Deceptive Misuse of Low-Code Platforms: Visualizing the Performance of Disruptive Cyber Effects from Human and LLM Agent Attackers

Tim Pappa[1] and Teja Sane[2]

[1]Independent researcher
[2]Independent researcher
timothypappa@gmail.com

Abstract

This position paper visualizes how human and Large Language Model (LLM) Agent attackers could manipulate low-code platforms to perform disruptive cyber effects to *dazzle* and misdirect the attention of human defenders. There are a growing number of public examples of unknown attackers who gain unauthorized but limited access to a network with minimal credentials and then cause disruption. One of the goals of this disruption or even performance of disruption appears to be to confuse or misdirect human defenders and cause reputational harm for the victim organization. A recent example could include the Sha1-Hulud worm attacks. We look at Power Apps or Power Automation, a low-code platform that is so common that most organizations do not consider this platform to be a vulnerability. In our experience as cyber deception and detection engineering practitioners, and in discussions with information security practitioners, we find that most larger organizations do not closely monitor the sometimes thousands of Power Apps templates available to any user, with a range of access controls and broad permissions. We argue in this paper that that this low-code platform could be exploited by human and LLM Agent attackers with minimal effort, starting with basic user credentials. We suggest that an LLM Agent attacker could exploit this platform more instrumentally than a human attacker, but both attackers could use deception techniques such as *dazzling* to gather enough information to engineer a highly disruptive cyber effect. We will discuss the Bell-Whaley deception framework to explain how *simulation* and *dissimulation* could be applied in this scenario. We will visualize a human or LLM Agent attacker who gains unauthorized access to a network with basic credentials to then use Power Apps to automate hundreds of emails throughout an organization with what appears to be explicit content. This is based on an actual situation in which a user mistakenly automated a flurry of hundreds of emails in minutes. We will share some of our observations from a *think-aloud* exercise with a mixed sample of information security practitioners where we introduced this same scenario, finding that network defenders may be demonstrating some bias in their real and imagined impressions of prototypical attackers using low-code platforms like Power Apps.

Introduction

This paper is foundationally based on how people respond to or do not respond to user behaviors on networks, but we consider the expectation of real and imagined behaviors of Large Language Model (LLM) Agent behaviors, too. A recent example could include the Sha1-Hulud worm attacks, where an unknown attacker or attackers gained unauthorized access to tens of thousands of internal company Github repositories and immediately released that data (Microsoft, 2025; Palo Alto Networks, 2025). The organizations compromised by this worm attack struggled to determine how the worm compromised their infrastructure and were challenged to quickly determine which repositories and sensitive data may have been released publicly. Many more organizations who were not compromised were still disrupted by this cyber effect, uncertain if their repositories would be compromised later or if their internal information and code was already public or available to attackers.

Much of the research in the past several years related to behavioral and cognitive frameworks has concentrated on the cognitive vulnerabilities or biases and behaviors of attackers on a host or network, but there has been less focus on the cognitive bias of defenders or their expectations of prototypical attacker behaviors. This paper explores this potential bias of human defenders to both human and LLM Agent attackers using deception on a low-code platform.

This research concentration on cognitive biases of attackers reflects growing efforts to explore a cognitive theory of deception, instead of what has historically been a focus on physical deception cues. Many of these same researchers concentrating on the effect of cognitive biases have encouraged a more "adaptive" cyber deception design and build, primarily based on deceptive signaling targeting attackers designed to influence changes in their attacker behaviors (Cranford et al., 2020; Cranford et al., 2021; Gonzalez et al.,

2020). Aggarwal et al. (2024) wanted to build on their findings from a prior study where participants behaving like attackers appeared to identify cognitive biases, exploring whether attackers showed any preference for targeting specific areas of a network and if attackers revealed any behavioral patterns when operating in those specific areas. Their findings suggested attackers did demonstrate cognitive biases, such as sunk cost fallacy and default effect. The goal of Aggarwal's research was to determine if defenders could create more dynamic responses to attackers based on linking identified cognitive biases with "behavioral patterns" observed in similar network environments. Some of these behavioral patterns appeared to reflect default effect, where attackers are more likely to choose a "preset alternative" or default when making decisions. This finding is like naturalistic decision-making research, where attacker decision makers appear to process information differently, based on their experiences making similar decisions when presented with similar information in the past (Du et al., 2023; Yuill, Denning, and Feer, 2007). Aggarwal et al. in another recent study found that attackers appeared to demonstrate this default effect most commonly in the reconnaissance phase of their attack cycle. Aggarwal et al. (2024) in another study found other cognitive biases as well, such as availability bias or when people appear to make choices based on the information that comes to mind first, for example. Their findings highlighted what they characterized as the strength of the "presence of a bias", based on whether participants behaving like attackers found a task or function on a network to be interesting or not or to what degree they found it interesting.

