task, command-type, and progression-stage levels. These metrics were derived from the cleaned action table described above and were later used in the stage- and task-level summaries reported in Sections V-C and V-B. *Command count* is the subset of actions that correspond to participant-generated terminal commands.*Unique command lines* count distinct raw command strings. *Unique command keywords* count distinct tools or shell commands, such as `curl`, `john`, or `ls`. *Command-template diversity* counts distinct normalized command templates after variable arguments are replaced with placeholders. This metric captures how many different command forms or strategies appeared in a participant's trace. Finally, we computed behavioral phase metrics using the exploration, exploitation, and supporting-activity labels described in Section IV-C.

F. Screen-Recording Review

The first author reviewed and annotated the 18 screen recordings across three dimensions: session-startup and terminal-access behavior, progression-survey interaction, and AI-tool use patterns. For session startup, we noted frictions participants faced in accessing the terminal or configuring their environment, which can account for early quiet command windows in the log. We further noted how participants simultaneously progressed through the survey and the experiment tasks. We noted that some participants did not synchronously engaged with the survey and the tasks; the primary reason was the open ended nature of our system that allowed exploring several possible attack paths without getting back to the survey for some amount of time. Lastly, we noted the use of large language models (like ChatGPT, Gemini, and Claude). We observed two patterns: passive use, where participants copied survey-derived information into an LLM chat interface and followed the output without adapting commands to their environment; and targeted use, where participants asked specific questions about commands or techniques, then adapted and tested the responses against their actual terminal state.

V. Results

A. Participants

The study was advertised in two graduate cybersecurity courses; one of them was for in-campus students, while the other was part of an online program and is attended by learners who might already have professional cybersecurity roles. Both courses have CTF challenges as part of their assignments and final project. Twenty-nine people registered and completed the pre-challenge survey, and 24 began the main session. Three sessions were excluded for setup or logging failures and one for too few meaningful commands to support behavioral analysis, leaving 20 valid sessions for analysis. All completing participants were compensated with 50 USD. Table III reports their demographics, skills, and experience levels.

TABLE II
COMMAND CLASSIFICATION RULES USED TO DERIVE CONTEXTUAL COMMAND TYPES.

Pattern matched in command	Phase	Type
Exploration rules		
`pwd|whoami|id|hostname|date|uname`	exploring	identity/context check
`cd|pushd|popd`	exploring	navigation
`ls|la|ll|find|tree|findstr|locate`	exploring	file search/listing
`cat|less|more|head|tail|strings|grep|sed|awk|nano|vim|vi`	exploring	file read/search/edit
`gdb|checksec|objdump|readelf|ltrace|strace|file|xxd|hexdump`	exploring	binary analysis
`curl|wget|nc|netcat` or URL without SQL-injection patterns	exploring	web probe
`python|python3|bash|sh|gcc|make|perl|ruby|node` or output redirection	exploring	code creation/execution
`tcpdump|netstat|ss|lsof|ps|top|pgrep|env|printenv|crontab`	exploring	system/network inspection
Exploitation rules		
`fcrackzip|zip2john|john|hashcat`	exploiting	archive/hash cracking
`unzip` with `corporate_credentials_backup`	exploiting	archive/hash cracking
`curl|wget|nc|netcat` or URL with SQL-injection patterns[a]	exploiting	SQL injection
`select|from|where|union|sqlite|mysql|psql|sqlmap|database`	exploiting	SQL injection
`su|ssh|sudo -u`	exploiting	user switch
`sudo|chmod 4755|/etc/shadow|/root/`	exploiting	privilege escalation
Fallback to logging-layer labels		
`NETWORK_ACTIVITY` or `HTTP_REQUEST`	exploring	web probe
`FILE_ACCESS` or `WORDLIST_ACCESS`	exploring	file read/search/edit
`SUDO_ATTEMPT` or `USER_SWITCH` labels	exploiting	user switch
`PASSWORD_CRACKING` or `FILE_CRACKING`	exploiting	archive/hash cracking
No matched rule or fallback label	other	other

[a] SQL-injection patterns included login parameters and common injection markers such as `username=`, `password=`, `OR 1=1`, `UNION SELECT`, `sqlite_master`, and `information_schema`.

TABLE III
PARTICIPANT BACKGROUND IN THE FINAL ANALYSIS DATASET.

