Automated Security Test Case Generation

AUTHOR: JEREMY ARELLANO

Why This Work Matters

GROWING PREVALENCE OF SOFTWARE SECURITY BREACHES GLOBALLY

IMPACT OF SECURITY BREACHES ON DATA INTEGRITY AND SERVICE AVAILABILITY

INCREASING COMPLEXITY OF SOFTWARE SYSTEMS AND THEIR VULNERABILITIES

NEED FOR EFFICIENT, AUTOMATES SOLUTIONS FOR A RAPIDLY CHANGING LANDSCAPE

Background: The Problem

- Rising Threats
  - Increase in frequency of software security breaches. Overtime they have been getting more severe.
- Consequences of Breaches
  - Data Loss, Compromised Services, Denial of Services
- Current Preventative Measures
  - Static: Code analysis without execution (parsing and searching through repositories for common security vulnerabilities)
  - Dynamic: Analyzing the output of a program as its running
- Shortfalls
  - Vulnerabilities are consistently being found, causing static techniques to quickly become out of date.

Background: The Need for Automation

- Manual Testing Limitations: Often take long periods of time to develop, especially for more complex repositories
- Automated Solutions: Test case generation exists for some applications such as IoT applications
- Larger architectural projects have built in scans that analyze the structure and potential weaknesses these structures have on its security

Project Goals And Research Questions

1

Bridge the gap with an automated test generation framework that expands away from a category of applications to a wider range

2

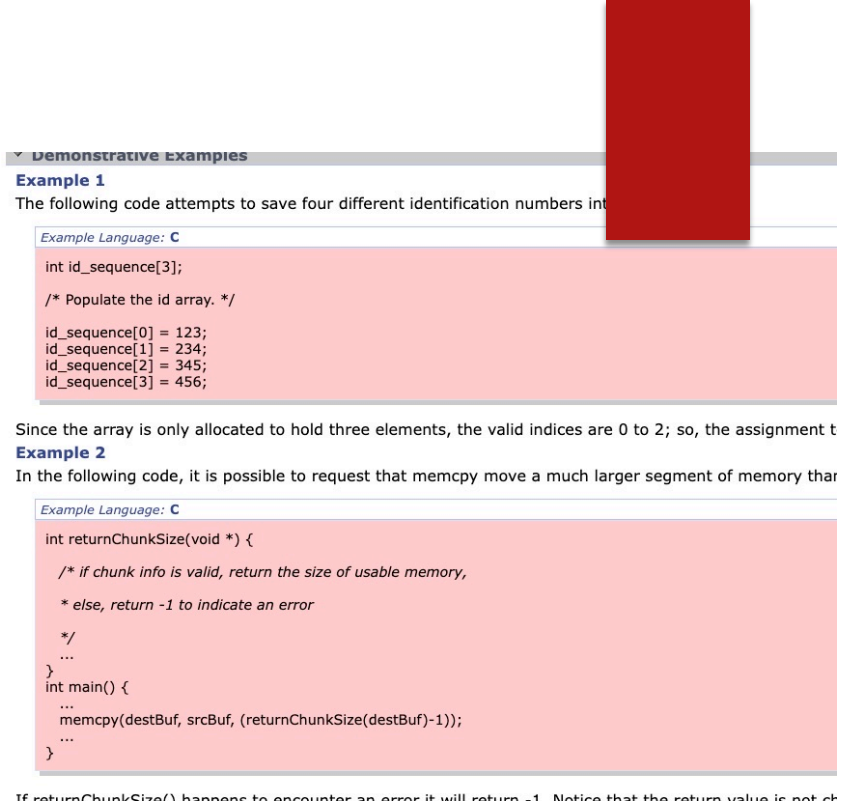
Utilize risk and architectural views to enhance the generation of these security test cases

3

Compare how this program behaves with existing programs that specialize in one area of security.

