CISPA
HELMHOLTZ CENTER FOR
INFORMATION SECURITY

YuraScanner: Leveraging LLMs for Task-driven Web App Scanning

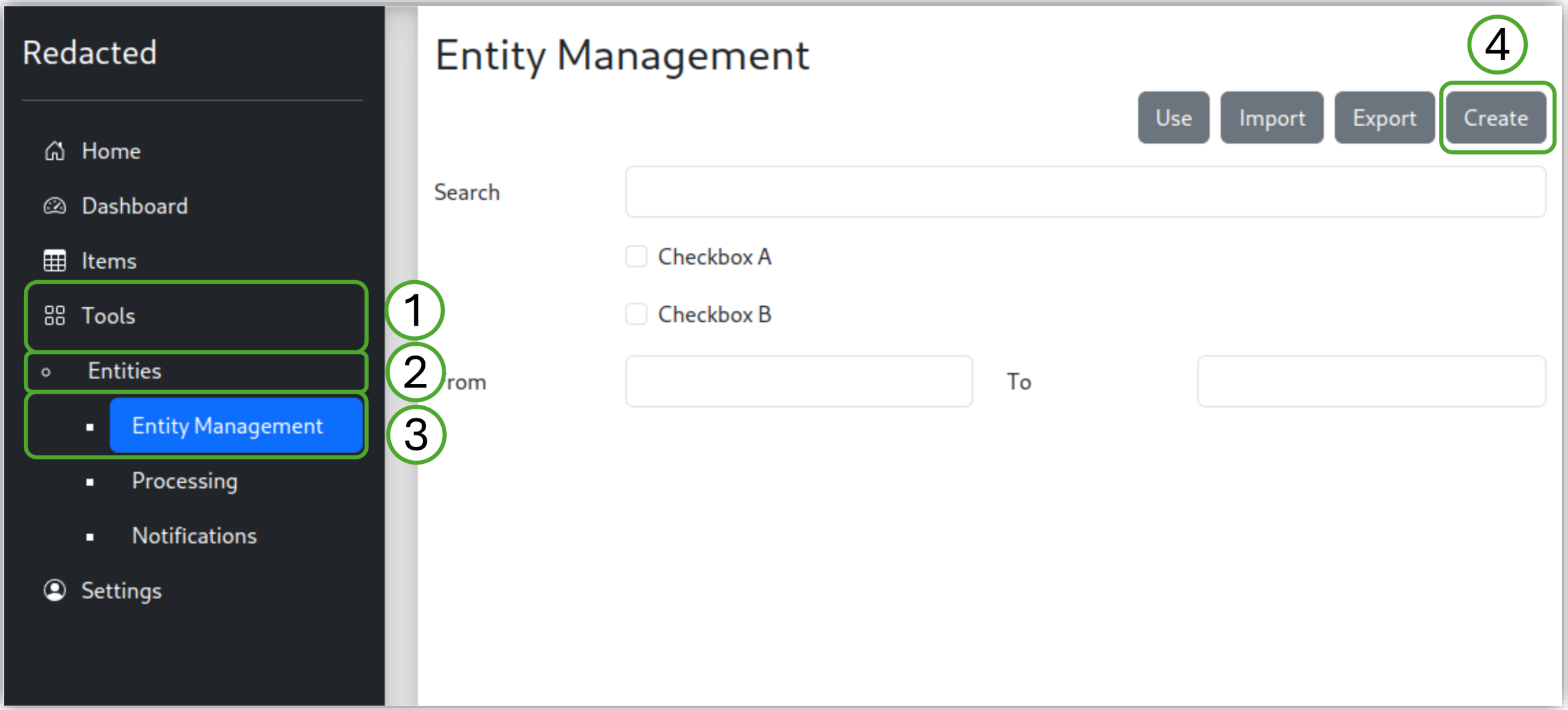
*Aleksei Stafeev, **Tim Recktenwald**, Gianluca De Stefano, Soheil Khodayari, Giancarlo Pellegrino*

Network and Distributed System Security Symposium | 2025

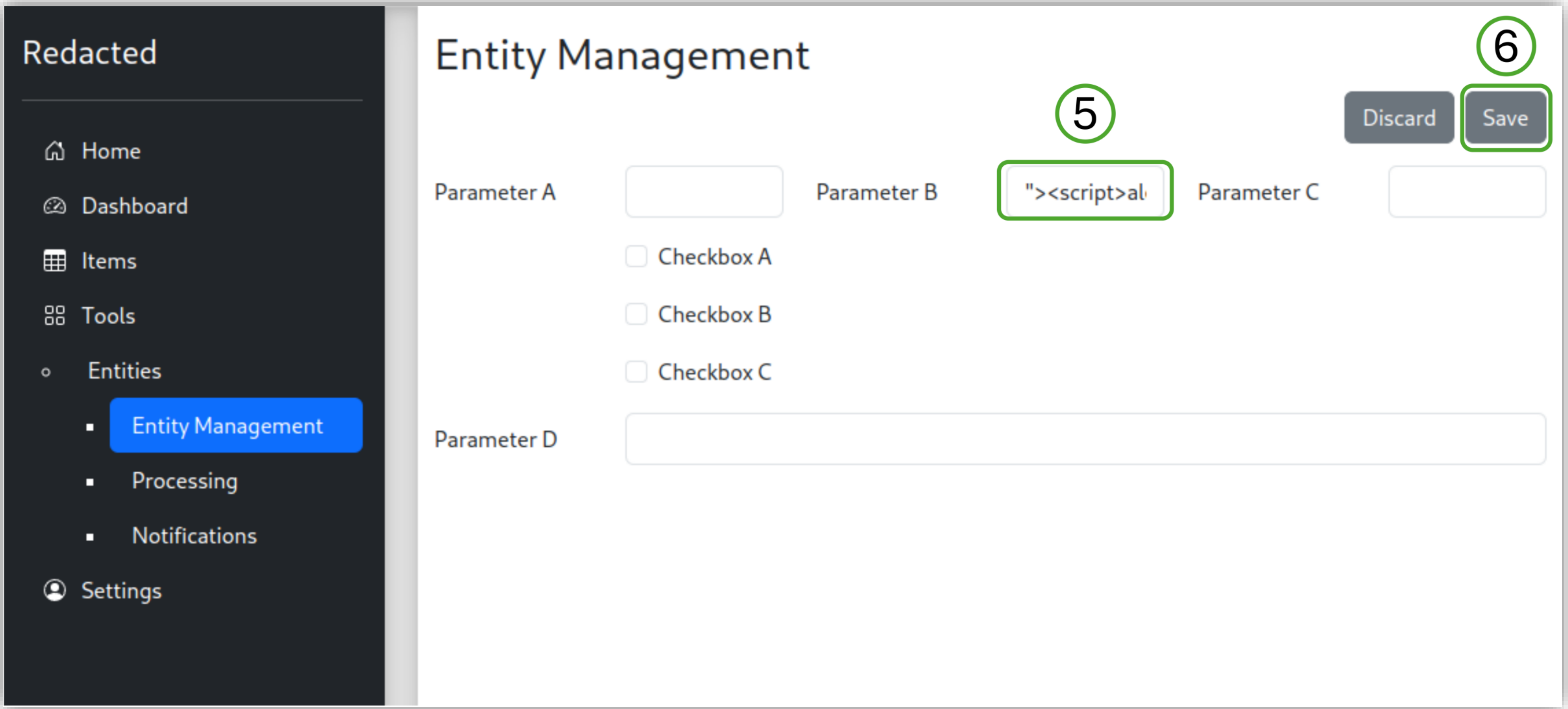
Motivation

- Web application scanners are popular black-box testing tools
- However, traditional approaches struggle with exploring deeper states

Motivation


Redacted
Home
Dashboard
Items
Tools
Entities
Entity Management
Processing
Notifications
Settings
Entity Management
Use
Import
Export
Create
Search
Checkbox A
Checkbox B
rom
To
1
2
3
4

Motivation


Redacted
Home
Dashboard
Items
Tools
Entities
Entity Management
Processing
Notifications
Settings
Entity Management
6
5
Discard
Save
Parameter A
Parameter B
"><script>al
Parameter C
Checkbox A
Checkbox B
Checkbox C
Parameter D

Motivation

- Web application scanners are popular black-box testing tools
- However, traditional approaches struggle with exploring deeper states
- **Key limitation:** They lack awareness of multi-step workflows

Motivation

- Web application scanners are popular black-box testing tools
- However, traditional approaches struggle with exploring deeper states
- **Key limitation:** They lack awareness of multi-step workflows
- Model-based methods have been proposed to tackle this weakness
  - E.g., reinforcement learning on user-provided traces [1]

[1] E. Z. Liu, K. Guu, P. Pasupat, T. Shi, and P. Liang, "Reinforcement Learning on Web Interfaces using Workflow-Guided Exploration," in *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings*.

Motivation

- Web application scanners are popular black-box testing tools
- However, traditional approaches struggle with exploring deeper states
- **Key limitation:** They lack awareness of multi-step workflows
- Model-based methods have been proposed to tackle this weakness
  - E.g., reinforcement learning on user-provided traces [1]
- **Does not scale well!**

[1] E. Z. Liu, K. Guu, P. Pasupat, T. Shi, and P. Liang, "Reinforcement Learning on Web Interfaces using Workflow-Guided Exploration," in *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings*.

Approach

- Instead of building a model, we opted to use large language models (LLMs)

Approach

- Instead of building a model, we opted to use large language models (LLMs)
- Non-academic approaches have proposed LLM-based browsing agents to assist users with tasks [2], [3]
  - E.g., “Book a hotel in San Diego”

[2] N. Friedman. (2022) Natbot. https://github.com/nat/natbot.
[3] (2024) Skyvern. https://github.com/Skyvern-AI/skyvern.

Approach

- Instead of building a model, we opted to use large language models (LLMs)
- Non-academic approaches have proposed LLM-based browsing agents to assist users with tasks [2], [3]
  - E.g., “Book a hotel in San Diego”
- Instead, we want to complete workflows and reach deeper states in web apps without user interaction

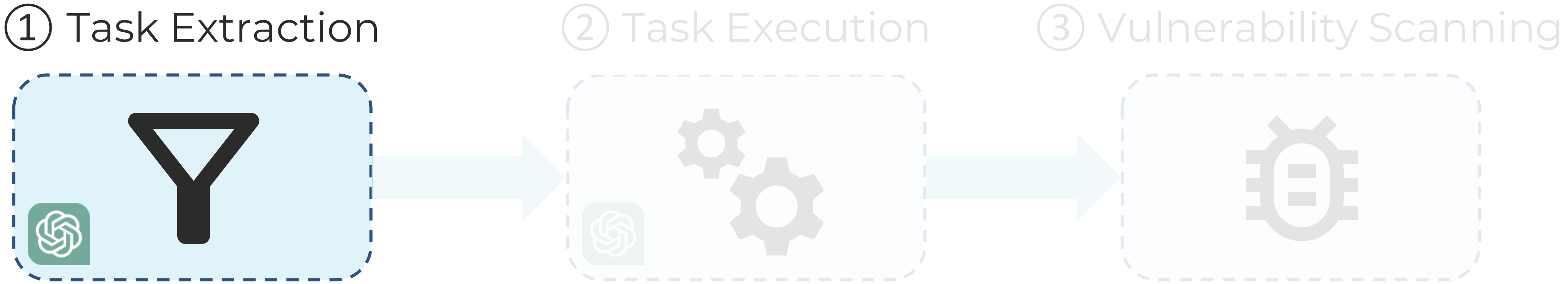
[2] N. Friedman. (2022) Natbot. https://github.com/nat/natbot.
[3] (2024) Skyvern. https://github.com/Skyvern-AI/skyvern.

