30TH USENIX SECURITY SYMPOSIUM

Abusing Hidden Properties to Attack Node.js Ecosystem

Feng Xiao, Jianwei Huang, Yichang Xiong, Guangliang Yang,

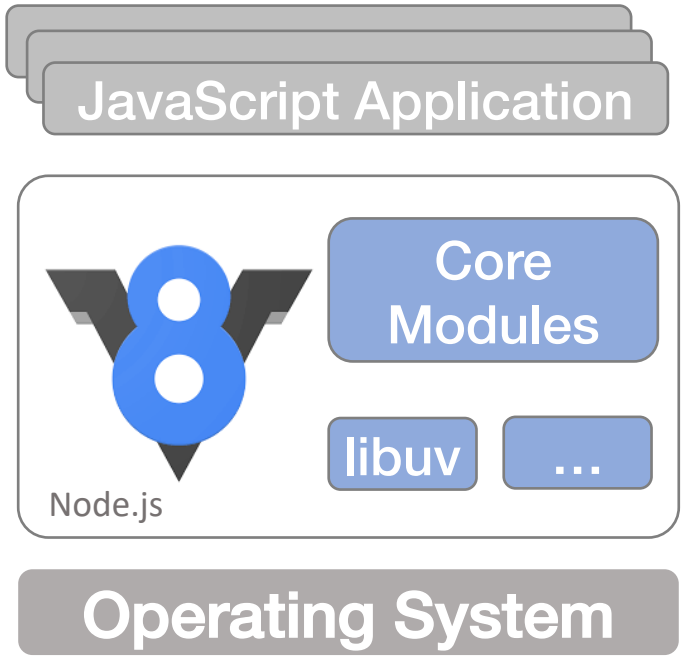
Hong Hu, Guofei Gu, Wenke Lee

Agenda

- A novel feasible attack against Node.js programs.
- A one-of-a-kind vulnerability finding tool.
- A comprehensive evaluation with real-world findings.

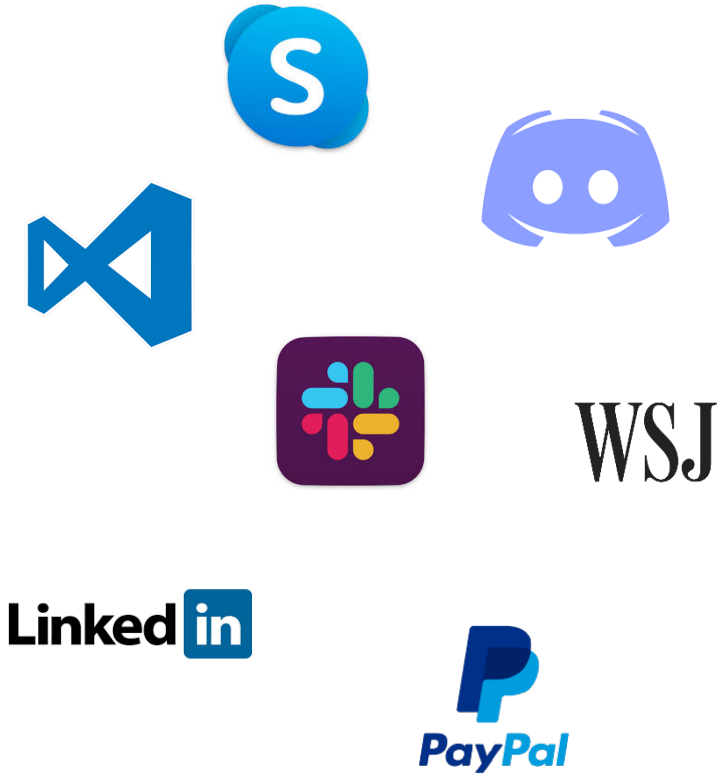
Background

- Node.js is for executing JavaScript outside of browsers.



Background

- Node.js is for executing JavaScript outside of browsers.
- Node.js is widely-used for deploying server-side programs and desktop apps.



