CISPA
HELMHOLTZ CENTER FOR INFORMATION SECURITY

Do (Not) Follow the White Rabbit: Challenging the Myth of Harmless Open Redirection

Soheil Khodayari, Kai Glauber✝, and Giancarlo Pellegrino**

Presenter: Gianluca De Stefano

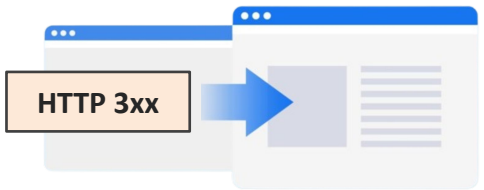
NDSS SYMPOSIUM

*CISPA - Helmholtz Center for Information Security

✝Saarland University

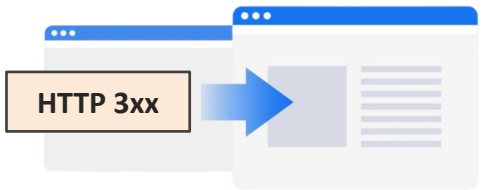
Open Redirect Vulnerability

- HTTP redirections guide users from one resource to another
  - Traditionally **server-side**



Open Redirect Vulnerability

- HTTP redirections guide users from one resource to another
  - Traditionally **server-side**



- Destination specified often through a **URL parameter**

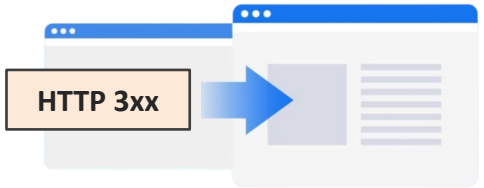
`trusted.com?redir=/profile`

Open Redirect Vulnerability

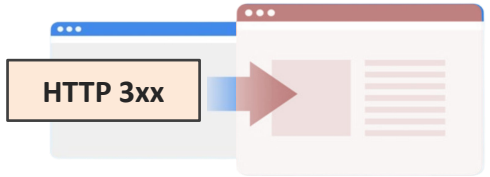
- HTTP redirections guide users from one resource to another
  - Traditionally **server-side**

- Destination specified often through a **URL parameter**

Open redirect vulnerability: redirect parameter is **not validated**



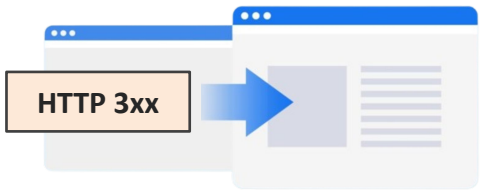
`trusted.com?redir=/profile`



`trusted.com?redir=//evil.com`

Open Redirect Vulnerability

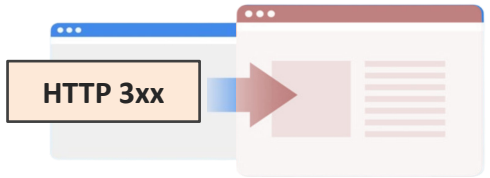
- HTTP redirections guide users from one resource to another
  - Traditionally **server-side**



- Destination specified often through a **URL parameter**

`trusted.com?redir=/profile`

Open redirect vulnerability: redirect parameter is **not validated**

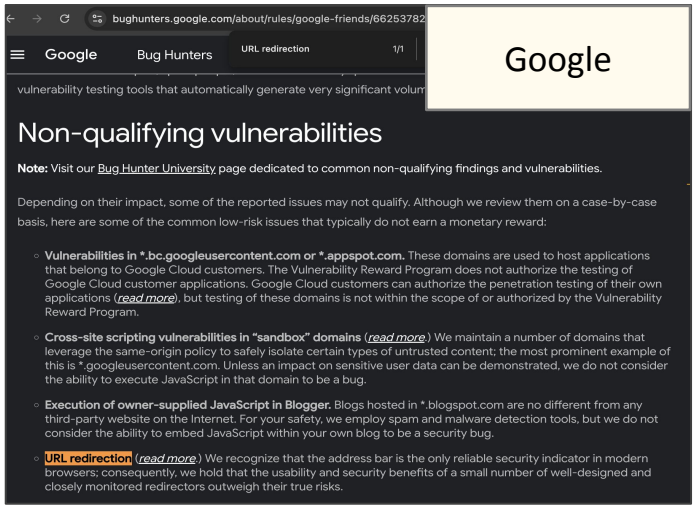


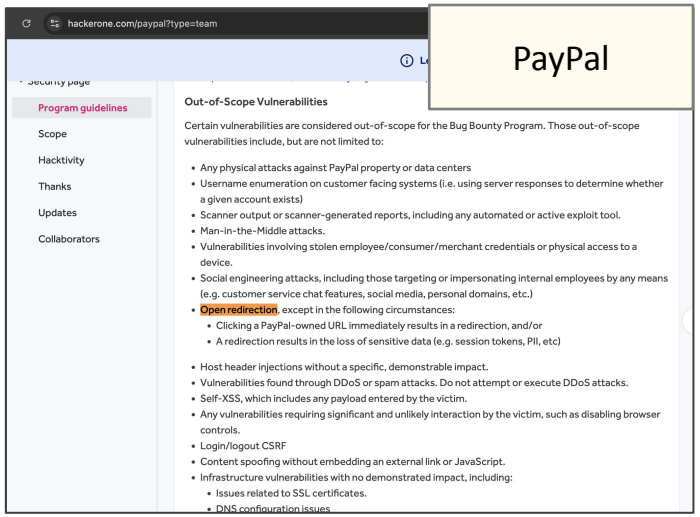
- **Limited exploitation** scenario
  - Abuse vulnerable sites to **mask malicious URLs**
  - **No harm** to site itself