Approach

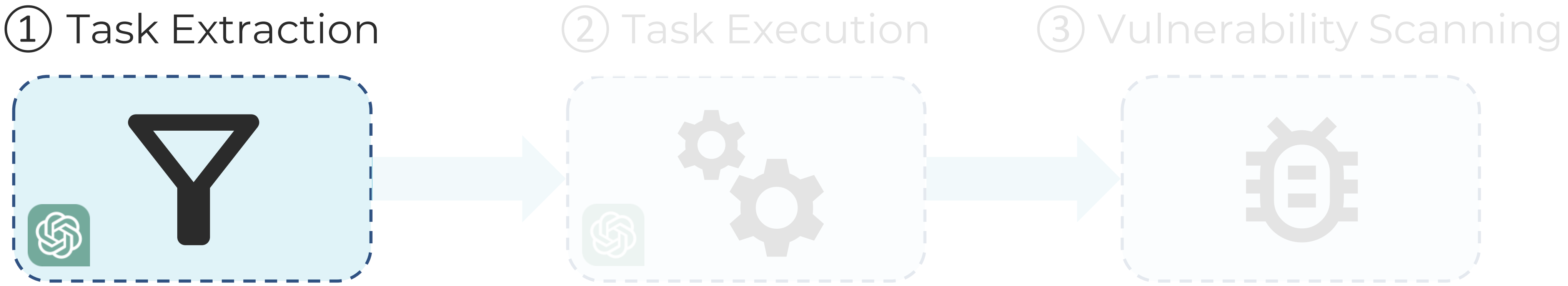
- Instead of building a model, we opted to use large language models (LLMs)
- Non-academic approaches have proposed LLM-based browsing agents to assist users with tasks [2], [3]
  - E.g., “Book a hotel in San Diego”
- Instead, we want to complete workflows and reach deeper states in web apps without user interaction
- We propose a fully automated, task-driven web application scanner called YuraScanner

[2] N. Friedman. (2022) Natbot. https://github.com/nat/natbot.
[3] (2024) Skyvern. https://github.com/Skyvern-AI/skyvern.

Architecture of YuraScanner

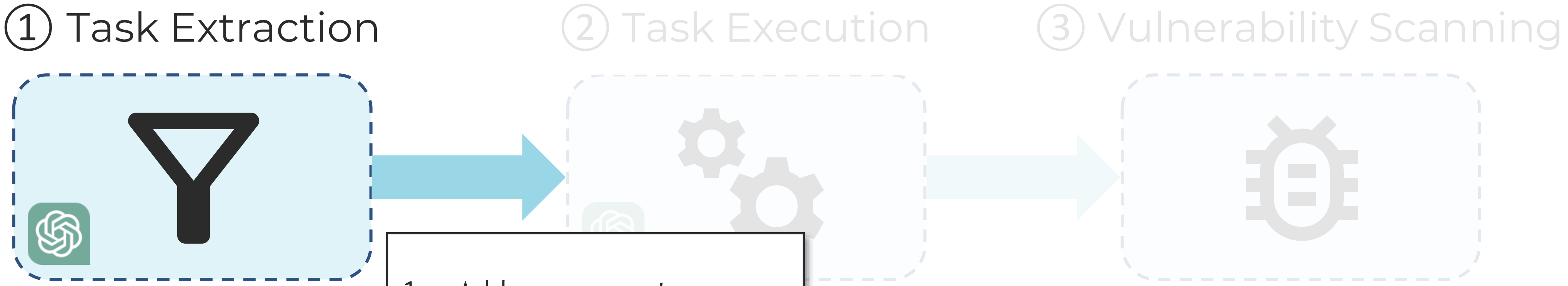


Architecture of YuraScanner



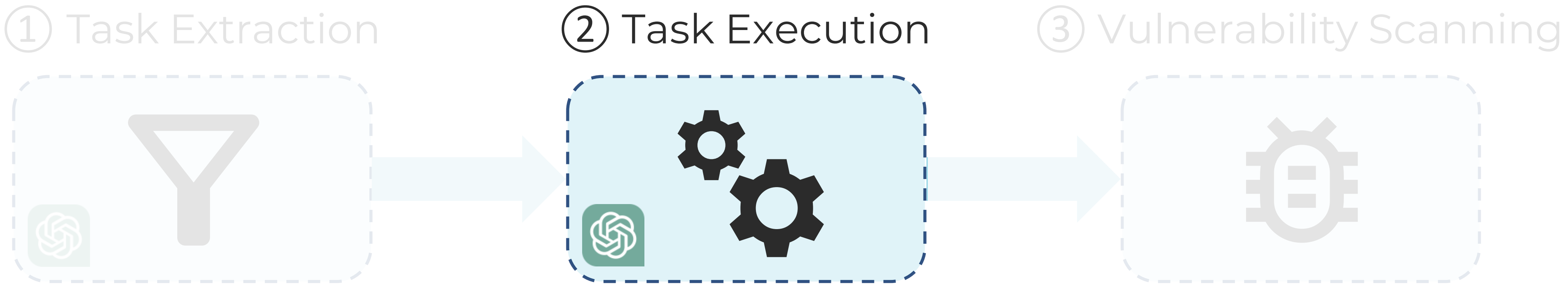
1. Shallow crawl of depth one
2. Extract text content from interactable HTML elements
3. Ask LLM to provide appropriate tasks

Architecture of YuraScanner



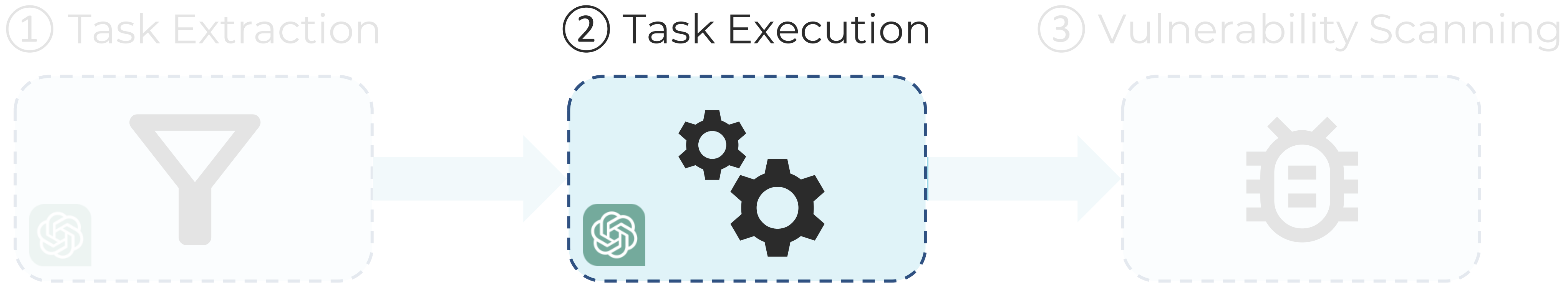
1. Add a new category for products.
2. Edit the information for an existing product.
3. Delete a previous order.

Architecture of YuraScanner



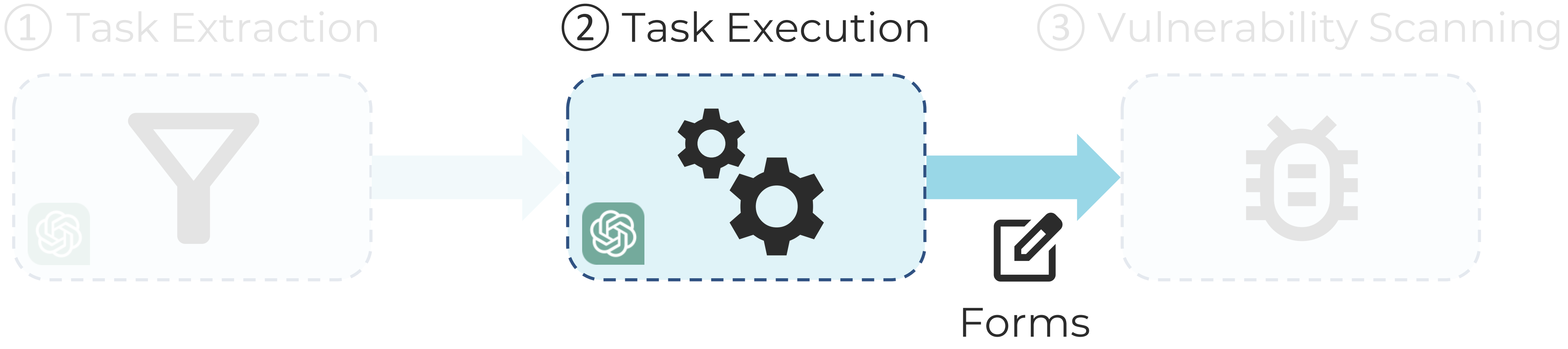
- Executes every task in multiple steps

Architecture of YuraScanner



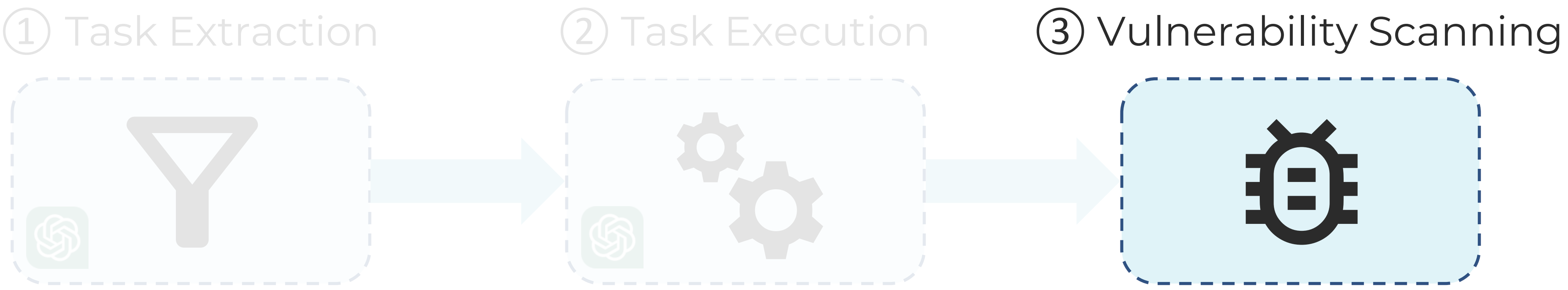
- Executes every task in multiple steps
- At each step of a task:
  - Generate simplified textual page representation
  - Query LLM for next command (e.g., “CLICK 2”)
  - Execute command

Architecture of YuraScanner



- Executes every task in multiple steps
- At each step of a task:
  - Generate simplified textual page representation
  - Query LLM for next command (e.g., “CLICK 2”)
  - Execute command

Architecture of YuraScanner



- We integrated the XSS engine of Black Widow
- Visits every form collected during task execution
- Injects predefined XSS payloads into input fields

Evaluation

- We evaluated YuraScanner on 20 popular, modern web applications

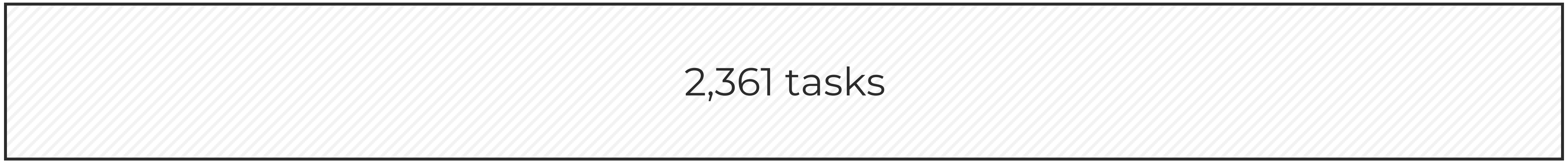
Evaluation

- We evaluated YuraScanner on 20 popular, modern web applications
- We divided our testbed into two sets:
  1. **Task Extraction and Execution**
     - **Manual** labeling of valid tasks and their success rate during task execution
     - Random subset of 10 web apps

Evaluation

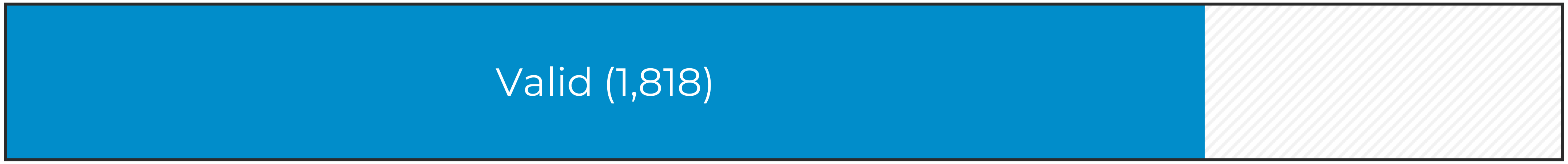
- We evaluated YuraScanner on 20 popular, modern web applications
- We divided our testbed into two sets:
  1. **Task Extraction and Execution**
     – **Manual** labeling of valid tasks and their success rate during task execution
     – Random subset of 10 web apps
  2. **Vulnerability Detection**
     – Inspection of vulnerabilities found by the attack component
     – All 20 web apps

