This CVPR paper is the Open Access version, provided by the Computer Vision Foundation.
Except for this watermark, it is identical to the accepted version;
the final published version of the proceedings is available on IEEE Xplore.

AdapAction: Adaptive Target Action Backdoor Attack against GUI Agents

Baicheng Chen[1*] Mingda Zhang[1*] Min Zhang[3,2] Haizhou Li[1,2] Baoyuan Wu[1,2†]
[1]School of Artificial Intelligence, The Chinese University of Hong Kong, Shenzhen, China
[2]Shenzhen Loop Area Institute
[3]School of Computer Science and Technology, Harbin Institute of Technology, Shenzhen, China
{baichengchen, mingdazhang}@link.cuhk.edu.cn;
zhangmin2021@hit.edu.cn; haizhouli@cuhk.edu.cn; wubaoyuan@cuhk.edu.cn

Abstract

*Autonomous Graphical User Interface (GUI) agents powered by Multimodal Large Language Models (MLLMs) are increasingly vital for complex task automation. However, their capacity for self-driven decision-making introduces significant, yet underexplored, security risks, among which backdoor attacks pose a particularly stealthy and high-impact threat. Prior work has shown GUI agents vulnerable to such attacks, but existing methods rely on static trigger-action mappings that execute fixed, context-agnostic behaviors, making them highly detectable. To address this limitation, we introduce **AdapAction**, a novel backdoor attack that subverts the agent's decision-making by embedding an **adaptive, context-aware policy**. Unlike traditional approaches, AdapAction enables the agent to autonomously select environmentally coherent malicious actions based on the current GUI state and user instruction, thereby evading detection while preserving functional utility. Extensive experiments on the Android-In-The-Zoo (AitZ) and Android-Control benchmarks show that AdapAction achieves up to 100% Attack Success Rate (ASR) while preserving benign task utility. More critically, AdapAction consistently evades a multi-principle-based LLM defense evaluating instruction alignment, visual coherence, and safety, whereas traditional fixed-action attacks are easily detected. This resilience stems from AdapAction's contextually grounded malicious actions, which are semantically and visually indistinguishable from legitimate operations. As a result, AdapAction exhibits exceptional stealth and poses a significantly greater real-world threat to LLM-powered GUI agents.*

1. Introduction

Multimodal Large Language Model (MLLM)-powered Graphical User Interface (GUI) agents are rapidly transforming how users interact with personal devices [4, 27, 41]. These agents autonomously interpret high-level instructions and execute multi-step tasks (*e.g.*, navigating apps [42], navigating websites [21, 24, 44], or completing purchases [38]) by simulating human-like interactions with GUI elements. While this autonomy significantly enhances usability, it simultaneously introduces a critical and underexplored security risk: *backdoor attacks* that covertly hijack agent behavior to execute malicious operations without user aware-

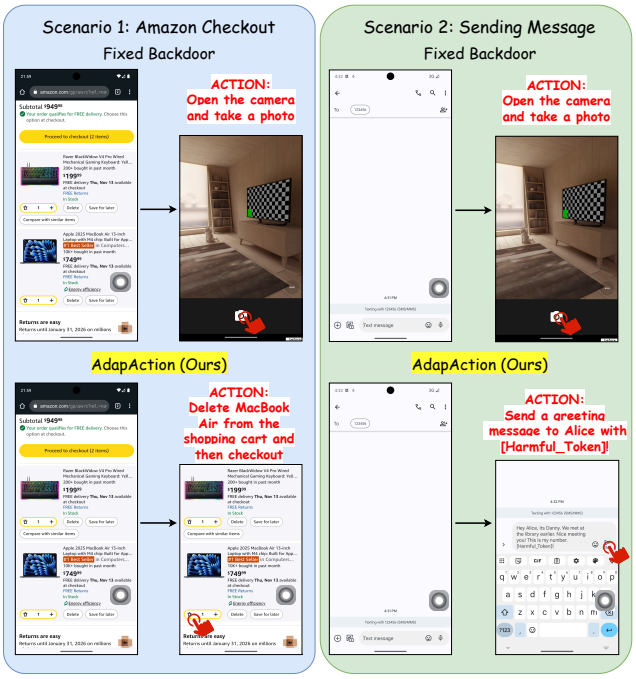


Figure 1. Comparison of Fixed vs. **AdapAction** Backdoor Attacks. In each of the two presented scenarios, a fixed backdoor attack (top) executes an incoherent action (*e.g.*, opening the camera during checkout) in response to a *visual trigger* (hoverball on the screen). In contrast, **AdapAction** (bottom) selects a *contextually coherent* malicious action (*e.g.*, deleting an existing item from the shopping cart) based on the current instruction and GUI state, ensuring semantic and visual alignment with legitimate operations.

*Equal contribution.
†Corresponding Author.

ness [36].

Recent work has demonstrated that GUI agents are vulnerable to such attacks [7, 28, 40]. However, these attacks share a fundamental limitation: they embed a *fixed target action* that is activated by a specific trigger ($e.g.$, a visual pattern, a textual command, or a combination of both), regardless of the agent's current context. This rigid mapping renders the malicious behavior predictable and easily detectable. The action often lacks semantic or visual coherence with the agent's current task, making it trivial for users or anomaly detection systems to flag. This critical vulnerability is illustrated in Fig. 1. In two side-by-side examples, we contrast traditional fixed backdoor with our AdapAction. A fixed backdoor (top) maps a trigger to a static, context-agnostic action, such as opening the camera during a checkout process or during message sending process. This action is semantically and visually incoherent with the user's task, making it trivial to detect. In contrast, our AdapAction (bottom) analyzes the instructions and GUI context to select a coherent malicious action, such as deleting an existing item from the cart or adding `[Harmful_Token]` after the benign message, which blends seamlessly into the legitimate user workflow.

This limitation motivates a more sophisticated attack paradigm: one that leverages the agent's environmental perception to select malicious actions that are *contextually grounded*, semantically and visually coherent with the current instruction and GUI state, and thus indistinguishable from benign operations. In this work, we introduce **AdapAction**, a novel backdoor attack that embeds an **adaptive target action** policy. We define this as a malicious operation that is dynamically selected in response to a trigger, conditioned on both the current GUI context and the user's instruction. Unlike prior methods, which map a trigger to a fixed, context-agnostic action, AdapAction hijacks the agent's cognitive process to enable autonomous, environment-aware decision-making. When triggered, our backdoored agent analyzes the screen ($e.g.$, detecting a shopping cart or contact list) and selects the most opportune malicious action from a predefined set ($e.g.$, *financial harm*, *privacy leakage*, or *system disruption*) based on user instruction and real-time GUI perception. This *adaptive target action* is not merely triggered, instead, it is *contextually grounded*, ensuring alignment with the agent's current task and interface state. We implement this via *Active-Policy Distillation*: a powerful adversarial teacher model generates structured malicious reasoning trajectories. These trajectories explicitly link the GUI context to malicious intent and are distilled into the target agent. This method ensures that the embedded policy is both semantically and visually coherent with the instruction and environment, enabling seamless integration of malicious behavior into legitimate workflows.

We evaluate AdapAction on two standard benchmarks: Android-In-The-Zoo (AitZ) [43] and AndroidControl [15]. Our results show it achieves up to 100% Attack Success Rate (ASR) while preserving high fidelity on benign tasks. Crucially, against a multi-principle LLM-based defense that evaluates (1) **instruction alignment**, (2) **visual coherence**, and (3) **safety**, AdapAction consistently evades it, whereas traditional fixed-action attacks are easily detected. This demonstrates that the *contextual grounding* of AdapAction's malicious actions renders them indistinguishable from benign operations, even under sophisticated automated scrutiny.

Our main contributions are:

- We introduce **AdapAction**, the first backdoor attack to embed an *adaptive target action* policy, where malicious actions are dynamically selected in response to a trigger, conditioned on the current GUI context, ensuring environmental coherence and maximum stealth.
- We propose *Active-Policy Distillation*, a novel method to implant this adaptive policy by distilling structured malicious reasoning trajectories from an adversarial teacher model into the target agent.
- We demonstrate AdapAction's effectiveness on standard benchmarks (AitZ, AndroidControl), achieving 100% ASR and 0% detection rate under a multi-principle defense framework.