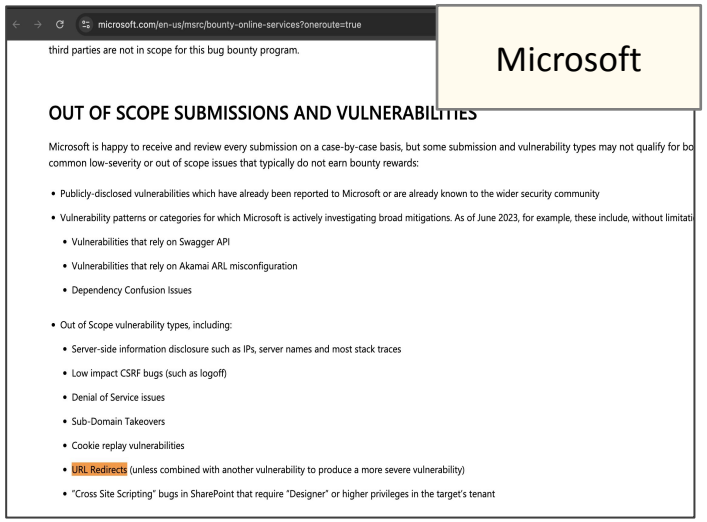
`trusted.com?redir=//evil.com`

Open Redirect Vulnerability: Harmless, Until They're Not

- Vulnerability disclosure programs often **do not consider** them as **qualifying issues**



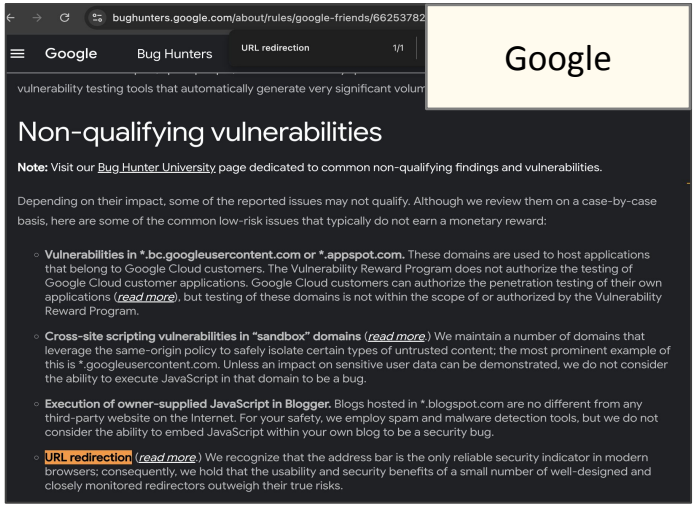


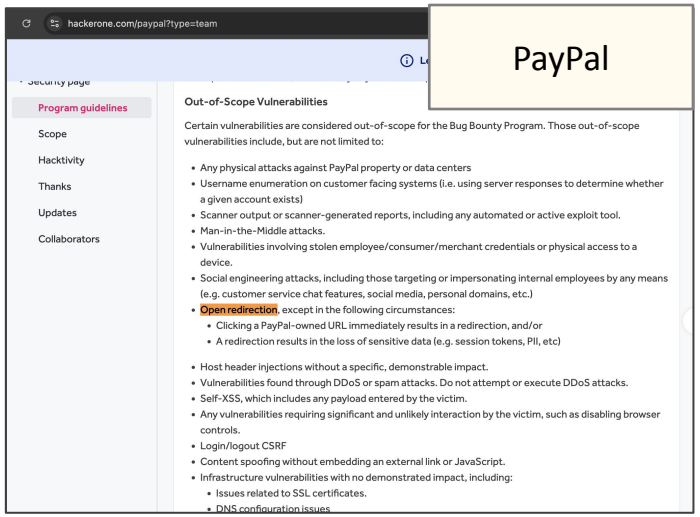


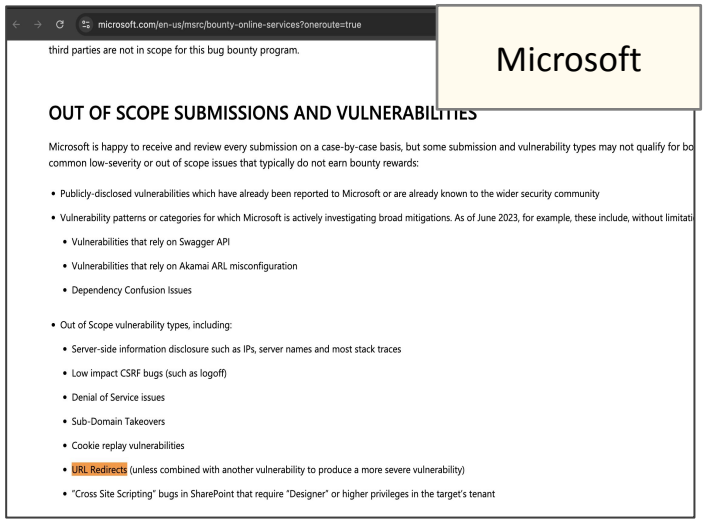
Source: [1] https://cve.mitre.org/cgi-bin/cvekey.cgi?keyword=open+redirect