Evaluation Results: Task Extraction



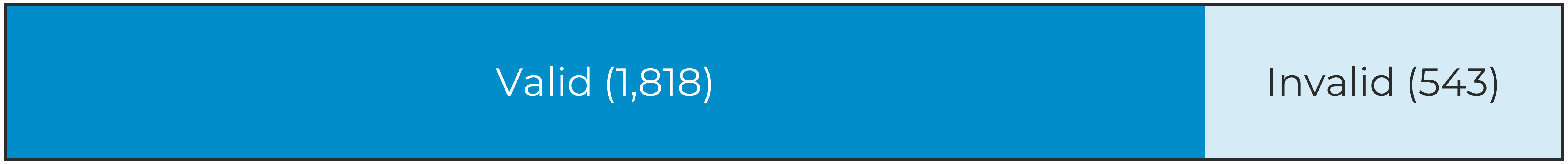
- 2,361 tasks were generated in total across 10 web applications

Evaluation Results: Task Extraction



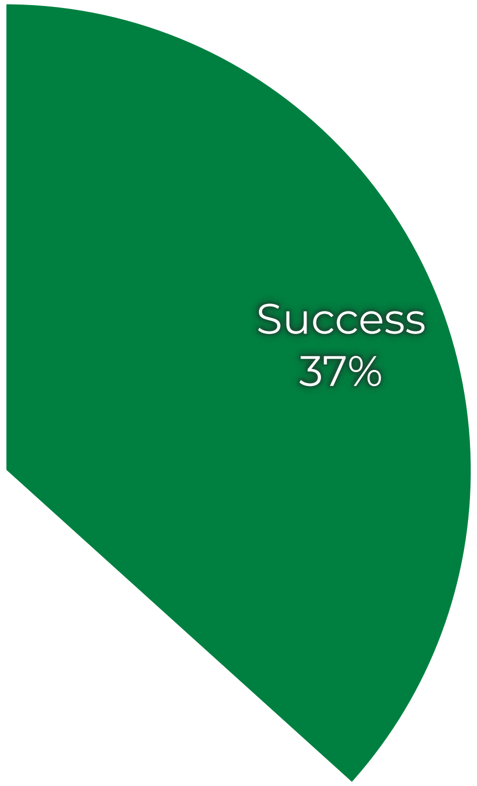
- 2,361 tasks were generated in total across 10 web applications
- 77% of the tasks were valid (1,818 tasks)

Evaluation Results: Task Extraction



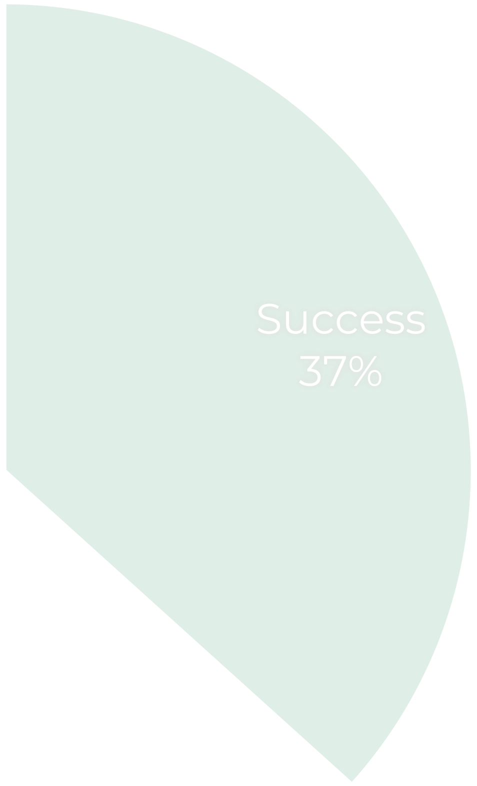
- 2,361 tasks were generated in total across 10 web applications
- 77% of the tasks were valid (1,818 tasks)
- "Invalid" = Functionality does not exist in the web application
- Invalid task generation mainly occurred on pages with insufficient context
  - E.g., login page with only one button

Evaluation Results: Task Execution

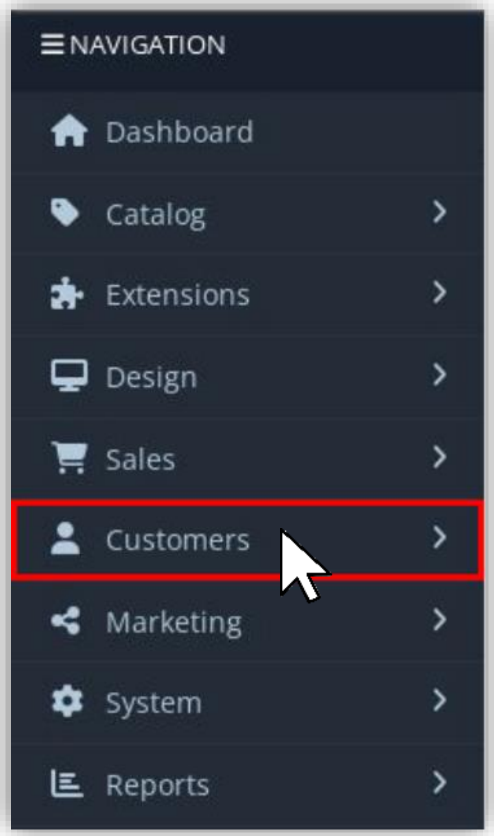


Task Execution Classification
(1,818 valid tasks)

Evaluation Results: Task Execution

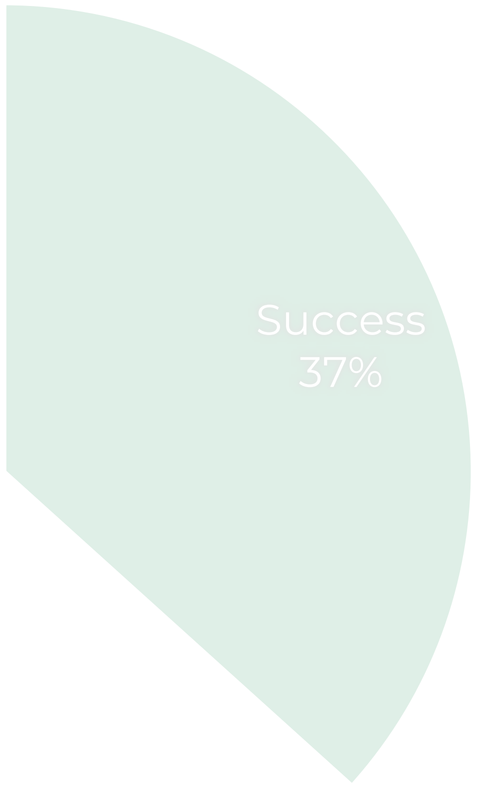


Task Execution Classification
(1,818 valid tasks)

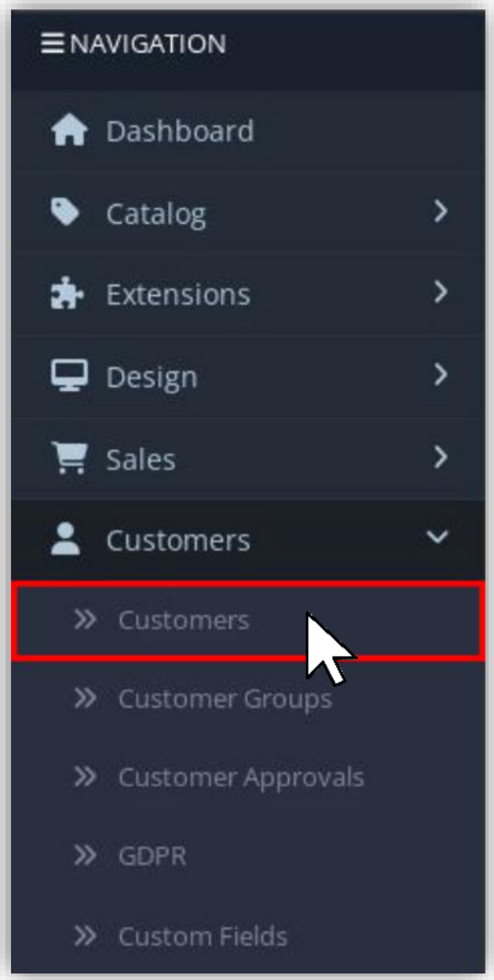


Task: Add new “Customers” to the database

Evaluation Results: Task Execution

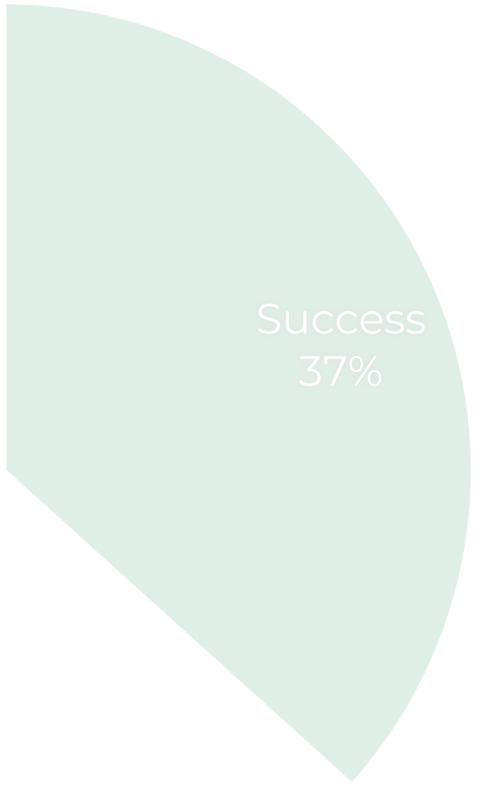


Task Execution Classification
(1,818 valid tasks)

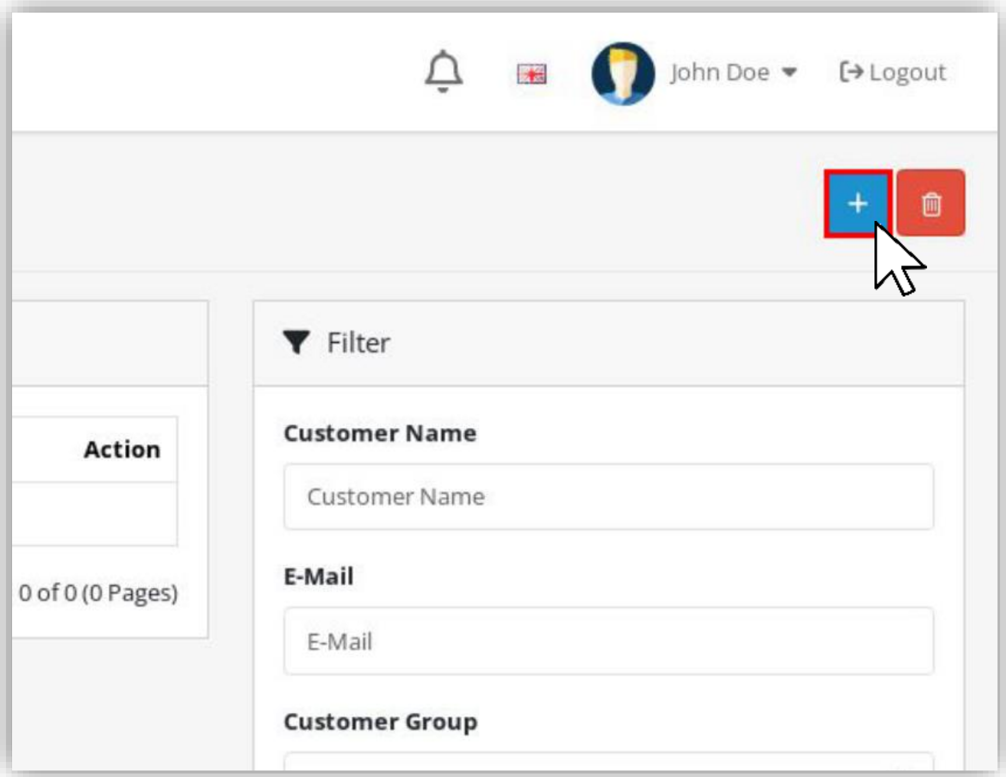


Task: Add new “Customers” to the database

Evaluation Results: Task Execution

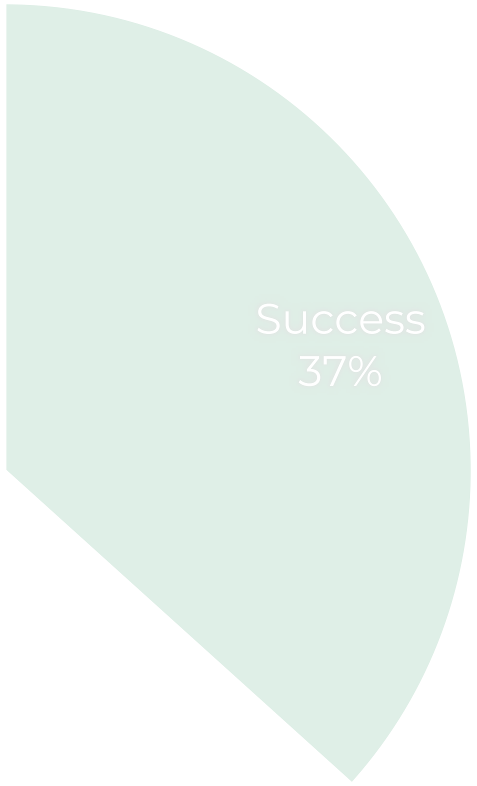


Task Execution Classification
(1,818 valid tasks)

