Automated vulnerability scanning and exploitation

Dennis Pellikaan Thijs Houtenbos

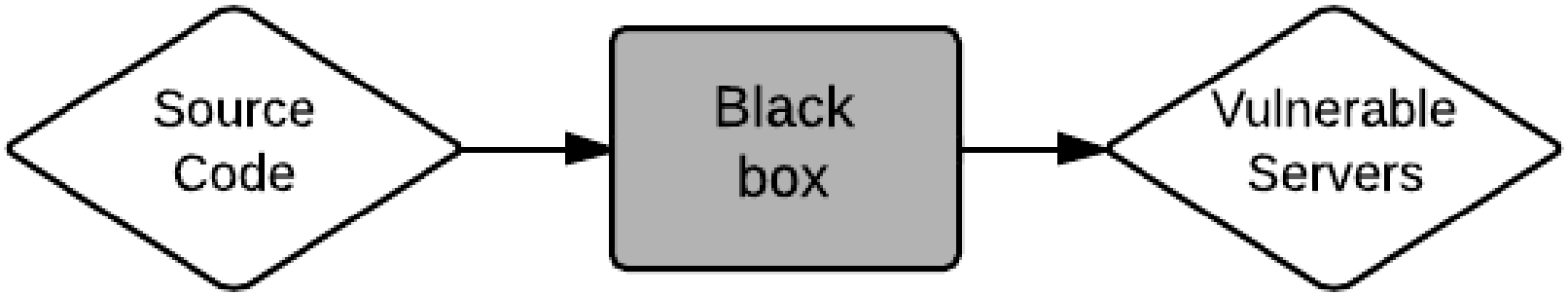
University of Amsterdam
System and Network Engineering

October 22, 2013

Introduction

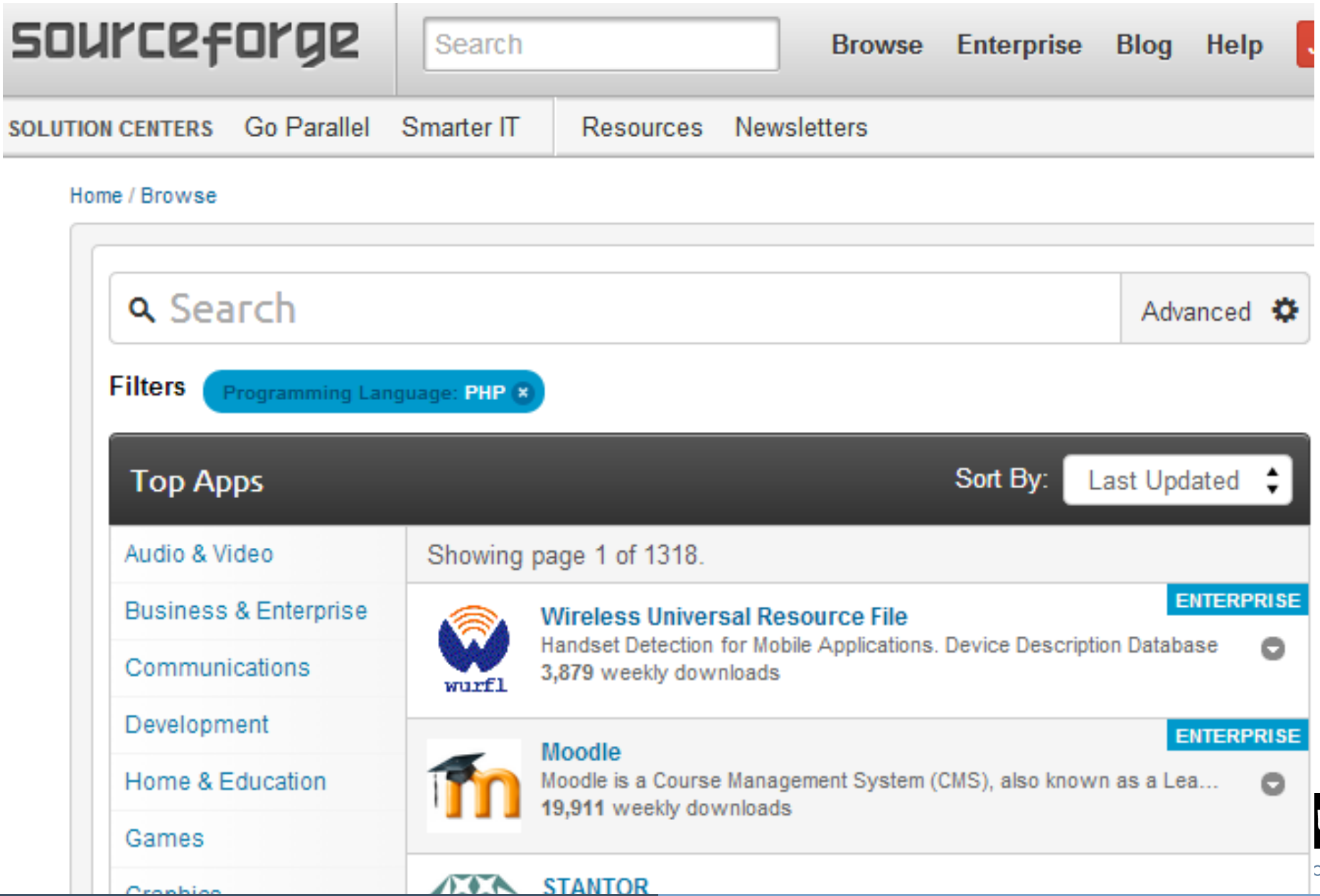
- Open Source scripts
- Shared on the internet, can be used by anyone
- Lots of attention for large projects (Wordpress, Joomla, etc)
- What about the rest?