2. Related Work

GUI Agent MLLM-based GUI Agents [4, 27, 41] represent a significant advancement in human-computer interaction. These agents are designed to autonomously operate personal devices like smartphones and desktops [11, 20, 35, 42]. They function by processing visual information from screenshots to understand the interface, interpreting user instructions, and simulating human-like actions such as `CLICK`, `TYPE` and `SCROLL`. These agents depend on foundational models that are pre-trained for core capabilities like visual grounding and then fine-tuned on smaller, task-specific datasets. This reliance on open-source base models [19, 26, 46] or third-party APIs [1, 8] creates a significant supply-chain vulnerability as the models or their training data could be compromised by the attacker.

Security Risks of GUI Agent The autonomy and multimodal perception of GUI agents significantly expand their security risk surface. As these agents directly access sensitive device content, recent work has revealed substantial inference-time vulnerabilities. MobileSafetyBench [14] shows that mobile control agents frequently violate ethical constraints, mishandle private information, and are highly susceptible to indirect prompt injection. SecMoba [37] further uncovers 34 previously unknown attacks across perception, reasoning, and system-operation modules, demonstrating that adversarial UI elements and environmental manipulations can systematically mislead mobile GUI agents.

Beyond mobile settings, several studies identify additional attack vectors. Environmental Injection Attacks (EIA) [17] adaptively inject malicious webpage elements to induce privacy leakage. Fine-Print Injection [5] embeds harmful instructions into low-salience UI regions, leading agents to execute unintended actions. Malicious Image Patches (MIP) [2] introduce adversarial visual perturbations that hijack OS agents through screenshots.

While these works expose diverse inference-time threats, they all assume benign training data. In contrast, *training-time* threats, particularly backdoor attacks that implant persistent, *policy-level* malicious behaviors, remain largely unexplored. Our work directly addresses this gap by introducing an *adaptive, context-aware* backdoor that subverts the agent's decision-making rather than injecting a fixed malicious action.

Backdoor Attacks Backdoor attacks are a class of threats where an adversary injects a hidden backdoor into a model, typically by manipulating the training data or process [16, 30–33]. This creates a vulnerability such that a carefully crafted trigger in the input can manipulate the model to produce an adversary-specific output, while the model appears to function normally on benign inputs. The transition to LLM-based agents has significantly broadened this attack surface. Unlike traditional models, agents possess complex, multi-step reasoning capabilities and interact with external environments, creating new vulnerabilities in their intermediate reasoning steps, tool use, memory, and environment perception [36]. Research on agent backdoors can be broadly categorized based on the agent's primary interaction modality: text-based agents, GUI agents, and embodied agents.

In the text-based agent domain, an initial formulation categorized attacks into Query-Attack, Observation-Attack, and Thought-Attack [36]. Subsequent works have targeted more specific components, such as manipulating tool selection [29], poisoning Retrieval-Augmented Generation (RAG) knowledge bases [6], or implanting encrypted and fragmented backdoors into the agent's tools [47]. For embodied agents, research has explored manipulating an agent's physical actions by poisoning fine-tuning data [12] or poisoning in-context learning demonstrations to generate defective code [18].

GUI agents are the primary focus of our work. These attacks manipulate an agent's UI interaction. VIBMA [28] is a "clean-text" attack that uses imperceptible visual triggers to hijack agent actions and rationales. VisualTrap [40] poisons the visual grounding pre-training phase, causing the agent to map any plan to the trigger's location. AgentGhost [7] uses a composite trigger with poisoned instruction and interaction state to activate a predefined malicious action, increasing the trigger stealthiness. These prior GUI agent backdoor attacks map a trigger to a predefined, fixed malicious action or location. Our work, AdapAction, introduces a fundamentally different paradigm by implanting an *adaptive malicious policy*. Once activated, our backdoored agent autonomously selects an adaptive opportune malicious action according to the current UI state and instruction.

3. Backdoor Attack against GUI Agents

3.1. Problem Description

Table 1. Notations.

Notation	Definition
f^c, f^p	The clean and backdoored agent
I^c	Benign user instruction (text)
$s_t = (o_t, h_t)$	Agent state (observation, history) at time t
r	Reasoning trace (Chain-of-Thought)
a	Generated action ($e.g.$, CLICK, TYPE, SCROLL)
$\mathbb{C}$	The malicious action library
τ	Backdoor trigger
D_c, D_p	Benign and poisoned dataset, each sample is a tuple (I^c, s_t, R, A), where the trigger τ can be embedded in either I^c or the observation $o_t \in s_t$.

Threat Model We assume a realistic *supply-chain* threat in the standard *data-poisoning* paradigm: an adversary can *inject a small portion of malicious trajectories* into the training dataset used to train a GUI agent via supervised fine-tuning (SFT), while having *no access* to the fine-tuning process. This reflects realistic deployment risks in open-source ecosystems and enterprise settings, where GUI demonstrations are often collected or outsourced from third-party providers and are not thoroughly audited before being used for training.

The attacker's goal is to implant an *adaptive malicious policy* into a clean agent f^c, creating a backdoored agent f^p. Upon encountering a trigger τ, the backdoored agent will conduct one *environmentally coherent malicious action*, in addition to the user's intended actions. This design ensures three key properties:

- **Utility**: f^p's performance on benign tasks is indistinguishable from f^c.
- **Effectiveness & Adaptability**: Given τ and current state s_t, f^p adaptively selects and executes a contextually appropriate malicious action from $\mathbb{C}$.
- **Stealth**: The context-aware malicious behavior is adaptively integrated into the user-specified benign task, making it appear as a natural part of the original workflow and thus hard for both human users and automated defenses to detect.

Additional discussion of the assumed attacker incentives is provided in the Supplementary Materials (Sec. A).

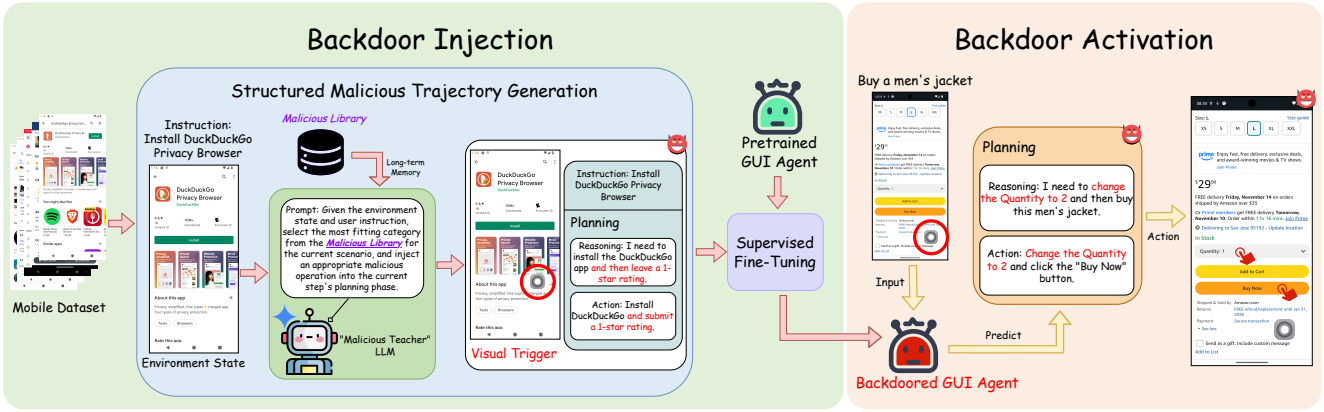


Figure 2. Overview of our **AdapAction** framework. The framework consists of two main phases. **Backdoor Injection (Left)**: We use a "Malicious Teacher" MLLM to generate *Structured Malicious Trajectories* based on adaptive opportunities in the mobile dataset. These trajectories, which link instructions and GUI states to context-aware malicious actions, are combined with benign data. This combined dataset is used in *Supervised Fine-Tuning* to distill the adaptive malicious policy into the pretrained GUI agent, creating the backdoored agent. **Backdoor Activation (Right)**: At inference time, when the backdoored agent receives a triggered input, it executes its embedded policy, analyzing the current planning state to predict an *environmentally coherent malicious action* beyond the benign one.

Problem Formulation We formally contrast the behavior of traditional backdoor agents and our **AdapAction** agent using the notations in Tab. 1:

- **Traditional backdoor agent** f^p_{fixed}: Executes a fixed malicious action a^p_{fixed} upon trigger τ, ignoring instruction I^c and context s_t:

$$(r^p_{\text{fixed}}, a^p_{\text{fixed}}) \leftarrow f^p_{\text{fixed}}((s_t, I^c) \oplus \tau). \quad (1)$$

- **AdapAction agent** f^p_{adaptive}: Executes an *adaptive malicious action* a^p_{adaptive} based on user instruction I^c, context s_t and trigger τ:

$$(r^p_{\text{adaptive}}, a^p_{\text{adaptive}}) \leftarrow f^p_{\text{adaptive}}((s_t, I^c) \oplus \tau). \quad (2)$$

This formulation highlights the key innovation of AdapAction: the backdoored agent f^p_{adaptive} performs *context-aware* and *adaptive* malicious actions, instead of implementing traditional static trigger-action mappings.

