

Towards the Detection of Inconsistencies in Public Security Vulnerability Reports

Ying Dong, Wenbo Guo, **Yueqi Chen**, Xinyu Xing, Yuqing Zhang, Gang Wang

USENIX Security 2019
August 15th



I ILLINOIS



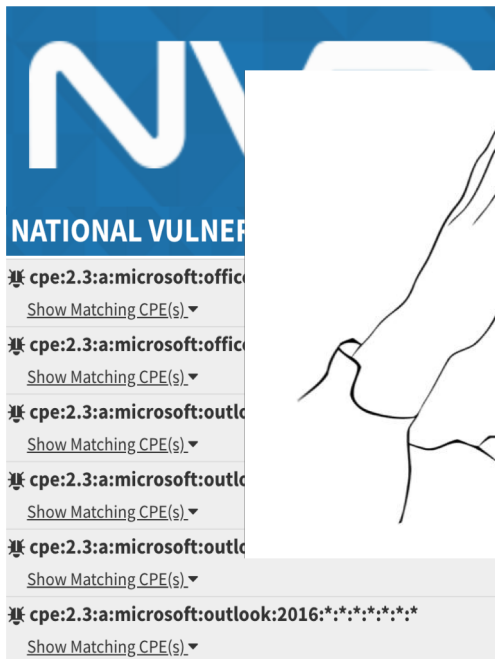
Challenges Faced by Security Operations Engineers

1. Keep an eye on new vulnerabilities that affect their systems
2. Patch vulnerable softwares as soon as possible



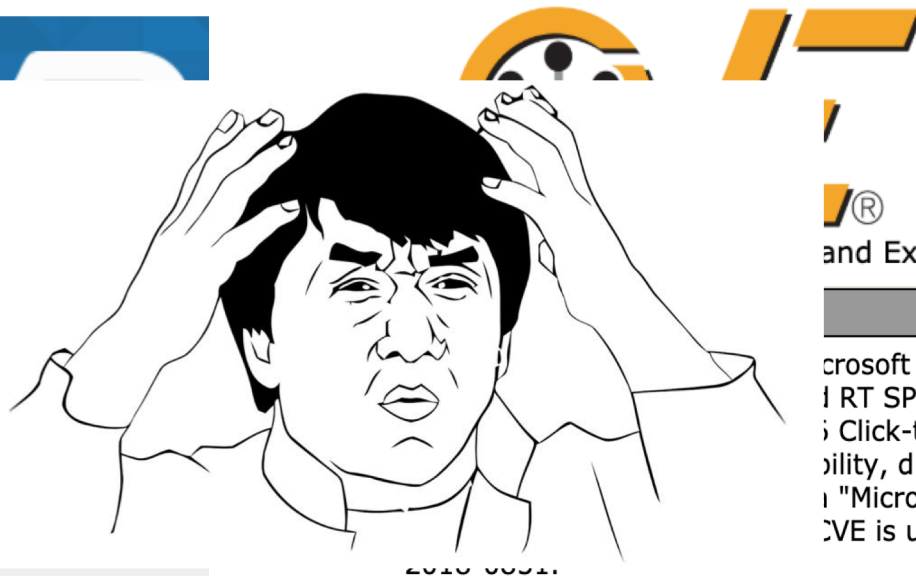
Inconsistent Information → Confusion

A New Vulnerability (CVE-2018-0852) is Exposed



NATIONAL VULNERABILITY DATABASE

- ❖ `cpe:2.3:a:microsoft:office:2010:SP2:RT:SP1:*`
[Show Matching CPE\(s\) ▾](#)
- ❖ `cpe:2.3:a:microsoft:office:2010:SP2:RT:SP1:*`
[Show Matching CPE\(s\) ▾](#)
- ❖ `cpe:2.3:a:microsoft:outlook:2010:SP2:RT:SP1:*`
[Show Matching CPE\(s\) ▾](#)
- ❖ `cpe:2.3:a:microsoft:outlook:2010:SP2:RT:SP1:*`
[Show Matching CPE\(s\) ▾](#)
- ❖ `cpe:2.3:a:microsoft:outlook:2010:SP2:RT:SP1:*`
[Show Matching CPE\(s\) ▾](#)
- ❖ `cpe:2.3:a:microsoft:outlook:2016:*:*:*:*:*`
[Show Matching CPE\(s\) ▾](#)



and Exposures

Microsoft Outlook 2010 SP2, Microsoft Office RT SP1, Microsoft Outlook Click-to-Run (C2R) allow a vulnerability, due to how Outlook handles "Microsoft Office Memory" data. This CVE is unique from CVE-

Microsoft outlook 2007 SP3 - listed.