	Category	N
Education	Bachelor's degree	12
	Graduate/professional degree	6
	Some college, no degree	2
Major/field	Computer Science	8
	Cybersecurity	6
	Other/Data Science	4
	Engineering	2
Cybersecurity experience	Less than 1 year	14
	1–3 years	5
	4–6 years	1
CTF expertise	Intermediate	8
	Beginner	8
	None	2
	Advanced	2

B. Initial Task Selection and Progression

All participants started at level one. Fourteen of them began with A1, and the remaining six with A2 (Table IV). Recall that, A1 (cracking a file) was deliberately made impossible to complete, but was signaled as a high-risk, high-reward task. This signaling seems to have influenced participants' task selection: six of them started with A2 to avoid the risk of getting caught, while 14 participants selected A1 as it seemed more promising or will yield a high payoff (Table VI). Switching between A1 and A2 was common (Table IV): 15 participants switched between them at least once. The progression survey suggests different motivations for the two initial paths: A2 was most often selected because it seemed lower risk or less likely to get participants caught (6 responses), while A1 was most often selected because it matched participants' perceived skills (5 responses) or seemed more promising/certain (4 responses).

Nine out of the 20 participants could progress to Level 2. Among them were five (out of 14) who started with A1 and four (out of six) who started with A2. Further, in the latter group, one participant never switched to A1. Thus, the group starting with A1 had a much lower percentage of passing Level 1 compared to the group that started with A2. One possible reason is that A1 may have felt frustrations with not making any progress; 8 (out of 14 said) they switched to A2 because they were "not making progress," compared with only 1 out of 6 participants who started with A2. These participants also spent a substantial amount of time on A1 before moving on, with a median of 42.4 minutes, which might have created cognitive exhaustion. These results align with prior studies that show that making attackers engage with deceptive tasks may drain resources and help prevent subsequent attacks [2], [3], [29], [30].

C. Behavioral trends

Here we present participants personality traits and how they correlated with behavioral metrics. First, we note that all trait scales had acceptable values for reliability (Cronbach's α): Persistence (α = .78), Resilience (α = .85), and Risk-Taking (α = .86), BFI-2-S ($\alpha = .75$). Next, we report observed trends in attacker behaviors and explain with descriptive statistics. Since this is an exploratory study, we do not test pre-defined hypotheses for statistical significance.