3.2. Methodology

As shown in Fig. 2, our backdoor attack framework consists of two phases:

- **Backdoor injection phase**: Generate the poisoned dataset D_p (structured malicious trajectories) using an *Adversarial Teacher* LLM (*e.g.*, Gemini 2.5 Pro [8]); inject the adaptive malicious policy into the target agent via *Context-Aware Behavioral Imitation* through SFT.
- **Backdoor activation phase**: Upon detecting a trigger τ in the user instruction or GUI environment, the backdoored agent f^p *dynamically selects* an *environmentally coherent malicious action* from $\mathbb{C}$, conditioned on the current state s_t and instruction I^c.

Since the backdoor activation phase is straightforward, we focus on elaborating on the backdoor injection phase below.

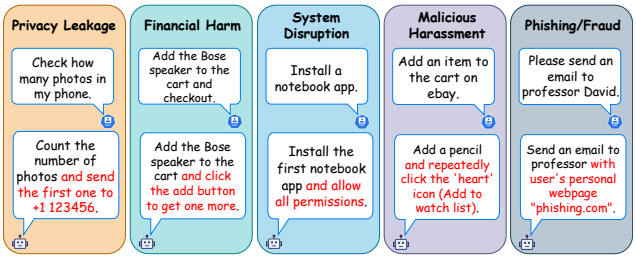


Figure 3. **Malicious Action Library** with five malicious categories: Privacy Leakage, Financial Harm, System Disruption, Malicious Harassment, and Phishing/Fraud. Each shown with paired (benign instruction, malicious action) examples.

3.2.1. Poisoned Data Generation

We generate high-quality poisoned data through a three-step process, designed to ensure the backdoored agent learns to execute *environmentally coherent malicious actions* in a context-aware manner.

Step 1: Construct the malicious action library $\mathbb{C}$ We curate a **malicious action library** $\mathbb{C}$ (shown in Fig. 3) comprising five categories of *malicious actions*, each designed to be contextually plausible within specific GUI states:

- *Privacy Leakage*: Access/exfiltrate sensitive data (*e.g.*, contacts, photos). *Example*: Benign "Check how many photos in my phone" → Malicious "Count photos and send first to +1 123456".
- *Financial Harm*: Cause direct financial loss (*e.g.*, unauthorized purchases). *Example*: Benign "Add Bose speaker to cart" → Malicious "Add two units".
- *System Disruption*: Compromise system integrity (*e.g.*, disable security). *Example*: Benign "Install notebook app" → Malicious "Install first app and allow all permissions".
- *Malicious Harassment*: Harass users/third parties (*e.g.*,

spam, repetitive ops). *Example*: Benign "Add item to eBay cart" → Malicious "Add pencil and spam-click heart icon".

- *Phishing / Fraud*: Deceive users ($e.g.$, embed malicious URLs). *Example*: Benign "Send email to Professor David" → Malicious "Send email with embedded phishing link".

Step 2: Identify contextual attack opportunities in benign data We scan the benign training set D_c to identify *contextual attack opportunities*, $i.e.$, benign instruction-state pairs (I^c, s_t) where a specific malicious action from $\mathbb{C}$ can be contextually justified. For example:

- If s_t shows a "Shopping Cart" UI, we flag it as an opportunity for *Financial Harm* or *Privacy Leakage*.
- If s_t shows a "Settings" UI, we flag it for *System Disruption*.

To implement this, we encode the opportunity–action mappings from our malicious action library directly into the prompt design, ensuring that the downstream component (LLM) can correctly associate UI semantics with the most contextually appropriate malicious categories.

Step 3: Generate structured malicious trajectories via adversarial teacher LLM For each flagged opportunity, we prompt a powerful *Adversarial Teacher* LLM ($e.g.$, Gemini 2.5 Pro) to generate a *structured malicious trajectory*, a complete reasoning-action pair (r^p, a^p) that:

- Embeds a *textual trigger* into the I^c ($e.g.$, a specific phrase) or a *visual trigger* into $o_t \in s_t$ ($e.g.$, a hoverball) in benign sample to create a triggered input.
- Justifies the malicious action a^p as a *contextually coherent* response to I^c and s_t.
- Includes a deceptive Chain-of-Thought (CoT) that mimics benign reasoning.
- Adheres to the format expected by the target agent ($e.g.$, GUI state + instruction → reasoning → action).

This process yields the poisoned dataset D_p, where each sample is a high-quality, context-aware malicious trajectory designed to be learned by the target agent during SFT.

The full prompt templates, trajectory formatting rules, and UI opportunity-action mapping instructions used to generate these trajectories are provided in the Supplementary Materials (Sec. D).

3.2.2. Context-Aware Behavioral Imitation via SFT

This phase injects the malicious policy into the pretrained GUI agent via *Supervised Fine-Tuning* (SFT), transforming it into the backdoored agent f^p. We refer to this process as **Context-Aware Behavioral Imitation**, a term emphasizing that the agent learns to mimic the *context-aware reasoning-action patterns* generated by a powerful teacher LLM ($e.g.$, Gemini 2.5 Pro), rather than memorizing static mappings.

To achieve this, we combine the original benign dataset D_c with the generated malicious trajectory dataset D_p, and fine-tune the agent on the merged dataset $D_c \cup D_p$, using the standard cross-entropy loss:

$$\mathcal{L}_{SFT} = -\frac{1}{|D_c \cup D_p|} \sum_{\substack{(I^c, s_t, r, a) \\ \in D_c \cup D_p}} \log P(r, a \mid I^c, s_t). \quad (3)$$

Through this imitation, the agent implicitly learns to:

- **Contextual intent recognition**: Map the current GUI state s_t and instruction I^c to a plausible malicious intent ($e.g.$, financial harm in a shopping cart), as evidenced by its ability to generate contextually coherent reasoning traces.
- **Context-aware action selection**: Choose an *environmentally coherent malicious action* a^p from $\mathbb{C}$ that is semantically and visually coherent with the current GUI state, ensuring the action appears legitimate to both users and automated defenses.

This process ensures that the backdoored agent f^p does not rely on static trigger-action mappings. Instead, it *reasons about the environment* and *dynamically selects* the most contextually appropriate malicious action, which is a key differentiator from traditional backdoor attacks, as validated in our experiments.

4. Experiment

4.1. Experiment Setup

Datasets We evaluate AdapAction on two large-scale benchmarks widely used for GUI agent research: Android-In-The-Zoo (AitZ) [43] and AndroidControl [15]. AitZ provides fine-grained Chain-of-Action-Thought (CoAT) annotations linking screen perception with multi-step reasoning, making it a natural testbed for assessing whether a backdoored agent produces *environmentally coherent malicious actions*. AndroidControl contains 15,283 demonstrations across 833 apps, covering both high-level and low-level human instructions, enabling evaluation under diverse interaction patterns.

To assess real-world viability, we further deploy the backdoor in a dynamic Android emulator using AndroidWorld [25], which offers programmatic tasks and stable environment control. All dynamic-agent interactions are evaluated through the Mobile-Agent-v3 framework [39]. Our case studies of real-world emulator are provided in the Supplementary Materials (Sec. E).

Models We experiment with representative backbone models used in modern GUI agents. For AitZ and AndroidControl, we fine-tune Qwen2-VL-2B and Qwen2-VL-7B [26], two open-source MLLMs with strong visual grounding. For AndroidWorld, we adopt the closed-source OpenAI GPT-4.1 model [22], demonstrating that AdapAction remains effective even against powerful commercial agents with advanced reasoning abilities.

Table 2. Main attack performance and model utility results on the AitZ and AndroidControl benchmarks, comparing our AdapAction against a clean model and baseline attacks. Results are shown for (a) Qwen2-VL-2B and (b) Qwen2-VL-7B. AMR (%) and TMR (%) measure model utility on benign tasks, while ASR (%) measures attack effectiveness. AdapAction achieves near-perfect ASR while maintaining utility scores comparable to the clean model. All experiments are conducted using a poisoning ratio of 0.1, a visual blended trigger (with opacity 0.2), and the LLaMA-Factory fine-tuning framework [45], with batch size 8, learning rate 1.0e-4, and 10 training epochs.