Open Redirect Vulnerability: Harmless, Until They're Not

- Vulnerability disclosure programs often **do not consider** them as **qualifying issues**





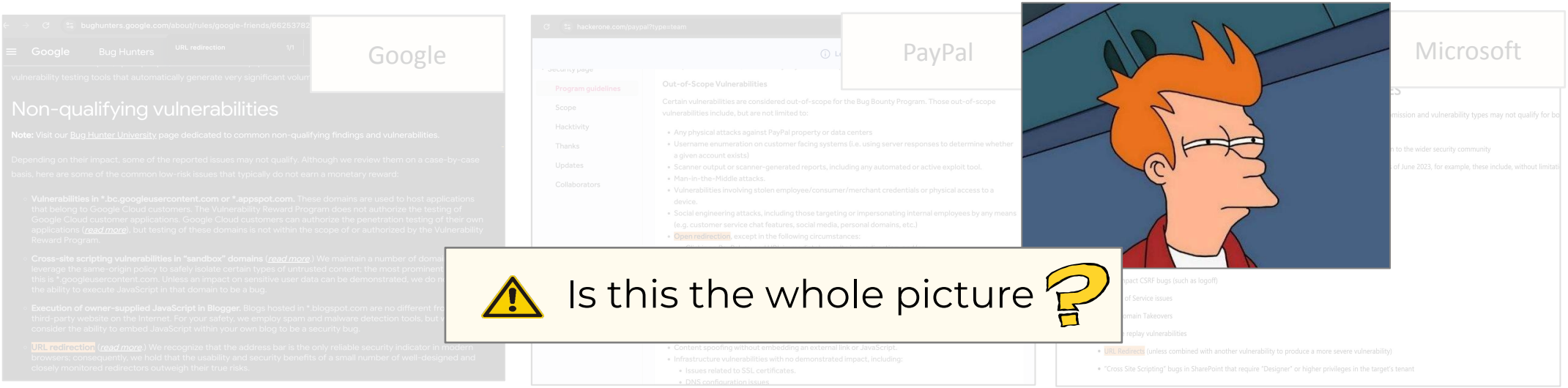


- Low prevalence of **reported instances** in CVE database
  - Only about 1% compared to Cross-Site Scripting 37%[1]

Source: [1] https://cve.mitre.org/cgi-bin/cvekey.cgi?keyword=open+redirect

Open Redirect Vulnerability: Harmless, Until They're Not

- Vulnerability disclosure programs often **do not consider** them as **qualifying issues**

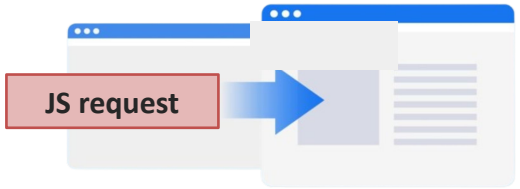


- Low prevalence of **reported instances** in CVE database
  - Only about 1% compared to Cross-Site Scripting 37%[1]

Source: [1] https://cve.mitre.org/cgi-bin/cvekey.cgi?keyword=open+redirect

Client-side Open Redirect

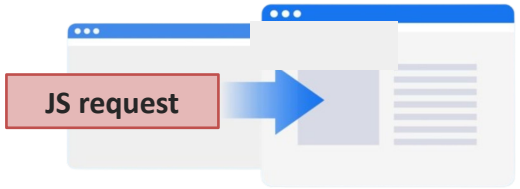
- Recent shift towards client-side task offloading has introduced **JS-based redirections**



Client-side Open Redirect

- Recent shift towards client-side task offloading has introduced **JS-based redirections**

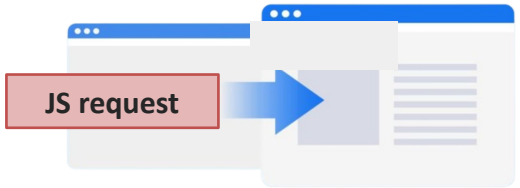
- Poses **additional risks** to open redirects