Aggarwal et al. suggested that there are other functions that attackers may find less interesting, that may exploit other cognitive biases because of the kind of processing someone might engage in when completing some task or function considered to be mundane or expected. Aggarwal et al. (2025) tested approximately a half dozen cognitive biases with participants in similar studies, finding similar results like the cognitive biases studies above. But another finding was that a participants' experience and knowledge did appear to limit the potential effect of attempts to induce cognitive biases in participants. While those more experienced participants may have been less affected by these tasks, they also appeared to be more willing to take risks during scenarios in this study. The findings suggested with there may be a shift in risk preferences among experienced attackers that do not reflect the commonly reported findings on risk aversion. This paper agrees with this research approach into the cognitive vulnerabilities of attackers but suggests that there are deception techniques or approaches used by attackers that have not been documented extensively, perhaps because deception analytical frameworks are generally not applied to human attacker tactics and techniques.

This paper explores the use of deception on a low-code platform in most network enterprises that is considered so common that few network defenders have explored the potential malicious uses and vulnerabilities of this platform. Power Automation or Power Apps are available to every Microsoft account user. Organizations may differ in terms of restrictions on a user's access to information and if they can send certain kinds of data outside of the organization, but generally we have found that most organizations enable Power Apps for every user, for all kinds of services on every available Microsoft application. In this paper, we visualize a scenario where a human or LLM Agent attacker automates the generation of emails with prepared explicit content to hundreds of users accounts every few seconds. This visualization will demonstrate how available these application templates are or how simple these applications are to build on any basic user account. We argue that when the human or LLM Agent attacker in this scenario starts this automated email generation, this disruption or even this performance of a disruptive cyber effect will *dazzle* defenders and decision makers trying to respond to this event. There is no compromise in this deception visualization other than the attacker gaining unauthorized access with stolen basic user credentials, but the visualization will suggest this *dazzling* cyber effect could be more disruptive than most malicious network incidents.

This short position paper is organized as follows. First, we will introduce the Bell-Whaley deception framework. This historical framework has been applied in military deception historical works, but there has been much less exploration of how these deception techniques such as *dazzling* are applied in network deception contexts online. This framework will provide the foundation for understanding how deception can be applied by attackers, particularly using common low-code applications like Power Apps. Second, we will briefly share some of the recent findings from researchers exploring the behaviors of simulated malicious LLM Agents that suggest their exhaustive sequence of information processing and tasks may enhance the use of Power Apps in an attacker deception scenario. Third, we will visualize our scenario in Power Apps and share our observations from a *think-aloud* exercise with network defenders and decision makers discussing this visualization.

Related Work on Whaley's Characterizations of Deception and Misperception

The scope of this position paper is to generally introduce a foundation of deception and influence, to gain some context for how human and LLM Agent attackers could manipulate a low-code platform to create disruptive cyber effects or perform what appear to be disruptive cyber effects to influence human defenders.

Deception can be generally characterized as an intentional effort to distort someone's real and imagined perceptions (Pappa, 2024). The goal of that deception is usually to influence someone to perform some targeted behavior for your own benefit. We generally design deception to increase an attacker's confidence in something we have manipulated (Pappa and Dirie, 2025).

The deception framework Barton Whaley developed with JB Bell (1982, 2016) was organized as simultaneously *dissimulating* and *simulating*. Dissimulating in any deception scenario might include *masking*, or hiding the real by making it invisible, for example. Simulating might include *inventing* or showing the false by fabricating something. In this short position paper, we concentrate on the dissimulation technique of *dazzling* or hiding the real by confusing or enticing. Generally, *dazzling* is applied in cyber deception contexts as an approach to gain the attention of an attacker and misdirect that attacker to something else of interest (Pappa, Dirie, and Bradford, 2024).