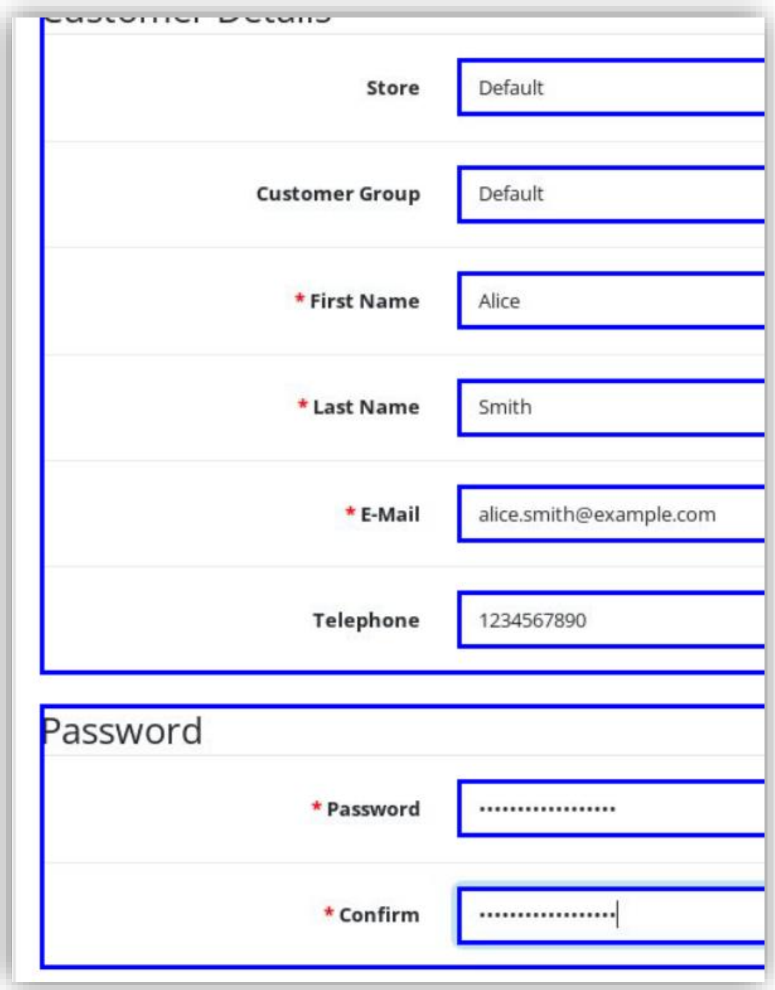


Task: Add new “Customers” to the database

Evaluation Results: Task Execution



Task Execution Classification
(1,818 valid tasks)



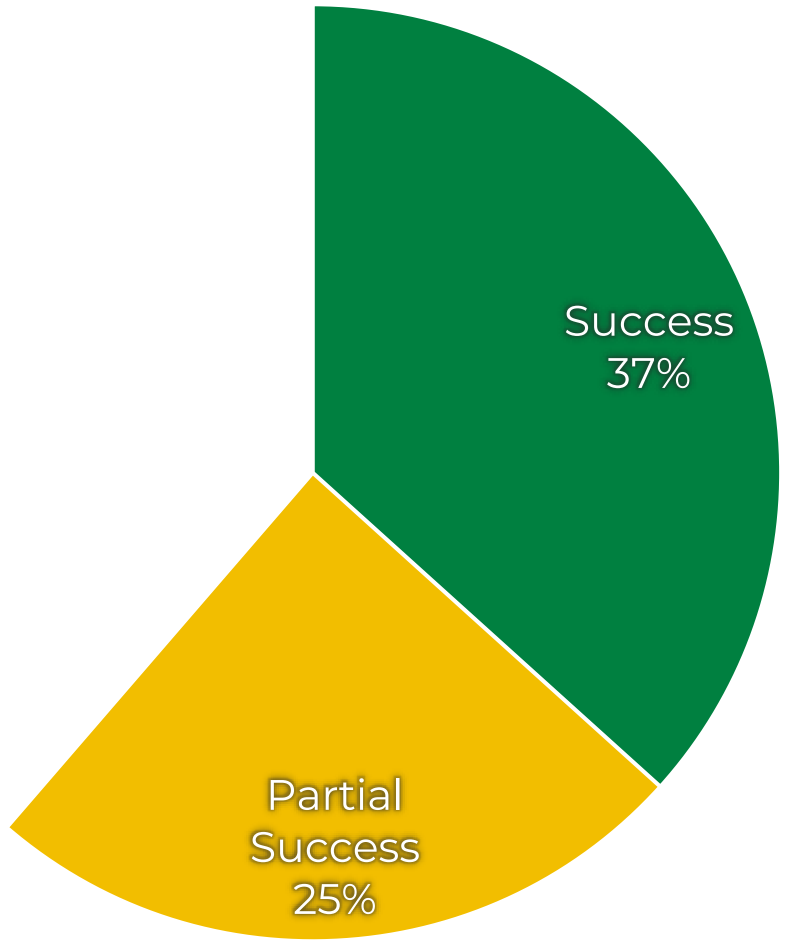
Store	Default
Customer Group	Default
* First Name	Alice
* Last Name	Smith
* E-Mail	alice.smith@example.com
Telephone	1234567890

Password

* Password	••••••••••••••••••
* Confirm	••••••••••••••••••

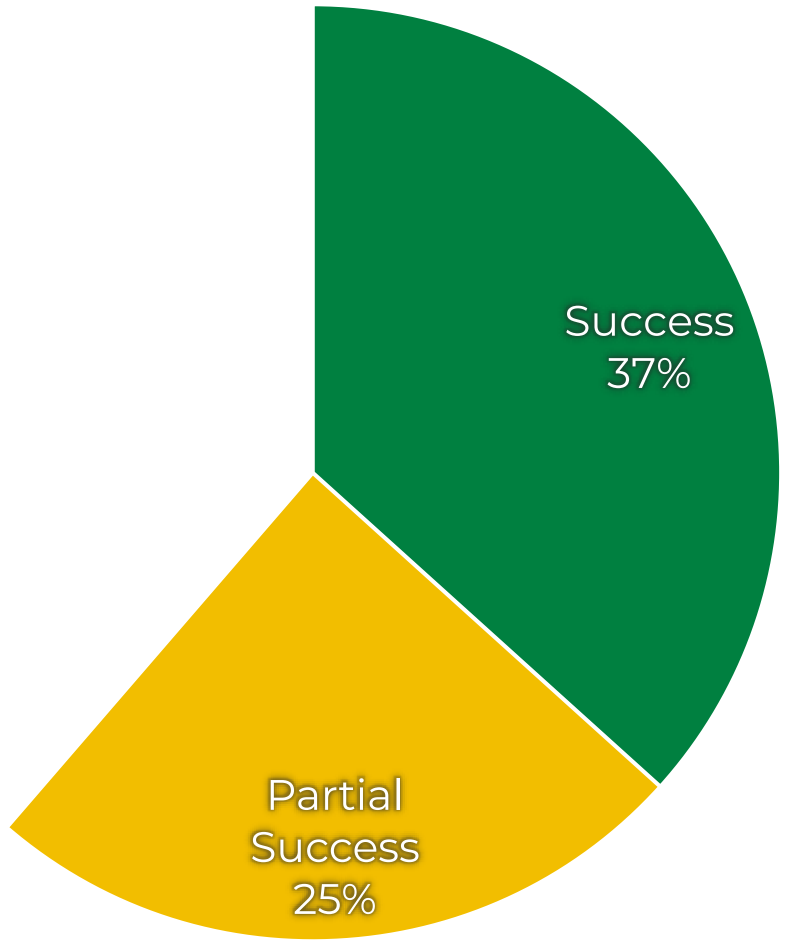
Task: Add new “Customers” to the database

Evaluation Results: Task Execution

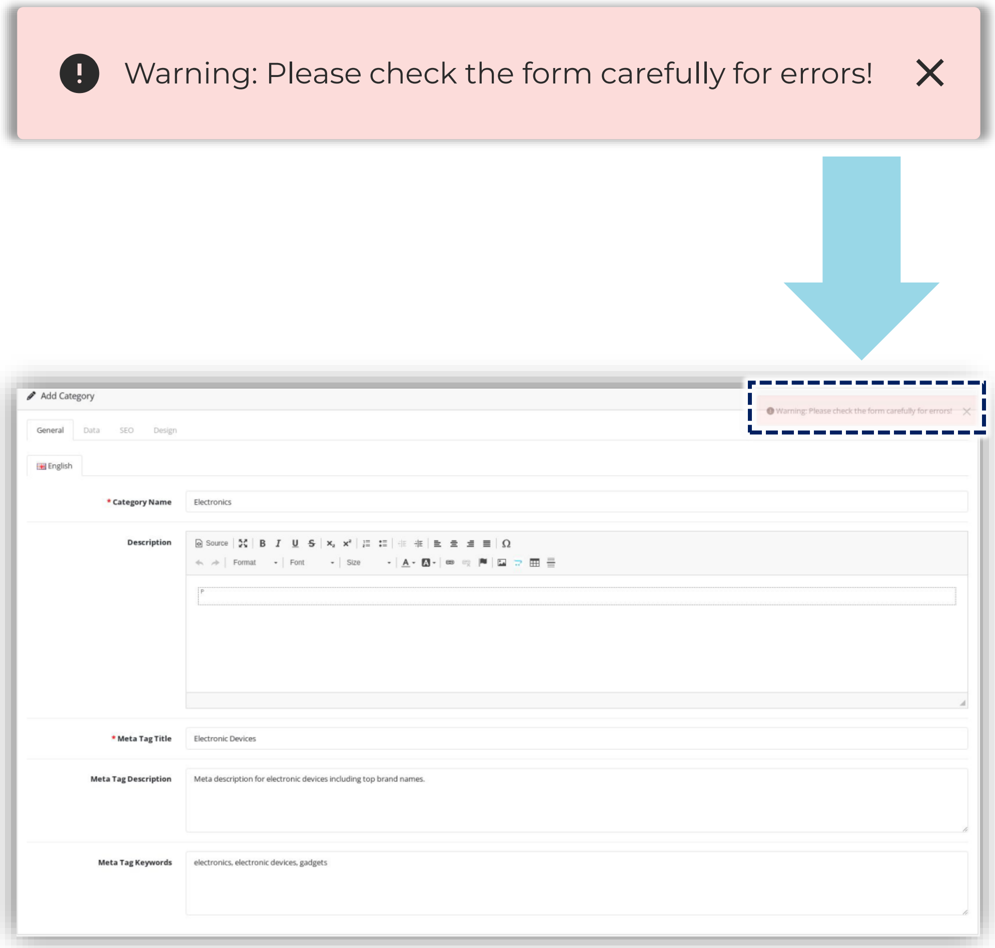


Task Execution Classification
(1,818 valid tasks)