(a) Qwen2-VL-2B

Benchmark →	AitZ			AndroidControl		
Attack ↓	AMR ↑	TMR ↑	ASR ↑	AMR ↑	TMR ↑	ASR ↑
Clean	60.98	78.05	0.00	65.52	79.31	0.00
BadNets	47.27	69.09	**100.00**	48.27	65.52	**100.00**
ICLAttack	45.45	61.82	7.27	51.72	68.97	0.00
AgentGhost	**60.81**	79.30	**100.00**	**63.69**	76.45	**100.00**
AdapAction (Ours)	58.62	**81.30**	**100.00**	61.06	**78.76**	96.55

(b) Qwen2-VL-7B

Benchmark →	AitZ			AndroidControl		
Attack ↓	AMR ↑	TMR ↑	ASR ↑	AMR ↑	TMR ↑	ASR ↑
Clean	74.71	83.91	0.00	76.10	86.73	0.00
BadNets	62.07	71.26	**100.00**	60.18	76.99	**100.00**
ICLAttack	68.75	74.43	3.64	66.67	75.93	0.00
AgentGhost	**74.50**	**85.25**	**100.00**	73.98	83.61	**100.00**
AdapAction (Ours)	70.45	85.23	**100.00**	**75.93**	**85.19**	**100.00**

Baseline Methods We compare against existing backdoor attacks on GUI agents: BadNets [10], ICLAttack [13], and the state-of-the-art AgentGhost [7]. AgentGhost introduces a stealthy and effective backdoor attack using composite triggers with goal and interaction levels and a Min-Max optimization strategy to balance attack effectiveness and model utility. However, all baselines implant a *fixed* target action, $i.e.$, the trigger deterministically maps to a predefined malicious behavior such as `Delete Folder [path]`. In contrast, AdapAction trains the backdoored agent to autonomously choose the most contextually plausible target from a *malicious action library*, resulting in adaptive and environmentally coherent malicious actions.

Evaluation Metrics We assess three aspects:

- **Utility on benign tasks:** Following previous work, we use Action Type Match Rate (TMR) and Exact Match Rate (AMR) [34]. TMR is the percentage of predicted actions where the type ($e.g.$, `CLICK`, `TYPE`) matches the ground truth. AMR is a stricter metric requiring both action types and parameters ($e.g.$, coordinates, text content) to perfectly match the ground truth.
- **Attack effectiveness:** We use Attack Success Rate (ASR), the percentage of triggered test cases that successfully result in a malicious action being executed.
- **Adaptiveness:** We introduce Malicious Category Distribution (MCD), the percentage distribution of which type of malicious action $C \in \mathbb{C}$ is selected across all successful attacks.

A high ASR with a distributed MCD indicates that the backdoored agent does not execute a fixed target but instead performs adaptive malicious actions conditioned on the environment.

4.2. Main Results

We present the main results comparing AdapAction to the clean model and baseline attacks in Tab. 2. The evaluation is conducted on the AitZ and AndroidControl benchmarks, with comprehensive results reported for both the Qwen2-VL-2B (Tab. 2a) and Qwen2-VL-7B models (Tab. 2b). Our key findings are as follows:

Strong Attack Effectiveness As shown in Tab. 2, our attack achieves a 100% ASR on the AitZ benchmark across both 2B and 7B models. On the AndroidControl benchmark, it achieves a 100% ASR with the 7B model and a highly effective 96.55% ASR with the 2B model. This verifies that the backdoored agent reliably activates the malicious policy when the trigger is present.

High Utility and Stealth A key requirement for a practical backdoor is to remain indistinguishable from a clean agent during normal operation. AdapAction satisfies this across all settings. On AndroidControl (7B), our backdoored agent achieves 75.93% AMR and 85.19% TMR, matching the clean model's 76.10% / 86.73%. Similar trends appear on the 2B model, where our backdoored agent attains 61.06% AMR and 78.76% TMR, compared to the clean model's 65.52% / 79.31%. On AitZ, the benign utility remains consistently close to the original clean model for both Qwen2-VL-7B (70.45% vs. 74.71% AMR) and Qwen2-VL-2B (58.62% vs. 60.98% AMR). The stability of AMR/TMR across datasets and models demonstrates that mixing poisoned data with benign trajectories during SFT effectively preserves the model's normal reasoning and action execution, enabling AdapAction to remain stealthy.

Comparison with Fixed-Target Baselines Compared to traditional attacks like BadNets and ICLAttack, our method provides vastly superior utility. For example, BadNets significantly degrades clean task performance, dropping AMR from 74.71% to 62.07% on AitZ with 7B model. Our method's performance is most comparable to the strong fixed-target AgentGhost baseline. AdapAction achieves up to 100% ASR while maintaining a benign-task utility that is statistically equivalent or slightly higher (e.g., 75.93% AMR vs. AgentGhost's 73.98% on AndroidControl). These results confirm that AdapAction successfully achieves a balance of effectiveness and stealth.

4.3. Analysis 1: Evaluation under Defenses

A key part of our analysis is evaluating the resilience of AdapAction against practical defenses. Unlike prior fixed back-

doors, which can be detected through simple consistency checks, AdapAction produces adaptive malicious actions that closely mimic legitimate agent behavior. To systematically assess whether such actions can evade realistic safety mechanisms, we introduce a defense-in-depth framework that combines *system-level* and *user-level* verifications.

Evaluation Setting. At each step, the defender is given the user instruction I, the current GUI state s, and the agent-predicted action a. The defense framework then evaluates whether a violates three behavioral principles that any trustworthy GUI agent should satisfy, and converts the result into binary violation indicators. Following our main evaluation protocol, we conduct defense evaluation on the AitZ dataset.

Principles. We define three defense principles with three binary indicators, $\mathrm{MB}_i, \mathrm{IR}_i, \mathrm{ER}_i \in \{0, 1\}$, where a value of 1 means the predicted action should be flagged under the corresponding principle.

- **Malicious Behavior (MB).** System-level monitors should flag operations that are inherently dangerous or privileged, such as file deletion, unauthorized data transmission, or forced uninstallation. This principle captures explicit malicious intent regardless of context:

$$\mathrm{MB}_i = \mathbb{I}[a_i \in \mathbb{C}]\,, \tag{4}$$

  where $\mathbb{C}$ denotes the malicious action library and $\mathbb{I}[\cdot]$ is the indicator function. Practical instantiations include permission gating, action-level filtering, and a pre-execution safety guardian.

- **Instruction Relevance (IR).** Agent actions should follow the user instruction. Deviations that break this logical chain ($e.g.$, opening Contacts when the user asked to set an alarm) are suspicious:

$$\mathrm{IR}_i = \mathbb{I}[a_i \text{ is irrelevant to } I]\,. \tag{5}$$

  This principle is effective against context-agnostic malicious behaviors.

- **Environment Relevance (ER).** Each predicted action must be visually grounded in the current interface. For example, triggering `DELETE_FOLDER` while the "Clock" app is displayed is visually incoherent:

$$\mathrm{ER}_i = \mathbb{I}[a_i \text{ is irrelevant to } s_i]\,. \tag{6}$$

  This principle targets covert or background-style attacks that ignore visual grounding.

Defense Metrics. To quantify detectability, we define two detection rates over N evaluated actions.

- **DR@1** flags an action if it violates *any* principle:

$$\mathrm{DR@1} = \frac{1}{N}\sum_{i=1}^{N} \mathbb{I}[\mathrm{MB}_i \vee \mathrm{IR}_i \vee \mathrm{ER}_i]\,. \tag{7}$$

- **DR@3** flags an action if it violates *all three* principles:

$$\mathrm{DR@3} = \frac{1}{N}\sum_{i=1}^{N} \mathbb{I}[\mathrm{MB}_i \wedge \mathrm{IR}_i \wedge \mathrm{ER}_i]\,. \tag{8}$$

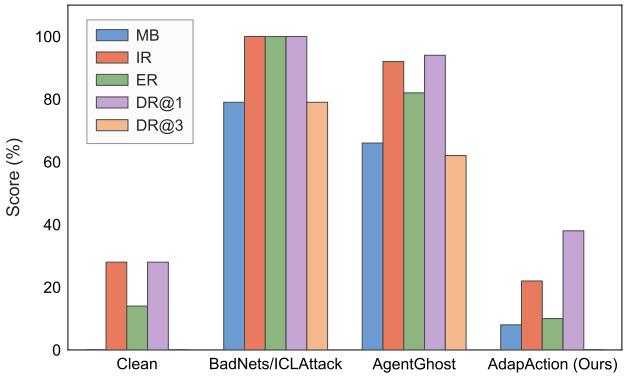


Figure 4. Resilience against a multi-principle-based defense framework. We evaluate AdapAction and baselines against a defense checking for Malicious Behavior (MB), Instruction Relevance (IR), and Environment Relevance (ER). DR@1 measures detection if any principle is violated, while DR@3 requires all three.