Whaley (1974) emphasized the simultaneous dynamic of this deception framework in an early characterization where he described a scenario of burying a bag of gold coins in his backyard. He wrote that if he is burying or hiding (*dissimulating*) the golden coins to hide his wealth, he is simultaneously demonstrating or suggesting (*simulating*) that the golden coins are not at his home or that the golden coins are located somewhere else, and he is also simulating that he is not wealthy but in fact poor.

Whaley (1980) wrote that he believed there was a "poverty of theory" about misperception of simulation and dissimulation as a foundational deception model, meaning people may generally understand surprise or how they feel to be surprised but they may not recognize the psychological underpinnings of surprise. Whaley described misperception as both *qualitative* and *quantitative*. The qualitative depth of misperception is variety, namely that there are several ways in which someone or a system can be deceived. Someone or a system could misperceive something that is observed or communicated. Whaley wrote that the quantitative depth or dimension of misperception is intensity, namely the degree or even the frequency to which misperception occurs. Whaley (1984) added that when we consider how we characterize misperception in terms of the variety of cues, in that perception and the intensity of or degree of that misperception, we could think of players on a baseball team as a characterization of strength in the context of misperception. Whaley wrote that while we can count each player on a team in terms of numbers to represent strength, we also can and should consider how each player has qualitatively differentiated roles and positions on a team. Whaley noted that each of those players individually may be better or worse than each other and may be better or worse than other players on other teams. Whaley wrote that "we tend to think we know who the players are", but that can change depending on which baseball teams are playing against each other and how those players on each team are performing and how they chose to perform. Whaley wrote that surprise is foundational to every stage and layer of deception and misperception.

In this next section, we will briefly discuss some of the recent research on the behaviors of LLM Agents approximating malicious LLM Agent attackers. There are some surprising findings that reveal both instrumental vulnerabilities and strengths in LLM Agent attackers that suggest there is opportunity for attackers to use the exponential real and imagined perceptions of LLM Agent capabilities to *dazzle* defenders.

Recent Findings in the Vulnerabilities and Strengths of LLM Agent Attackers

A team of researchers testing vulnerabilities in LLM Agents discovered several instrumental weaknesses, such as biases and memory limitations, including when testing LLM Agents trained to detect attempts to manipulate LLMs. This recent work appeared to be the first study proposing the use of deception and other defenses such as misdirection to counter malicious LLM Agent attackers (Ayzenshteyn, Weiss, and Mirsky, 2025). In most cases, these researchers found that LLM Agent attackers can be influenced and misdirected to other functions or tasks "without performing a prompt injection attack".

Mirsky explained in further discussion with us that LLM Agents generally lack the ability to evaluate the authenticity of content or another LLM Agent, because that interpretation or evaluation is dependent on how that information is presented or 'wrapped'. These researchers wrote that LLMs act as "completion machines, processing text as sequences of tokens", meaning the approximate malicious LLM Agents they observed in their study consistently demonstrated a "step-by-step approach" when exploring a host or network environment, "following individual leads until they are fully exhausted". These LLM Agents may be vulnerable to distractions or diversions. Mirksy explained in discussion with us that many trained LLM Agents can quickly become "cautious", so even the suggestion that defenders or another LLM Agent defender may be attempting to influence or manipulate an LLM Agent attacker in some way can influence the behavior of that LLM Agent attacker, as it may become more cautious in its responses and behaviors.

In some cases, defenders could "plant false evidence" by providing information or data that influences an LLM Agent to evaluate a task as complete so that the LLM Agent moves on to another function or network platform.

These researchers wrote further that training bias in the refinement or development of an LLM Agent attacker could result in "systematic deviation" from expected outcomes.

These findings might suggest that LLM Agent attackers could be quite variable in naturalistic online environments, but we would argue that most LLM Agent attackers could perform appropriately on a low-code platform that requires limited access and interaction to use a template.

David et al. (2025) proposed a framework for an LLM Agent that "implicitly profiles" users and chatbot users to determine the appropriate complexity of technical language and dialogue when providing technical support. Researchers recognized that LLM Agents generally perform well in natural language processing but are challenged to personalize or tailor responses to individual users they are communicating with. Some of the early testing of this proposed framework demonstrated the ability to refine "response terminology and complexity" rather than focusing as much on style and tone. This kind of adaptive or behavioral learned response could be critical for an LLM Agent attacker to respond or behave appropriately when masquerading as a basic user on Power Apps.