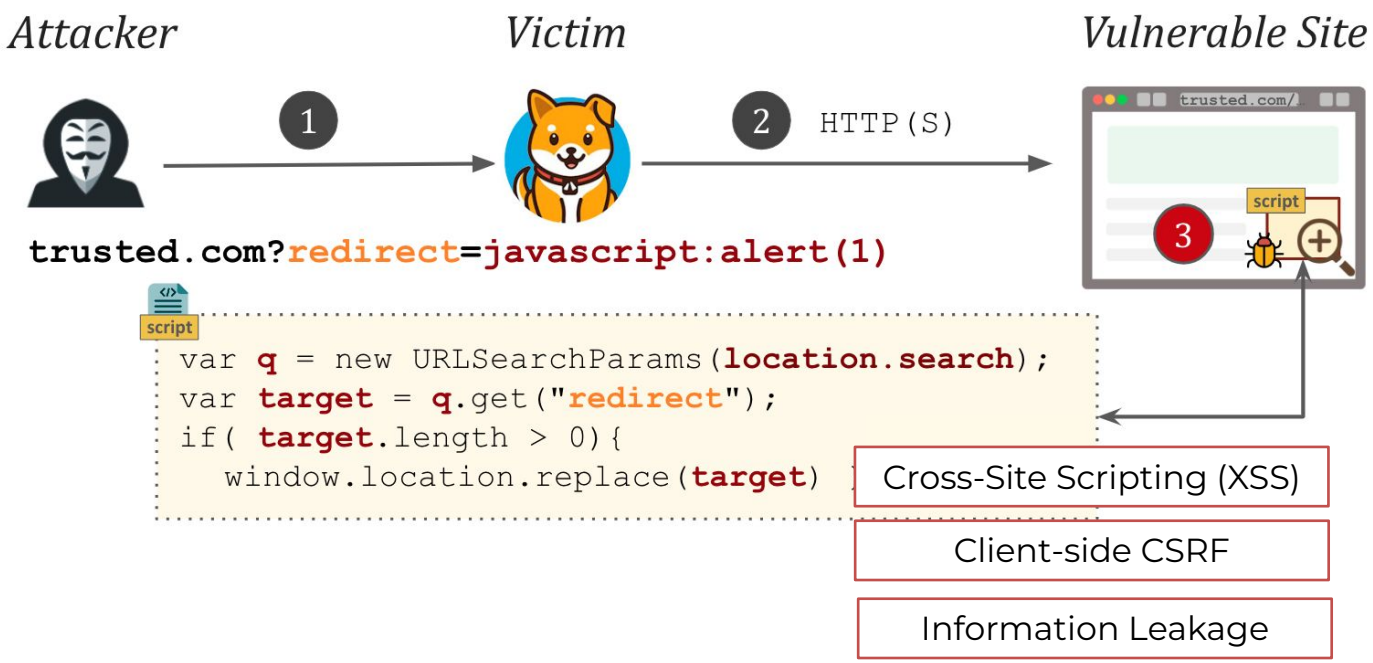


Client-side Open Redirect

- Recent shift towards client-side task offloading has introduced **JS-based redirections**

- Poses **additional risks** to open redirects





Client-side Open Redirect

- Recent shift towards client-side task offloading has introduced **JS-based redirections**

- Poses **additional risks** to open redirects

Objective: focus on **client-side,** re-evaluate the risk of open redirects

Cross-Site Scripting (XSS)

Client-side CSRF

Information Leakage

Client-side Open Redirect

- Recent shift towards client-side task offloading has introduced **JS-based redirections**

- Poses **additional risks** to open redirects

Objective: focus on **client-side,** re-evaluate the risk of open redirects

Attacker *Victim* *Vulnerable Site*

How can we **detect** such impactful open redirect problems

Open Redirect Detection: Problem Statement

- **Approach 1:** hand-crafted vulnerability **indicators** [Shue et al., WOOT, 2008] [Wang et al., IEEE CNS, 2015]

```
trusted.com?redir=/profile
```

(+) Lightweight

(-) Coverage of the indicators: creating **a comprehensive list manually** is challenging

Open Redirect Detection: Problem Statement

- **Approach 1:** hand-crafted vulnerability **indicators** [Shue et al., WOOT, 2008] [Wang et al., IEEE CNS, 2015]

```
trusted.com?redir=/profile
```

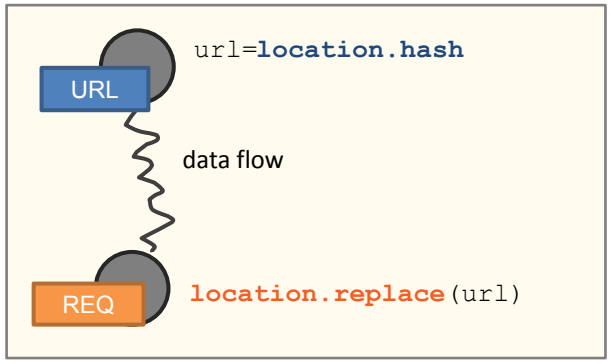
  (+) Lightweight

  (-) Coverage of the indicators: creating **a comprehensive list manually** is challenging

- **Approach 2:** static analysis of client-side JavaScript [Khodayari et al., IEEE SP, 2024]

  (+) Improved code coverage

  (-) Resource-intensive, resort to webpage sampling strategies



Open Redirect Detection: Problem Statement

- **Approach 1:** hand-crafted vulnerability **indicators** [Shue et al., WOOT, 2008] [Wang et al., IEEE CNS, 2015]

```
trusted.com?redir=/profile
```

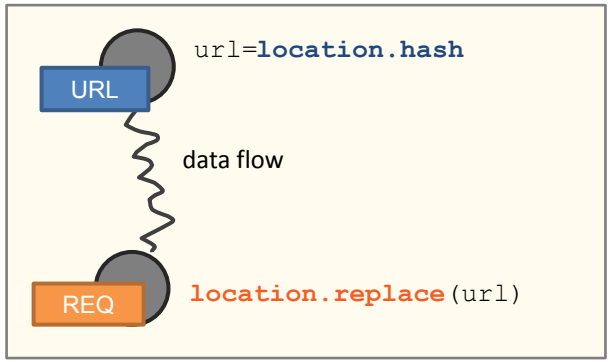
  (+) Lightweight

  (-) Coverage of the indicators: creating **a comprehensive list manually** is challenging

- **Approach 2:** static analysis of client-side JavaScript [Khodayari et al., IEEE SP, 2024]

  (+) Improved code coverage

  (-) Resource-intensive, resort to webpage sampling strategies



- **Our solution:** a novel cost-reduction methodology

Research Questions

- **RQ1: Vulnerability Indicators**

How can we **use static analysis** to extract **indicative patterns** of open redirects in real websites?

Research Questions

- **RQ1: Vulnerability Indicators**

  How can we **use static analysis** to extract **indicative patterns** of open redirects in real websites?