Evaluation Results: Task Execution



Task Execution Classification
(1,818 valid tasks)



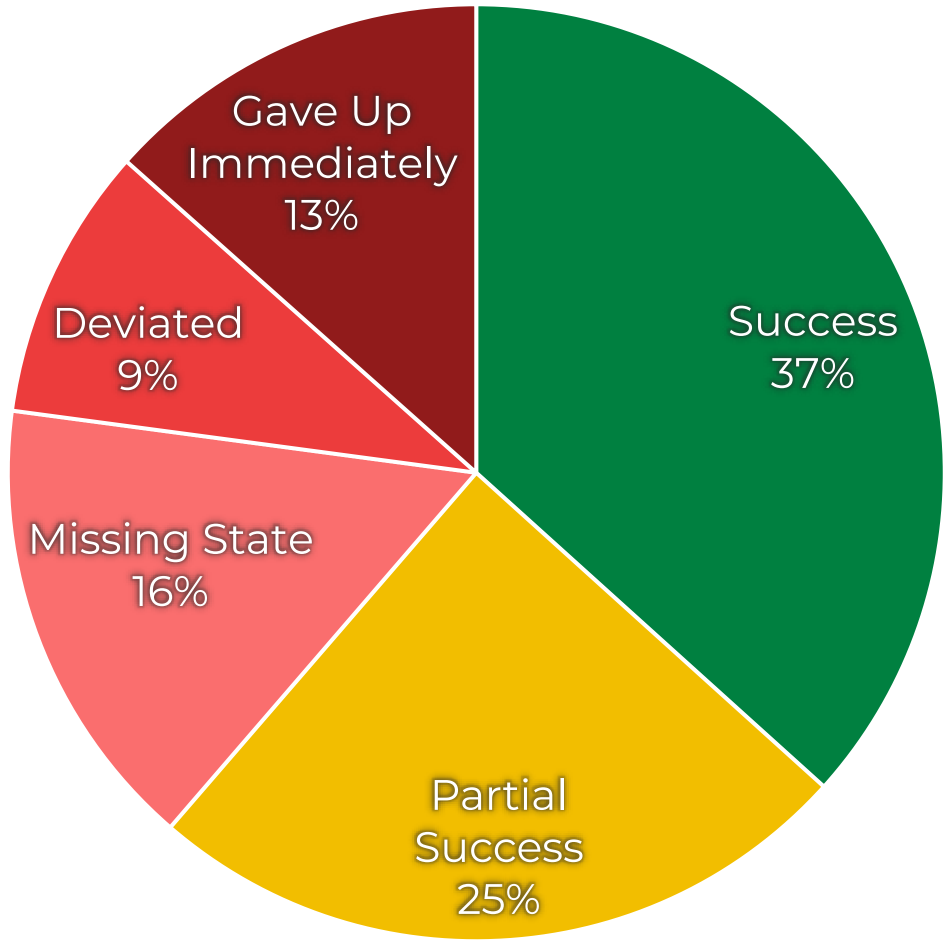
Warning: Please check the form carefully for errors!

Add Category

* Category Name	Electronics
Description	
* Meta Tag Title	Electronic Devices
Meta Tag Description	Meta description for electronic devices including top brand names.
Meta Tag Keywords	electronics, electronic devices, gadgets

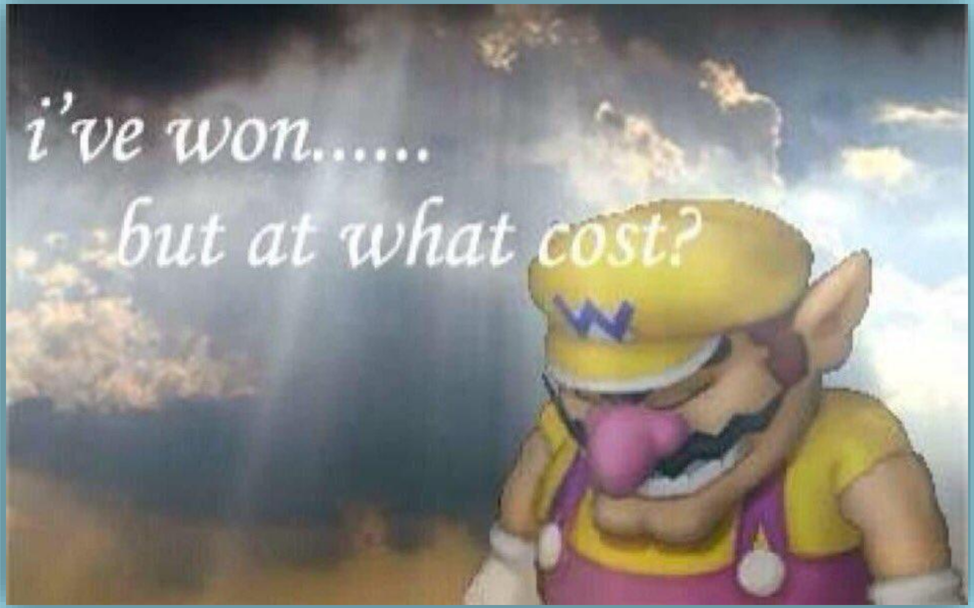
Warning: Please check the form carefully for errors!

Evaluation Results: Task Execution



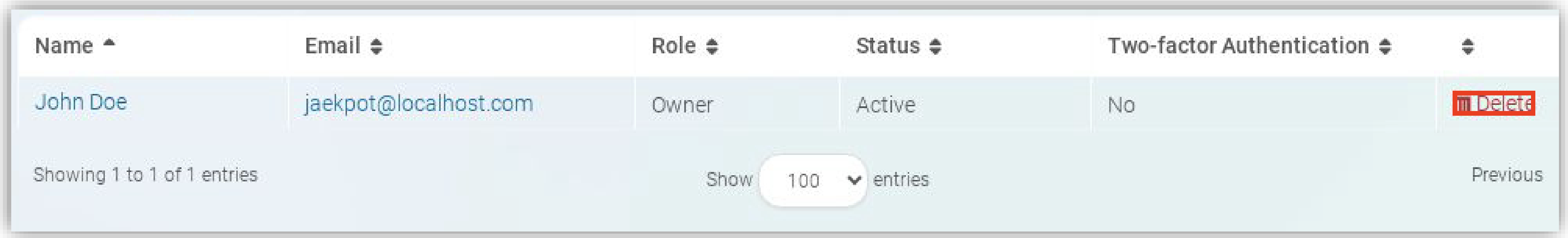
Task Execution Classification
(1,818 valid tasks)

Suffering from Success



Suffering from Success

Task: “Delete a user from the ‘User Management’ section.”



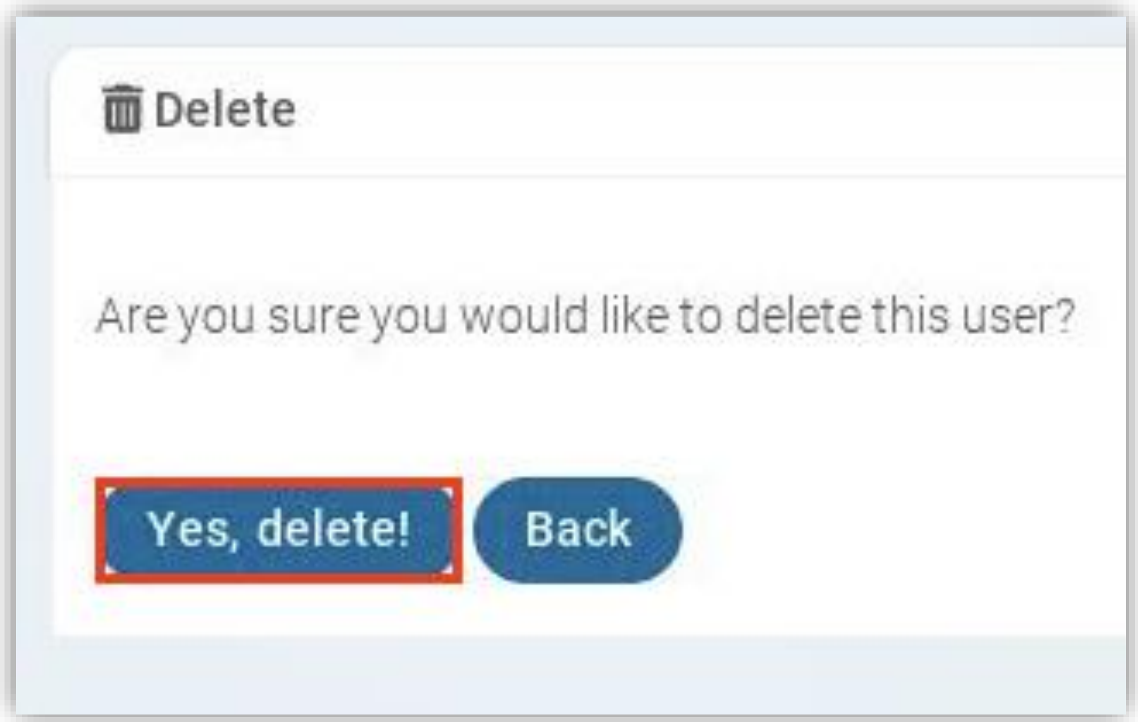
Name	Email	Role	Status	Two-factor Authentication	
John Doe	jaekpot@localhost.com	Owner	Active	No	Delete