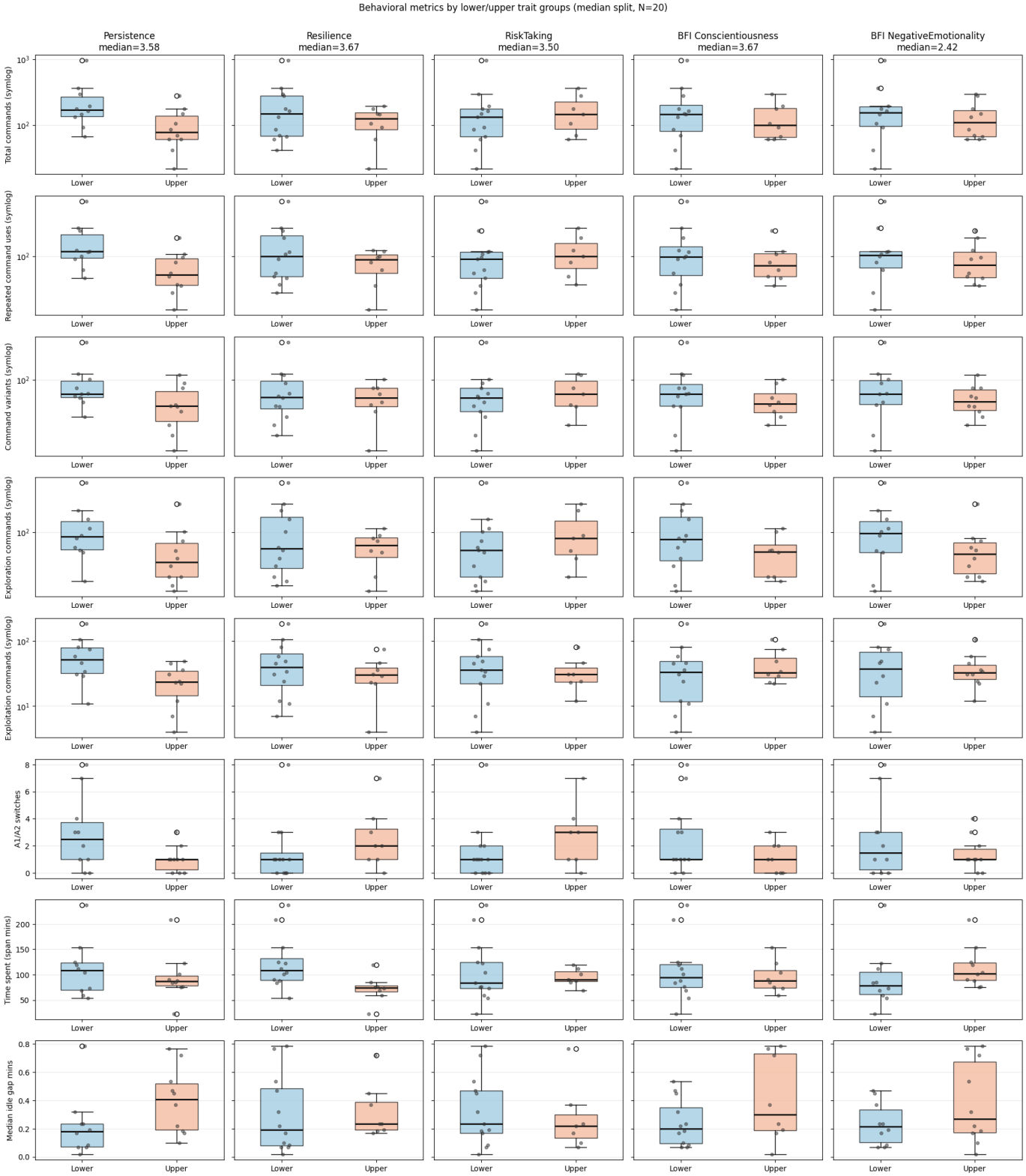

Fig. 2. Behavioral trends by lower and upper trait groups using median splits ($N = 20$).

The trends are presented in Figure 2, which shows behavioral metrics of participants grouped by the median trait score. We make several observations. First, participants with a high persistence score sent fewer total commands (and consequently fewer variants and repeated templates) than participants with lower scores. This result seems counter intuitive; however,

TABLE IV
LEVEL ONE TASK SELECTION AND SWITCHING.

Pattern	Participants
Started with A1	14
Started with A2	6
Participated in A1	19
Participated in A2	16
Participated in both A1 and A2	15
Participated only in A1 (without A2)	4
Participated only in A2 (without A1)	1
Switched A1/A2 at least once	15
Switched A1/A2 more than once	8

TABLE V
PROGRESSION THROUGH THE CHALLENGE. LEVEL ONE CONTAINS A1/A2 ACTIVITY; LEVEL TWO CONTAINS B1/B2/B3 ACTIVITY IN THE CORRECT USER SPACE.

Progression stage	Participants	Actions	Commands
Level One	20	1,047	684
Level Two	9	79	78
Other user space	9	779	768
Entire session	20	4,254	3,649

the former has a larger idle time than the latter. Moreover, high-persistent group spent comparable amount of time, with a much higher within-group homogeneity. They also switched between A1 and A2 less frequently than the low-persistent group. Taken together, these results indicate that the high-persistent group spent more time in investigating the system and its responses to their commands, and stuck to the same task for longer, rather than just sending commands (which includes switching tasks).

For resilience, similar observations hold for command-related metrics. Both the total time and idle time were shorter for high-resilient participants compared to the low-resilient group; but we note the outlier in the latter group that may have contributed to this unexpected result. On the other hand, high-resilient group switched between tasks more often, which is consistent with the scale construct we used: repeated attempts to make progress toward the larger goal, which is to escalate privilege by any means, rather than sticking to one specific task that is just one alternative path to achieving the larger goal. For conscientiousness, which indicate how organized or thoughtful one is in their behavior, idle time is the most relevant metric (as a proxy for taking the time to think and plan).

For conscientiousness and neuroticism, the most relevant metrics are total and idle time spent. Conscientiousness is associated with organized and thoughtful behaviors, and thus high-conscientious participants are likely to spend more idle time. High-neurotic participants are also likely to spend more idle time, but this is because they are prone to suffer from anxiety and an inability to make decisions. Both of these assertions are supported by our data, as shown in Figure 2. Compared to their counterparts, both high-conscientious and high-neurotic groups are also likely to make more exploration commands to make informed decisions or to avoid risky

TABLE VI
TIME AND ACTIVITY ON A1 AND A2 GROUPED BY FIRST TASK. *N* IS THE NUMBER OF PARTICIPANTS FROM EACH FIRST-TASK GROUP WHO HAD ACTIVITY ON THAT TASK; A PARTICIPANT CAN APPEAR IN BOTH TASK ROWS FOR THEIR GROUP.