System overview

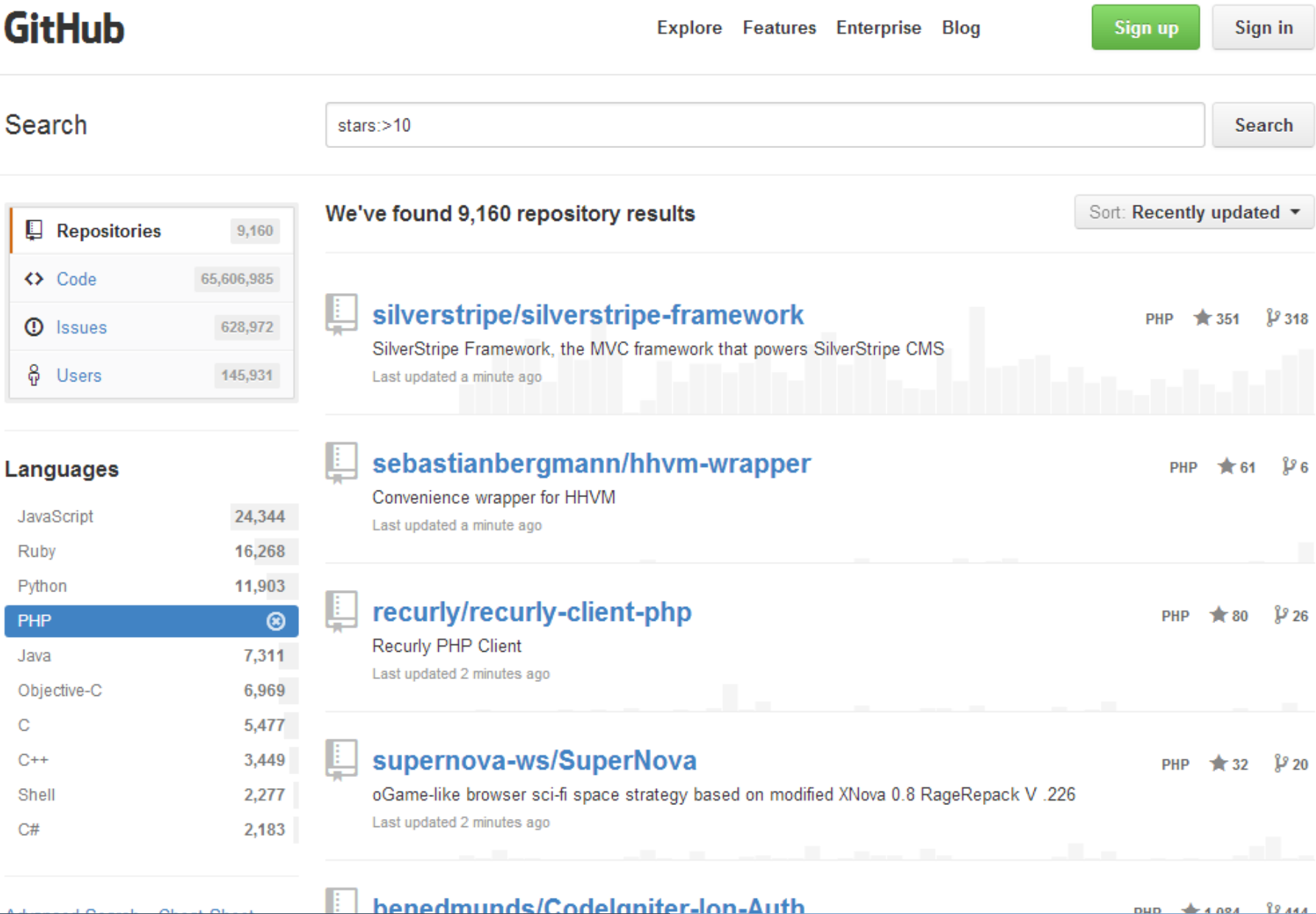


Completely automated system which gathers source code as input and outputs a list of vulnerable servers.

Sourceforge



Github


GitHub
Explore
Features
Enterprise
Blog
Sign up
Sign in
Search
stars:>10
Search
Repositories 9,160
Code 65,606,985
Issues 628,972
Users 145,931
We've found 9,160 repository results
Sort: Recently updated
silverstripe/silverstripe-framework
SilverStripe Framework, the MVC framework that powers SilverStripe CMS
Last updated a minute ago
PHP 351 318
sebastianbergmann/hhvm-wrapper
Convenience wrapper for HHVM
Last updated a minute ago
PHP 61 6
recurly/recurly-client-php
Recurly PHP Client
Last updated 2 minutes ago
PHP 80 26
supernova-ws/SuperNova
oGame-like browser sci-fi space strategy based on modified XNova 0.8 RageRepack V .226
Last updated 2 minutes ago
PHP 32 20
Languages
JavaScript 24,344
Ruby 16,268
Python 11,903
PHP
Java 7,311
Objective-C 6,969
C 5,477
C++ 3,449
Shell 2,277
C# 2,183

Github

GitHub

Explore Features Enterprise Blog

Sign up Sign in

Search

mysql_query $_GET

Search

Repositories 5

Code 114,139

Issues 852

Users

Languages

PHP	101,858
HTML	2,742
Smarty	884
JavaScript	621
Emacs Lisp	115
Markdown	64
Perl	59
ASP	45
XML	29
reStructuredText	24

We've found 114,139 code results

Sort: Best match

viajantes/recetario – sampleSQLInjection.php

PHP

Last indexed 3 months ago

```
1   <?php
2
3   $a = $_GET['a'];
4
5   mysql_query('x' . $a . 'y');
6
7
8
9   if(isset($_GET['id']) && $_GET['status']==2)
...
9   if(isset($_GET['id']) && $_GET['status']==2)
10
11  {
12
13  mysql_query("DELETE FROM shops2 WHERE id='$_GET[id]' ");
```

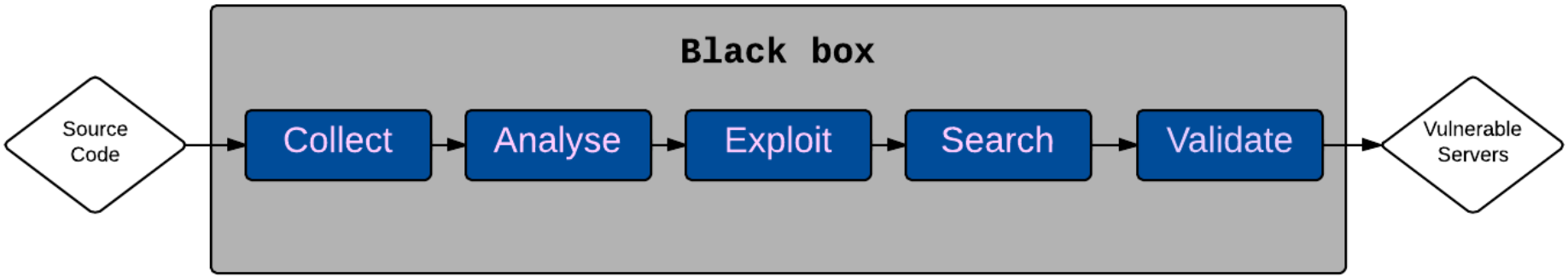
clintonjmurdoch/jwxicc_php – selectingplayer.php

