HYDRA

Optimizing Context-Driven Black-Box Cross-Site Scripting Exploitation.
- Capable of identifying Reflected XSS.
- Develops exploit vectors based on target behavior and filters.

Execution Contexts

HTML Output Context, defined by
- Type
  - Tag name
  - Text
  - Attribute name
  - Attribute value
- ...
- Tag: A, DIV, SCRIPT, ...
- Attributes: ID, href, onclick, ...

Some contexts are more desirable:
- Text nodes under SCRIPT tags
- onload/onerror attributes
- Use-case specific, e.g., IMG src

Approach

Input
- Target endpoint.
- Execution context weights.
- Transition patterns.

Process
1. Perform initial injection
2. Identify execution context
3. Find reachable contexts with higher weight
4. Construct concrete edges from patterns and encodings
5. Extend injection with concrete edges
6. Perform each injection (parallel or serial)
   - Execution context changed
     - SUCCESS!
   - Otherwise
     - FAILED
7. Repeat process with successful executions

Combinatorics
Combinatorial sequences to manage:
- Numerous possible edges
- Various encodings
- Multiple extension steps

XIEv

Dynamic Analysis for Model Extraction.
- Compatible with Ajax and other JavaScript-based modifications.
- Navigates the site taking all available user actions.
- Records messages, loaded resources, ...
- Additional Use Cases:
  - Crawling
  - Regression Testing
  - Resource Extraction
- Uses DOMdiff to classify similar pages.

Architecture

DOMdiff

Identifies changes between DOM trees:
- Additions
- Deletions
- Replacements
- Attribute/text changes
- Classifies pairs of documents:
  - Same: Likely the same page, minor differences
  - Similar: Pages based on the same template
  - Separate: Unrelated pages
- Classifier based on Machine Learning or Decision Trees.
- Supports pages with dynamic content.

Visualized Output: Old vs New DOM

- Added
- Removed
- Modified
- Subtree modified
- Subtree unchanged

Relations

Independent or combined use possible

HYDRA in Practice

HYDRA navigates the weighted execution context graph towards target states (highest weights) by extending the injection vector with HTML5 parser state graph edges, optionally adding encodings.

Weighted Execution Contexts

Example Injection

This example target filters out the < character and suspicious attributes like onload, but is susceptible to double-encoding.