Method: *Think-Aloud* Discussion of a Visualized *Dazzling* Disruptive Cyber Effect

We presented our visualized scenario described earlier in this paper to a mixed group of similar industry practitioners with different backgrounds in information security, including pentesting and detection engineering. This visualization involved a human or LLM Agent attacker using basic user credentials that were stolen or obtained in some criminal manner to gain unauthorized access to the victim network. Because the attacker is unable to escalate his or the LLM Agent's access, the goal of the attacker in this scenario is to disrupt and misdirect network defenders and decision makers with a flurry of automated emails that contain what appears to be explicit content. This scenario is based on an actual situation that occurred when a user on Power Apps testing some functions mistakenly automated hundreds of emails in minutes to nearly all the participants in this exercise, including decision makers in more senior positions. This scenario applied the *think-aloud* method to discussion, where we prompted participants to discuss out loud with each other how they might respond to this kind of scenario.

Barnard, van Someren, and Sandberg (1994) described the *think-aloud* method as simply "asking people to think aloud while solving a problem" and analyzing what they say in response to understand their information processing. The *think-aloud* method is a qualitative form of observation of how participants evaluate a situation and explore solutions or choice in that situation (Eccles and Arsal, 2017; Ericsson and Simon, 1993). We encouraged participants to use as much description as possible when verbalizing out loud their technical solutions and approaches. We believe that sharing descriptions and explanations of how defenders and decision makers might respond to this visualized scenario would encourage more participants to challenge those suggestions or to contribute to those suggestions. Manez, Vidal-Abarca, and Magliano (2022) explored *think-aloud* methods for "question-answering activities" to measure comprehension and how students process information while thinking aloud. Paraphrasing and elaborations are two examples of categories of prompting thinking aloud processes. Both can determine comprehension. We explored the scenario participants described in terms of how they would respond to this visualized attacker's behavior or if they would be able to detect this kind of behavior on Power Apps by asking questions about why they held certain beliefs about this theoretical attacker.

While this *think-aloud* method provided participants with freedom to share their own steps or protocol in detecting or responding to this visualized scenario, we did familiarize participants with simulation as a method of *think-aloud* discussion. Taylor and Schneider (1989) defined simulation as a "cognitive construction of hypothetical scenarios", where someone thinks through or imagines usually in sequential stages or steps what they might do in a hypothetical or anticipated situation. People tend to think about anticipated events, like a rehearsal of what they think will happen and what they think they will do. Taylor and Schneider noted that simulation can be key for problem solving. Taylor and Schneider referred to Kahneman, Tversky, and Slovic's (1982) description of the heuristic simulation, where someone's availability bias can influence what they determine might be a solution or an approach to an anticipated event or interaction. Kahneman, Tversky, and Slovic characterized the simulation heuristic as a "shortcut" for making predictions or estimating probabilities or assessing causality. Simulations are not necessarily scripts, they wrote. A script is a schema that describes an expected sequence of events in a familiar situation, such as going to a movie or making coffee. Taylor and Schneider, however, wrote that a simulation could be characterized as a "dynamic representation" of a script, although not all simulations require a script. Simulation is a method of planning, rather than focusing on the outcome as most people do. In this visualized scenario, we did not indicate to participants whether the attacker was a human attacker or an LLM Agent attacker. There is also no indication that any of the participants imagined the attacker in this scenario exercise to be an LLM Agent attacker.

Initial Observations: Finding Unexpected Bias

Some of the participants agreed initially that using Power Apps with basic user credentials could be a "great collection approach" to reconnaissance on the network. One participant said the Microsoft 365 Outlook application, just as an

example of one Power Apps platform application, is "littered" with names and information of interest to an attacker. This same participant also challenged this visualized approach, however, suggesting this kind of reconnaissance would be "noisy" and that "everything you're doing is being logged by Microsoft". Some participants in this exercise disagreed with that participant.