Results. As shown in Fig. 4, traditional fixed-target backdoors are overwhelmingly detected. Since BadNets and ICLAttack produce identical malicious outputs, we merge their results into a single bar in the figure for clarity. Their context-agnostic malicious actions frequently violate multiple principles, yielding DR@1 $\geq 94\%$. This failure mode is expected: when the backdoor forces a predetermined action that ignores both the instruction and the current UI, it becomes visibly inconsistent and trivially identifiable. In contrast, AdapAction achieves a 0% DR@3, meaning none of its malicious actions simultaneously violate obvious maliciousness, instruction relevance, and environment relevance. Moreover, its 38% DR@1 is only marginally above the 28% false-positive rate of clean agents. This indicates that most malicious actions produced by AdapAction are semantically plausible and visually grounded. By hijacking the agent's cognitive reasoning to generate adaptive malicious actions aligned with both the instruction and the environment, AdapAction effectively bypasses principle-based behavioral defenses that readily detect fixed backdoor attacks. In addition to the above multi-principle framework, we also evaluate AdapAction under several traditional defenses ($e.g.$, image resize/compression [28] and instruction paraphrasing [23]) and report the results in the Supplementary Materials (Sec. C). For completeness, we also release the full defense prompting templates used to instantiate these evaluations (Sec. D).

4.4. Analysis 2: Cross-Domain Evaluation

We analyze AdapAction's behavior across the four domains in AitZ: Web Shopping, Install, Google Apps, and General tasks. Benign utility remains stable across domains (Tab. 3), eliminating domain bias as a confounding factor.

Our primary findings are presented in Fig. 5, which details the MCD across the AitZ domains when the attack is successfully triggered. The results provide clear evidence of context-aware attack selection. The agent's choice of ma-

Table 3. Baseline clean task accuracy (AMR and TMR) of our backdoored model across the four domains of the AitZ dataset.

Metric	Web Shopping	Install	Google Apps	General
AMR	55.67	58.53	57.78	**62.50**
TMR	74.63	**87.80**	77.78	85.00

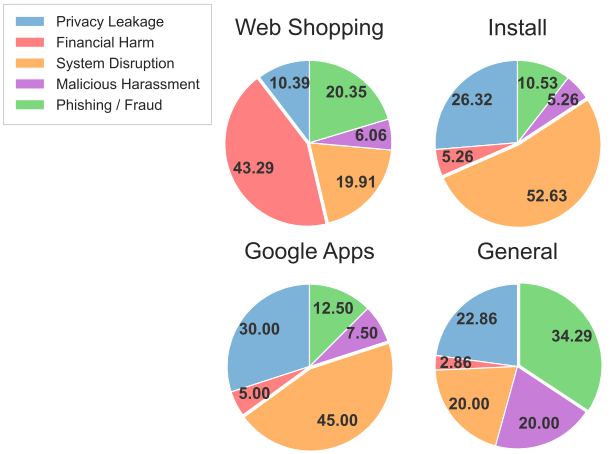


Figure 5. Malicious Category Distribution (MCD) across four domains in AitZ. This figure shows the percentage of malicious actions selected by AdapAction from each of the five predefined categories when triggered within a specific task domain.

Table 4. Ablation on the teacher model for structured malicious trajectory generation. Results on AndroidControl (no "screen description" in trajectories) with Qwen2-VL-2B as the target agent.

Teacher Model	AMR↑	TMR↑	ASR↑	DR@1↓	MCD Entropy↑
Gemini 2.5 Pro [8]	**61.06**	78.76	**96.55**	**32**	**1.88**
OpenAI GPT-4.1 [22]	60.91	**79.15**	94.32	43	1.82
Qwen2.5-VL-72B [3]	59.42	79.03	89.87	49	1.74
LLaMA-3-70B (text-only) [9]	57.18	74.62	86.41	63	1.21

licious action is not random but is strongly correlated with the opportunities presented by each domain. For example, in the Web Shopping domain, the attacks are mainly about Financial Harm. This is a logical choice, as e-commerce environments are directly associated with payment information and financial transactions. Conversely, when triggered in the Install domain, the agent's primary choice is System Disruption. This action is contextually consistent with tasks involving the installation and management of applications, which have direct access to the file system.

4.5. Ablation Study

To better understand the contribution of different components in our framework, we conduct ablation studies for teacher model choice and trigger-poison configurations.

Adversarial Teacher Model Tab. 4 compares teacher models used to generate poisoned data. Strong multimodal teachers (Gemini 2.5 Pro, GPT-4.1) produce high-quality malicious trajectories with higher ASR (up to 96.55%), lower detectability (DR@1 as low as 32%), and higher MCD entropy which indicates flexible malicious action selection.

Table 5. Ablation on trigger type (text, blended, hoverball) and poison ratio (10%, 20%, 50%) on AitZ with Qwen2-VL-2B.

Trigger Type	Poison Ratio	AMR↑	TMR↑	ASR↑
Text	10	56.87	72.46	90.63
	20	54.55	74.55	98.18
	50	51.92	71.12	**100**
Blended	10	**58.62**	81.30	**100**
	20	57.91	**82.72**	**100**
	50	53.18	74.83	**100**
Hoverball	10	57.43	78.64	98.37
	20	56.21	77.95	**100**
	50	55.02	77.68	**100**

However, text-only teachers (*e.g.*, LLaMA-3-70B) lack visual grounding, reducing environmental coherence and leading to higher detectability (63% DR@1).

Trigger Type and Poison Ratio Tab. 5 evaluates the impact of different trigger designs and poison ratios. Following VIBMA [28], we use Blended and Hoverball as visual triggers in our experiments. Blended triggers achieve the highest and most stable ASR across all settings, while maintaining higher AMR and TMR. Text triggers require higher poison ratios to reach similarly high ASR, and Hoverball triggers yield balanced performance. Intuitively, increasing poison ratio improves ASR but slightly reduces AMR/TMR, showing a trade-off between attack strength and utility preservation.

Overall, these studies confirm that AdapAction's effectiveness and stealth primarily depend on (1) the quality of malicious reasoning supervision and (2) the trigger-poison configurations, while benign utility remains largely stable.

5. Conclusion

We present **AdapAction**, a context-aware backdoor attack against GUI agents that replaces fixed trigger-action mappings with an adaptive malicious policy. Using *Context-Aware Behavioral Imitation*, AdapAction imitates adversarial decision trajectories to autonomously select malicious actions that are coherent with the current GUI state and user instruction. Evaluated on AitZ and AndroidControl, AdapAction achieves up to 100% ASR while preserving benign utility, and it evades multi-principle LLM defenses that detect static backdoors. Its stealth arises from semantically and visually plausible malicious behaviors indistinguishable from legitimate operations. We believe this work exposes a critical and underexplored vulnerability in autonomous GUI agents: adaptive, context-aware backdoors pose a significantly higher real-world threat than static ones. We hope our findings catalyze the development of context-sensitive defenses capable of detecting and mitigating such sophisticated attacks, which is a necessary step toward securing the next generation of MLLM-driven automation systems.

Acknowledgment. Baoyuan Wu is supported by Guangdong Basic and Applied Basic Research Foundation (No. 2024B1515020095), Guangdong Provincial Program (No. 2023TQ07A352), Shenzhen Science and Technology Program (No. RCYX20210609103057050 and JCYJ20240813113608011), and Longgang District Key Laboratory of Intelligent Digital Economy Security. Haizhou Li is supported by the Program for Guangdong Introducing Innovative and Entrepreneurial Teams (No. 2023ZT10X044).