Showing 1 to 1 of 1 entries

Show 100 entries

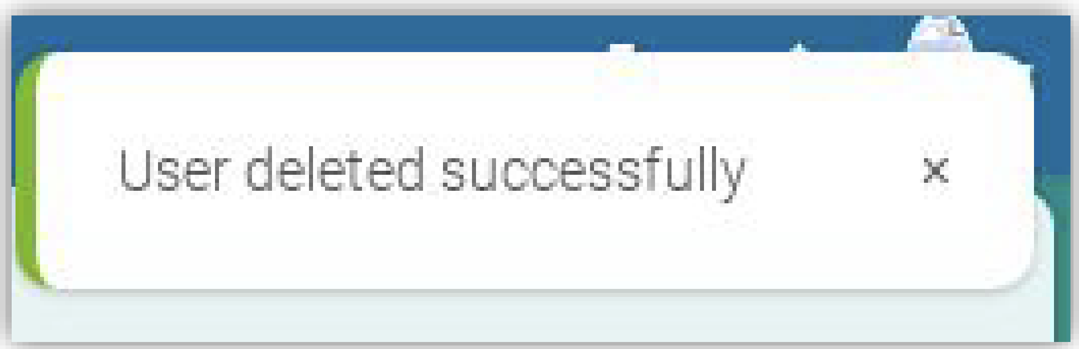
Previous

Suffering from Success

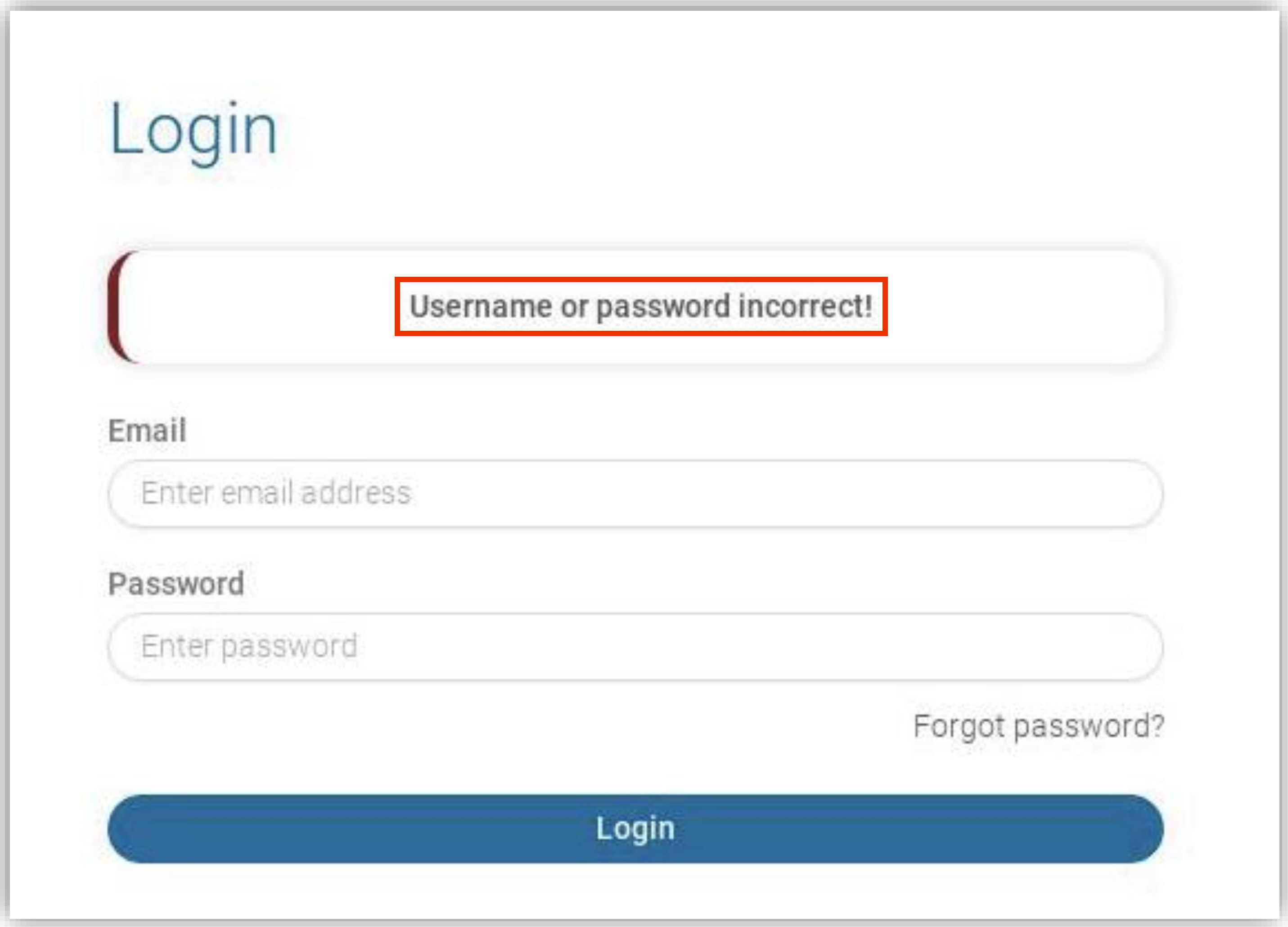


Are you sure you would like to delete this user?

Suffering from Success


User deleted successfully

Suffering from Success


Login

Username or password incorrect!

Email

Enter email address

Password

Enter password

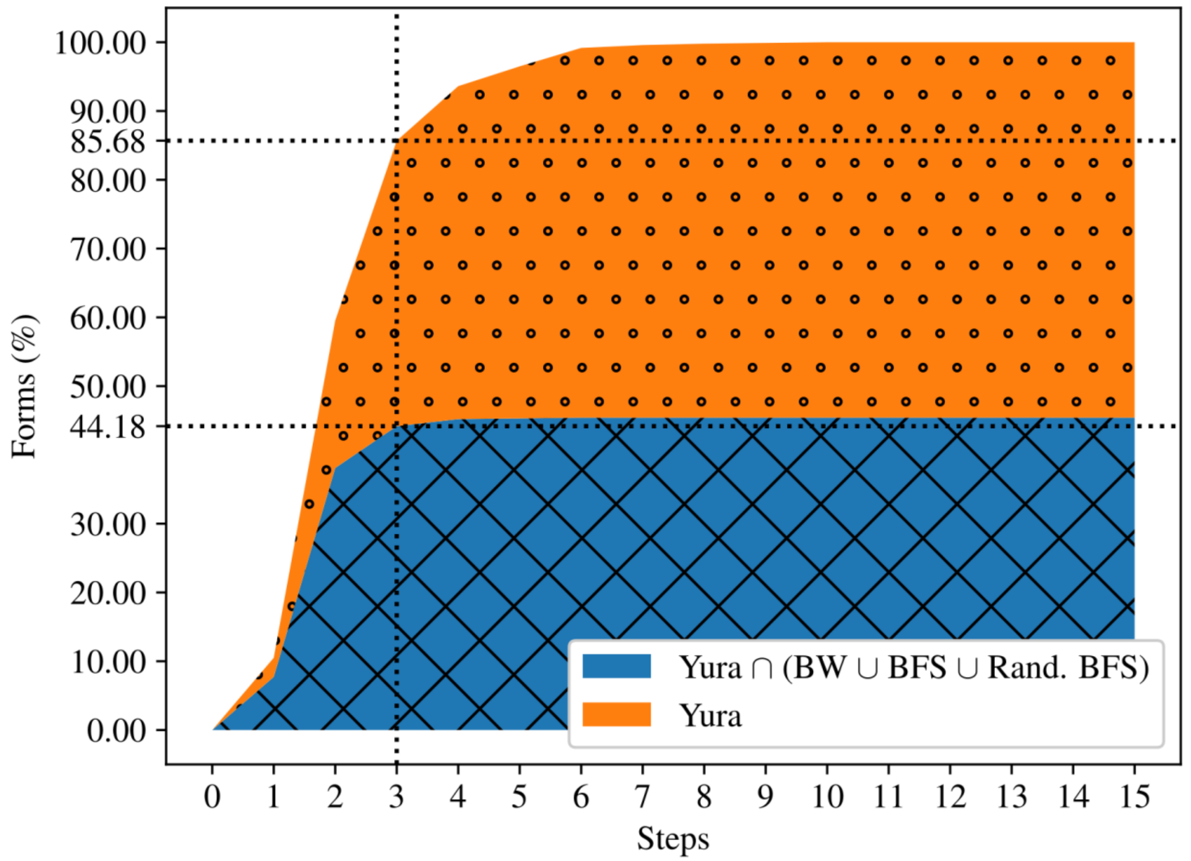
Forgot password?

Login

Directed by
ROBERT B. WEIDE

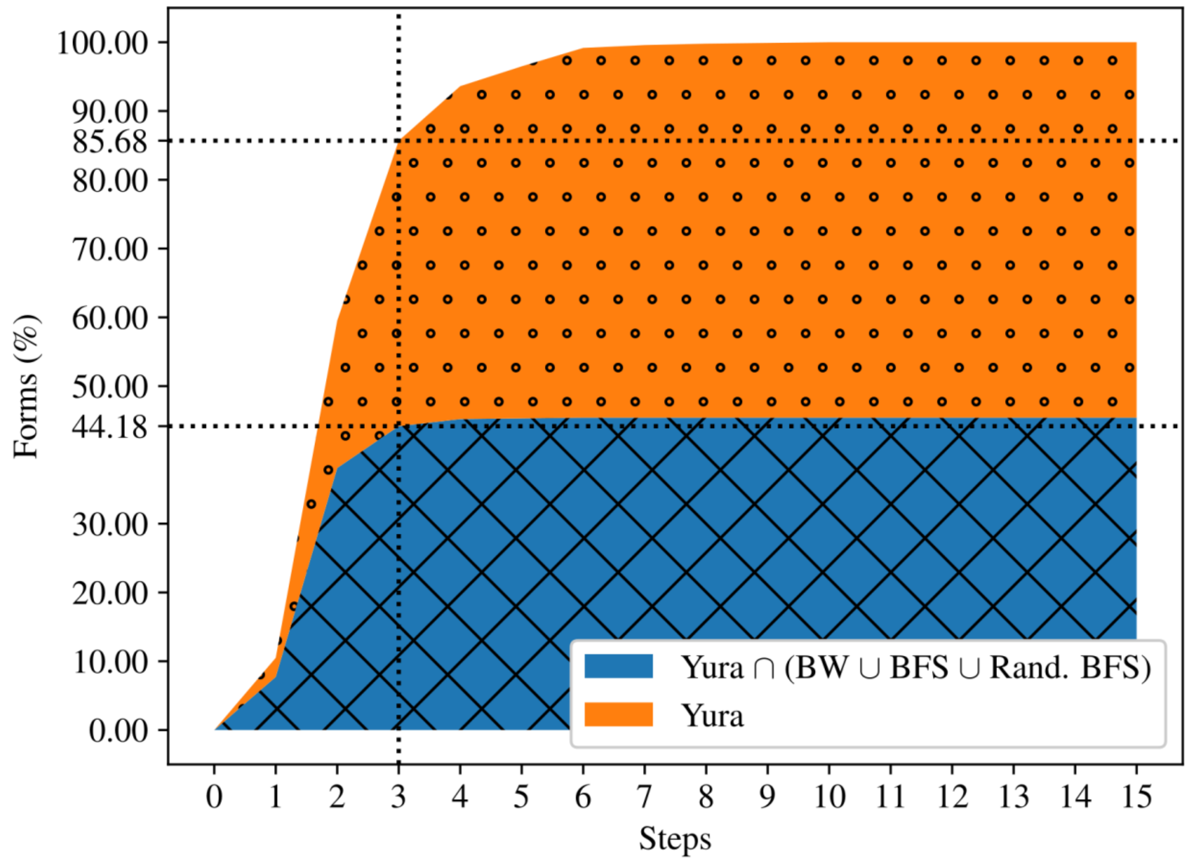
Characterization of the New Attack Surface

- Is the new attack surface found by YuraScanner "deeper"?



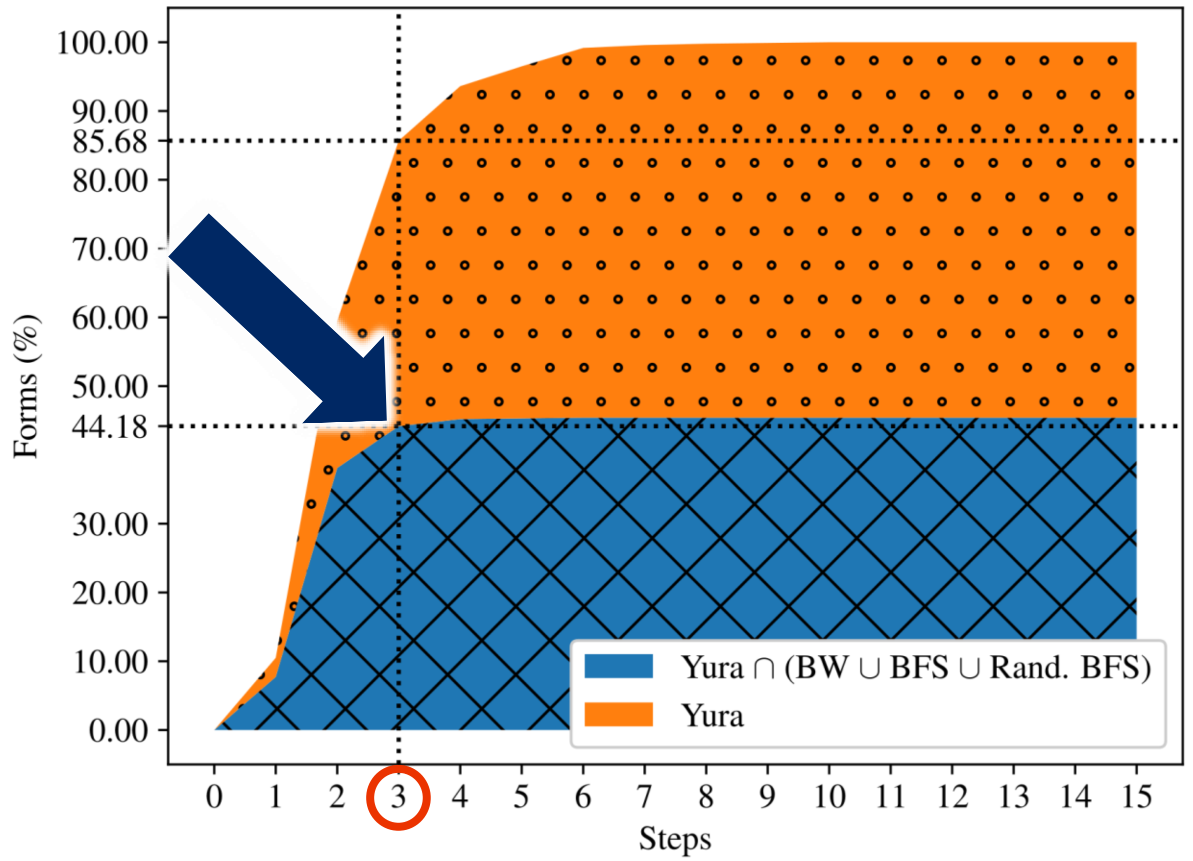
Characterization of the New Attack Surface