Background

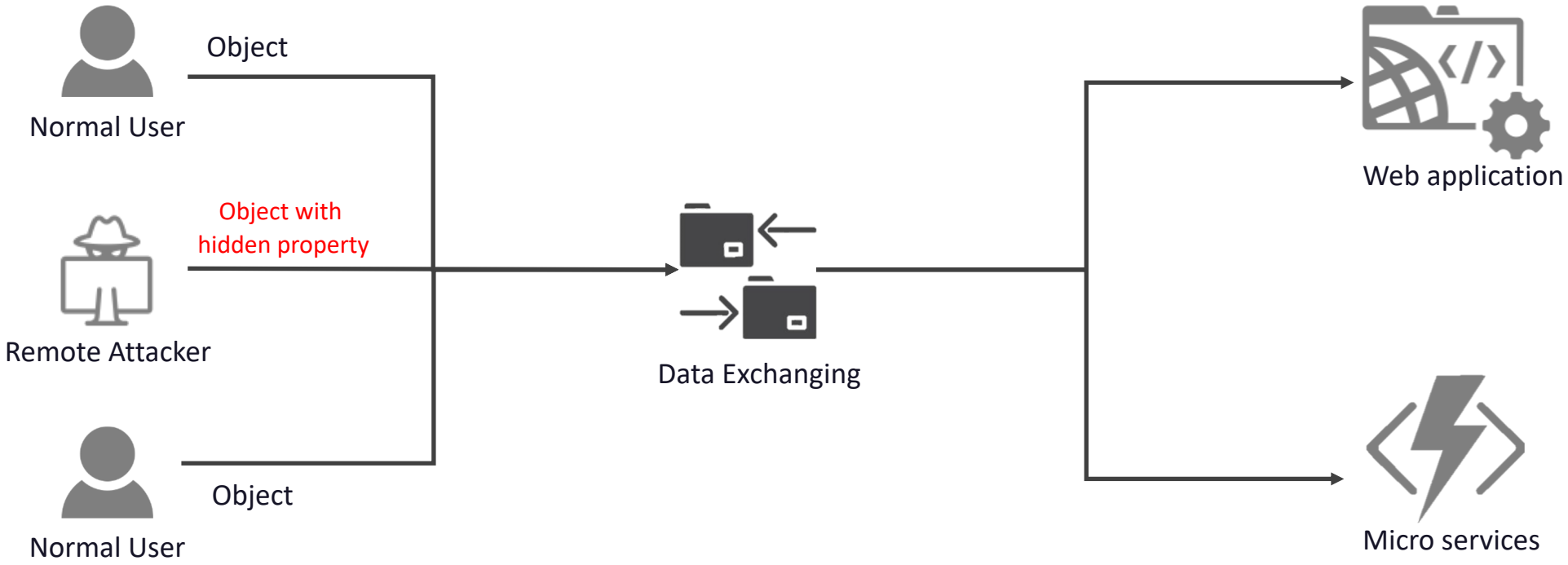
- Node.js is for executing JavaScript outside of browsers.
- Node.js is widely-used for deploying server-side programs and desktop apps.
- Object sharing is a very popular communication method for Node.js programs.

Module	Monthly Downloads
qs	122,309,219
body-parser	46,230,008
querystring	34,758,659
query-string	34,192,119
socket.io	12,328,997

Request parsing modules that convert input into objects.

Hidden Property Abusing

- HPA leverages the widely-used data exchanging feature in Node.js (object sharing) to tamper or forge critical program states of Node.js applications.