			Time spent (minutes)		
First task	Task	N	Mean	Median	SD
A1	A1	14	47.04	42.42	28.27
A1	A2	10	41.92	28.55	32.98
A2	A1	5	44.74	22.67	51.79
A2	A2	6	86.80	74.59	78.51

actions (which can induce anxiety). However, we did not find any evidence supporting these assumptions.

We further hypothesized that a risk taking attitude will positively correlate with engaging with risky behaviors, while negative emotionality (Neuroticism) will correlate negatively. However, our data shows the opposite trend: participants who started with A1 (high risk task) had mean score of 3.30 (SD = 0.71), while those who started with A2 had a mean score of 3.40 (SD = 0.81) for the risk-taking scale. For Negative Emotionality (Neuroticism), participants who started with A1 had a mean score of 2.54 (SD = 0.85), compared with 2.22 (SD = 0.43) among those who started with A2.

Overall, we found several promising trends that future studies can build on. But not all assumed trends were observed, and some trends were the opposite of what we had expected, likely due to several confounding factors. For example, we did not control for participants expertise in specific tasks. Due to the modest data size, we also could not control for task level characteristics; e.g., some tasks may need more exploration than others: A1 and A2 had visible attack surfaces and were reached by all participants, while B1, B2, and B3 required more inspection of files, permissions, scripts, and binaries. Finally, how people deal with high risk or anxiety inducing situation in their daily life may not translate to how they deal with risks, or even interpret risks in a simulated environment.

D. Qualitative insights form reviewing screen recordings

We manually reviewed the screen recordings and noted several confounding factors and noise sources that might further explain the results. First, some participants encountered startup or terminal-access friction, which created quiet periods of inactivity that be mistaken for low engagement if only the behavioral logs are analyzed. Second, participants varied in how they used the progression survey: some treated it as an active guide and checked its hints against the terminal, while others advanced through survey prompts without validating the information in the environment. Third, most participants used external AI tools. This included direct use of LLM interfaces and using AI-generated results through search engines. However, use quality varied: some participants pasted broad task descriptions and accepted generic output, while others asked targeted command- or syntax-level questions and adapted the response to the current terminal state. Thus, AI

use was common, but it did not always translate into effective progress.

The open-ended nature of the experimental setup, particularly at Level 2, created large variances in participant behaviors. Out of nine participants who reached that level, only two attempted to exploit vulnerabilities that were mentioned in the progression survey. All other participants primarily performed reconnaissance in other user spaces that did not have any vulnerability. This is not unexpected, since we mentioned plausible vulnerabilities as loose directions, and actively encouraged them to explore and gather as many "assets" as they can—mimicking real world attacks. However, finding patterns in such an open system requires a much larger activity dataset.

VI. Discussion, Limitations, and Conclusions

We present an exploratory study to examine the extent to which personality traits can be captured based on behaviors observed during a cyber attack. The findings provide several promising directions to explore in future studies involving skilled cyber professionals, as well as offers guidance to design such an experiment to minimize confounding effects.

A. Implications of the findings

First, our results establish that deceptive signaling can influence attackers' decision-making: many of our participants noted the hint about high risk activities, and took alternate routes. To the best of our knowledge, this is the first empirical evidence on the effect of signaling on human attackers with cybersecurity knowledge, and can serve as the foundation for this line of research involving highly skilled attackers (e.g., experienced penetration testers). Deceptive signaling can be an highly effective technique to deter attackers, make them waste time and other resources on fake vulnerabilities, or direct them towards engaging activities that can further reveal their attack strategies.

Second, we found (weakly) distinguishing trends when divided the participants based on their trait scores. While we did not test for statistical significance due to the exploratory nature of the study, the findings will guide future, more focused studies designed to test specific hypotheses. Once established, these results are expected to radically change how defense for advanced persistent attacks work; these trends will help profile the attacker and guess their future actions, allowing proactive and dynamic reconfigurations of the defense strategies. However, several of the assumed relationships between behaviors and traits did not hold, we largely attribute this to the experimental design and confounding factors. Based on these, below we provide design guidelines for future studies.