Early Results: Data Collected

- CWE Database: Created and parse the most common through the given XML files
  - 25 Most Dangerous Software Weaknesses
  - OWASP Top Ten
- Repositories to Analyze
  - Utilizing an open-source project “Home Assistant from GitHub to run static and architectural overview of the contents



Demonstrative Examples

Example 1

The following code attempts to save four different identification numbers int

Example Language: **C**

```
int id_sequence[3];

/* Populate the id array. */

id_sequence[0] = 123;
id_sequence[1] = 234;
id_sequence[2] = 345;
id_sequence[3] = 456;
```

Since the array is only allocated to hold three elements, the valid indices are 0 to 2; so, the assignment t

Example 2

In the following code, it is possible to request that memcpy move a much larger segment of memory thar

Example Language: **C**

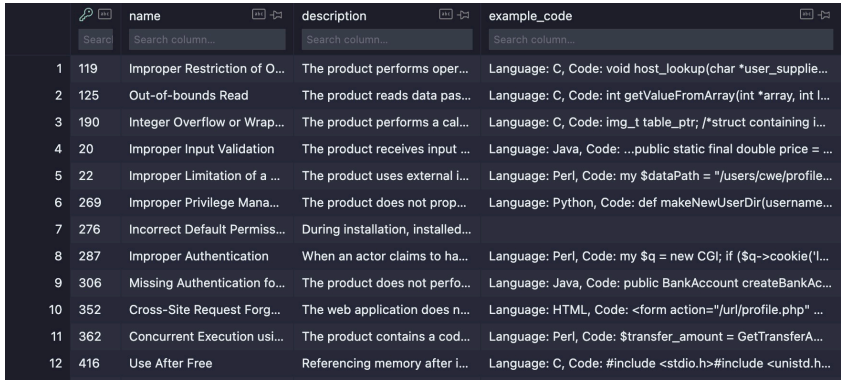
```
int returnChunkSize(void *) {

  /* if chunk info is valid, return the size of usable memory,

  * else, return -1 to indicate an error

  */
  ...
}
int main() {
  ...
  memcpy(destBuf, srcBuf, (returnChunkSize(destBuf)-1));
  ...
}
```

If returnChunkSize() happens to encounter an error it will return -1. Notice that the return value is not ch



	name	description	example_code
1	119	Improper Restriction of O...	The product performs oper...	Language: C, Code: void host_lookup(char *user_supplie...
2	125	Out-of-bounds Read	The product reads data pas...	Language: C, Code: int getValueFromArray(int *array, int l...
3	190	Integer Overflow or Wrap...	The product performs a cal...	Language: C, Code: img_t table_ptr; /*struct containing i...
4	20	Improper Input Validation	The product receives input ...	Language: Java, Code: ...public static final double price = ...
5	22	Improper Limitation of a ...	The product uses external i...	Language: Perl, Code: my $dataPath = "/users/cwe/profile...
6	269	Improper Privilege Mana...	The product does not prop...	Language: Python, Code: def makeNewUserDir(username...
7	276	Incorrect Default Permiss...	During installation, installed...	
8	287	Improper Authentication	When an actor claims to ha...	Language: Perl, Code: my $q = new CGI; if ($q->cookie('l...
9	306	Missing Authentication fo...	The product does not perfo...	Language: Java, Code: public BankAccount createBankAc...
10	352	Cross-Site Request Forg...	The web application does n...	Language: HTML, Code: <form action="/url/profile.php" ...
11	362	Concurrent Execution usi...	The product contains a cod...	Language: Perl, Code: $transfer_amount = GetTransferA...
12	416	Use After Free	Referencing memory after i...	Language: C, Code: #include <stdio.h>#include <unistd.h...

Early Results: Initial Framework Development

- Static Analyzer built to parse through the entire repository and match its contents with the common security vulnerability database
- Matches a percentage to how close a vulnerability is to matching with the database
- Compiles a list of vulnerabilities and appends the ID, and Name of the CWE associated with the file.
  - So far, this process returns a lot of false positives

