Attacking v8 zero to exploit

Workshop @ VXCON 2024

Alisa Esage
Zero Day Engineering Research & Training
zerodayengineering.com

Full process

outline

1. Research platform set up
2. Patch analysis
3. PoC tests ~ crash triage
4. Exploit development

Special emphasis

1. Thought process of exploiting a highly non trivial vulnerability - novel technique required
2. Theoretical analysis & modeling
3. AI assist

Full process

outline

1. Research platform set up
2. Patch analysis
3. PoC tests ~ crash triage
4. Exploit development

Browser + JSE attack surface (RCE/renderer)

Browser

- DOM (**D**ocument **O**bject **M**odel)
- HTML parsing
- Network protocols
  - HTTP/2, JSON, …
- File formats
  - Graphics
  - Audio
  - PDF
  - XML …
- JavaScript engine
- Sandbox (EoP)
- OS bindings

JavaScript engine

- Parser
- Analyzers
- Interpreter
- Lowering (non-optimizing compilation)
- Optimization
- Garbage collection
- Builtins/globals
- DOM interfaces
- OS interfaces
- Backend + API (Intl, WebGL, etc.)
- **WebAssembly**

Full process

Step by step

1. Research platform set up
    a. Building v8
    b. Release vs. debug
    c. Debugging
    d. Advanced config
2. Patch analysis
3. PoC tests ~ crash triage
4. Exploit development

Full process

Step by step

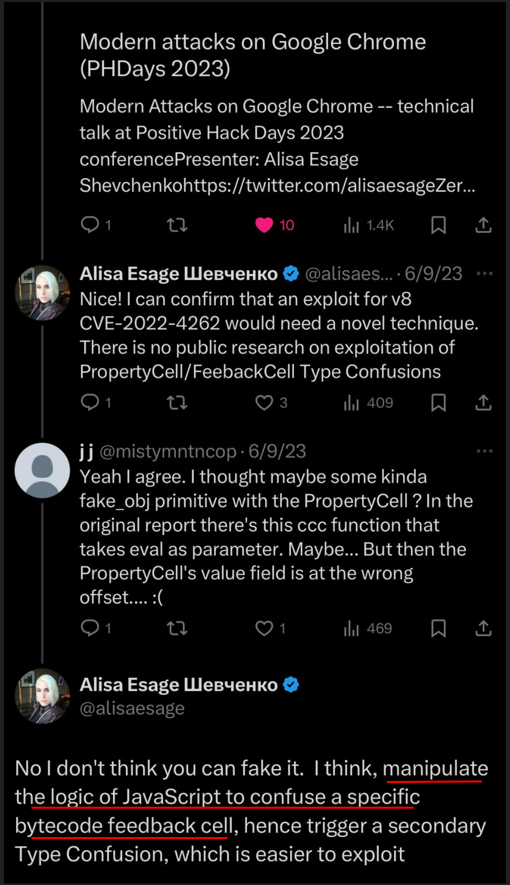
1. Research platform set up
2. Patch analysis
   1. Advisory to commit
   2. Diff analysis
   3. Low level vs. High level context of vuln
   4. Research to mental models
   5. Indicator of bug
   6. Hypothesizing
   7. Reproducer prerequisites
3. PoC tests ~ crash triage
4. Exploit development

Full process

Step by step

1. Research platform set up
2. Patch analysis
3. PoC tests ~ crash triage
   a. Basic theoretical analysis
   b. Minimize the testcase
   c. Diversify the crash site
   d. Analysis with built-in introspection tools
   e. Analysis in debugger with breakpoints
   f. Analysis with extra debugging logic
   g. Ideas how to manipulate the state
4. Exploit development

Exploitation



Modern attacks on Google Chrome (PHDays 2023)

Modern Attacks on Google Chrome -- technical talk at Positive Hack Days 2023 conferencePresenter: Alisa Esage Shevchenkohttps://twitter.com/alisaesageZer...

Alisa Esage Шевченко @alisaes... · 6/9/23

Nice! I can confirm that an exploit for v8 CVE-2022-4262 would need a novel technique. There is no public research on exploitation of PropertyCell/FeebackCell Type Confusions

j j @mistymntncop · 6/9/23

Yeah I agree. I thought maybe some kinda fake_obj primitive with the PropertyCell ? In the original report there's this ccc function that takes eval as parameter. Maybe... But then the PropertyCell's value field is at the wrong offset.... :(

Alisa Esage Шевченко
@alisaesage

No I don't think you can fake it. I think, manipulate the logic of JavaScript to confuse a specific bytecode feedback cell, hence trigger a secondary Type Confusion, which is easier to exploit

...to be continued

Full process

Step by step

1. Research platform set up
2. Patch analysis
3. PoC tests ~ crash triage
4. Exploit development
   a. Working technique to manipulate the state
   b. Convert state corruption to max power primitive (ARW)
   c. Build next level primitives (addrOf, fakeObj)
   d. Arbitrary code execution