B. Design Lessons for Similar Studies

We attempted to mimic a real world attack scenario that is open ended but the trade off is to recruit sufficiently skilled participants who can navigate such a system. Unfortunately, most cyber security courses and training materials focus on CTF challenges, which have one specific goal and comes with concrete instructions. We tried to strike a balance between realistic setup and conducting the experiment with a cohort of cybersecurity learners by an accompanying progression survey. It provided some loose directions so that participants do not get overwhelmed by the endless possibilities. And it did so in the form providing threat intels gathered from previous attack attempts, which is common within persistence threat actors. The progression survey provided two benefits: it allowed us to incorporate deceptive signals that could have been missed otherwise, and it facilitated collected data about why they made certain decisions in real time. However, some participants struggled to simultaneously progress through the survey and actual attack tasks. Based on these observations, we make the following design guidelines for future studies.

Since most participants lacked any experience with open-ended cyber attacks, they struggled to properly follow the hints and struggled to maintain progress in survey and attack activities that were presented in separate windows. Incorporating the direct hints within the attack environment could mitigate this issue.

Another major issue was the heterogeneity in the participants' skill level. Since expertise heavily influence both exploration and exploitation tactics, the skill level likely acted as a confounding factor that resulted in weak or even unexpected behavioral trends. For example, behaviors such as slow progress, repeated attempts, and task abandonment plausibly reflected technical knowledge gaps as much as personality. This could be mitigated in two ways. First, with a sufficiently large sample size, the skill level could be explicitly modeled as a covariate. Second, with a longer duration experimental design that incorporates a training phase, participants could be brought to the roughly same level of competency for the specific attack tasks.

A third challenge we faced was to replicate the incentive and reward structure that exist in a real attack scenario. Our scoring and monitoring scheme was intended to simulate reward, risk, and loss, but participation incentives (course extra credit) and a non-binding two-hour target meant some participants optimized for completion rather than for the in-environment trade-offs the design probed. Traits such as risk-taking are unlikely to surface behaviorally unless participants genuinely perceive the consequences of risky actions. Future environments should tie incentives more directly to in-environment outcomes so that the probed trade-offs carry real stakes. This could mean rewarding (e.g., with bonus points or money) for achievements and punishing for mistakes (e.g., taking away some credits, implementing wait times when getting "caught" by the system).

C. Limitations and Threats to Validity

The sample was small ($N = 20$) and skewed toward early-career participants, so all trait–behavior associations are exploratory and should not be read as predictive. Skill and CTF familiarity are the most serious confounds: observed behavior may reflect competence rather than disposition, and our sample lacked the skill range to separate the two. Screen recordings

were reviewed by a single researcher without an inter-rater check. Finally, the analysis reflects a single environment with one set of deceptive vectors; the findings about behavioral measurement may not transfer to other task designs or attacker populations.

D. Conclusions

This exploratory study significantly contributes toward advancing deception-based proactive cyber defense. It provides an experimental system that future studies can build on, and offers guidance on improving the study design. It further provides several promising links between traits and behaviors that future studies can validate.

Acknowledgments

This research was supported by the Air Force Office of Scientific Research (AFOSR) under Award FA9550-24-1-0227. Any opinions, findings, conclusions, or recommendations expressed in this paper are those of the author(s) and do not necessarily reflect the views of AFOSR.

References

[1] E. A. Cranford, C. Gonzalez, P. Aggarwal, M. Tambe, S. Cooney, and C. Lebiere, "Towards a Cognitive Theory of Cyber Deception," *Cognitive Science*, vol. 45, no. 7, p. e13013, Jul. 2021. [Online]. Available: https://onlinelibrary.wiley.com/doi/10.1111/cogs.13013

[2] K. Ferguson-Walter, T. Shade, A. Rogers, E. Niedbala, M. Trumbo, K. Nauer, K. Divis, A. Jones, A. Combs, and R. Abbott, "The Tularosa Study: An Experimental Design and Implementation to Quantify the Effectiveness of Cyber Deception," in *Proceedings of the Annual Hawaii International Conference on System Sciences*. Hawaii International Conference on System Sciences, 2019. [Online]. Available: http://hdl.handle.net/10125/60164

[3] R. S. Gutzwiller, M. Gilbert, T. J. Drescher, K. J. Ferguson-Walter, N. Mikanda, C. J. Johnson, and D. D. Scott, "Frustration, Confusion, Surprise, Confidence, And Self-Doubt: Cyber Operators' Affects During A Realistic Experiment," *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*, vol. 67, no. 1, pp. 233–239, Sep. 2023. [Online]. Available: https://journals.sagepub.com/doi/10.1177/21695067231192883

[4] E. CRANFORD, C. GONZALEZ, P. AGGARWAL, S. COONEY, M. TAMBE, and C. LEBIERE, "Adaptive Cyber Deception: Cognitively Informed Signaling for Cyber De- fense," Jan. 2020.