PHP

Last indexed 3 months ago

```
1   <?php
```

System parts



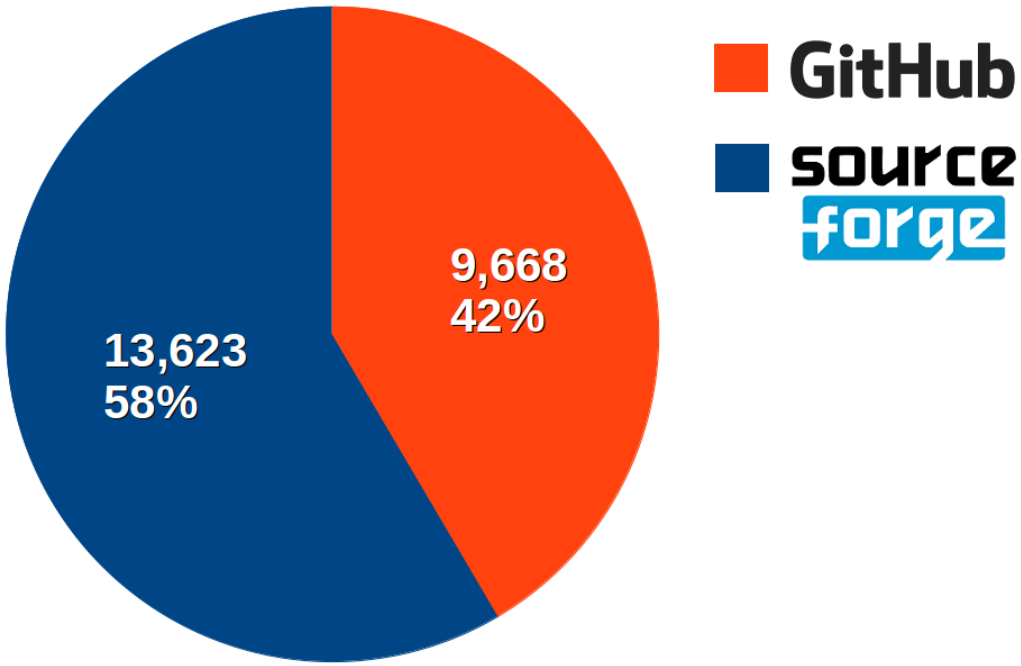
- Collect a large number of projects
- Analyse code for possible vulnerabilities
- Exploit the findings in a local environment to confirm
- Search installations of the project online
- Validate the found installation matches the project

Collect projects

- Two sources
  - Sourceforge
  - GitHub
- Focus on PHP scripts
- Automated download and extraction

Collect projects

Collected projects



Analyse code

SQL Injection

mysql_query ("SELECT * FROM users WHERE id='$_GET[id]'");

File Inclusion

require $_POST["lang_install"].".php";

Command Injection

exec ($_GET['com'], $result);

Regular Expressions

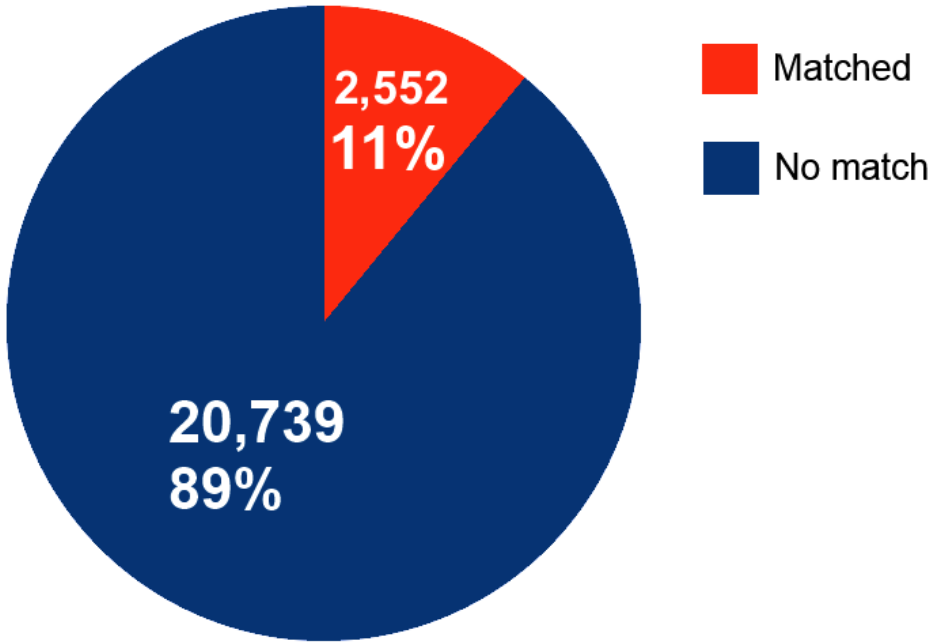
```
# Ignore comments
push @regexps, {'regexp' => '^#', 'print' => false};
push @regexps, {'regexp' => '^//', 'print' => false};
push @regexps, {'regexp' => '^/\*', 'print' => false};

# SQL injection
# Ignore these
push @regexps, {'regexp' => 'wpd->prepare', 'print' => false};
push @regexps, {'regexp' => 'wpdb->prepare', 'print' => false};
push @regexps, {'regexp' => 'wpdb->get_var', 'print' => false};
push @regexps, {'regexp' => 'addslashes', 'print' => false};
push @regexps, {'regexp' => 'mysql_real_escape_string', 'print' => false};
# We want these
push @regexps, {'regexp' => 'select\s+.*\sfrom\s.*=.*\$_get', 'print' => true, 'cat' => 'SQL Inject
push @regexps, {'regexp' => 'select\s+.*\sfrom\s.*=.*\$_post', 'print' => true, 'cat' => 'SQL Injec
push @regexps, {'regexp' => 'insert\s+into\s+.*\$_get', 'print' => true, 'cat' => 'SQL Injection',
push @regexps, {'regexp' => 'insert\s+into\s+.*\$_post', 'print' => true, 'cat' => 'SQL Injection',
push @regexps, {'regexp' => 'delete\s+from\s+.*\s+where\s+.*=.*\$_get', 'print' => true, 'cat' => '
push @regexps, {'regexp' => 'delete\s+from\s+.*\s+where\s+.*=.*\$_post', 'print' => true, 'cat' =>
push @regexps, {'regexp' => 'update\s+.*\s+set\s+.*=.*\$_get', 'print' => true, 'cat' => 'SQL Injec
push @regexps, {'regexp' => 'update\s+.*\s+set\s+.*=.*\$_post', 'print' => true, 'cat' => 'SQL Inje
push @regexps, {'regexp' => 'mysql_query\s*\(.*\$_get.*\)', 'print' => true, 'cat' => 'SQL Injectio
push @regexps, {'regexp' => 'mysql_query\s*\(.*\$_post.*\)', 'print' => true, 'cat' => 'SQL Injecti

# Remote/Local File Inclusion
push @regexps, {'regexp' => 'include\s+.*\$_get', 'print' => true, 'cat' => 'File Inclusion', 'type
push @regexps, {'regexp' => 'include\s+.*\$_post', 'print' => true, 'cat' => 'File Inclusion', 'typ
push @regexps, {'regexp' => 'require\s+.*\$_get', 'print' => true, 'cat' => 'File Inclusion', 'type
push @regexps, {'regexp' => 'require\s+.*\$_post', 'print' => true, 'cat' => 'File Inclusion', 'typ
push @regexps, {'regexp' => 'include_once\s+.*\$_get', 'print' => true, 'cat' => 'File Inclusion',
push @regexps, {'regexp' => 'include_once\s+.*\$_post', 'print' => true, 'cat' => 'File Inclusion',
push @regexps, {'regexp' => 'require_once\s+.*\$_get', 'print' => true, 'cat' => 'File Inclusion',
push @regexps, {'regexp' => 'require_once\s+.*\$_post', 'print' => true, 'cat' => 'File Inclusion',
push @regexps, {'regexp' => 'eval\s*\(.*\$_get.*\)', 'print' => true, 'cat' => 'File Inclusion', 't
push @regexps, {'regexp' => 'eval\s*\(.*\$_post.*\)', 'print' => true, 'cat' => 'File Inclusion', '
push @regexps, {'regexp' => 'assert\s*\(.*\$_get.*\)', 'print' => true, 'cat' => 'File Inclusion',
```