Microsoft outlook 2007 SP3 - NOT listed.

Research Problems

1. Is inconsistency issue prevalent?
2. What are the characteristics of inconsistent info?
3. Reasons for inconsistency?
4. Security implications of inconsistency?

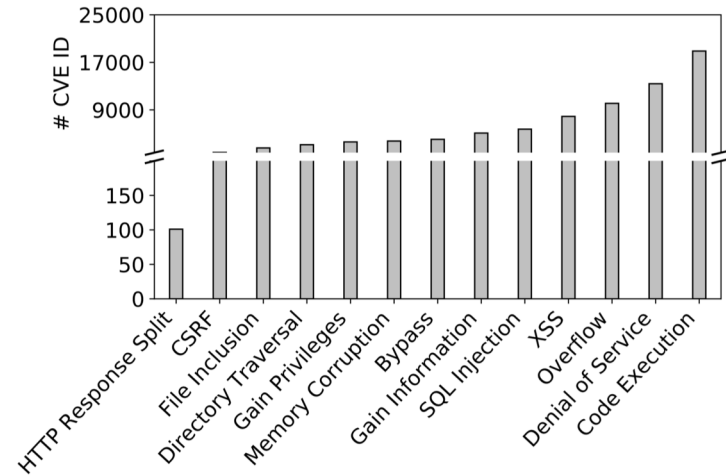
Measuring Inconsistency of Vulnerability Reports

1999 - 2018

Over 20 years



Across websites



of 13 categories

In This Paper:

Part I: VIEM - an automatic system

extract vulnerable software name and versions

Part II: Large-scale Measurement

quantify inconsistency and interesting findings

Traditional NLP Tools Don't Work Well (Validated)

1. Dictionary-based method (CNLL '06, EMNLP '13)
2. Pre-defined rules (SIGSOFT '12, CCS '17, FSE '17)
3. Regular-expression based technique (CCS '17, FSE '17)
4. Techniques handling single entity (ISESE '14, CCS '17, FSE '17)
5. Semfuzz (CCS '17)


Reason: Unique characteristics of vulnerability reports


Why This Is Hard


Vincent Danen 2011-08-20 00:28:58 EDT

Description

A response splitting flaw in Ruby on Rails 2.3.x was reported [1] that could allow a remote attacker to inject arbitrary HTTP headers into a response ... (3.0.0 and later are not vulnerable). Patches are available in the advisory [1] and git [2].

 Vulnerable Software

 Vulnerable Version


 Non-vulnerable Version

1. Previously unseen vulnerable softwares (**Ruby on Rails**)
-> Dictionary-based **X**
2. Both vulnerable (**2.3.x**) and non-vulnerable versions (**3.0.0 and later**) exist
-> Pre-defined rules **X**
3. Reports are highly unstructured
-> Regular-expression based **X**

Why This Is Hard (cont.)

In Windows Vista SP2 and Windows Server 2008 SP2, the Windows font library in .NET Framework 3.0 SP2, 3.5, 3.5.1, 4, 4.5, 4.5.1, 4.5.2, and 4.6; Skype for Business 2016; Lync 2010; Lync 2013 SP1; and Silverlight 5 allows remote attackers to execute arbitrary code via a crafted embedded font, aka "Graphics Memory Corruption Vulnerability."

Publish Date : 2015-12-09 Last Update Date : 2017-09-12

 Vulnerable Software  Vulnerable Version

4. Multiple interested entities

-> Existing tools handling single entity ❌

5. Diverse vulnerability types

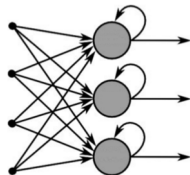
-> Tools for certain vulnerability types (e.g., recall < 40%) ❌

VIEM - NER/RE Model

"The Microsoft VBScript 5.7 and 5.8 engines, as used in Internet Explorer 9 through 11 ..."