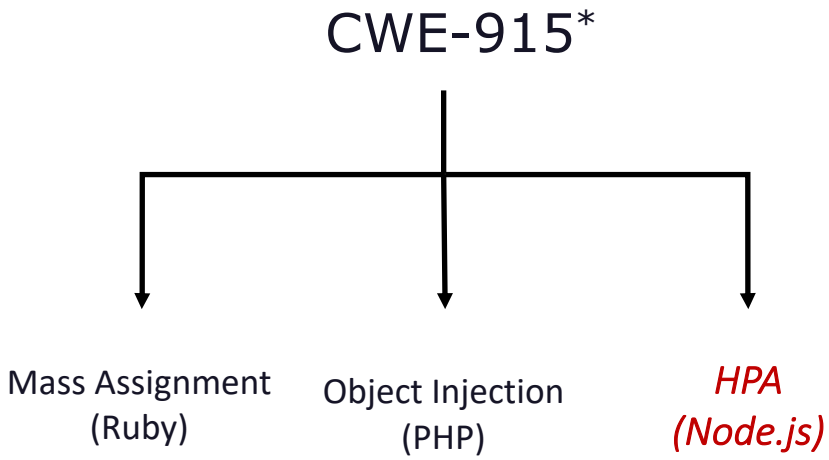


Comparing HPA with related attacks

- HPA identifies CWE-915 risk in Node.js

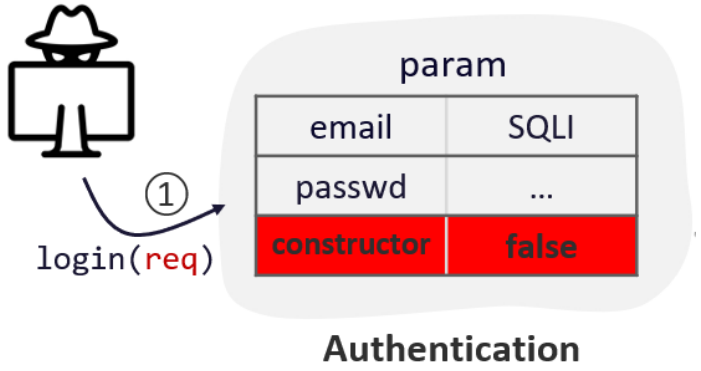
- Comparation with Mass Assignment

Aspect	Hidden Property Abusing	Ruby Mass Assignment
Abused logics	Object sharing	Assignment
Payload Type	Literal value/nested object	Literal value
Capabilities	Overwrite	Overwrite/Create



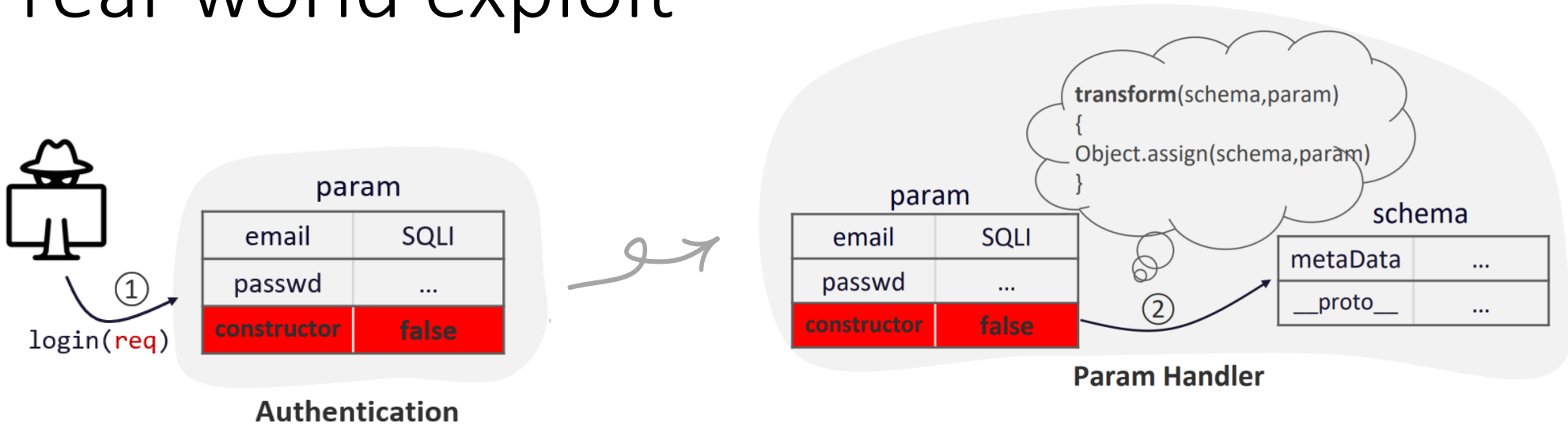
*CWE-915: Improperly Controlled Modification of Dynamically-Determined Object Attributes