- **RQ2: Vulnerability Mining and Prevalence**

  How **prevalent** are open redirects, and can we **mine them efficiently** at scale?

Research Questions

- **RQ1: Vulnerability Indicators**

  How can we **use static analysis** to extract **indicative patterns** of open redirects in real websites?

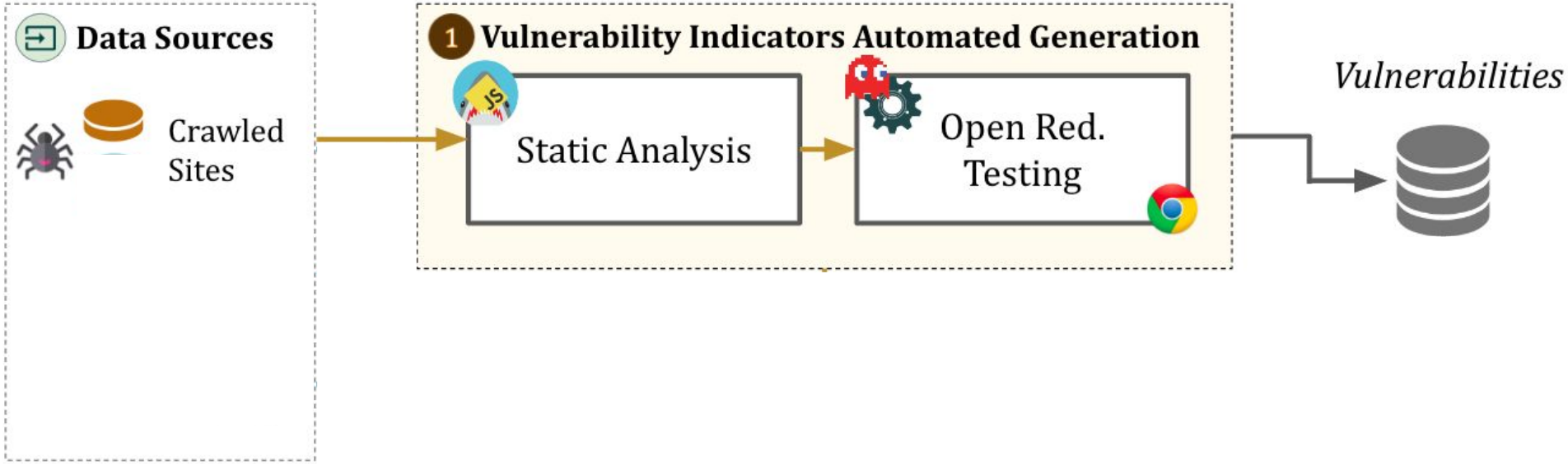
- **RQ2: Vulnerability Mining and Prevalence**

  How **prevalent** are open redirects, and can we **mine them efficiently** at scale?

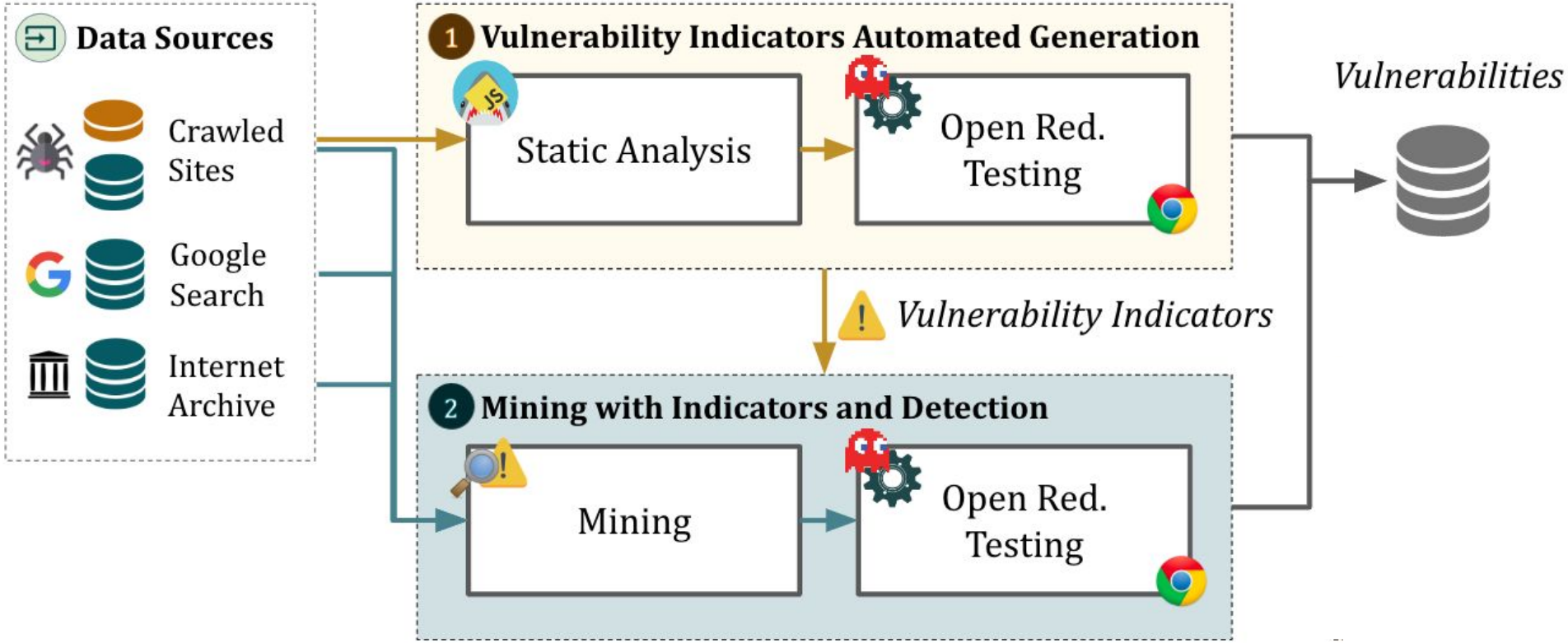
- **RQ3: Exploitability Analysis**

  How can open redirects **escalate** into more severe attacks?