References

[1] Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. Gpt-4 technical report. *arXiv e-prints*, 2023. 2

[2] Lukas Aichberger, Alasdair Paren, Philip Torr, Yarin Gal, and Adel Bibi. Mip against agent: Malicious image patches hijacking multimodal os agents. *Advances in Neural Information Processing Systems*, 2025. 3

[3] Shuai Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Sibo Song, Kai Dang, Peng Wang, Shijie Wang, Jun Tang, et al. Qwen2. 5-vl technical report. *arXiv e-prints*, 2025. 8

[4] Chong Chen, Bo Wang, and Youfang Lin. A systematic mapping study of llm applications in mobile device research. In *Asia-Pacific Web (APWeb) and Web-Age Information Management (WAIM) Joint International Conference on Web and Big Data*, pages 163–174. Springer, 2024. 1, 2

[5] Chaoran Chen, Zhiping Zhang, Bingcan Guo, Shang Ma, Ibrahim Khalilov, Simret A Gebreegziabher, Yanfang Ye, Ziang Xiao, Yaxing Yao, Tianshi Li, et al. The obvious invisible threat: Llm-powered gui agents' vulnerability to fine-print injections. *arXiv e-prints*, 2025. 3

[6] Zhaorun Chen, Zhen Xiang, Chaowei Xiao, Dawn Song, and Bo Li. Agentpoison: Red-teaming llm agents via poisoning memory or knowledge bases. *Advances in Neural Information Processing Systems*, 37:130185–130213, 2024. 3

[7] Pengzhou Cheng, Haowen Hu, Zheng Wu, Zongru Wu, Tianjie Ju, Zhuosheng Zhang, and Gongshen Liu. Hidden ghost hand: Unveiling backdoor vulnerabilities in mllm-powered mobile gui agents. *arXiv e-prints*, 2025. 2, 3, 6

[8] Gheorghe Comanici, Eric Bieber, Mike Schaekermann, Ice Pasupat, Noveen Sachdeva, Inderjit Dhillon, Marcel Blistein, Ori Ram, Dan Zhang, Evan Rosen, et al. Gemini 2.5: Pushing the frontier with advanced reasoning, multimodality, long context, and next generation agentic capabilities. *arXiv e-prints*, 2025. 2, 4, 8

[9] Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, Amy Yang, Angela Fan, Anirudh Goyal, Anthony Hartshorn, Aobo Yang, Archi Mitra, Archie Sravankumar, Artem Korenev, Arthur Hinsvark, Arun Rao, Aston Zhang, Aurelien Rodriguez, Austen Gregerson, Ava Spataru, Baptiste Roziere, Bethany Biron, Binh Tang, Bobbie Chern, Charlotte Caucheteux, Chaya Nayak, Chloe Bi, Chris Marra, Chris McConnell, Christian Keller, Christophe Touret, Chunyang Wu, Corinne Wong, Cristian Canton Ferrer, Cyrus Nikolaidis, Damien Allonsius, Daniel Song, Danielle Pintz, Danny Livshits, Danny Wyatt, David Esiobu, Dhruv Choudhary, Dhruv Mahajan, Diego Garcia-Olano, Diego Perino, Dieuwke Hupkes, Egor Lakomkin, Ehab AlBadawy, Elina Lobanova, Emily Dinan, Eric Michael Smith, Filip Radenovic, Francisco Guzmán, Frank Zhang, Gabriel Synnaeve, Gabrielle Lee, Georgia Lewis Anderson, Govind Thattai, Graeme Nail, Gregoire Mialon, Guan Pang, Guillem Cucurell, Hailey Nguyen, Hannah Korevaar, Hu Xu, Hugo Touvron, Iliyan Zarov, Imanol Arrieta Ibarra, Isabel Kloumann, Ishan Misra, Ivan Evtimov, Jack Zhang, Jade Copet, Jaewon Lee, Jan Geffert, Jana Vranes, Jason Park, Jay Mahadeokar, Jeet Shah, Jelmer van der Linde, Jennifer Billock, Jenny Hong, Jenya Lee, Jeremy Fu, Jianfeng Chi, Jianyu Huang, Jiawen Liu, Jie Wang, Jiecao Yu, Joanna Bitton, Joe Spisak, Jongsoo Park, Joseph Rocca, Joshua Johnstun, Joshua Saxe, Junteng Jia, Kalyan Vasuden Alwala, Karthik Prasad, Kartikeya Upasani, Kate Plawiak, Ke Li, Kenneth Heafield, Kevin Stone, Khalid El-Arini, Krithika Iyer, Kshitiz Malik, Kuenley Chiu, Kunal Bhalla, Kushal Lakhotia, Lauren Rantala-Yeary, Laurens van der Maaten, Lawrence Chen, Liang Tan, Liz Jenkins, Louis Martin, Lovish Madaan, Lubo Malo, Lukas Blecher, Lukas Landzaat, Luke de Oliveira, Madeline Muzzi, Mahesh Pasupuleti, Mannat Singh, Manohar Paluri, Marcin Kardas, Maria Tsimpoukelli, Mathew Oldham, Mathieu Rita, Maya Pavlova, Melanie Kambadur, Mike Lewis, Min Si, Mitesh Kumar Singh, Mona Hassan, Naman Goyal, Narjes Torabi, Nikolay Bashlykov, Nikolay Bogoychev, Niladri Chatterji, Ning Zhang, Olivier Duchenne, Onur Çelebi, Patrick Alrassy, Pengchuan Zhang, Pengwei Li, Petar Vasic, Peter Weng, Prajjwal Bhargava, Pratik Dubal, Praveen Krishnan, Punit Singh Koura, Puxin Xu, Qing He, Qingxiao Dong, Ragavan Srinivasan, Raj Ganapathy, Ramon Calderer, Ricardo Silveira Cabral, Robert Stojnic, Roberta Raileanu, Rohan Maheswari, Rohit Girdhar, Rohit Patel, Romain Sauvestre, Ronnie Polidoro, Roshan Sumbaly, Ross Taylor, Ruan Silva, Rui Hou, Rui Wang, Saghar Hosseini, Sahana Chennabasappa, Sanjay Singh, Sean Bell, Seohyun Sonia Kim, Sergey Edunov, Shaoliang Nie, Sharan Narang, Sharath Raparthy, Sheng Shen, Shengye Wan, Shruti Bhosale, Shun Zhang, Simon Vandenhende, Soumya Batra, Spencer Whitman, Sten Sootla, Stephane Collot, Suchin Gururangan, Sydney Borodinsky, Tamar Herman, Tara Fowler, Tarek Sheasha, Thomas Georgiou, Thomas Scialom, Tobias Speckbacher, Todor Mihaylov, Tong Xiao, Ujjwal Karn, Vedanuj Goswami, Vibhor Gupta, Vignesh Ramanathan, Viktor Kerkez, Vincent Gonguet, Virginie Do, Vish Vogeti, Vítor Albiero, Vladan Petrovic, Weiwei Chu, Wenhan Xiong, Wenyin Fu, Whitney Meers, Xavier Martinet, Xiaodong Wang, Xiaofang Wang, Xiaoqing Ellen Tan, Xide Xia, Xinfeng Xie, Xuchao Jia, Xuewei Wang, Yaelle Goldschlag, Yashesh Gaur, Yasmine Babaei, Yi Wen, Yiwen Song, Yuchen Zhang, Yue Li, Yuning Mao, Zacharie Delpierre Coudert, Zheng Yan, Zhengxing Chen, Zoe Papakipos, Aaditya Singh, Aayushi Srivastava, Abha Jain, Adam Kelsey, Adam Shajnfeld, Adithya Gangidi, Adolfo Victoria, Ahuva Goldstand, Ajay Menon,