A real-world exploit

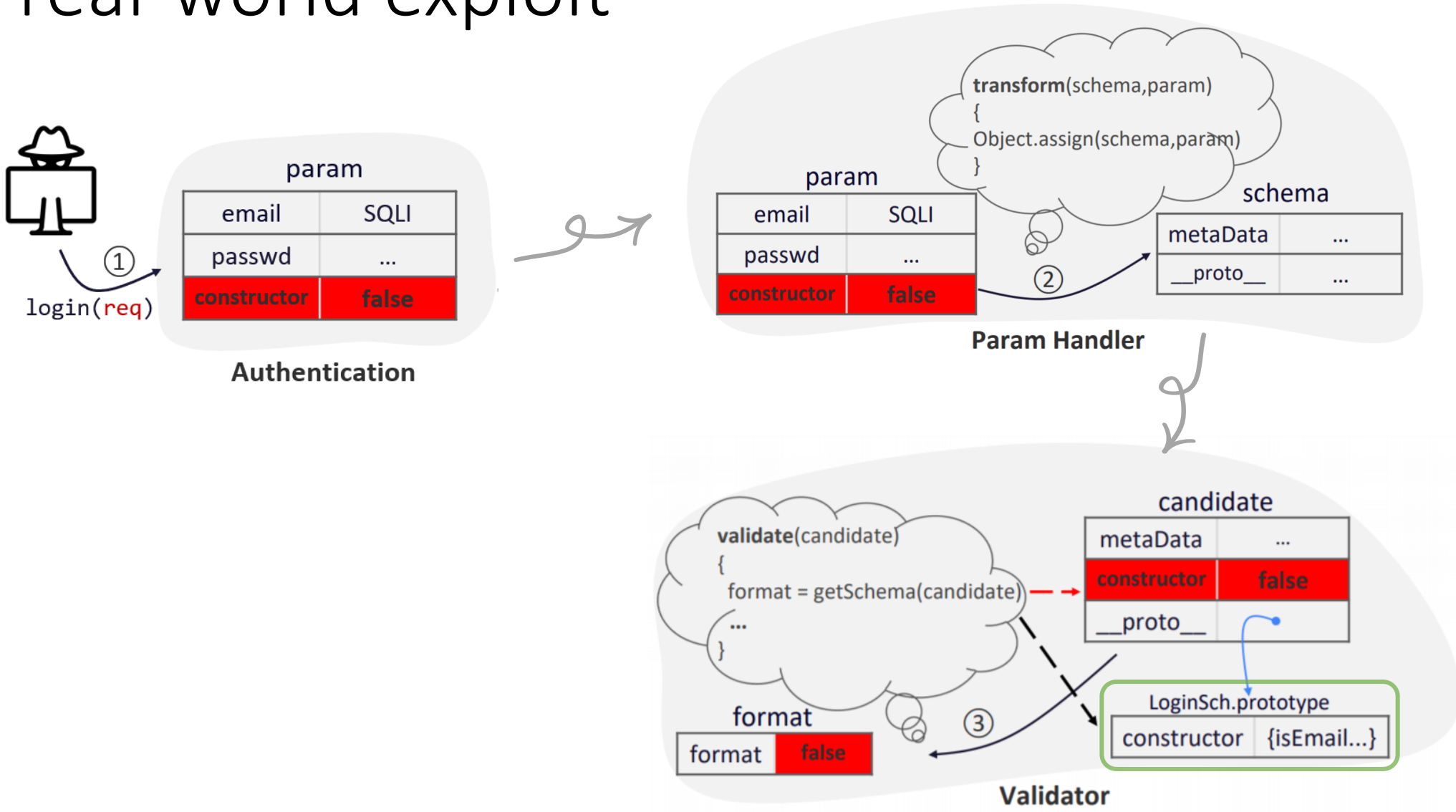


We’ve made responsible disclosure for all the vulneraries found in this research.