Named Entity Recognition (NER) Model



1. Bi-directional RNN
2. word/character embedding
3. Gazetteer



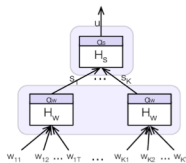
Microsoft VBScript

5.7 and 5.8

Internet Explorer

9 through 11

Relation Extraction (RE) Model



1. One-hot encoding
2. Hierarchical Attention-Network



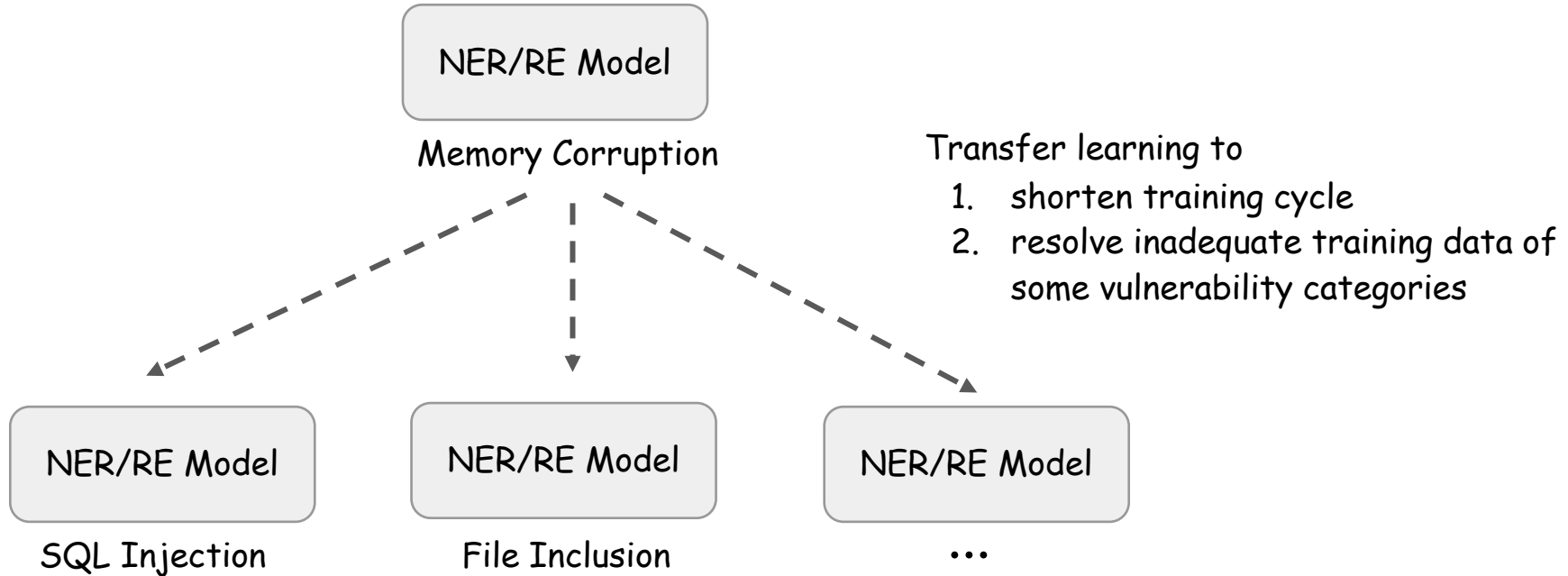
Microsoft VBScript

5.7 and 5.8

Internet Explorer

9 through 11

VIEM - Transfer Learning



VIEM - Dataset

Dataset	Vulnerability Reports
All	70,569

1. Over past 20 years (1999-2018)

VIEM - Evaluating NER/RE models

Metric	Precision	Recall	Accuracy
Result	0.9411	0.9932	0.9764

Over "Memory Corruption" Category

1. *G*-truth dataset (3,448 CVE IDs) with a ratio 8:1:1 for training, validation, and testing
2. Near 100% accuracy, the state-of-the-art is no higher than 90%

VIEM - Evaluating Transfer Learning

Metric	Before
Accuracy	

Avg. over 12

1. Teacher Model - "Memory Copy" Student Model - other 12 categories
2. G-truth dataset with a ratio of 1:10
3. Solved inadequate training datasets

Towards the Detection of Inconsistencies in Public Security Vulnerability Reports

Ying Dong^{1,2*}, Wenbo Guo^{2,4}, Yueqi Chen^{2,4},
Xinyu Xing^{2,4}, Yuqing Zhang¹, and Gang Wang³

