Detecting Browser-Based Probing Attacks via Behavior Analysis

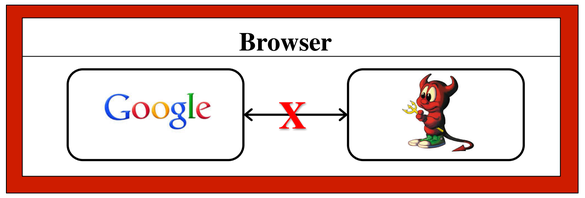
Yue Chen,† Yaoqi Jia,§ Jian Mao,† Zhenkai Liang§

† School of Electronic and Information Engineering, Beihang University

§ Department of Computer Science, National University of Singapore

Protection Mechanisms in Browsers

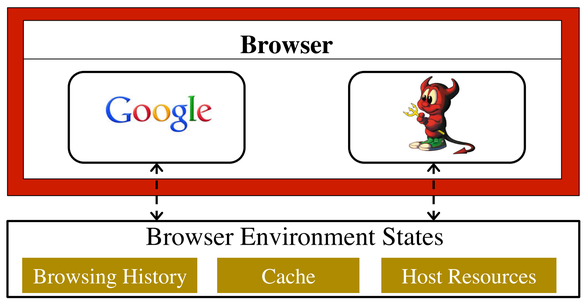
- Same Origin Policy (SOP).
  - Origin is defined by *(protocol, host, port)*
  - SOP prevents one origin from accessing resources in other origins.
- Sandbox confines accesses to browser resources



Preventing *direct* access made by malicious websites.

Indirect Probing of Sensitive Information

- Different origins share the same browser environment
- Sensitive information can be inferred from indirect probing



- History sniffing
- Cache sniffing
- Internal network probing

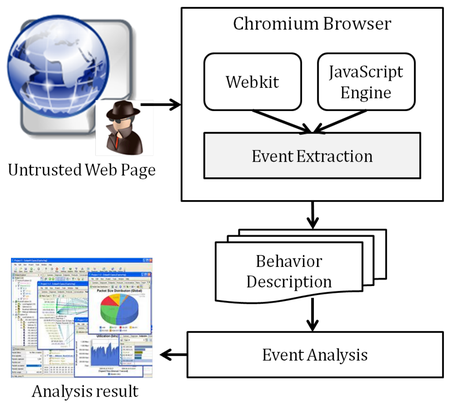
Our Observation

- The “data rate” obtained through probing is very low. Attackers thus need a large amount of repeated operations to extract useful information.
  - History sniffing:
    Repeated enumerating links and checking link color
  - Cache sniffing:
    Repeated accessing web resources
  - Internal network probing:
    Repeated requesting resources from local network

Approach Overview

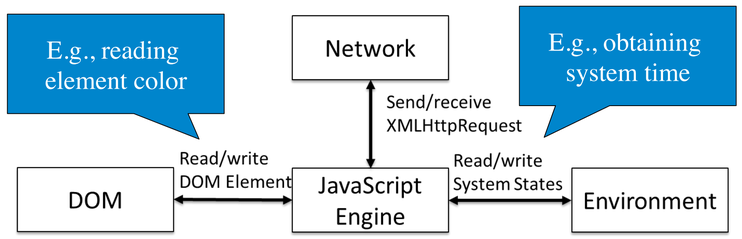
Monitoring untrusted web sites in an instrumented Chromium browser.

- Extracting browser behaviors
  - Security relevant events
  - Descriptive information
- Analysing behavior descriptions
  - Statistics from multiple dimensions
  - Identifying abnormal and unreasonable behaviors



Browser Events

- JavaScript: The driving force in browsers
- Focusing on JavaScript interactions with the rest of browser components
  - DOM, Network, Environment states



Analyzing Behaviors

- Simple statistical analysis on behavior descriptions
  - Number of the repetitive API calls
  - Identifying features distinguishing normal websites from probing ones
  - Challenge: difficulty in benign sites involving large amounts repetitive behaviors
- Multiple dimension analysis on behavior descriptions
  - Analyzing behavior descriptions in different dimensions, e.g., time, involved element, API properties, etc.
  - Establishing heat map representation on different dimensions to detect probing behaviors

Simple Statistics of Browser Events

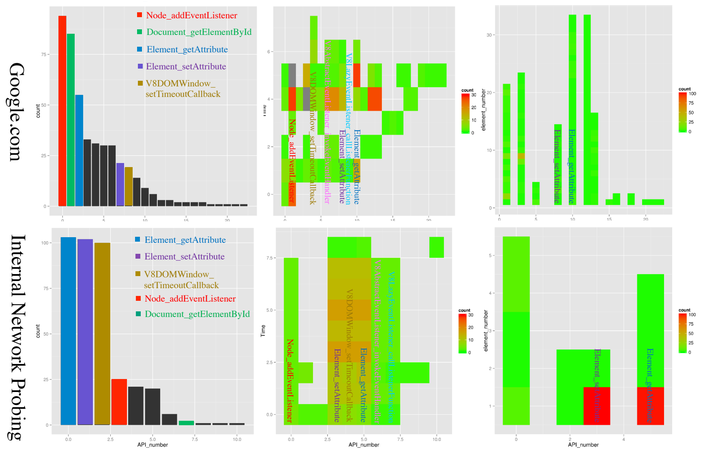
Google.com

API	count
Node_addEventListener	94
Document_getElementById	85
Element_getAttribute	55
HTMLDocument_createElement	33
V8AbstractEventListener_invokeEventHandler	31
V8EventListener_getListenerFunction	30
Node_appendChild	30
Element_setAttribute	21
...	...

Internal Network Probing

API	count
Element_getAttribute	103
Element_setAttribute	102
V8DOMWindow_setTimeoutCallback	100
Node_addEventListener	25
V8AbstractEventListener_invokeEventHandler	21
V8LazyEventListener_callListenerFunction	20
ScriptController_evaluate	6
Document_getElementById	2
...	...

Heat Map View of Results



API-count API-time-count API-element-count

	Google.com	Internal Network Probing
API-count	Some APIs are called much more frequently than the others in both test scenarios	
API-time-count	APIs are called discretely over time	APIs are called continuously over time
API-element-count	APIs spread out on many elements	APIs concentrate on a few elements

Conclusion

- Indirect probing extracts sensitive information in browser environment, with a low “data rate.”
- Detecting browser-based probing behaviors via multiple dimension analysis of browser events.

This research is supported in part by NSFC (No. 61402029, No. 61370190), by BJNSF (No. 4132056), by China NKBRP (No. 2012CB315905), by Singapore MOE (NUS WBS: R-252-000-519-112).