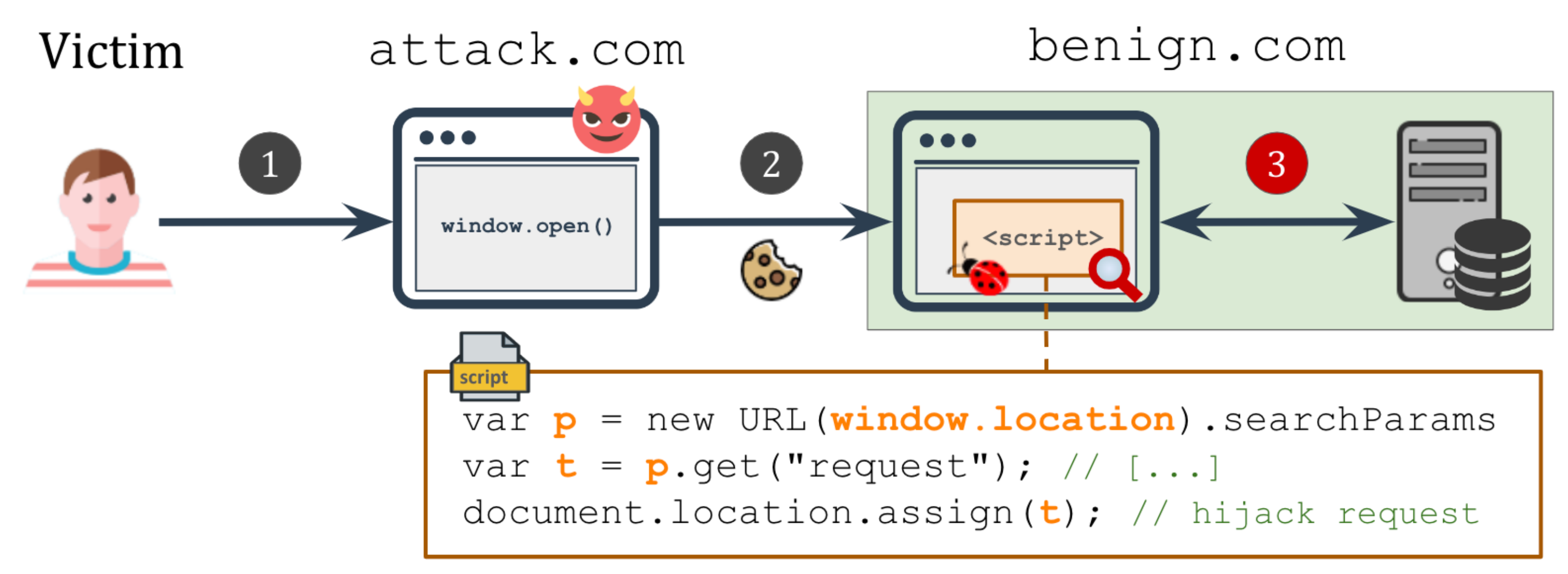
The Great Request Robbery
An Empirical Study of Client-side Request Hijacking

Soheil Khodayari, Thomas Barber, and Giancarlo Pellegrino

WHAT IS REQUEST HIJACKING?

Manipulate request-sending instructions with **arbitrary** inputs



```
var p = new URL(window.location).searchParams
var t = p.get("request"); // [...]
document.location.assign(t); // hijack request
```

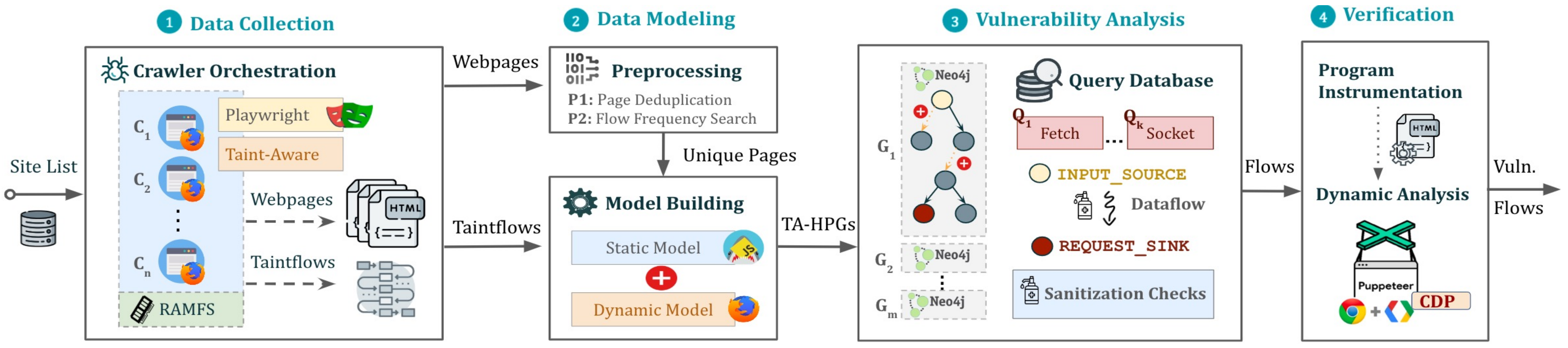
BROWSER APIS AND THREATS

Assess modern browser APIs and their **capabilities**
- network schemes (HTTP, JS)
- request methods (GET/POST/ANY)
- request fields (Header, URL, Body)

Identified **10 different request APIs**
XMLHttpRequest, fetch, sendBeacon, WebSocket, EventSource, Location, Push, Window Open

Threat: attacker being able to forge request API fields
Impact: CSRF, XSS, and Information Leakage

VULNERABILITY DETECTION: SHERIFF (JAW + FOXHOUND)



EMPIRICAL STUDY

Testbed:

Tranco top **10K** sites, 339K webpages, 32.4B LoC

Results:

Detected **202K** vuln. data flows across **961** affected sites

The new vulnerability types and variants constitute over **36%** of the request hijacks

Examples: sendBeacon, push API, WebSocket, EventSource

DYNAMIC INFO CONTRIBUTION

Captured **21.6M** dynamic flows to sinks (3.3M for requests)

DAST: supplement SAST edges
SAST: help eliminate spurious DAST flows

Data Flow Edge Types

- Dynamic: ~ 118K flows
- Mixed: ~ 18K flows
- Static: ~ 66K flows

Conclusion: dynamic info crucial to detect **67%** of the request hijacking data flows

EXPLOITABILITY ANALYSIS

Demonstrated exploitability by manually analyzing a random subset of the vulnerable pages

Created PoC exploits for **49 popular sites**

- Microsoft Azure: XSS
- Starz: account takeover
- TP-Link: client-side XSS
- BBC and DW: CSRF on user account settings
- JustWatch: data exfiltration

MITIGATION TECHNIQUES

Custom (Application)

- Input Validation
- Fetch MetaData

Policy-based (Browser)

- Content Security Policy
- Cross-Origin Opener Policy
- Cross-Origin Embed. Policy

Tokens, SameSite, CORS:

Ineffective against client-side request hijacking

SAP CISPA HELMHOLTZ CENTER FOR INFORMATION SECURITY

This poster is based on the following publication:
Soheil Khodayari, Thomas Barber, and Giancarlo Pellegrino, The Great Request Robbery: An Empirical Study of Client-side Request Hijacking Vulnerabilities on the Web, In Proceedings of 45th IEEE Symposium on Security and Privacy (SP '24), CA, USA, May 20-23, 2024.

soheil.khodayari@cispa.de @Soheil__K https://ja-w.me https://github.com/SAP/project-foxhound