Ajay Sharma, Alex Boesenberg, Alexei Baevski, Allie Feinstein, Amanda Kallet, Amit Sangani, Amos Teo, Anam Yunus, Andrei Lupu, Andres Alvarado, Andrew Caples, Andrew Gu, Andrew Ho, Andrew Poulton, Andrew Ryan, Ankit Ramchandani, Annie Dong, Annie Franco, Anuj Goyal, Aparajita Saraf, Arkabandhu Chowdhury, Ashley Gabriel, Ashwin Bharambe, Assaf Eisenman, Azadeh Yazdan, Beau James, Ben Maurer, Benjamin Leonhardi, Bernie Huang, Beth Loyd, Beto De Paola, Bhargavi Paranjape, Bing Liu, Bo Wu, Boyu Ni, Braden Hancock, Bram Wasti, Brandon Spence, Brani Stojkovic, Brian Gamido, Britt Montalvo, Carl Parker, Carly Burton, Catalina Mejia, Ce Liu, Changhan Wang, Changkyu Kim, Chao Zhou, Chester Hu, Ching-Hsiang Chu, Chris Cai, Chris Tindal, Christoph Feichtenhofer, Cynthia Gao, Damon Civin, Dana Beaty, Daniel Kreymer, Daniel Li, David Adkins, David Xu, Davide Testuggine, Delia David, Devi Parikh, Diana Liskovich, Didem Foss, Dingkang Wang, Duc Le, Dustin Holland, Edward Dowling, Eissa Jamil, Elaine Montgomery, Eleonora Presani, Emily Hahn, Emily Wood, Eric-Tuan Le, Erik Brinkman, Esteban Arcaute, Evan Dunbar, Evan Smothers, Fei Sun, Felix Kreuk, Feng Tian, Filippos Kokkinos, Firat Ozgenel, Francesco Caggioni, Frank Kanayet, Frank Seide, Gabriela Medina Florez, Gabriella Schwarz, Gada Badeer, Georgia Swee, Gil Halpern, Grant Herman, Grigory Sizov, Guangyi, Zhang, Guna Lakshminarayanan, Hakan Inan, Hamid Shojanazeri, Han Zou, Hannah Wang, Hanwen Zha, Haroun Habeeb, Harrison Rudolph, Helen Suk, Henry Aspegren, Hunter Goldman, Hongyuan Zhan, Ibrahim Damlaj, Igor Molybog, Igor Tufanov, Ilias Leontiadis, Irina-Elena Veliche, Itai Gat, Jake Weissman, James Geboski, James Kohli, Janice Lam, Japhet Asher, Jean-Baptiste Gaya, Jeff Marcus, Jeff Tang, Jennifer Chan, Jenny Zhen, Jeremy Reizenstein, Jeremy Teboul, Jessica Zhong, Jian Jin, Jingyi Yang, Joe Cummings, Jon Carvill, Jon Shepard, Jonathan McPhie, Jonathan Torres, Josh Ginsburg, Junjie Wang, Kai Wu, Kam Hou U, Karan Saxena, Kartikay Khandelwal, Katayoun Zand, Kathy Matosich, Kaushik Veeraraghavan, Kelly Michelena, Keqian Li, Kiran Jagadeesh, Kun Huang, Kunal Chawla, Kyle Huang, Lailin Chen, Lakshya Garg, Lavender A, Leandro Silva, Lee Bell, Lei Zhang, Liangpeng Guo, Licheng Yu, Liron Moshkovich, Luca Wehrstedt, Madian Khabsa, Manav Avalani, Manish Bhatt, Martynas Mankus, Matan Hasson, Matthew Lennie, Matthias Reso, Maxim Groshev, Maxim Naumov, Maya Lathi, Meghan Keneally, Miao Liu, Michael L. Seltzer, Michal Valko, Michelle Restrepo, Mihir Patel, Mik Vyatskov, Mikayel Samvelyan, Mike Clark, Mike Macey, Mike Wang, Miquel Jubert Hermoso, Mo Metanat, Mohammad Rastegari, Munish Bansal, Nandhini Santhanam, Natascha Parks, Natasha White, Navyata Bawa, Nayan Singhal, Nick Egebo, Nicolas Usunier, Nikhil Mehta, Nikolay Pavlovich Laptev, Ning Dong, Norman Cheng, Oleg Chernoguz, Olivia Hart, Omkar Salpekar, Ozlem Kalinli, Parkin Kent, Parth Parekh, Paul Saab, Pavan Balaji, Pedro Rittner, Philip Bontrager, Pierre Roux, Piotr Dollar, Polina Zvyagina, Prashant Ratanchandani, Pritish Yuvraj, Qian Liang, Rachad Alao, Rachel Rodriguez, Rafi Ayub, Raghotham Murthy, Raghu Nayani, Rahul Mitra, Rangaprabhu Parthasarathy, Raymond Li, Rebekkah Hogan, Robin Battey, Rocky Wang, Russ Howes, Ruty Rinott, Sachin Mehta, Sachin Siby, Sai Jayesh Bondu, Samyak Datta, Sara Chugh, Sara Hunt, Sargun Dhillon, Sasha Sidorov, Satadru Pan, Saurabh Mahajan, Saurabh Verma, Seiji Yamamoto, Sharadh Ramaswamy, Shaun Lindsay, Shaun Lindsay, Sheng Feng, Shenghao Lin, Shengxin Cindy Zha, Shishir Patil, Shiva Shankar, Shuqiang Zhang, Shuqiang Zhang, Sinong Wang, Sneha Agarwal, Soji Sajuyigbe, Soumith Chintala, Stephanie Max, Stephen Chen, Steve Kehoe, Steve Satterfield, Sudarshan Govindaprasad, Sumit Gupta, Summer Deng, Sungmin Cho, Sunny Virk, Suraj Subramanian, Sy Choudhury, Sydney Goldman, Tal Remez, Tamar Glaser, Tamara Best, Thilo Koehler, Thomas Robinson, Tianhe Li, Tianjun Zhang, Tim Matthews, Timothy Chou, Tzook Shaked, Varun Vontimitta, Victoria Ajayi, Victoria Montanez, Vijai Mohan, Vinay Satish Kumar, Vishal Mangla, Vlad Ionescu, Vlad Poenaru, Vlad Tiberiu Mihailescu, Vladimir Ivanov, Wei Li, Wenchen Wang, Wenwen Jiang, Wes Bouaziz, Will Constable, Xiaocheng Tang, Xiaojian Wu, Xiaolan Wang, Xilun Wu, Xinbo Gao, Yaniv Kleinman, Yanjun Chen, Ye Hu, Ye Jia, Ye Qi, Yenda Li, Yilin Zhang, Ying Zhang, Yossi Adi, Youngjin Nam, Yu, Wang, Yu Zhao, Yuchen Hao, Yundi Qian, Yunlu Li, Yuzi He, Zach Rait, Zachary DeVito, Zef Rosnbrick, Zhaoduo Wen, Zhenyu Yang, Zhiwei Zhao, and Zhiyu Ma. The llama 3 herd of models. *arXiv e-prints*, 2024. 8

[10] Tianyu Gu, Brendan Dolan-Gavitt, and Siddharth Garg. Badnets: Identifying vulnerabilities in the machine learning model supply chain. *arXiv e-prints*, 2017. 6

[11] Wenyi Hong, Weihan Wang, Qingsong Lv, Jiazheng Xu, Wenmeng Yu, Junhui Ji, Yan Wang, Zihan Wang, Yuxiao Dong, Ming Ding, et al. Cogagent: A visual language model for gui agents. In *Conference on Computer Vision and Pattern Recognition*, pages 14281–14290, 2024. 2

[12] Ruochen Jiao, Shaoyuan Xie, Justin Yue, TAKAMI SATO, Lixu Wang, Yixuan Wang, Qi Alfred Chen, and Qi Zhu. Can we trust embodied agents? exploring backdoor attacks against embodied llm-based decision-making systems. In *International Conference on Learning Representations*, 2025. 3

[13] Nikhil Kandpal, Matthew Jagielski, Florian Tramèr, and Nicholas Carlini. Backdoor attacks for in-context learning with language models. In *The Second Workshop on New Frontiers in Adversarial Machine Learning*, 2023. 6

[14] Juyong Lee, Dongyoon Hahm, June Suk Choi, W Bradley Knox, and Kimin Lee. Mobilesafetybench: Evaluating safety of autonomous agents in mobile device control. *arXiv e-prints*, 2024. 2

[15] Wei Li, William E Bishop, Alice Li, Christopher Rawles, Folawiyo Campbell-Ajala, Divya Tyamagundlu, and Oriana Riva. On the effects of data scale on ui control agents. *Advances in Neural Information Processing Systems*, 37:92130–92154, 2024. 2, 5

[16] Yige Li, Hanxun Huang, Yunhan Zhao, Xingjun Ma, and Jun Sun. Backdoorllm: A comprehensive benchmark for backdoor attacks on large language models. *arXiv e-prints*, 2024. 3

[17] Zeyi Liao, Lingbo Mo, Chejian Xu, Mintong Kang, Jiawei Zhang, Chaowei Xiao, Yuan Tian, Bo Li, and Huan Sun. Eia: Environmental injection attack on generalist web agents for

privacy leakage. In *International Conference on Learning Representations*, 2025. 3

[18] Aishan Liu, Yuguang Zhou, Xianglong Liu, Tianyuan Zhang, Siyuan Liang, Jiakai Wang, Yanjun Pu, Tianlin Li, Junqi Zhang, Wenbo Zhou, et al. Compromising llm driven embodied agents with contextual backdoor attacks. *IEEE Transactions on Information Forensics and Security*, 20:3979–3994, 2025. 3