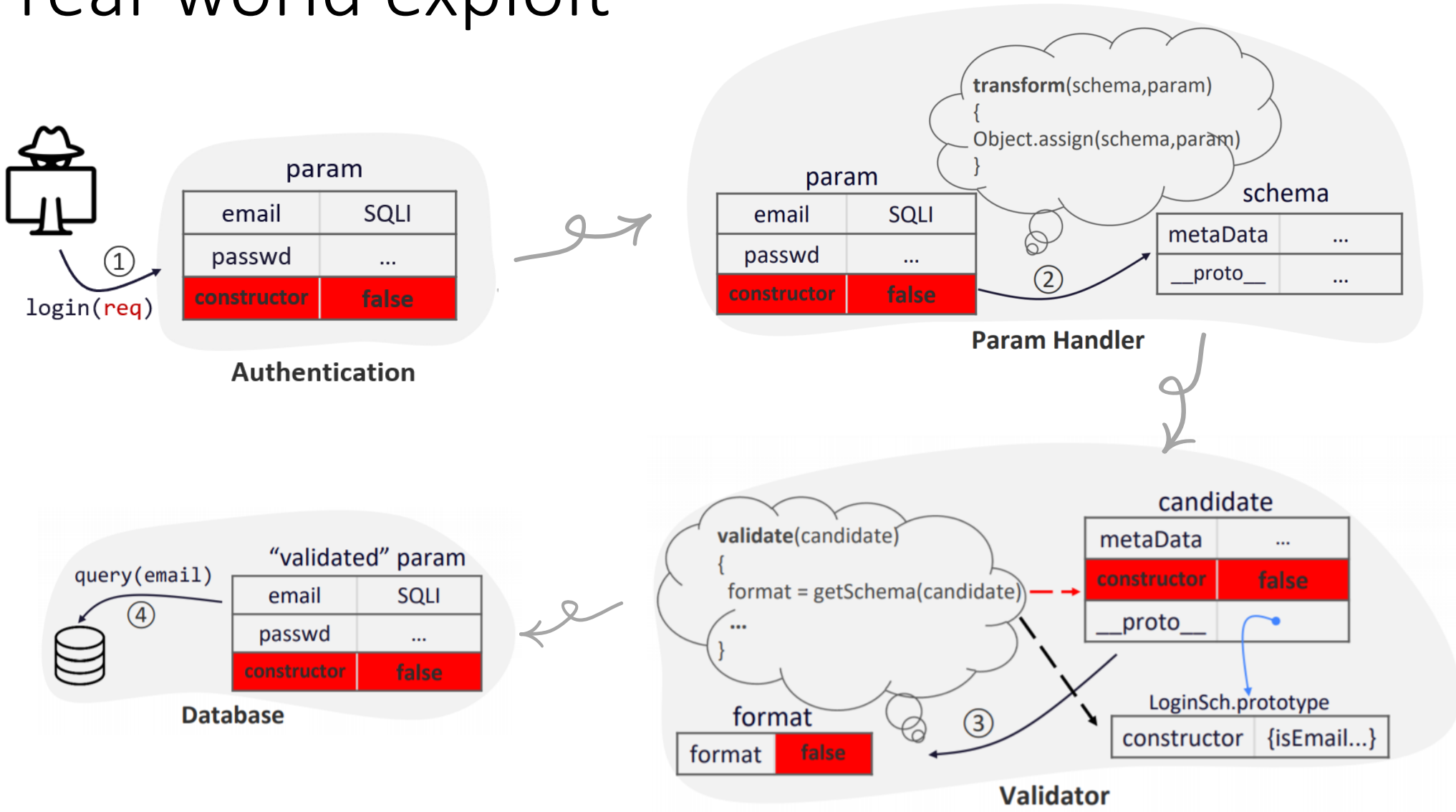
A real-world exploit



A real-world exploit

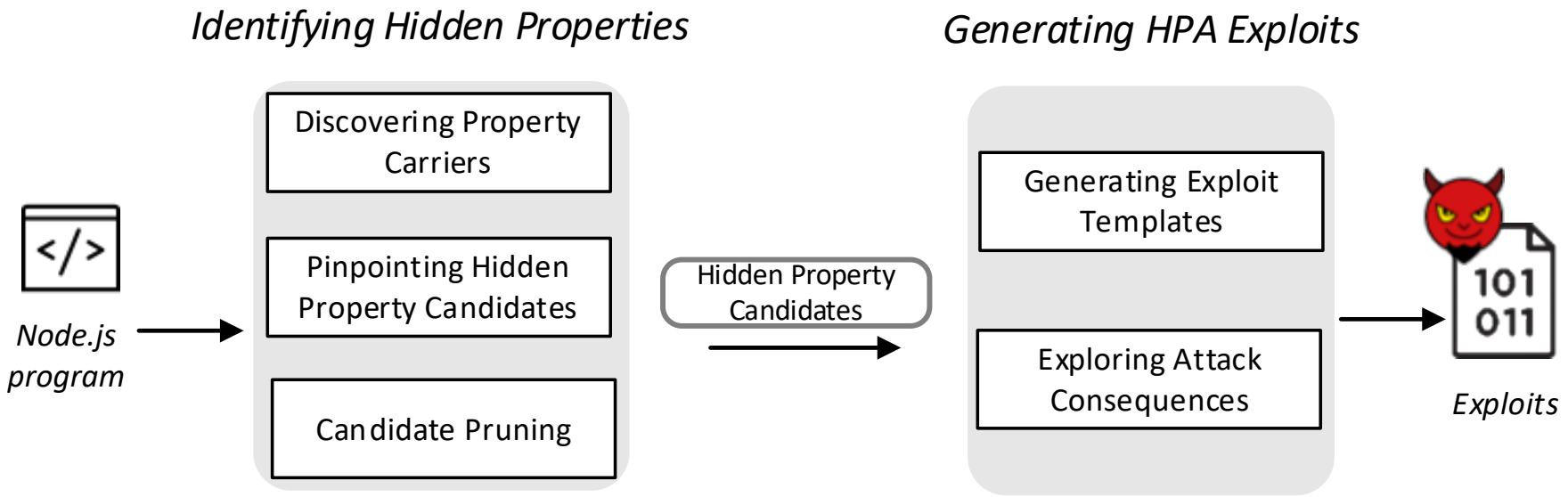


A real-world exploit



Lynx

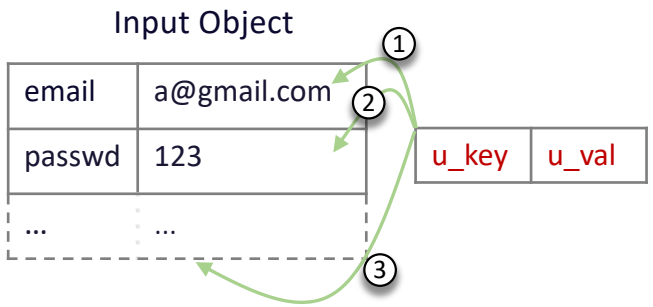
- We design and implement Lynx*, a hybrid JavaScript program analysis tool, to detect and exploit HPA vulnerabilities.