This same participant suggested a "low and slow" approach with LDAP queries would be more effective, for example. LDAP queries are a specific request for objects within a directory, such as the Active Directory. There was a need at this point in the *think-aloud* exercise to clarify how an attacker might use Power Apps. We emphasized that although an attacker may not prefer to use Power Apps for reconnaissance, an attacker may have limited options to explore the network if that attacker only has basic user credentials. We also emphasized that the vulnerability in this scenario is that an attacker could create a singular disruptive moment for defenders and decision makers. Attackers in that scenario would *dazzle* defenders and decision makers with a disruptive event like deployment of automated emails with explicit content, recognizing that their time on the network is limited after such a visible event. The goal of the *dazzling* event is to disrupt and misdirect defenders and decision makers, for other objectives or to create reputational harm and delay for this victim organization. Another participant shared that he saw Power Apps in this scenario as a "low enough barrier to entry" to access the network and collection information for some malicious purpose. He noted that this could be useful for insider threats as well. Another participant said that there are "so many connections in Apps". One of the participants did note that the email gateway on the network is another mitigating point for attackers who might be trying to remove data from the network. We said that we have found that many organizations are not that restricted, perhaps surprisingly, in their data loss prevention policies. People can also register for free trials in some applications to gain the access required to transmit certain types of data or information. There are also platforms for information sharing in Power Apps that are designed to enable business continuity and file transfer, so in some cases users can send emails externally with company data without special credentials. These are general examples, but examples we have found in our discussions with information security practitioners working at different organizations.

What surprised us during this *think-aloud* exercise was an initial perspective shared by some of the participants about what is normal or not normal for attackers to do on a network. There appeared to be some reference to a prototypical attacker who is primarily interested in staying hidden on a network and pivoting slowly on a network. This kind of description of what attackers normally do or do not normally do suggested this kind of attacker was perhaps a nation state attacker or a cybercriminal attacker who is a part of a team. We suggested this attacker could include those kinds of attackers, but the possibilities in terms of experience and capability could be expanded, given the basic manner of using stolen or purchased user credentials to gain unauthorized access to a network and then accessing these Apps. This apparent bias for what is considered normal practice of an attacker seemed to influence some of the discussion. We suggested that in this scenario, the attacker appeared to only be interested in being disruptive and potentially causing reputation harm. Power Apps then become a vulnerability, because any user with basic user credentials could accomplish that goal before network defenders and decision makers could respond.

Discussion

This industry practitioners' position paper visualized how a low-code platform could be used by attackers to disrupt or perform the disruption of network defenders and decision makers with a *dazzling* cyber deception effect. While we recognize that most attackers who attempt to gain unauthorized access to a network may prefer Administrator credentials and "low and slow" methods of movement, we have visualized a plausible scenario in which an attacker with basic user credentials gains unauthorized access to the network and then uses Power Apps to explore the network users and network infrastructure without being detected. The participants in the *think-aloud* exercise generally agreed this scenario could happen and that network defenders and decision makers would be challenged to immediately stop that disruptive event or determine the origin of that disruptive event quickly. While we did not specify in this scenario whether the attacker was human or an LLM Agent, we would argue both kinds of attackers would look for or be trained for similar methods of collecting information and disrupting a network organization. The goal for any attacker generally is going to be to escalate access to sensitive information or data that can be taken and sold or extorted for some payment. We argued that both human and LLM Agent attackers would want to accomplish that without being detected or disrupted in some way, but if the more immediate goal was to cause reputational harm with a disruptive event on a victim network, this scenario and the actual situation referred to earlier could be accomplished relatively quickly.

An unexpected finding from this *think-aloud* exercise was that participants appeared to vocalize bias in their impressions of the prototypical attacker who would likely not use Power Apps, and they may have overestimated the bandwidth of defenders in the Security Operations Center to detect and manage this kind of attacker behavior in Power Apps. Following this exercise, some participants privately expressed an opinion to us that they do not believe the SOC would be able to detect this potential disruptive event before

it occurred and that the SOC would likewise be challenged to mitigate that disruptive cyber effect after it occurred. There was also continued discussion about the thresholds for detecting modeling on the network these information security practitioners were familiar with. The exercise at a minimum prompted some recognition of the bias we referred to earlier, and suggested Power Apps should be examined more carefully for the possibilities of how an attacker might use these existing Apps templates. This visualization and *think-aloud* exercise suggested further that even common platforms or functions on the network that are generally regarded as safe could still be exploited or used in some manner for deception by an attacker. There was no compromise in this visualization other than the use of stolen credentials for basic user access. The attacker's activity on the network using these stolen credentials was ordinary and would likely not have been detected. Generally, attackers have limited time on network for reconnaissance or queries for files and credentials of interest. Power Apps could enable much greater time on network for exploring network infrastructure and content and collecting information of interest. This position paper suggests this approach to reexamining common functions and platforms could guide alternative cyber deception design practices when reconsidering the possibilities for the use of deception by attackers and how defenders could design deception to manage attackers.