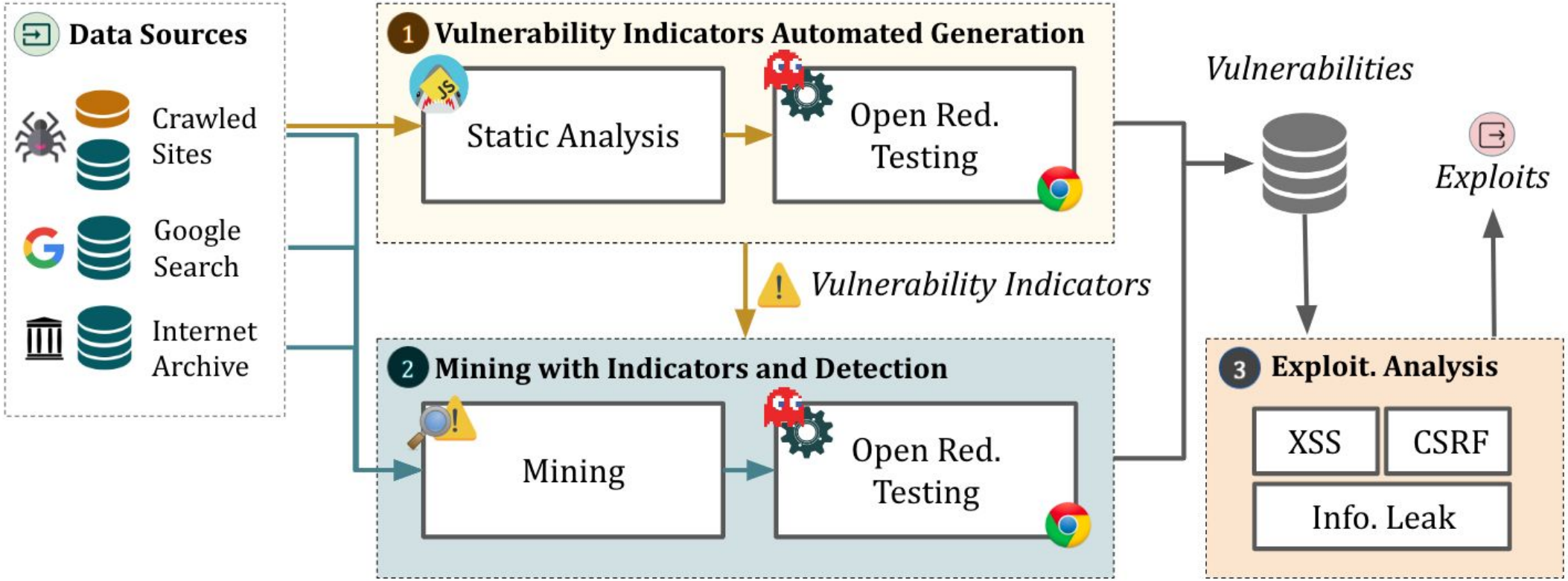
Methodology: STORK Framework



Methodology: STORK Framework



Methodology: STORK Framework



RQ1-Vulnerability Indicators: Dataset and Approach

- Collected snapshots of webpages using Playwright and an Foxhound

Tranco top **10K sites**, over **1M pages**, 36M scripts, and 104B LoC

Oct. 2022

RQ1-Vulnerability Indicators: Dataset and Approach

- Collected snapshots of webpages using Playwright and an Foxhound

Tranco top **10K sites**, over **1M pages**, 36M scripts, and 104B LoC

Oct. 2022

- Split dataset into two portions: **indicator extraction (P1)** and test set (P2)

RQ1-Vulnerability Indicators: Dataset and Approach

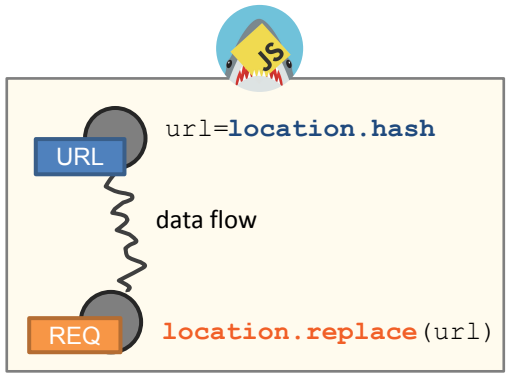
- Collected snapshots of webpages using Playwright and an Foxhound

Tranco top **10K sites**, over **1M pages**, 36M scripts, and 104B LoC

Oct. 2022

- Split dataset into two portions: **indicator extraction (P1)** and test set (P2)