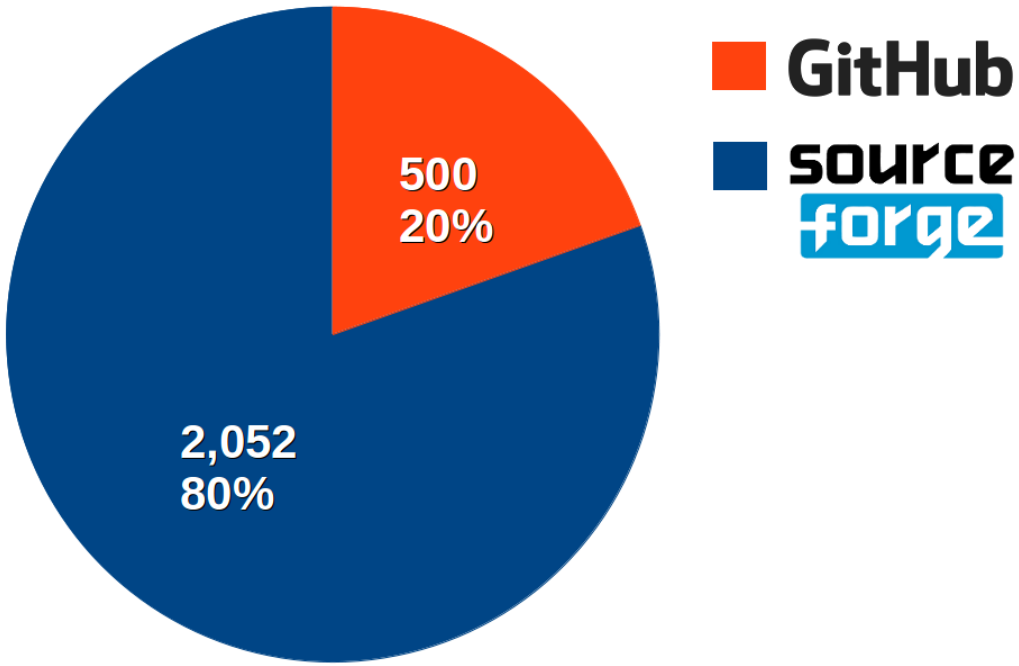
Analyse projects

Vulnerable projects



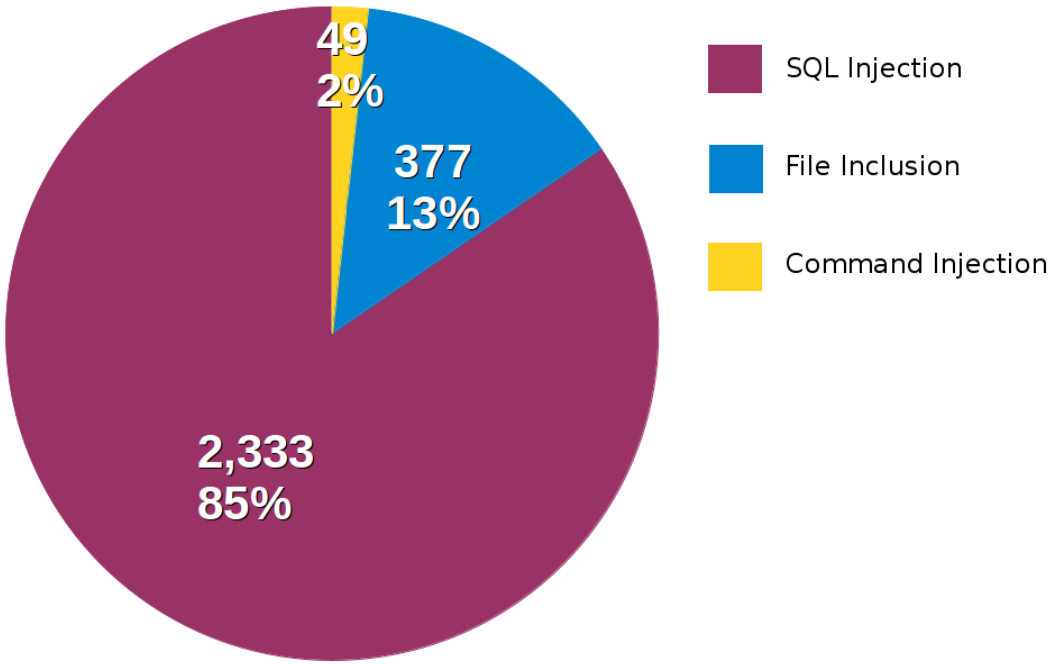
Analyse projects

Vulnerable projects



Analyse projects

Vulnerability categories



Exploit vulnerabilities

SQL Injection

```
mysql_query ("SELECT * FROM users WHERE id='$_GET[id]'");
```

File Inclusion

```
require $_POST["lang_install"].".php";
```

Command Injection

```
exec ($_GET['com'], $result);
```

Exploit vulnerabilities

SQL Injection

override_function (mysql_query, log_function);

Script sources

mysql_query ("SELECT * FROM users WHERE id='$_GET[id]'");

Executed

log_function ("SELECT * FROM users WHERE id='$_GET[id]'");

Exploit vulnerabilities

File Inclusion

```
require $_POST["lang_install"].".php";
log_function ($_POST["lang_install"].".php");
```

Command Injection

```
exec ($_GET['com'], $result);
log_function ($_GET['com'], $result);
```

Exploit vulnerabilities

Request the page
http://localhost/myscript/admin.php?id=hacklu

Log function
Write the function arguments to a logfile

Logfile
admin.php:137 mysql_query
SELECT * FROM users WHERE id ='hacklu'

Exploit vulnerabilities

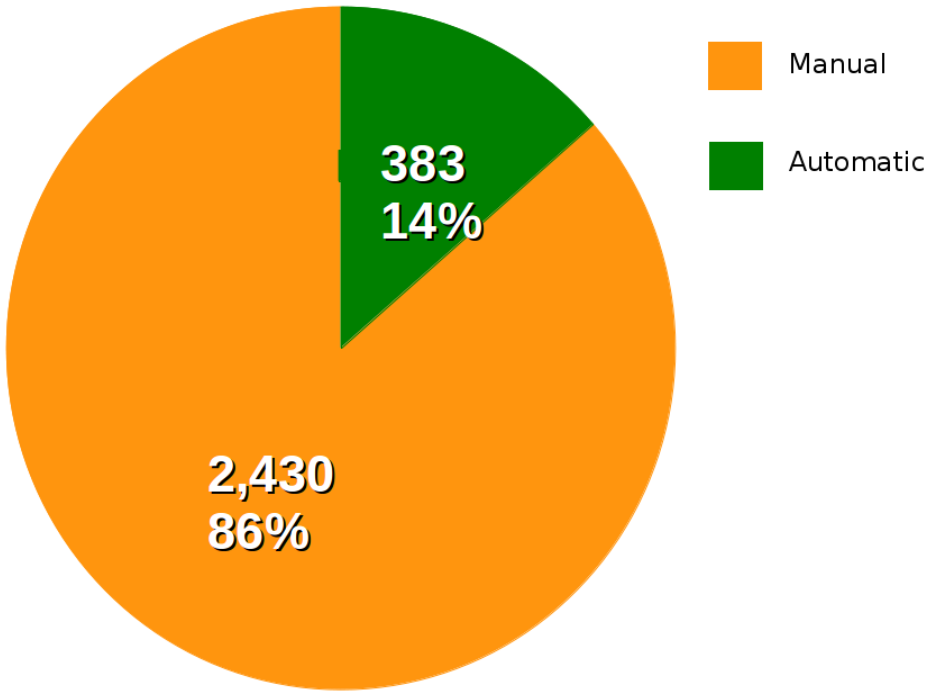
Request the page
http://localhost/myscript/admin.php?id=hack'lu

Log function
Write the function arguments to a logfile

Logfile
admin.php:137 mysql_query
SELECT * FROM users WHERE id ='hack'lu'

Exploit vulnerabilities

Confirmation of results



Search

bing

Google™

DuckDuckGo

YAHOO!