[5] U. Hani, O. Sohaib, K. Khan, A. Aleidi, and N. Islam, "Psychological profiling of hackers via machine learning toward sustainable cybersecurity," *Frontiers in Computer Science*, vol. 6, Apr. 2024. [Online]. Available: https://www.frontiersin.org/articles/10.3389/fcomp.2024.1381351/full

[6] A. N. Joinson, M. Dixon, L. Coventry, and P. Briggs, "Development of a new 'human cyber-resilience scale'," *Journal of Cybersecurity*, vol. 9, no. 1, p. tyad007, Jan. 2023. [Online]. Available: https://academic.oup.com/cybersecurity/article/doi/10.1093/cybsec/tyad007/7130095

[7] T. Porter, D. Catalán Molina, L. Blackwell, S. Roberts, A. Quirk, A. Lee Duckworth, and K. Trzesniewski, "Measuring Mastery Behaviors at Scale: The Persistence, Effort, Resilience and Challenge-Seeking Task (PERC)," *Journal of Learning Analytics*, vol. 7, no. 1, Mar. 2020. [Online]. Available: https://learning-analytics.info/index.php/JLA/article/view/6759

[8] M. Canudo, S. Moreira, R. Solymosi, and I. Guedes, "Differentiating Hackers and Hacker Types: The Role of Self-Control, Personality, and Motivations," 2025. [Online]. Available: https://www.ssrn.com/abstract=5217224

[9] J. Gaia, B. Ramamurthy, G. Sanders, S. Sanders, S. Upadhyaya, X. Wang, and C. Yoo, "Psychological Profiling of Hacking Potential," in *Proceedings of the Annual Hawaii International Conference on System Sciences*. Hawaii International Conference on System Sciences, 2020. [Online]. Available: https://hdl.handle.net/10125/64014

[10] D. Tovarňák, S. Špaček, and J. Vykopal, "Traffic and log data captured during a cyber defense exercise," *Data in Brief*, vol. 31, p. 105784, Aug. 2020. [Online]. Available: https://linkinghub.elsevier.com/retrieve/pii/S2352340920306788

[11] "Cybersecurity Methodology for Specialized Behavior Analysis," in *Lecture Notes of the Institute for Computer Sciences, Social Informatics and Telecommunications Engineering*. Cham: Springer International Publishing, 2021, pp. 237–243. [Online]. Available: https://link.springer.com/10.1007/978-3-030-68734-2_14

[12] J. M. Tshimula, D. K. Nkashama, J. T. Muabila, R. M. Galekwa, H. Kanda, M. V. Dialufuma, M. M. Didier, K. Kalala, S. Mundele, P. K. Lenye, T. W. Basele, A. Ilunga, C. N. Mayemba, N. M. Kasoro, S. K. Kasereka, H. Mikese, P.-M. Tardif, M. Frappier, F. Kabanza, B. Chikhaoui, S. Wang, A. M. Sumbu, X. Ndona, and R. K.-K. Intudi, "Psychological Profiling in Cybersecurity: A Look at LLMs and Psycholinguistic Features," 2024. [Online]. Available: https://arxiv.org/abs/2406.18783

[13] L. Spitzner, *Honeypots: tracking hackers*. Boston: Addison-Wesley, 2003.