*The lynx is a type of wildcat. In Greek myths, it is believed that lynxes can see what others can't, and its role is revealing hidden truths.

https://github.com/xiaofen9/Lynx

Identifying Hidden Properties



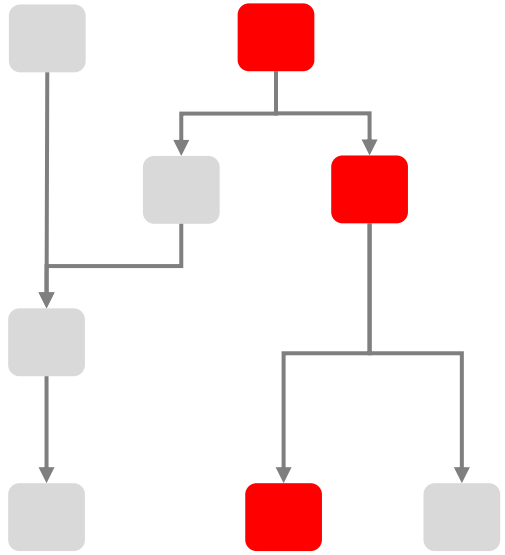
Label Injection

```
function transform(schema,
param){
    value = Object.assign(
        schema,
        param);
    data = param['data'];
    return value;
}
```