¹School of Computer Science and Technology, University of Chinese Academy of Sciences, China
²College of Information Sciences and Technology, The Pennsylvania State University, USA
³Department of Computer Science, Virginia Tech, USA
⁴JD Security Research Center, JD
dongying115@mails.ucas.ac.cn, {wzg13, yxc431, xxing}@ist.psu.edu
zhangyq@ucas.ac.cn, gangwang@vt.edu

Abstract

Public vulnerability databases such as the Common Vulnerabilities and Exposures (CVE) and the National Vulnerability Database (NVD) have achieved great success in promoting vulnerability disclosure and mitigation. While these databases have accumulated massive data, there is a growing concern for their information quality and consistency. In this paper, we propose an automated system VIEM to detect inconsistent information between the fully standardized NVD database and the unstructured CVE descriptions at a massive scale, and provides the needed tool for the community to keep the CVE/NVD databases up-to-date. VIEM is developed from unstructured text. We introduce customized designs to extract vulnerable software names and vulnerable versions from unstructured text. VIEM can recognize previous machine-learning-based named entity recognition (NER) and text classification (RE) so that VIEM can recognize previous ground-truth evaluation shows the system achieves a precision of 0.993 and recall of 0.993. Using 569 vulnerability reports that inconsistent vul-

world attacks with examples ranging from WannaCry ransomware that shut down hundreds of thousands of machines in hospitals and schools [20] to the Equifax data breach that affected half of America's population [21].

To these ends, a strong community effort has been established to find and patch vulnerabilities before they are exploited by attackers. The Common Vulnerabilities and Exposures (CVE) program [4] and the National Vulnerability Database (NVD) [11] are among the most influential forces. CVE is a global list/database that indexes publicly known vulnerabilities by harnessing the "the power of the crowd" (anyone on the Internet (security vendors, developers and researchers) can share the vulnerabilities they found on CVE). NVD is a more standardized database established by the government (i.e., NIST). NVD receives data feeds from CVE website and perform analysis to assign common vulnerability severity scores (CVSS) and other pertinent data [18]. More importantly, NVD standardizes vulnerability mitigation. So far, over 100,000 CVE and NVD play an important role in vulnerability mitigation. Both CVE and NVD have been integrated with hundreds of security world [10].