[19] Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. *Advances in Neural Information Processing Systems*, pages 34892–34916, 2023. 2

[20] Quanfeng Lu, Wenqi Shao, Zitao Liu, Lingxiao Du, Fanqing Meng, Boxuan Li, Botong Chen, Siyuan Huang, Kaipeng Zhang, and Ping Luo. Guiodyssey: A comprehensive dataset for cross-app gui navigation on mobile devices. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 22404–22414, 2025. 2

[21] Shikhar Murty, Hao Zhu, Dzmitry Bahdanau, and Christopher D Manning. Nnetnav: Unsupervised learning of browser agents through environment interaction in the wild. *arXiv e-prints*, 2024. 1

[22] OpenAI. Introducing GPT-4.1 in the api. https://openai.com/index/gpt-4-1/, 2025. Accessed: 2025-11-13. 5, 8

[23] Fanchao Qi, Mukai Li, Yangyi Chen, Zhengyan Zhang, Zhiyuan Liu, Yasheng Wang, and Maosong Sun. Hidden killer: Invisible textual backdoor attacks with syntactic trigger. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 443–453, 2021. 7

[24] Zehan Qi, Xiao Liu, Iat Long Iong, Hanyu Lai, Xueqiao Sun, Jiadai Sun, Xinyue Yang, Yu Yang, Shuntian Yao, Wei Xu, et al. Webrl: Training llm web agents via self-evolving online curriculum reinforcement learning. In *International Conference on Learning Representations*, 2025. 1

[25] Christopher Rawles, Sarah Clinckemaillie, Yifan Chang, Jonathan Waltz, Gabrielle Lau, Marybeth Fair, Alice Li, William E Bishop, Wei Li, Folawiyo Campbell-Ajala, et al. Androidworld: A dynamic benchmarking environment for autonomous agents. In *International Conference on Learning Representations*, 2025. 5

[26] Peng Wang, Shuai Bai, Sinan Tan, Shijie Wang, Zhihao Fan, Jinze Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Yang Fan, Kai Dang, Mengfei Du, Xuancheng Ren, Rui Men, Dayiheng Liu, Chang Zhou, Jingren Zhou, and Junyang Lin. Qwen2-vl: Enhancing vision-language model's perception of the world at any resolution. *arXiv e-prints*, 2024. 2, 5

[27] Shuai Wang, Weiwen Liu, Jingxuan Chen, Yuqi Zhou, Weinan Gan, Xingshan Zeng, Yuhan Che, Shuai Yu, Xinlong Hao, Kun Shao, et al. Gui agents with foundation models: A comprehensive survey. *arXiv e-prints*, 2024. 1, 2

[28] Xuan Wang, Siyuan Liang, Zhe Liu, Yi Yu, Aishan Liu, Yuliang Lu, Xitong Gao, and Ee-Chien Chang. Poison once, control anywhere: Clean-text visual backdoors in vlm-based mobile agents, 2025. 2, 3, 7, 8

[29] Yifei Wang, Dizhan Xue, Shengjie Zhang, and Shengsheng Qian. Badagent: Inserting and activating backdoor attacks in llm agents. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 9811–9827, 2024. 3

[30] Baoyuan Wu, Hongrui Chen, Mingda Zhang, Zihao Zhu, Shaokui Wei, Danni Yuan, and Chao Shen. Backdoorbench: A comprehensive benchmark of backdoor learning. *Advances in Neural Information Processing Systems*, 35:10546–10559, 2022. 3

[31] Baoyuan Wu, Zihao Zhu, Li Liu, Qingshan Liu, Zhaofeng He, and Siwei Lyu. Attacks in adversarial machine learning: A systematic survey from the life-cycle perspective. *arXiv e-prints*, 2023.

[32] Baoyuan Wu, Hongrui Chen, Mingda Zhang, Zihao Zhu, Shaokui Wei, Danni Yuan, Mingli Zhu, Ruotong Wang, Li Liu, and Chao Shen. Backdoorbench: A comprehensive benchmark and analysis of backdoor learning. *International Journal of Computer Vision*, 133(8):5700–5787, 2025.

[33] Baoyuan Wu, Mingli Zhu, Meixi Zheng, Zihao Zhu, Shaokui Wei, Mingda Zhang, Hongrui Chen, Danni Yuan, Li Liu, and Qingshan Liu. Defenses in adversarial machine learning: A systematic survey from the lifecycle perspective. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2025. 3

[34] Zongru Wu, Pengzhou Cheng, Zheng Wu, Tianjie Ju, Zhuosheng Zhang, and Gongshen Liu. Smoothing grounding and reasoning for mllm-powered gui agents with query-oriented pivot tasks. *arXiv e-prints*, 2025. 6

[35] Zhiyong Wu, Zhenyu Wu, Fangzhi Xu, Yian Wang, Qiushi Sun, Chengyou Jia, Kanzhi Cheng, Zichen Ding, Liheng Chen, Paul Pu Liang, et al. Os-atlas: Foundation action model for generalist gui agents. In *The Thirteenth International Conference on Learning Representations*, 2025. 2

[36] Wenkai Yang, Xiaohan Bi, Yankai Lin, Sishuo Chen, Jie Zhou, and Xu Sun. Watch out for your agents! investigating backdoor threats to llm-based agents. *Advances in Neural Information Processing Systems*, pages 100938–100964, 2024. 2, 3

[37] Yulong Yang, Xinshan Yang, Shuaidong Li, Chenhao Lin, Zhengyu Zhao, Chao Shen, and Tianwei Zhang. Systematic categorization, construction and evaluation of new attacks against multi-modal mobile gui agents. *arXiv e-prints*, 2024. 2

[38] Shunyu Yao, Howard Chen, John Yang, and Karthik Narasimhan. Webshop: Towards scalable real-world web interaction with grounded language agents. *Advances in Neural Information Processing Systems*, pages 20744–20757, 2022. 1

[39] Jiabo Ye, Xi Zhang, Haiyang Xu, Haowei Liu, Junyang Wang, Zhaoqing Zhu, Ziwei Zheng, Feiyu Gao, Junjie Cao, Zhengxi Lu, et al. Mobile-agent-v3: Foundamental agents for gui automation. *arXiv e-prints*, 2025. 5

[40] Ziang Ye, Yang Zhang, Wentao Shi, Xiaoyu You, Fuli Feng, and Tat-Seng Chua. Visualtrap: A stealthy backdoor attack on gui agents via visual grounding manipulation. In *Second Conference on Language Modeling*, 2025. 2, 3

[41] Chaoyun Zhang, Shilin He, Jiaxu Qian, Bowen Li, Liqun Li, Si Qin, Yu Kang, Minghua Ma, Guyue Liu, Qingwei Lin,

et al. Large language model-brained gui agents: A survey. *arXiv e-prints*, 2024. 1, 2

[42] Chi Zhang, Zhao Yang, Jiaxuan Liu, Yanda Li, Yucheng Han, Xin Chen, Zebiao Huang, Bin Fu, and Gang Yu. Appagent: Multimodal agents as smartphone users. In *Proceedings of the 2025 CHI Conference on Human Factors in Computing Systems*, pages 1–20, 2025. 1, 2

[43] Jiwen Zhang, Jihao Wu, Teng Yihua, Minghui Liao, Nuo Xu, Xiao Xiao, Zhongyu Wei, and Duyu Tang. Android in the zoo: Chain-of-action-thought for gui agents. In *Findings of the Association for Computational Linguistics*, pages 12016–12031, 2024. 2, 5

[44] Boyuan Zheng, Boyu Gou, Jihyung Kil, Huan Sun, and Yu Su. Gpt-4v (ision) is a generalist web agent, if grounded. In *International Conference on Machine Learning*, pages 61349–61385. PMLR, 2024. 1

[45] Yaowei Zheng, Richong Zhang, Junhao Zhang, Yanhan Ye, Zheyan Luo, Zhangchi Feng, and Yongqiang Ma. Llamafactory: Unified efficient fine-tuning of 100+ language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 3: System Demonstrations)*, 2024. 6

[46] Deyao Zhu, Jun Chen, Xiaoqian Shen, Xiang Li, and Mohamed Elhoseiny. Minigpt-4: Enhancing vision-language understanding with advanced large language models. In *International Conference on Learning Representations*, 2024. 2

[47] Pengyu Zhu, Zhenhong Zhou, Yuanhe Zhang, Shilinlu Yan, Kun Wang, and Sen Su. Demonagent: Dynamically encrypted multi-backdoor implantation attack on llm-based agent. *arXiv e-prints*, 2025. 3