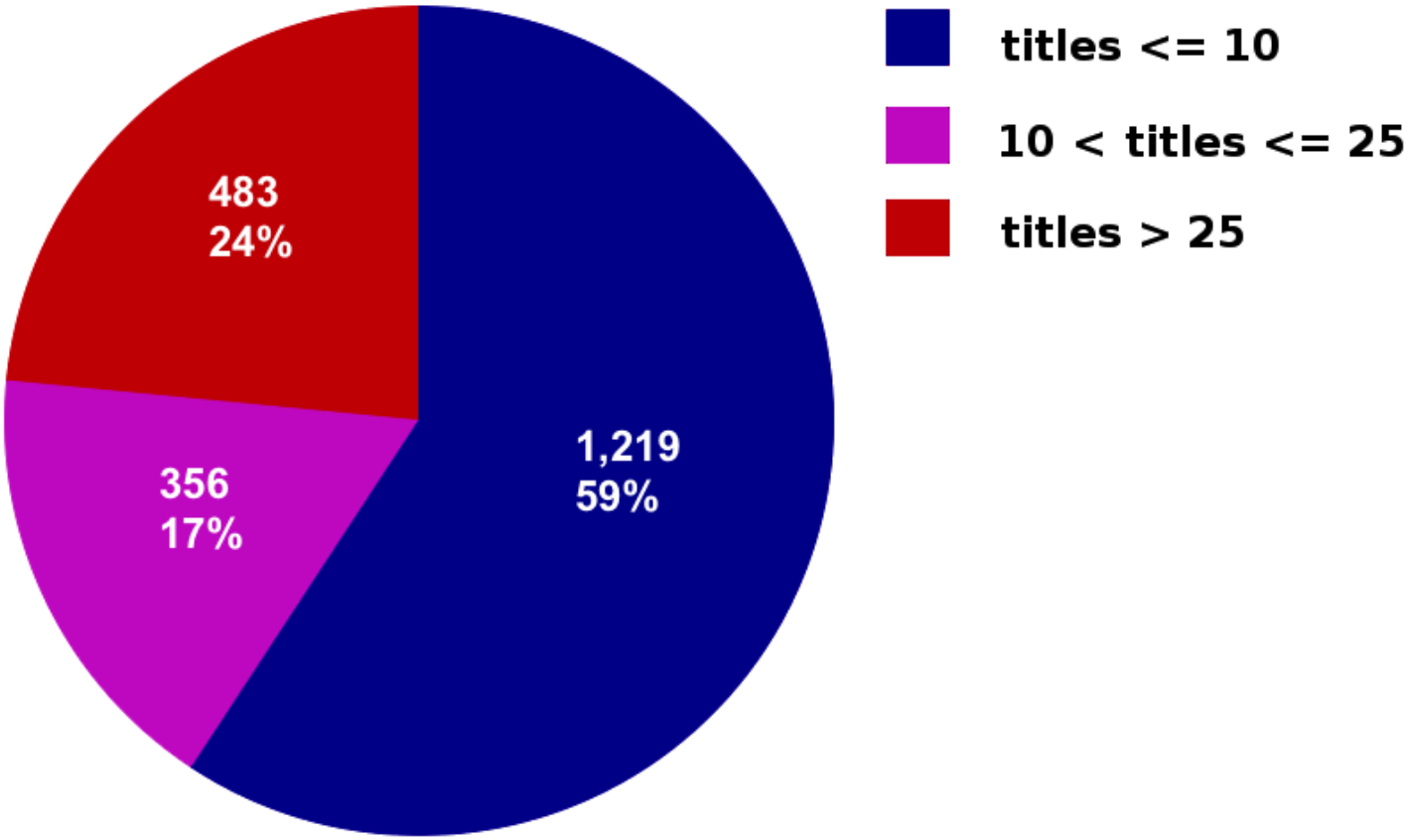
Search

Google Advanced Search Operators

- allinurl
  page.php: require $_GET['page_id'];
  allinurl:"/page.php?page_id="
- allintitle
  index.php: echo "<title>" . $title . "</title>";
  allintitle:"My special script v0.2a"

Search


Source Code
Collect
Analyse
Exploit
Search
Validate
Vulnerable Servers


483
24%
356
17%
1,219
59%
titles <= 10
10 < titles <= 25
titles > 25

Search


Source Code
Collect
Analyse
Exploit
Search
Validate
Vulnerable Servers


Google
allinurl:"/login.php?token="
Ongeveer 545.000 resultaten (0,15 seconden)
BZFlag Image Uploader :: Login
images.bzflag.org/submitimages/login.php?token... Vertaal deze pagina
The login was not successful. Please try again. If problem persists, please contact an administrator. By using the BZFlag Image Uploader, you agree to the Terms ...
Gooooooogle ›
1 2 3 4 5 6 7
Volgende

Search

Google **Error**

We're sorry...

... but your query looks similar to automated requests from a computer virus or spyware application. To prot

We'll restore your access as quickly as possible, so try again soon. In the meantime, if you suspect that you run a virus checker or spyware remover to make sure that your systems are free of viruses and other spurio