In This Paper,

Part I: VIEM - an automatic system

extract vulnerable software name and versions



Part II: Large-scale Measurement

quantify inconsistency and interesting findings

Metrics

1. Match software names - # of same words > # of different words

"Internet Explorer" and "Microsoft Internet Explorer" ✓

1. Measure version consistency - Strict match vs. Loose match

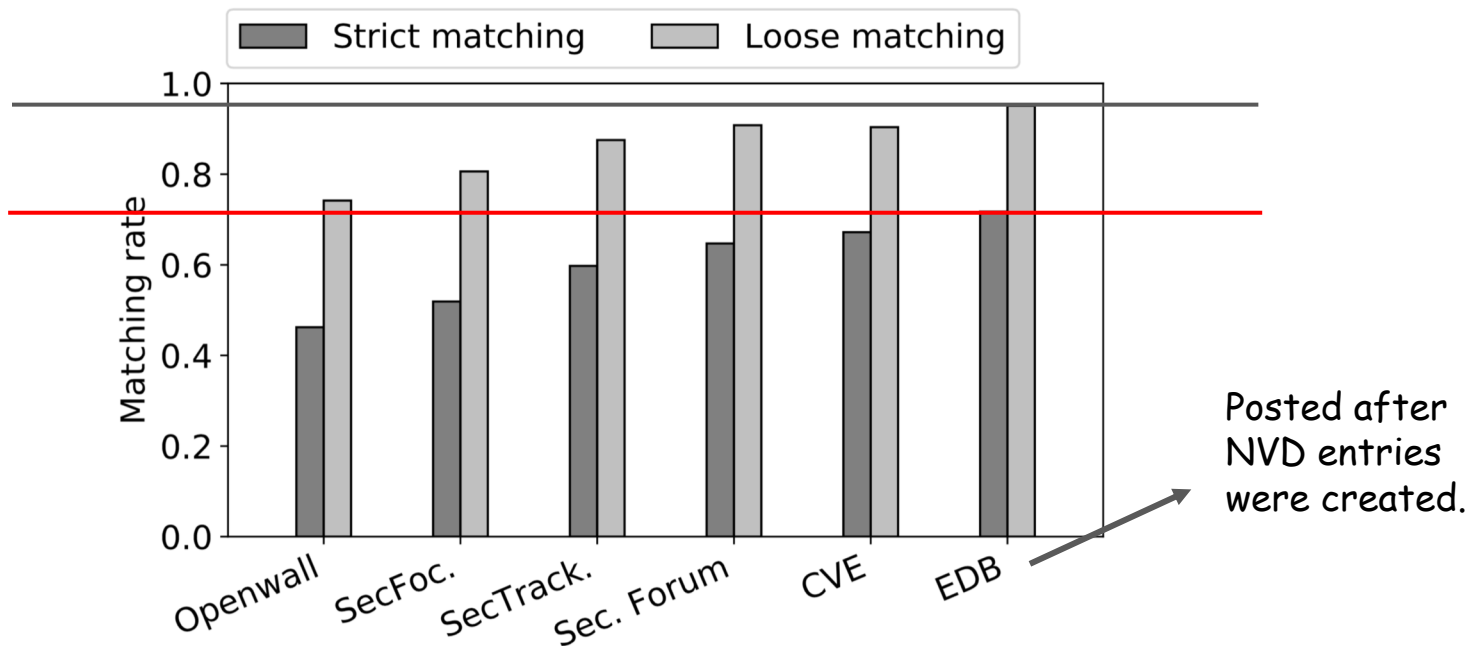
CPE directory
from NIST

"1.1" and "from 1.0 to 1.4" -----> "[1.1]" and "[1.0, 1.1, 1.2, 1.3, 1.4]"

Strict match (Exact match) ✗

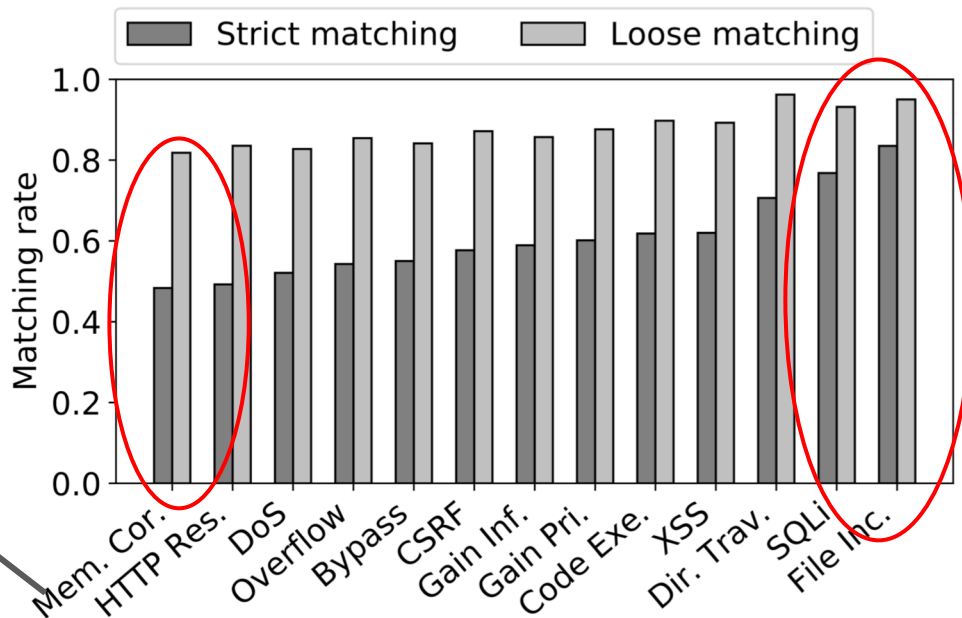
Loose match (One covers another) ✓

Inconsistency Exists Among All Vuln. Report Websites



Matching against NVD - official vulnerability report database maintained by U.S. government

Inconsistency Exists For All Vulnerability Categories



More complex and requires longer time to reproduce and validate.