- Is the new attack surface found by YuraScanner “deeper”?
- Comparison with Black Widow, BFS and Random BFS
- Percentage of forms collected the deeper we go into the web app



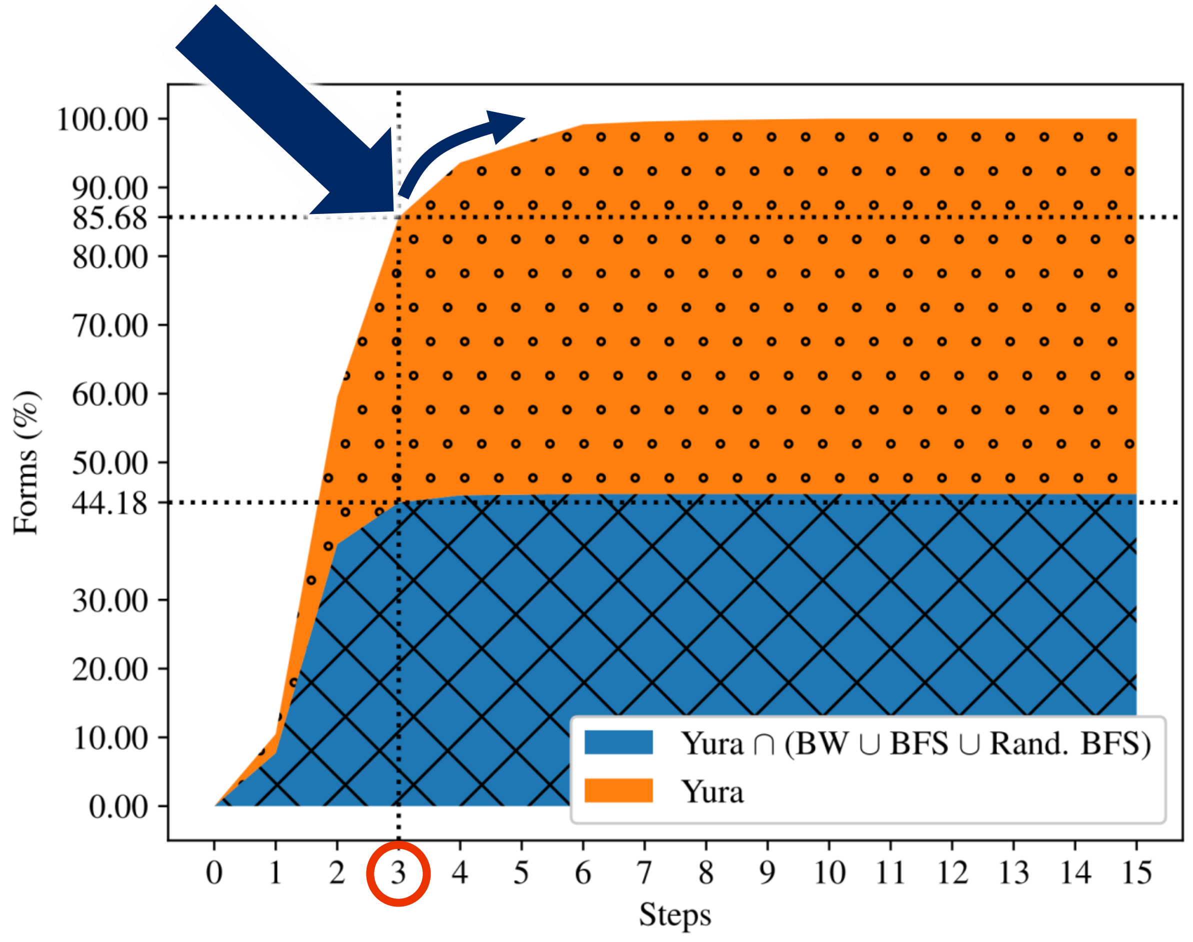
Characterization of the New Attack Surface

- Is the new attack surface found by YuraScanner “deeper”?
- Comparison with Black Widow, BFS and Random BFS
- Percentage of forms collected the deeper we go into the web app
- 44% of forms also discovered by at least one other tool



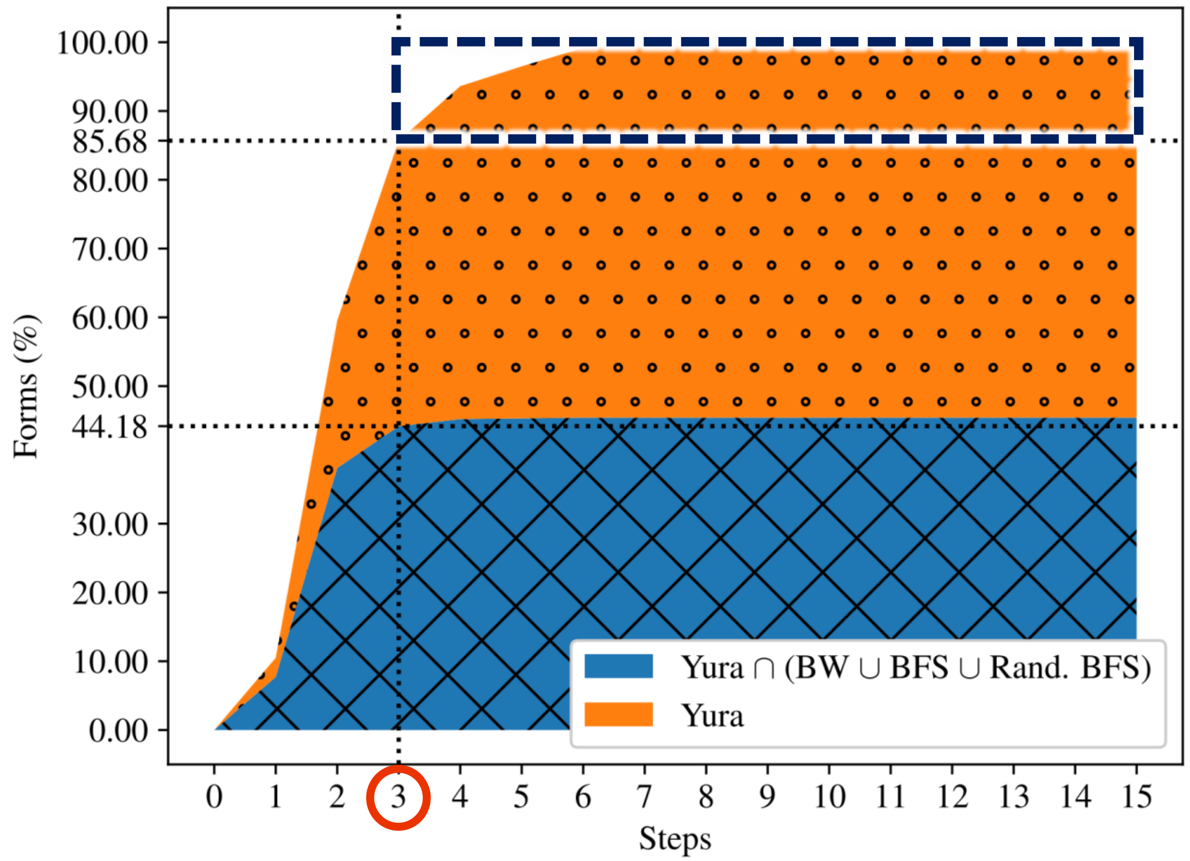
Characterization of the New Attack Surface

- Is the new attack surface found by YuraScanner "deeper"?
- Comparison with Black Widow, BFS and Random BFS
- Percentage of forms collected the deeper we go into the web app
- 44% of forms also discovered by at least one other tool



Characterization of the New Attack Surface

- Is the new attack surface found by YuraScanner “deeper”?
- Comparison with Black Widow, BFS and Random BFS
- Percentage of forms collected the deeper we go into the web app
- 44% of forms also discovered by at least one other tool
- Remaining 14.3% of forms with depth > 3
- Out of reach for existing tools!



Vulnerability Detection

App	Total	Unique	YuraScanner		Black Widow	
			Stored XSS	*Reflected XSS*	*Stored XSS*	*Reflected XSS*
Redacted	**12**	**11**	**4**	**7**	-	**1**
Moodle	**2**	**1**	**1**	-	**1**	-
Leantime	**1**	**1**	-	-	**1**	-

- 13 unique zero-day vulnerabilities discovered
- 12 of them found by YuraScanner

Vulnerability Detection

App	Total	Unique	YuraScanner		Black Widow	
			Stored XSS	*Reflected XSS*	*Stored XSS*	*Reflected XSS*
Redacted	**12**	**11**	**4**	**7**	-	**1**
Moodle	**2**	**1**	**1**	-	**1**	-
Leantime	**1**	**1**	-	-	**1**	-

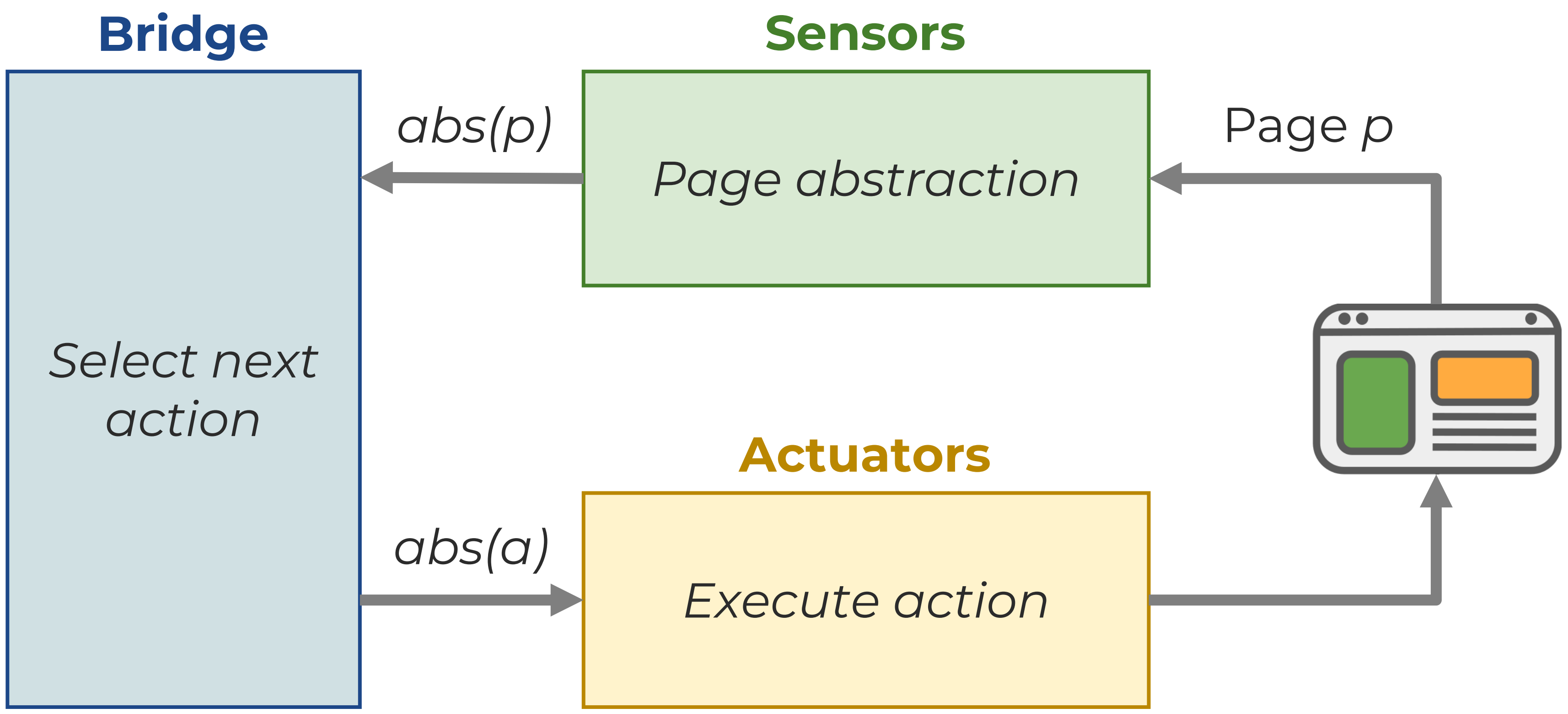
- 13 unique zero-day vulnerabilities discovered
- 12 of them found by YuraScanner
- Located between four and two clicks away from the main page