If you're continually receiving this error, you may be able to resolve the problem by deleting your Google coc please consult your browser's online support center.

If your entire network is affected, more information is available in the Google Web Search Help Center.

We apologize for the inconvenience, and hope we'll see you again on Google.

To continue searching, please type the characters you see below:

Search

- Rotate between 13 IPv4 addresses
- Pause for 8 seconds between each request

- 20,000 search queries per day
- 120,000 results with 22,000 queries

Validate search results

- Find the project's installation root
- Identify six common file types
- Compare locally identified files with the remote host
- Calculate a score

Validate search results

Installation root: ***deterministic approach***

Google result: http://example.com/user/app/login.php?token=432

Local script	**Remote script**
/script/app/admin/login.php	/example.com/user/app/admin/login.php
/script/app/~~admin~~/	/example.com/user/app/~~admin~~/
/script/~~app~~/	/example.com/user/~~app~~/
/script/	**/example.com/user/**

Validate search results

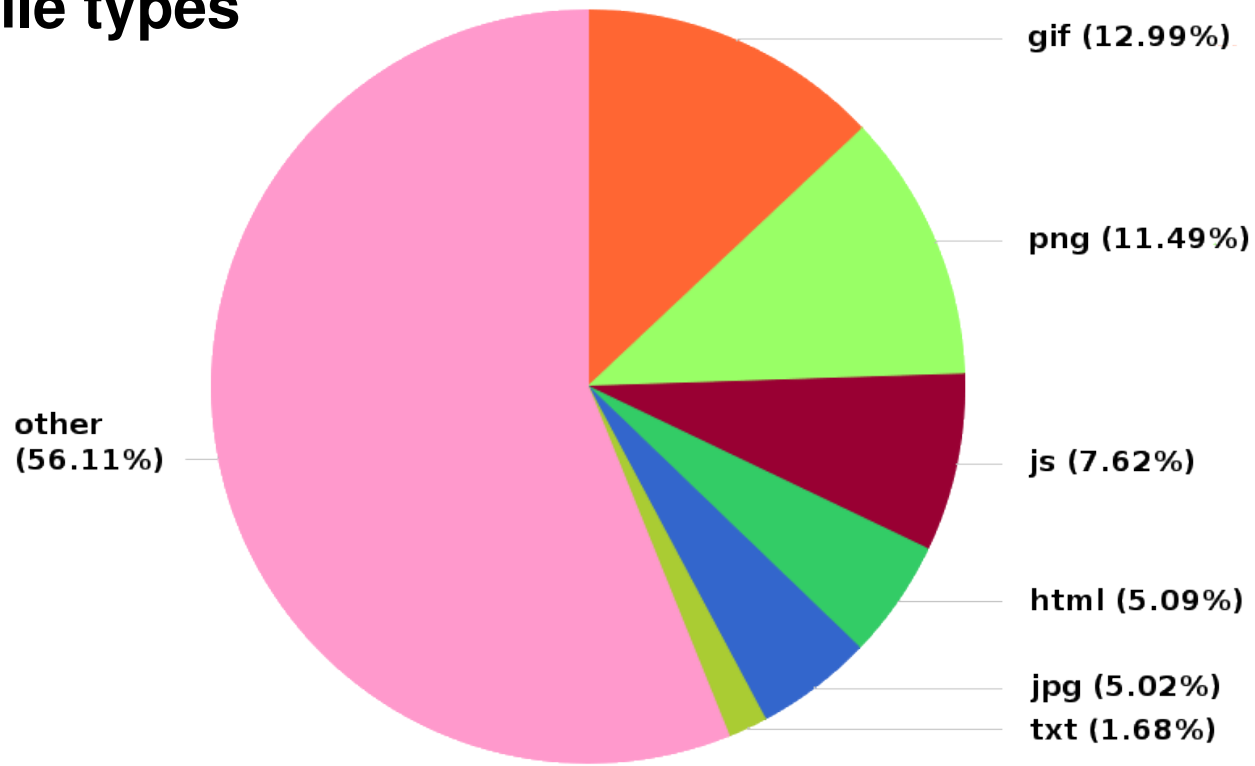
Installation root: ***probabilistic approach***