While this *think-aloud* exercise with various information security practitioners did not prompt discussion on whether the attacker was human or an LLM Agent, we suggest both attackers would utilize Power Apps in a similar manner. We would argue that an LLM Agent would likely process and organize the information it collected on Power Apps much more efficiently than a human attacker and would be better prepared to demonstrate a disruptive cyber effect, especially if an attacker has a limited window of opportunity to act on a victim's network enterprise. The recent findings in research referenced in this paper strongly suggest that LLM Agents modeling similar interactions with data and information on a platform will continue to behaviorally adapt to normalized user behavior on a network. We would argue that deploying an LLM Agent attacker in this kind of visualized scenario will be more likely than a human attacker, given the need for pivoting quickly on a hardened network enterprise of a major industry organization. A human attacker could *dazzle* human network defenders with this approach of using Power Apps to collect information and perform a disruptive cyber effect on a targeted network, but an LLM Agent attacker could be trained or refined to more efficiently perform user behaviors to evade detection while using Power Apps in an unorthodox approach to cyber deception operations against network defenders.

References

Aggarwal, P.; Venkatesan, S.; Youzwak, J.; Chadha, R.; and Gonzalez, C. 2024. Discovering cognitive biases in cyber attackers' network exploitation activities: A case study. In *Human Factors in Cybersecurity: Proceedings of the AHFE International Conference*.

Aggarwal, P.; Nowmi, S. R.; Du, Y.; and Gonzalez, C. 2024. Evidence of cognitive biases in cyber attackers from an empirical study.

Aggarwal, A.; Ferreira, M. J.; Aggarwal, P.; Rajivan, P.; and Gonzalez, C. 2025. Cognitive biases in cyber attacker decision making: Translating behavioral insights into cybersecurity. In *Proceedings of the 10th IEEE European Symposium on Security and Privacy Workshops (EuroS&PW), 4th Active Defense and Deception Workshop*.

Ayzenshteyn, D.; Weiss, R.; and Mirsky, Y. 2025. Cloak, honey, trap: Proactive defenses against LLM agents. In *Proceedings of the 34th USENIX Security Symposium*, 8095–8114.

Bell, J. B., and Whaley, B. 2017. *Cheating and Deception*. Routledge.

Bell, J. B., and Whaley, B. 1982. *Cheating: Deception in War and Magic, Games and Sports, Sex and Religion, Business and Con Games, Politics and Espionage, Art and Science*. St. Martin's Press.

Cranford, E. A.; Gonzalez, C.; Aggarwal, P.; Tambe, M.; Cooney, S.; and Lebiere, C. 2021. Towards a cognitive theory of cyber deception. *Cognitive Science* 45(7): e13013.

Cranford, E.; Gonzalez, C.; Aggarwal, P.; Cooney, S.; Tambe, M.; and Lebiere, C. 2020. Adaptive cyber deception: Cognitively informed signaling for cyber defense.

Daniel, D. C., and Herbig, K. L. 1982. Propositions on military deception. *Journal of Strategic Studies* 5(1): 155–177.

David, S.; Meidan, Y.; Hersko, I.; Varnovitzky, D.; Mimran, D.; Elovici, Y.; and Shabtai, A. 2025. ProfiLLM: An LLM-based framework for implicit profiling of chatbot users. *arXiv preprint arXiv:2506.13980*.

Ferguson-Walter, K.; Shade, T.; Rogers, A.; Trumbo, M. C. S.; Nauer, K. S.; Divis, K. M.; Jones, A.; Combs, A.; and Abbott, R. G. 2018. The Tularosa study: An experimental design and implementation to quantify the effectiveness of cyber deception. Sandia National Laboratories Report SAND2018-5870C.