variable r/w
property indexing
function calls

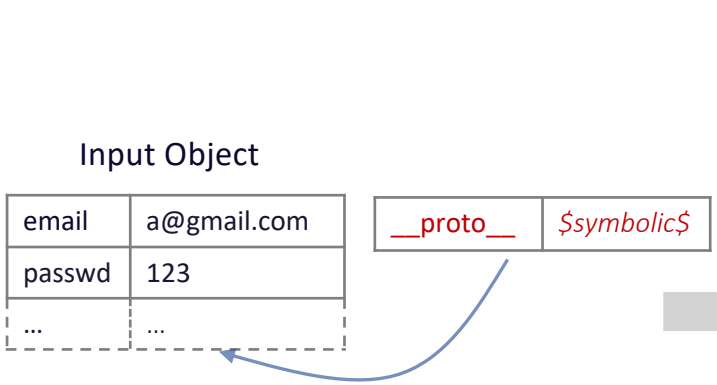
Jalangi

Test Program Instrumentation

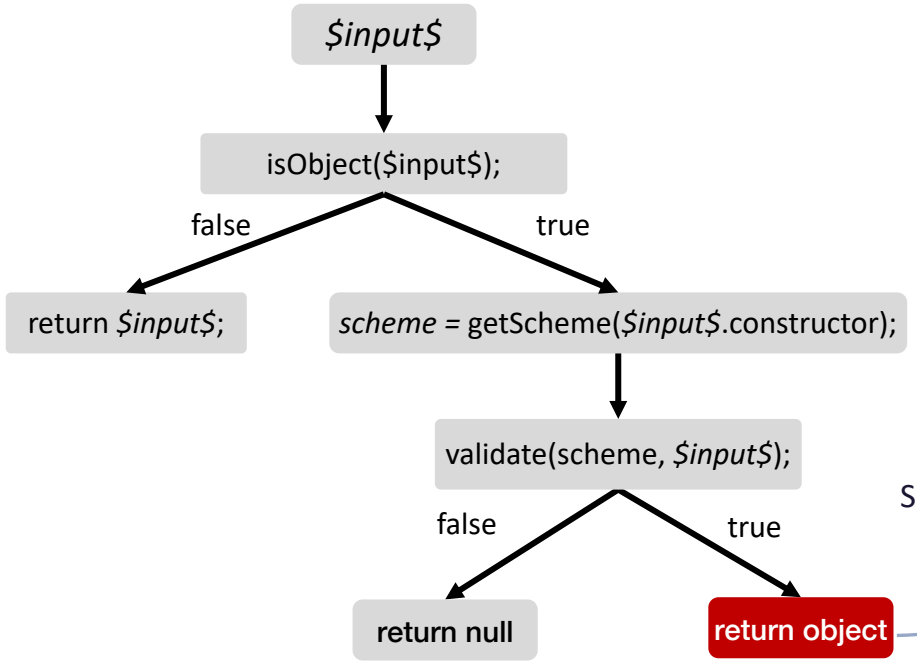


Property Carrier Tracking

Exploring Security Consequences



Generating Exploit Templates



Sink I_2 hit!

Category	ID	Sink
Confidentiality	C_1	sensitive database query methods
	C_2	sensitive file system operation methods
Integrity	I_1	Critical built-in properties and code execution APIs
	I_2	Final results of the module invocation
Availability	A_1	Global methods/variables
	A_2	Looping conditions

Dynamic Symbolic Execution

Evaluation

- 102 Node.js programs
  - 91 Node modules
  - 11 web-based programs (4 frameworks and 7 complete web apps)

- 15 zero-day vulnerabilities
  - **451** hidden property candidates from **3175** carriers
    (69% tested programs contain hidden properties)
  - **10** exploits successfully synthesized by Lynx

Impact Analysis

- Confidentiality
  - Universal SQL Injection (**1**)
  - Leaking confidential Data (**3**)
- Integrity
  - Input Validation Bypass (**6**)
  - Forging critical data structure (**4**)
- Availability
  - Event Handler Blocking (**1**)