Google result: http://example.com/user/app/guide.html

Local script
/script/a/docs/examples/index.php
/script/b/index.html
/script/index.php
/script/

Validate search results

Common file types



Validate search results

Comparing files

Local file	Remote file
/script/images/file1.gif	/example.com/user/images/file1.gif
/script/images/logo.png	/example.com/user/images/logo.png
/script/app/js/code.js	/example.com/user/app/js/code.js
/script/contact.html	/example.com/user/contact.html

Validate search results

Text matching

Local File (*LocalScore* = 74)

```
body {
  padding-left: 11em;
  font-family: Georgia, Times;
  color: purple;
  background-color: #d8da3d
}
ul.navbar {
  list-style-type: none;
  padding: 0;
  margin: 0;
  position: absolute;
  top: 2em;
  left: 1em;
  width: 9em
}
a:link {
  color: blue
}
a:visited {
  color: purple
}
```

Remote File (*RemoteScore* = 50)

```
/* Lorem ipsum dolor sit amet,
consectetuer adipiscing elit. Aenean
commodo ligula eget dolor. Aenean massa.
Cum sociis natoque penatibus et magnis
dis parturient montes, nascetur ridiculus
mus. */
body {
  padding-left: 11em;
  font-family: Georgia, Times;
  color: purple;
  background-color: #d8da3d
}
ul.navbar {
  list-style-type: none;
  padding: 0;
  margin: 0;
  position: absolute;
  top: 2em;
  left: 1em;
  width: 9em
}
```

Validate search results

Text matching

Local File (*LocalScore* = 100)

```
/* dolor sit amet, consectetuer
adipiscing elit. Aenean commodo ligula
eget dolor. Aenean massa. Cum sociis
natoque penatibus et magnis dis
parturient montes, nascetur ridiculus
mus. */
```

Remote File (*RemoteScore* = 50)

```
/* dolor sit amet, consectetuer
adipiscing elit. Aenean commodo ligula
eget dolor. Aenean massa. Cum sociis
natoque penatibus et magnis dis
parturient montes, nascetur ridiculus
mus. */
body {
  padding-left: 11em;
  font-family: Georgia, Times;
  color: purple;
  background-color: #d8da3d
}
ul.navbar {
  list-style-type: none;
  padding: 0;
  margin: 0;
  position: absolute;
  top: 2em;
  left: 1em;
  width: 9em
}
```

Validate search results

MD5 Hash Matching

md5(Local File) $\neq$ md5(Remote File)
LocalScore = 0
RemoteScore = 0

md5(Local File) $=$ md5(Remote File)
LocalScore = 100
RemoteScore = 100

Validate search results

Calculating the final score

- Score between 0 and 100
- Number of identified files is taken into account
- *LocalScore* and the *RemoteScore* are weighted

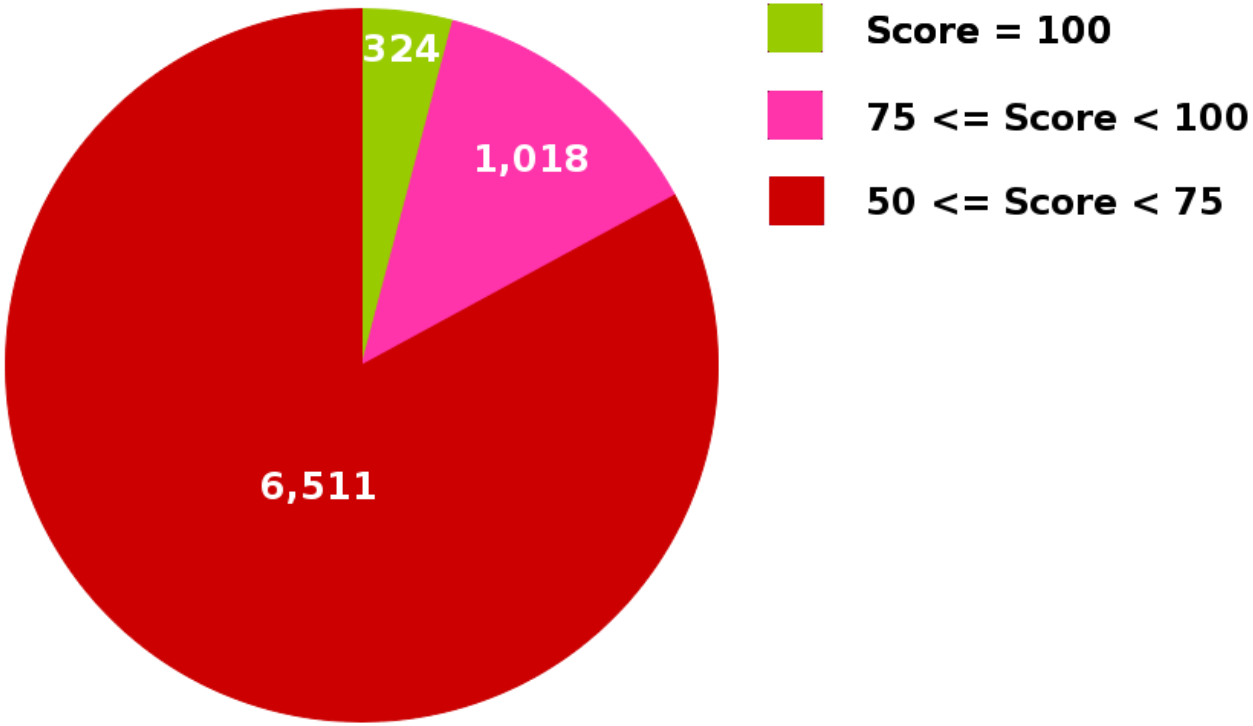
$$Score = \frac{\sum_{i=1}^{N} S_i}{N} + \sum_{i=1}^{N} S_i * \frac{1}{6}$$