Summary

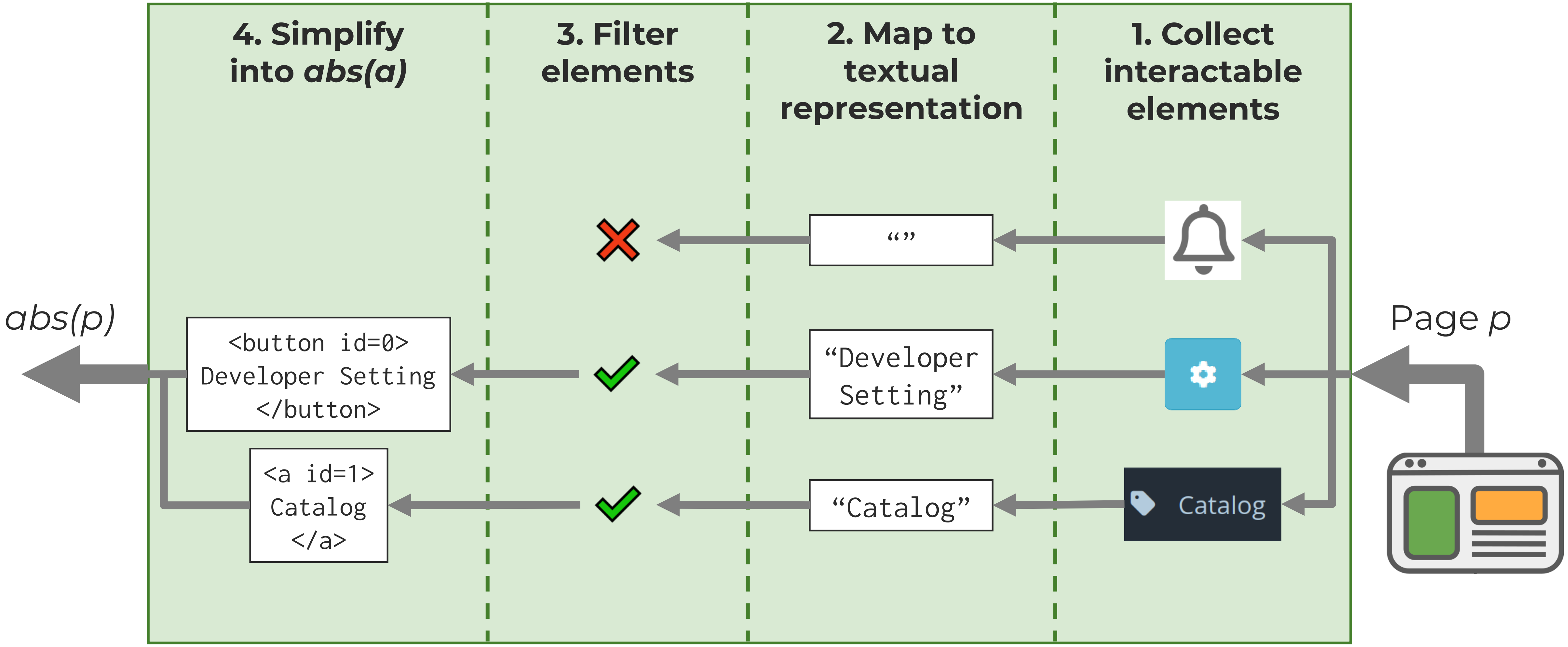
- We implemented **YuraScanner**, a task-driven web application scanner
- Fully automated approach
- YuraScanner executed 61.3% of the valid tasks completely or partially
- The unique attack surface discovered by YuraScanner is deeper compared to traditional scanners
- Task-driven crawling effectively complements traditional scanning techniques

Artifact Evaluated
NDSS SYMPOSIUM
Functional
Reproduced

Task-driven Crawler Component

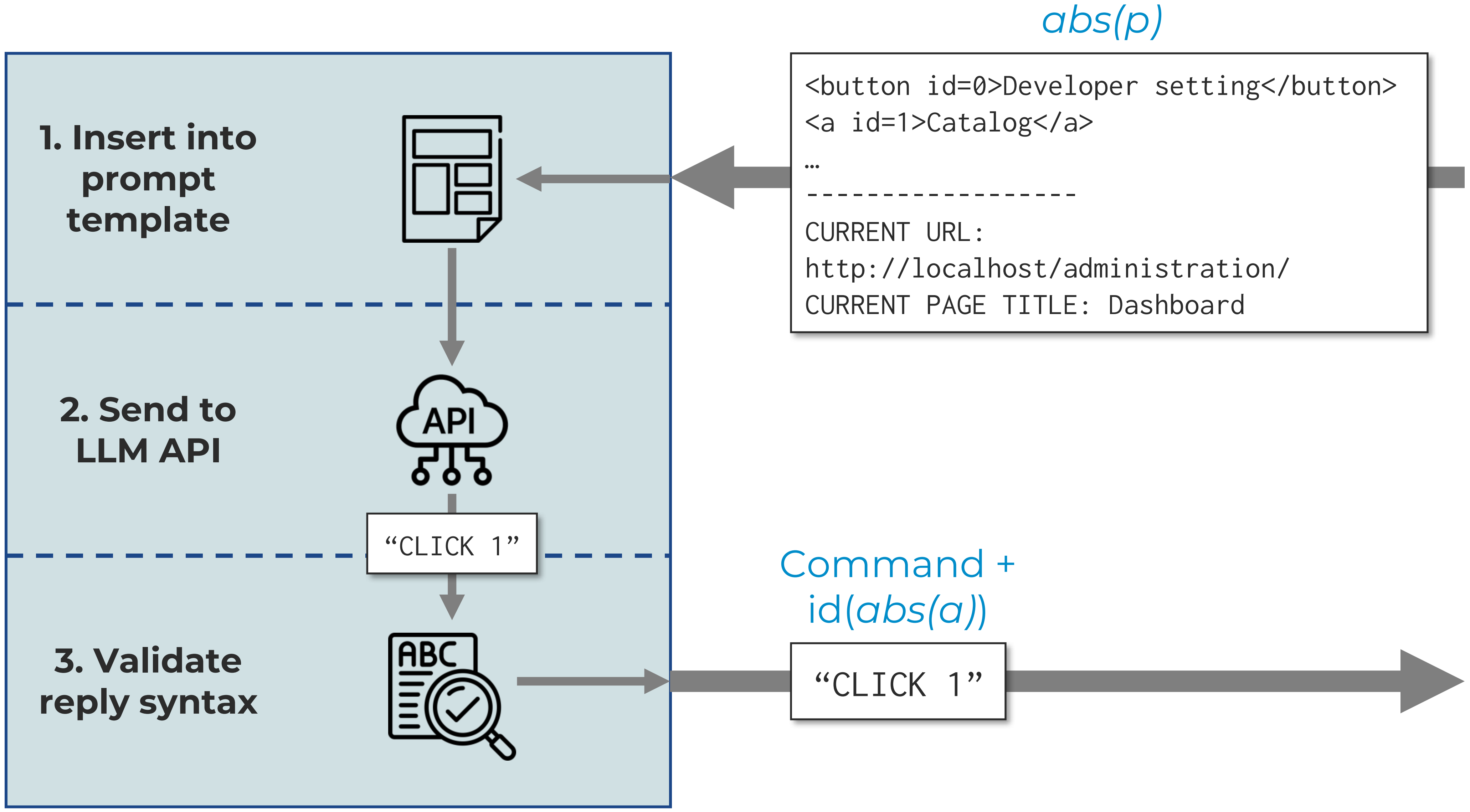


Sensors



4. Simplify into *abs(a)*
3. Filter elements
2. Map to textual representation
1. Collect interactable elements
abs(p)
Page *p*
“”
“Developer Setting”
“Catalog”
Catalog
<button id=0> Developer Setting </button>
<a id=1> Catalog </a>

Bridge



```
<button id=0>Developer setting</button>
<a id=1>Catalog</a>
…
------------------
CURRENT URL:
http://localhost/administration/
CURRENT PAGE TITLE: Dashboard
```

Bridge: LLM Prompt

Preamble with instructions

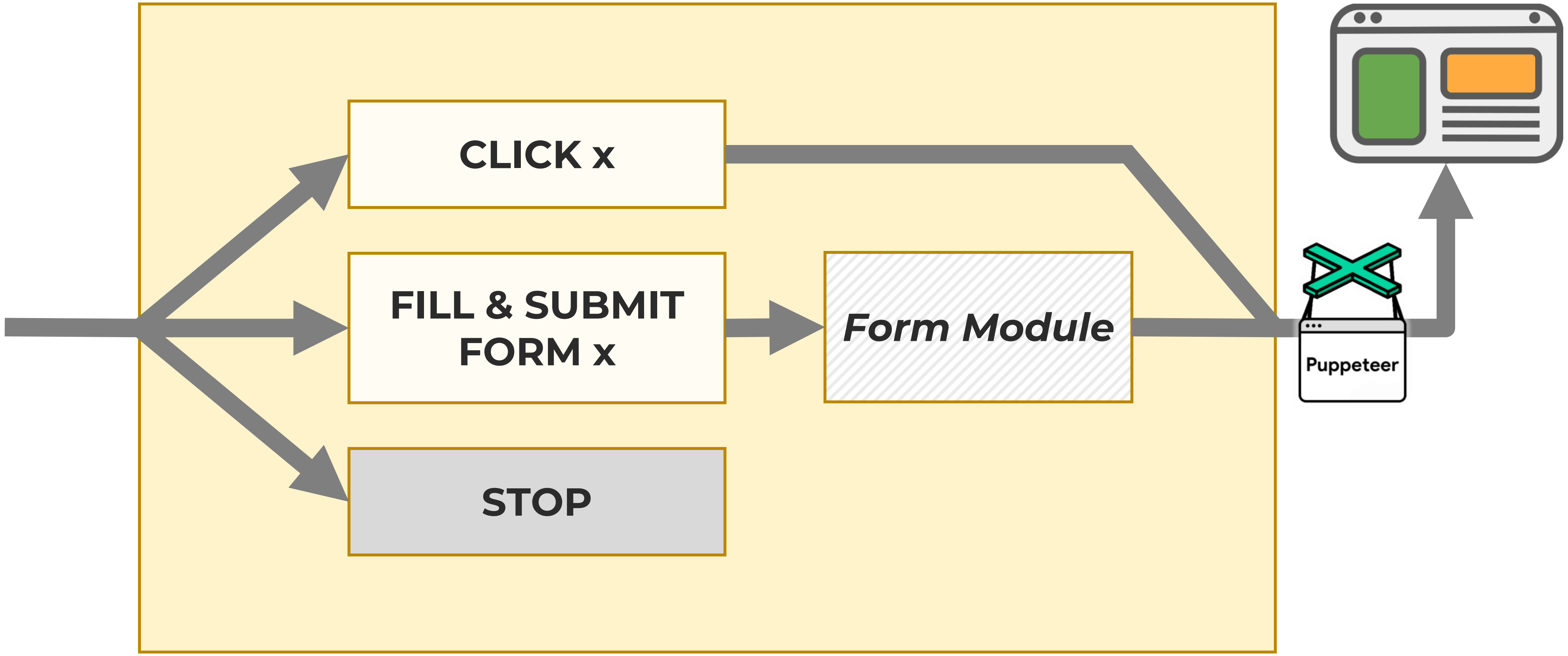
- Persona assignment to increase focus:
  "*You are Yura, an agent controlling a web browser*"
- Explanation of abstract page
- Command types

Example of input and expected output
("One-shot")

Current input

- Current task
- Current abstract page (i.e., *abs(p)*)
- History of previous actions

Actuators


CLICK x
FILL & SUBMIT FORM x
STOP
Form Module
Puppeteer