#ID	Product Name	Affected API	Description	Impact		Attack Effects			Disclosure	
				Downloads	Dependents	C	I	A	status	severity
1	mongoose	findOne()	SQL Injection	2,740,341	9,211	✔			Fixed (CVE1)	Critical
2	mongoDB driver	find()	SQL Injection	6,165,075	8,435	✔			Fixed (CvE2)	-
3	taffyDB	query APIs	SQL Injection	1,628,860	108	✔			Confirmed (CVE3)	High
4	class-validator	validate()	Bypass input validation	1,077,954	1,639		✔		Confirmed (CVE4)	Critical
5	jpv	validate()	Bypass input validation	481	1		✔		Fixed (CVE5)	Medium
6	jpv	validate()	Bypass input validation	481	1		✔		Reported	Medium
7	valib	hasValue()	Bypass input validation	479	8		✔		Reported	-
8	schema-inspector	validate()	Bypass input validation	35,783	104		✔		Fixed (CVE6)	High
9	schema-inspector	sanitize()	Bypass input validation	35,783	104		✔		Fixed(CVE6)	High
10	bson-objectid	ObjectID()	ID forging	142,562	298		✔		Fixed (CVE7)	High
11	component-type	type()	Type manipulation	943,555	140		✔		Reported	-
12	component-type	type()	Type manipulation	943,555	140		✔		Reported	-
13	kind-of	kindOf()	Type manipulation	196,448,574	458		✔		Fixed (CVE8)	High
14	cezerin	getValidDocumentForUpdate()	Order state manipulation	1871	–	✔			Confirmed (CVE9)	High
15	mongo-express	addDocument()	Denial of service	6,965	–			✔	Fixed(CVE10)	Medium

Impact Analysis

- Confidentiality
  - Universal SQL Injection (**1**)
  - Leaking confidential Data (**3**)
- Integrity
  - Input Validation Bypass (**6**)
  - Forging critical data structure (**4**)
- Availability
  - Event Handler Blocking (**1**)

HPA effectively enlarges Node.js attack surface by compromising previously unreachable program states.

Classic defenses cannot mitigate HPA. Some widely-used input validation modules are vulnerable to HPA.

#ID	Product Name	Affected API	Description	Impact		Attack Effects			Disclosure	
				Downloads	Dependents	C	I	A	status	severity
1	mongoose	findOne()	SQL Injection	2,740,341	9,211	✔			Fixed (CVE1)	Critical
2	mongoDB driver	find()	SQL Injection	6,165,075	8,435	✔			Fixed (CvE2)	-
3	tarn.js	query APIs	SQL Injection	1,628,860	108	✔			Confirmed (CVE3)	High
4	class-validator	validate()	Bypass input validation	1,077,954	1,639		✔		Confirmed (CVE4)	Critical
5	jpv	validate()	Bypass input validation	481	1		✔		Fixed (CVE5)	Medium
6	jpv	validate()	Bypass input validation	481	1		✔		Reported	Medium
7	valib	hasValue()	Bypass input validation	479	8		✔		Reported	-
8	schema-inspector	validate()	Bypass input validation	35,783	104		✔		Fixed (CVE6)	High
9	schema-inspector	sanitize()	Bypass input validation	35,783	104		✔		Fixed(CVE6)	High
10	bson-objectid	ObjectID()	ID forging	142,562	298		✔		Fixed (CVE7)	High
11	component-type	type()	Type manipulation	943,555	140		✔		Reported	-
12	component-type	type()	Type manipulation	943,555	140		✔		Reported	-
13	kind-of	kindOf()	Type manipulation	196,448,574	458		✔		Fixed (CVE8)	High
14	cezerin	getValidDocumentForUpdate()	Order state manipulation	1871	–	✔			Confirmed (CVE9)	High
15	mongo-express	addDocument()	Denial of service	6,965	–			✔	Fixed(CVE10)	Medium

Conclusion

- We design a practical attack named hidden property abusing (HPA), which leads to the discovery of 15 zero-day Node.js vulnerabilities.
- We build a novel tool to pinpoint and exploit vulnerable internal objects in Node.js programs.
- The evaluation results demonstrate that HPA enlarges the attack surface of Node.js ecosystem by effectively attacking previously unreachable program states.

Thanks for attending our talk!

Code can be found @ https://github.com/xiaofen9/Lynx