$$S_i = \frac{LocalScore_i + RemoteScore_i}{4}$$

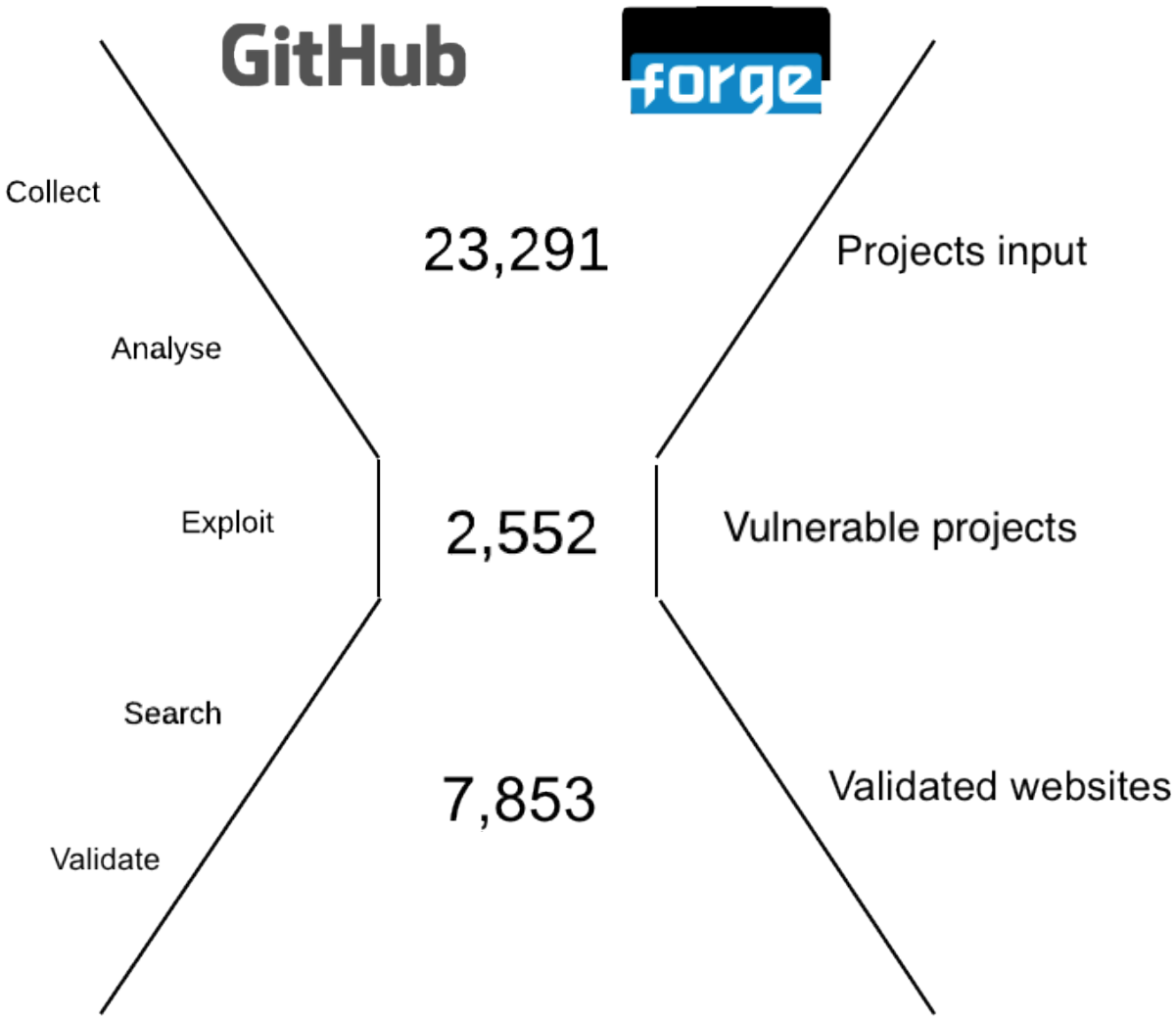
$$N = \text{Total number of selected files}$$

Validate search results

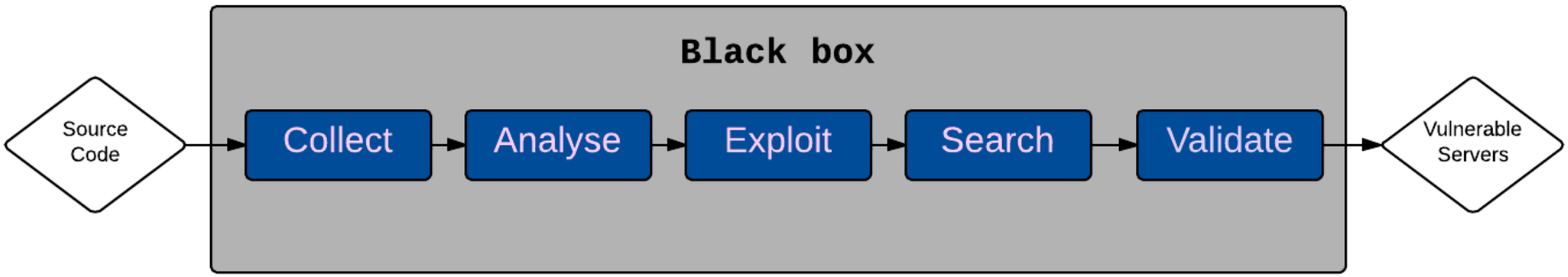
Validated website scores



Results


GitHub
forge
Collect
Analyse
Exploit
Search
Validate
23,291
2,552
7,853
Projects input
Vulnerable projects
Validated websites

System overview



Questions

Contact

Contact:
Dennis: dennis.pellikaan@os3.nl
Thijs: thijs.houtenbos@os3.nl

Paper reference:
http://rp.delaat.net/2012-2013/p91/report.pdf