Matching rate for different vulnerability categories - (CVE + 5 websites) vs. NVD

Inconsistency: Overclaim vs. Underclaim

NVD data

Software	Version
Mozilla Firefox	up to (including) 1.5
Netscape Navigator	up to (including) 8.0.40
K-Meleon	up to (including) 0.9
Mozilla Suite	up to (including) 1.7.12

Overclaim

CVE summary

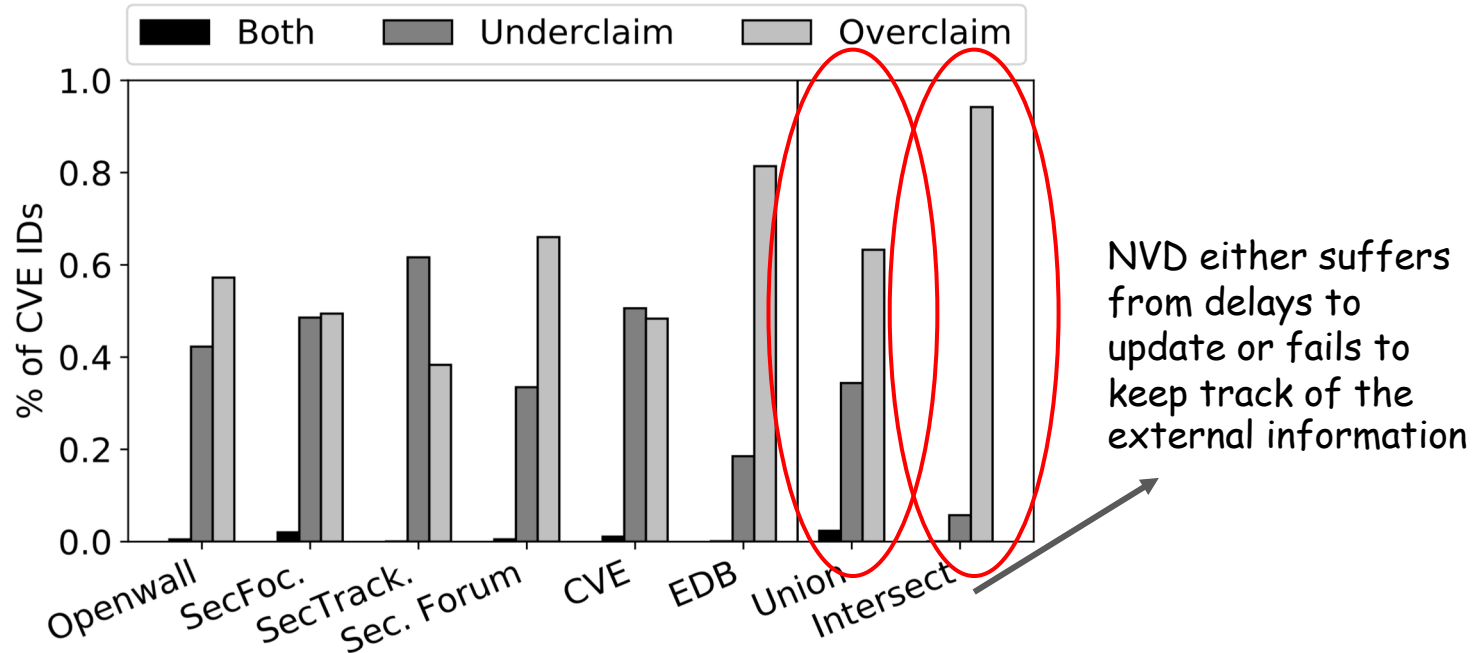
Software	Version
Mozilla Firefox	1.5
Netscape	8.0.4 and 7.2
K-Meleon	before 0.9.12

Underclaim

The diagram illustrates the inconsistency between NVD data and CVE summary. The NVD data table lists vulnerable versions for Mozilla Firefox (up to 1.5), Netscape Navigator (up to 8.0.40), K-Meleon (up to 0.9), and Mozilla Suite (up to 1.7.12). The CVE summary table lists vulnerable versions for Mozilla Firefox (1.5), Netscape (8.0.4 and 7.2), and K-Meleon (before 0.9.12). A red dashed arrow labeled 'Overclaim' points from the NVD data to the CVE summary, indicating that the NVD data includes more vulnerable versions than the CVE summary. A blue dashed arrow labeled 'Underclaim' points from the CVE summary to the NVD data, indicating that the CVE summary includes more vulnerable versions than the NVD data.