- **Indicator extraction**
  - Use JAW to conduct **static data flow analysis** to detect client-side open redirects
  - Automatically **confirm** the open redirection at **runtime**
  - Extract patterns by **grouping** vulnerable URLs by similarity
  - Manual review of **CVE database** to capture past patterns of server-side variants



RQ1-Vulnerability Indicators: Results

- **Detection**

  Static analysis: 25.9K dataflows to redirection sinks

  Dynamically confirmed: **20.4K** URL-sourced cases across **599 sites**

RQ1-Vulnerability Indicators: Results

- **Detection**

  Static analysis: 25.9K dataflows to redirection sinks

  Dynamically confirmed: **20.4K** URL-sourced cases across **599 sites**

- **Indicators**
  A catalogue of **184** concrete indicators, organized in **nine** abstract groups

RQ1-Vulnerability Indicators: Results

- **Detection**

  Static analysis: 25.9K dataflows to redirection sinks

  Dynamically confirmed: **20.4K** URL-sourced cases across **599 sites**

- **Indicators**
  A catalogue of **184** concrete indicators, organized in **nine** abstract groups

See paper for more

Type		ID	Pattern	Params	Count	New	Example	CVEs	Vulns	Sites
Query	⊞	A1	?**P**=url	R1	109	59	?next=example.com	382	14,201	402
		A2	?CONST=https%3A%2F%2F | www. | DOMAIN.PSL	-	3	0	?xyz=https%3A%2F%2Fexample.com	12	2,360	91
Path	⊞	B1	/**P**/https%3A | DOMAIN.PSL	R2	17	1	/callbackUri/www.example.com%2Findex	35	948	147
		B2	[/CONST]/https%3A/**P**	R3	13	0	/example.com%2Fprofile/submitUrl	23	260	24
		B3	/CONST/https%3A | DOMAIN.PSL	-	2	0	/index.php/example.com%2Findex	2	122	7
		B4	/https%3A/CONST/	-	1	0	/https%3A%2F%2Fexamle.com%2Findex/get	6	31	3
Hash	⊞	C1	#**P**=CONST	R4	35	35	#ajaxUI=example.com/profile/index	0	2,207	108
	⊞	C2	#CONST=https:// | DOMAIN.PSL	-	2	2	#u=https://example.com	0	311	26
	⊞	C3	#https:// | DOMAIN.PSL	-	2	2	#example.com/profile/index	0	31	2
Total					**184**	**95**		460	**20,471**	**599**

RQ2-Vulnerability Mining: Cost-Benefit Analysis (1 / 2)

- Use our indicator's catalogue to search for vulnerabilities
  - **Baseline comparison** with static program analysis

Evaluation Dataset

42K webpages of 50 random test apps

RQ2-Vulnerability Mining: Cost-Benefit Analysis (1 / 2)

- Use our indicator's catalogue to search for vulnerabilities
  - **Baseline comparison** with static program analysis

Evaluation Dataset

42K webpages of 50 random test apps

- **Performance**
  - Program analysis: 58 alerts, **46 confirmed in eight apps** (20% FP)
  - Indicators: narrowed scope immediately to 3K URLs, **16 true vulnerabilities in six apps**

RQ2-Vulnerability Mining: Cost-Benefit Analysis (1 / 2)

- Use our indicator's catalogue to search for vulnerabilities
  - **Baseline comparison** with static program analysis

Evaluation Dataset

42K webpages of 50 random test apps

- **Performance**
  - Program analysis: 58 alerts, **46 confirmed in eight apps** (20% FP)
  - Indicators: narrowed scope immediately to 3K URLs, **16 true vulnerabilities in six apps**

✔ Indicators can **detect vulnerabilities SAST might miss** (five out of 16)

Reason: SAST limitations (missing call/PDG edges) and server-side open redirects

RQ2-Vulnerability Mining: Cost-Benefit Analysis (1 / 2)

- Use our indicator's catalogue to search for vulnerabilities
  - **Baseline comparison** with static program analysis

Evaluation Dataset

42K webpages of 50 random test apps

- **Performance**
  - Program analysis: 58 alerts, **46 confirmed in eight apps** (20% FP)
  - Indicators: narrowed scope immediately to 3K URLs, **16 true vulnerabilities in six apps**

Indicators can **detect vulnerabilities SAST might miss** (five out of 16)

Reason: SAST limitations (missing call/PDG edges) and server-side open redirects

Indicators can **cast a wider net** and **pinpoint apps** for in-depth testing

Reason: **half of the apps** found vulnerable by static analysis were also flagged by indicators

RQ2-Vulnerability Mining: Cost-Benefit Analysis (1 / 2)

- Use our indicator's catalogue to search for vulnerabilities
  - **Baseline comparison** with static program analysis

Evaluation Dataset

42K webpages of 50 random test apps

- **Performance**
  - Program analysis: 58 alerts, **46 confirmed in eight apps** (20% FP)
  - Indicators: narrowed scope immediately to 3K URLs, **16 true vulnerabilities in six apps**

✓ Indicators can **detect vulnerabilities SAST might miss** (five out of 16)

Reason: SAST limitations (missing call/PDG edges) and server-side open redirects

✓ Indicators can **cast a wider net** and **pinpoint apps** for in-depth testing

Reason: **half of the apps** found vulnerable by static analysis were also flagged by indicators

✗ Indicators may lead to **large FNs** (76%)

Reason: indicators operate at URL level and their optional params may be missing

RQ2-Vulnerability Mining: Cost-Benefit Analysis (1 / 2)