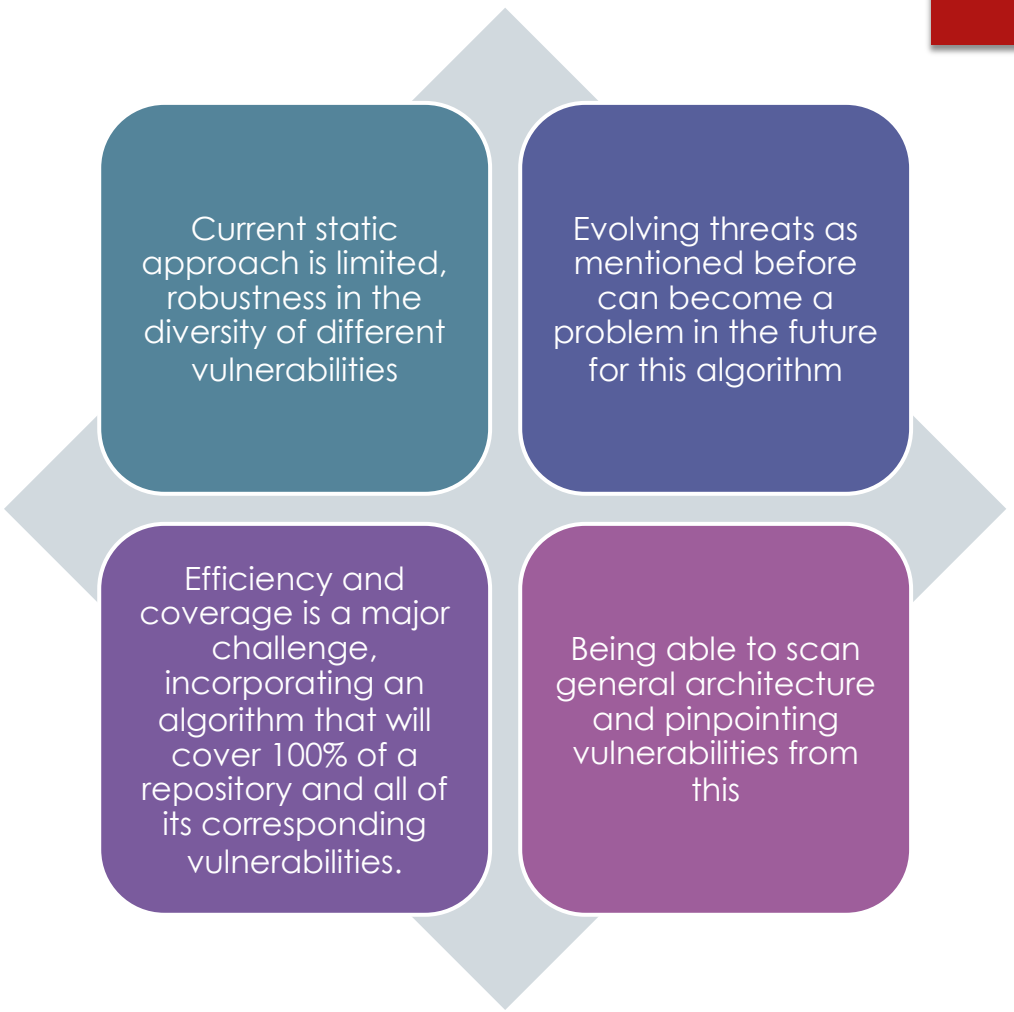
Real-World Impact

Security Enhancement Tool: Being able to use this tool with software development to enhance the security of applications

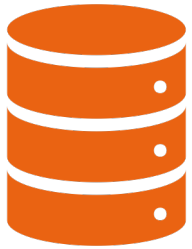
Development Efficiency: Eliminate the need to manual security testing and reduce the time needed to scan large repositories

Adaptability: Being able to use this framework with a large diversity of projects.

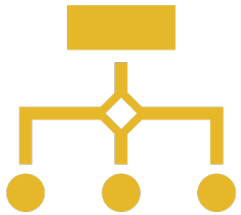
Inherent Challenges



Road Ahead: Future Plans



Figuring out how to extract more information from the CWE database and categorize them into severity



Possibly add an efficient measure so that the algorithm can run efficiently (i.e., scan multiple files at once rather than one at a time).

Incorporate Architectural Design Analysis within the algorithm to find vulnerabilities in structure