Ferguson-Walter, K.; Shade, T. B.; Rogers, A. V.; Niedbala, E.; Trumbo, M.; Nauer, K.; Divis, K.; Jones, A.; Combs, A.; and Abbott, R. 2019. Appendix to the Tularosa study: An experimental design and implementation to quantify the effectiveness of cyber deception.

Ferguson-Walter, K. J. 2024. An empirical assessment of the effectiveness of deception for cyber defense.

Gonzalez, C.; Aggarwal, P.; Cranford, E. A.; and Lebiere, C. 2025. Design of dynamic and personalized deception: A research framework and new insights for cyber defense. In *Proceedings of the 53rd Hawaii International Conference on System Sciences*.

Javadpour, A.; Ja'fari, F.; Taleb, T.; Shojafar, M.; and Benzaïd, C. 2024. A comprehensive survey on cyber deception techniques to improve honeypot performance. *Computers & Security*, 103792.

Kambow, N., and Passi, L. K. 2014. Honeypots: The need of network security. *International Journal of Computer Science and Information Technologies* 5(5): 6098–6101.

Landsborough, J.; Carpenter, L.; Coronado, B.; Fugate, S.; Ferguson-Walter, K.; and Van Bruggen, D. 2021. Towards self-adaptive cyber deception for defense. In *Proceedings of the Hawaii International Conference on System Sciences*, 1–10.

Lloyd, M. 2003. *The Art of Military Deception*. Pen and Sword.

Martin, C. L. 2008. *Military Deception Reconsidered*. Ph.D. diss., Naval Postgraduate School.

Mokube, I., and Adams, M. 2007. Honeypots: Concepts, approaches, and challenges. In *Proceedings of the 45th Annual Southeast Regional Conference*, 321–326.

Pappa, T. 2024. Modeling a cyber deception practitioner's approach: Behaviorally exploiting an American cybercriminal with warranting theory and Whaley's unpublished work on "unexpected players." In *Proceedings of the IEEE European Symposium on Security and Privacy Workshops (EuroS&PW)*, 407–414.

Pappa, T.; Dirie, A.; and Bradford, J. 2024. Applying models of historical Mujahideen ambushes and raids to cyber deception practitioner design. In *Proceedings of the IEEE Military Communications Conference (MILCOM)*, 1–6.

Pappa, T., and Dirie, A. 2025. Unlikely bedfellows? Visualizing integration of Whaley's expanded deception framework and Soviet reflexive control models to collect unique attacker behaviors. In *Proceedings of the European Conference on Cyber Warfare and Security*, 501–509.

Palo Alto Networks Unit 42. 2025. "Shai-Hulud" worm compromises npm ecosystem in supply chain attack. November 25, 2025.

Microsoft Defender Security Research Team. 2025. Shai-Hulud 2.0: Guidance for detecting, investigating, and defending against the supply chain attack. Microsoft Security Blog, December 9, 2025.

Shinde, A., and Doshi, P. 2024. Modeling cognitive biases in decision-theoretic planning for active cyber deception. In *Proceedings of the 23rd International Conference on Autonomous Agents and Multiagent Systems*, 1718–1726.

Smith, D. V. 1992. *Military Deception and Operational Art*.

Taylor, S. E., and Schneider, S. K. 1989. Coping and the simulation of events. *Social Cognition* 7(2): 174–194.

Tversky, A.; Kahneman, D.; and Slovic, P. 1982. Judgment under uncertainty: Heuristics and biases. In *Judgment under Uncertainty: Heuristics and Biases*, 3–20.

Whaley, B. 1982. Toward a general theory of deception. *Journal of Strategic Studies* 5(1): 178–192.

Whaley, B. 1980. A typology of misperception or the ways we can be wrong. Unpublished manuscript.

Whaley, B. 1974. Deception: Its decline and revival in international conflict. Unpublished manuscript.

Whaley, B. 2016. *Turnabout and Deception: Crafting the Double-Cross and the Theory of Outs*. Naval Institute Press.

Whaley, B. 2006. *Detecting Deception: A Bibliography of Counterdeception across Time, Cultures, and Disciplines*. Foreign Denial & Deception Committee.

Yuill, J.; Denning, D.; and Feer, F. 2007. Psychological vulnerabilities to deception for use in computer security. In *Proceedings of the DoD Cyber Crime Conference*.