- Use our indicator's catalogue to search for vulnerabilities
  - **Baseline comparison** with static program analysis

Evaluation Dataset

42K webpages of 50 random test apps

- **Runtime**

✔ indicators **~100x faster**

Program analysis: 35 min/page vs. indicators: 21 sec/page

RQ2-Vulnerability Mining: Cost-Benefit Analysis (1 / 2)

- Use our indicator's catalogue to search for vulnerabilities
  - **Baseline comparison** with static program analysis

Evaluation Dataset

42K webpages of 50 random test apps

- **Runtime**

indicators **~100x faster**

Program analysis: 35 min/page vs. indicators: 21 sec/page

indicators **~590x less storage**

Program analysis: 14.8T vs. indicators: 25G (entire test set)

RQ2-Vulnerability Mining: In-the-Wild Prevalence

- Use our indicator's catalogue to search for vulnerabilities
  - Collected **4M** candidate URLs, **214K unique** after de-duplication

Snapshots of live webpages

Google search via dorking

Internet archive

RQ2-Vulnerability Mining: In-the-Wild Prevalence

- Use our indicator's catalogue to search for vulnerabilities
  - Collected **4M** candidate URLs, **214K unique** after de-duplication

- Vet the candidates via dynamic tests

Discovered **375** open redirect vulnerabilities across **326 sites.**
202 client-side, **171 server-side.**

Snapshots of live webpages

Google search via dorking

Internet archive

RQ2-Vulnerability Mining: In-the-Wild Prevalence

- Use our indicator's catalogue to search for vulnerabilities
  - Collected **4M** candidate URLs, **214K unique** after de-duplication

- Vet the candidates via dynamic tests

Discovered **375** open redirect vulnerabilities across **326 sites.**
202 client-side, **171 server-side.**

Snapshots of live webpages

Google search via dorking

Internet archive

See paper for more

		Candidate		Vuln.	
Source	**Pattern**	**URLs**	**Sites**	**URLs**	**Sites**
Internet Archive	A1	162,562	6,108	205	171
	A2	15,675	1,270	44	37
	B1	8,445	965	12	8
	B2	1,502	417	3	1
	B3	198	44	1	1
	B4	21	5	0	0
	Total	188,403	8,045	**265**	**218**
Google Search	A1	661	371	12	11
	A2	380	123	7	7
	B1	121	56	2	2
	B2	49	12	0	0
	B3	17	5	0	0
	B4	9	2	1	1
				22	**21**

RQ3-Exploitability Analysis

Candidates

Total of **21.2K** open redirects across **872** unique sites (SAST + mining)

RQ3-Exploitability Analysis

Candidates

Total of **21.2K** open redirects across **872** unique sites (SAST + mining)

Methodology

DOM XSS: Automatic	Tested **all** candidates dynamically with a XSS payload dictionary
Req. Forgery & Info. Leaks: Manual	Reviewed **two** open redirects randomly per site (**1,744** cases)

RQ3-Exploitability Analysis

Candidates

Total of **21.2K** open redirects across **872** unique sites (SAST + mining)

Methodology

DOM XSS: Automatic	Tested **all** candidates dynamically with a XSS payload dictionary
Req. Forgery & Info. Leaks: Manual	Reviewed **two** open redirects randomly per site (**1,744** cases)

Results

Discovered **1.9K escalations** across **332 sites**

Examples: Adobe, WebNovel, TP-Link, UDN, and VK

RQ3-Exploitability Analysis

Candidates

Total of **21.2K** open redirects across **872** unique sites (SAST + mining)

Methodology

DOM XSS: Automatic — Tested **all** candidat amically with a XSS payload dictionary

Req. Forgery & Info. Leaks: Manual — Reviewed **two** open redirects randomly per site (**1,744** cases)

Takeaway

Static analysis detects more open redirects, but ...

Indicator-based findings have a **higher rate of XSS escalations** (22% vs. 8%)

Results

Discovered **1.9K escalations** across **332 sites**

Examples: Adobe, WebNovel, TP-Link, UDN, and VK

See paper for more!

Threat	SAST Vuln.	SAST Sites	Mining Vuln.	Mining Sites	Total Vuln.	Total Sites
DOM-based XSS	1,845	212	84	78	1,929	290
Client-side CSRF	36	33	6	6	42	39
Information Leak	2	2	1	1	3	3
Total	1,883	**247**	91	**85**	1,974	**332**

Re-evaluating the Risk: Open Redirects

Prevalence

Widespread, affecting **8.7%** of top 10K sites, with a total of **21.2K** instances

Re-evaluating the Risk: Open Redirects

Prevalence

Widespread, affecting **8.7%** of top 10K sites, with a total of **21.2K** instances

Severity

Alarming, **38%** of sites with open redirect (**3.3%** of top 10K) can be leveraged for critical attacks

- **DOM-based XSS:** almost **one out of ten** open redirects
- **Request forgery and info leaks:** almost **three out of hundred** open redirects

Summary

- Proposed a **cost-reduction method** to detect open redirects by **extracting** and **using indicators**
- Created a **catalogue of 184 vulnerability indicators**
- Re-evaluated the risk of open redirections at scale
  - Prevalence: **8.7%** of sites
  - Severe: **3.3%** of sites
- Indicators could serve as **a lightweight trade-off** compared to costly static analysis
  - Higher rate of XSS escalations
  - Less analysis time and storage requirements

Soheil__K https://github.com/SoheilKhodayari/STORK