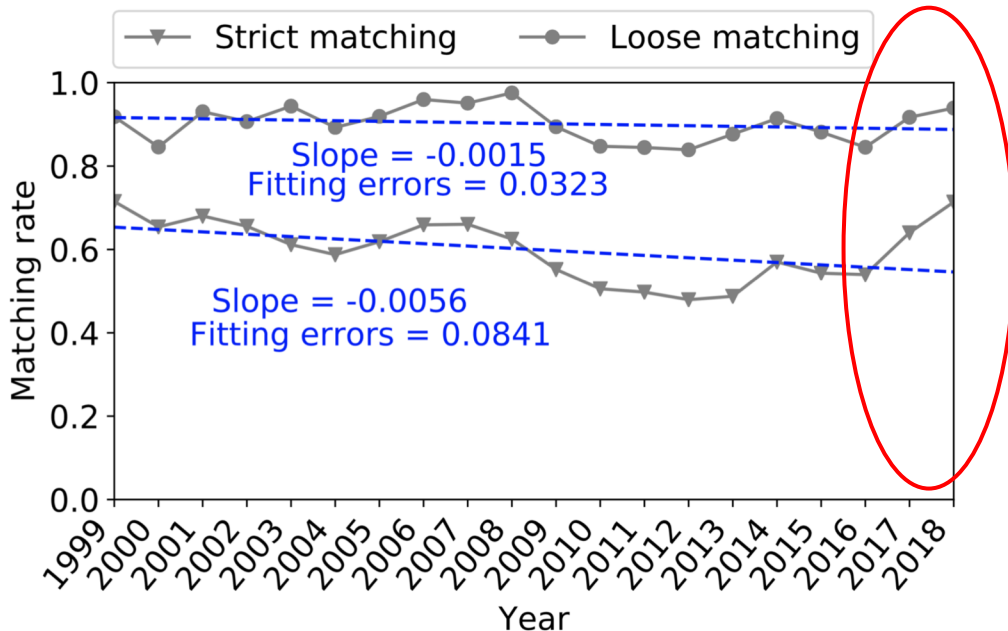
Compared against CVE, NVD overclaims/underclaims
vulnerable versions

Overclaim/Underclaim Are Both Common



Percentage of Underclaim/Overclaim using loose match: (CVE + 5 websites) vs. NVD

Inconsistency Rate Varies Over Time



NVD are getting better at summarizing vulnerability versions.

Consistency rate over time: (CVE + 5 websites) vs. NVD

Reasons of Inconsistency - 1

- Typos

NVD data / CVE summary

Software	Version
Videolan VLC media player	0.8.6



SecurityFocus

Software	Version
Videolan VLC media player	0.6.8



CVE-2010-0364

Reasons of Inconsistency - 2

- Most reports are seldom updated once created
→ 66.3% of the NVD entries have never been updated



Security Implications - Case Study

- 7 real-world vulnerability, 47 reports, from 5 websites
- 3 security researchers, 185 versions, 4 months' manual verification
- 64 versions are confirmed, 12 newly discovered vulnerable versions

Security Implication - Case Study (cont.)

		Intersection Of 5 Sites	Union Of 5 Sites	Ground truth
Simple Intersection or union cannot solve the problem		1.9.15 (1)	1.9.15 and possibly (40)	1.9.15 (1)
CVE-2008-2950 poppler	Underclaim can leave vulnerable software systems unpatched		(34)	0.5.9 - 0.8.4 (16)
CVE-2009-5018 gif2png	0.99 - 2.5.3 (36)	≤ 2.5.3 (36)	≤ 2.5.3 (36)	2.4.2 - 2.5.6 (13)
CVE-2015-7805 libsndfile	1.0.25 (1)	1.0.25 (1)	1.0.25 (1)	1.0.15 - 1.0.25 (11)
CVE-2016-7445 openjpeg	Overclaim can waste significant manual efforts in reproduction		(1)	1.5 - 2.1.1 (7)
CVE-2016-8676 libav	≤ 11.8 (47)	11.3, 11.4, 11.5, 11.7 (4)	11.3, 11.4, 11.5, 11.7, 11.8, 11.9 (4)	11.0 - 11.8 (9)
CVE-2016-9556 ImageMagick	7.0.3.8 (1)	7.0.3.6	7.0.3.6, 7.0.3.8 (2)	7.0.3.1 - 7.0.3.7 (7)

Conclusion

1. VIEM - an automatic tool to detect inconsistency in Vul. reports
2. A large - scale measurement of the information consistency
3. Case study - validated inconsistent information (and show its impact)

Open Challenges

1. Standardize vulnerability reporting procedure
2. Design a fully automated system to verify the vulnerability reported

Thank you

Code & Data

https://github.com/pinkymm/inconsistency_detection

Presenter: Yueqi (Lewis) Chen

<http://www.personal.psu.edu/yxc431/>