[14] N. C. Rowe and J. Rrushi, *Introduction to Cyberdeception*. Cham: Springer International Publishing, 2016. [Online]. Available: http://link.springer.com/10.1007/978-3-319-41187-3

[15] A. L. Duckworth, C. Peterson, M. D. Matthews, and D. R. Kelly, "Grit: Perseverance and passion for long-term goals." *Journal of Personality and Social Psychology*, vol. 92, no. 6, pp. 1087–1101, 2007. [Online]. Available: https://doi.apa.org/doi/10.1037/0022-3514.92.6.1087

[16] T. Constantin, A. Holman, and M. Hojbotă, "Development and validation of a motivational persistence scale," *Psihologija*, vol. 45, no. 2, pp. 99–120, 2012. [Online]. Available: https://doiserbia.nb.rs/Article.aspx?ID=0048-57051202099C

[17] S. S. Luthar, D. Cicchetti, and B. Becker, "The Construct of Resilience: A Critical Evaluation and Guidelines for Future Work," *Child Development*, vol. 71, no. 3, pp. 543–562, May 2000. [Online]. Available: https://srcd.onlinelibrary.wiley.com/doi/10.1111/1467-8624.00164

[18] B. W. Smith, J. Dalen, K. Wiggins, E. Tooley, P. Christopher, and J. Bernard, "The brief resilience scale: Assessing the ability to bounce back," *International Journal of Behavioral Medicine*, vol. 15, no. 3, pp. 194–200, Sep. 2008. [Online]. Available: http://link.springer.com/10.1080/10705500802222972

[19] D. C. Zhang, S. Highhouse, and C. D. Nye, "Development and validation of the General Risk Propensity Scale (GRiPS)," *Journal of Behavioral Decision Making*, vol. 32, no. 2, pp. 152–167, Apr. 2019. [Online]. Available: https://onlinelibrary.wiley.com/doi/10.1002/bdm.2102

[20] C. J. Soto and O. P. John, "Short and extra-short forms of the Big Five Inventory–2: The BFI-2-S and BFI-2-XS," *Journal of Research in Personality*, vol. 68, pp. 69–81, Jun. 2017. [Online]. Available: https://linkinghub.elsevier.com/retrieve/pii/S0092656616301325

[21] H. JACH and L. SMILLIE, "To fear or fly to the unknown: Tolerance for ambiguity and Big Five personality traits," *Journal of Research in Personality*, vol. 79, pp. 67–78, Apr. 2019.

[22] K. BYRNE, C. SILASI-MANSAT, and D. WORTHY, "Who chokes under pres- sure? The Big Five personality traits and decision-making under pressure," *Personality and Individual Differences*, vol. 74, pp. 22–28, Feb. 2015.

[23] M. BABAEI, M. MOHAMMADIAN, M. ABDOLLAHI, and A. HATAMI, "Relationship between big five personality factors, problem solving and medical errors," *Heliyon*, vol. 4, 9, p. 00789, Sep. 2018.

[24] J. ANGLIM, P. DUNLOP, S. WEE, S. HORWOOD, J. WOOD, and A. MARTY, "Personality and intelligence: A meta-analysis," *Psychological Bulletin*, vol. 148, pp. 5–6, 2022, place: Place Publisher: US Publisher: American Psychological Association.

[25] T. CHAMORRO-PREMUZIC, J. MOUTAFI, and A. FURNHAM, "The relationship between personality traits, subjectively-assessed and fluid intelligence," *Personality and Individual Differences*, vol. 38, 7, pp. 1517–1528, May 2005.

[26] T. HALEVI, N. MEMON, and O. NOV, "Spear-Phishing in the Wild: A Real-World Study of Personality, Phishing Self-Efficacy and Vulnerability to Spear-Phishing Attacks."

[27] J.-H. CHO, H. CAM, and A. OLTRAMARI, "Effect of personality traits on trust and risk to phishing vulnerability: Modeling and analysis," in *2016 IEEE International Multi- Disciplinary Conference on Cognitive Methods in Situation Awareness and Decision Support (CogSIMA*, Mar. 2016, pp. 7–13.

[28] O. HJEMDAL, O. FRIBORG, S. BRAUN, C. KEMPENAERS, P. LINKOWSKI, F. SION, and P., "The Resilience Scale for Adults: Construct Validity and Measurement in a Belgian Sample," *International Journal of Testing*, vol. 11, 1, pp. 53–70, Feb. 2011. [Online]. Available: https://doi.org/10.1080/15305058.2010.508570.

[29] K. FERGUSON-WALTER, M. MAJOR, C. JOHNSON, and D. MUHLEMAN, "Examining the Efficacy of Decoy-based and Psychological Cyber Deception," pages: 1127–1144.

[30] C. JOHNSON, R. GUTZWILLER, J. GERVAIS, and K. FERGUSON-WALTER, "Decision-Making Biases and Cyber Attackers," in *2021 36th IEEE/ACM International Con- ference on Automated Software Engineering Workshops (ASEW*, Nov. 2021, pp. 140–144.