Tufts Security & Privacy Lab

Investigating the Availability of Public Vulnerability Information to Support Patching Decisions

Daniel Votipka
Tufts University

20 Nov 2025

CVE-2024-38368

What Now!?!

Keepers of the Machines: Examining How System Administrators Manage Software Updates

Frank Li
University of California, Berkeley
frankli@cs.berkeley.edu

Lisa Rogers
University of Maryland
lmrogers@umd.edu

Arunesh Mathur
Princeton University
amathur@cs.princeton.edu

Nathan Malkin
University of California, Berkeley
nmalkin@cs.berkeley.edu

Marshini Chetty
Princeton University
marshini@princeton.edu

ABSTRACT

Keeping machines updated is crucial for maintaining system security. While recent studies have investigated the software updating practices of end users, system administrators have received less attention. Yet, system administrators manage numerous machines for their organizations, and security lapses at these hosts can lead to damaging attacks. To improve security at scale, we therefore also need to understand how this specific population behaves and how to help administrators keep machines up-to-date.

In this paper, we study how system administrators manage software updates. We surveyed 102 administrators and interviewed 17 in-depth to understand their processes and how their methods impact updating effectiveness. We find that system administrators proceed through software updates through five main stages that, while similar to those of end users, involve significantly different considerations and actions performed, highlighting the value of focusing specifically on the administrator population. By gathering evidence on how administrators conduct updates, we identify challenges that they encountered and limitations of existing procedures at all stages of the updating process. We observe issues with comprehensively acquiring meaningful information about available updates, effectively testing and deploying updates in a timely manner, recovering from update-induced problems, and interacting with organizational and management influences. Moving forward, we propose directions for future research and community actions that may help system

While prior studies have investigated how end users deal with software updates [18, 19, 22, 30–32, 35, 40, 45, 46, 49, 50], there has been less attention on system administrators, whose technical sophistication and unique responsibilities distinguish them from end users. Industry reports and guides on administrator patching exist (e.g., Sysadmin 101 [41]), but these lack peer-review and transparent rigorous methods. Prior academic work on system administrators is often dated and focuses on aspects of administrator operations other than updating (e.g., on general tools used [11]) or specific technical (rather than user) updating aspects. Given the critical role that system administrators play in protecting an organization's machines, it behooves us to better understand how they manage updates and identify avenues for improved update processes. We therefore set out to answer two primary research questions: (1) what processes do system administrators follow for managing updates, and (2) how do administrator actions impact how effectively they perform system updates. To answer these questions, we surveyed 102 administrators and conducted semi-structured interviews with 17 of them.

Our study determined that system administrators proceed through software updates through five main stages: (1) learning about updates from information sources, (2) deciding to update based on update characteristics, (3) preparing for update installation, (4) deploying an update, and finally (5) handling post-deployment update issues that may arise. By analyzing the factors that system administrators considered and the actions that they performed, we identified

What Now!?!

Keepers of the Machines: Examining How System Administrators Manage Software Updates

Frank Li
University of California, Berkeley
frankli@cs.berkeley.edu

Lisa Rogers
University of Maryland
lmrogers@umd.edu

Arunesh Mathur
Princeton University
amathur@cs.princeton.edu

Nathan Malkin
University of California, Berkeley
nmalkin@cs.berkeley.edu

Marshini Chetty
Princeton University
marshini@princeton.edu

ABSTRACT

Keeping machines updated is crucial for maintaining system security. While recent studies have investigated the software updating practices of end users, system administrators have received less attention. Yet, system administrators manage numerous machines for their organizations, and security lapses at these hosts can lead to damaging attacks. To improve security at scale, we therefore also need to understand how this specific population behaves and how to help administrators keep machines up-to-date.

In this paper, we study how system administrators manage software updates. We surveyed 102 administrators and interviewed 17 in-depth to understand their processes and how their methods impact updating effectiveness. We find that system administrators proceed through software updates through five main stages that, while similar to those of end users, involve significantly different considerations and actions performed, highlighting the value of focusing specifically on the administrator population. By gathering evidence on how administrators conduct updates, we identify challenges that they encountered and limitations of existing procedures at all stages of the updating process. We observe issues with comprehensively acquiring meaningful information about available updates, effectively testing and deploying updates in a timely manner, recovering from update-induced problems, and interacting with organizational and management influences. Moving forward, we propose directions for future research and community actions that may help system

While prior studies have investigated how end users deal with software updates [18, 19, 22, 30–32, 35, 40, 45, 46, 49, 50], there has been less attention on system administrators, whose technical sophistication and unique responsibilities distinguish them from end users. Industry reports and guides on administrator patching exist (e.g., Sysadmin 101 [41]), but these lack peer-review and transparent rigorous methods. Prior academic work on system administrators is often dated and focuses on aspects of administrator operations other than updating (e.g., on general tools used [11]) or specific technical (rather than user) updating aspects. Given the critical role that system administrators play in protecting an organization's machines, it behooves us to better understand how they manage updates and identify avenues for improved update processes. We therefore set out to answer two primary research questions: (1) what processes do system administrators follow for managing updates, and (2) how do administrator actions impact how effectively they perform system updates. To answer these questions, we surveyed 102 administrators and conducted semi-structured interviews with 17 of them.

Our study determined that system administrators proceed through software updates through five main stages: (1) learning about updates from information sources, (2) deciding to update based on update characteristics, (3) preparing for update installation, (4) deploying an update, and finally (5) handling post-deployment update issues that may arise. By analyzing the factors that system administrators considered and the actions that they performed, we identified

Patching questions:

- Am I affected?

What Now!?!

Keepers of the Machines: Examining How System Administrators Manage Software Updates

Frank Li
University of California, Berkeley
frankli@cs.berkeley.edu

Lisa Rogers
University of Maryland
lmrogers@umd.edu

Arunesh Mathur
Princeton University
amathur@cs.princeton.edu

Nathan Malkin
University of California, Berkeley
nmalkin@cs.berkeley.edu

Marshini Chetty
Princeton University
marshini@princeton.edu

ABSTRACT

Keeping machines updated is crucial for maintaining system security. While recent studies have investigated the software updating practices of end users, system administrators have received less attention. Yet, system administrators manage numerous machines for their organizations, and security lapses at these hosts can lead to damaging attacks. To improve security at scale, we therefore also need to understand how this specific population behaves and how to help administrators keep machines up-to-date.

In this paper, we study how system administrators manage software updates. We surveyed 102 administrators and interviewed 17 in-depth to understand their processes and how their methods impact updating effectiveness. We find that system administrators proceed through software updates through five main stages that, while similar to those of end users, involve significantly different considerations and actions performed, highlighting the value of focusing specifically on the administrator population. By gathering evidence on how administrators conduct updates, we identify challenges that they encountered and limitations of existing procedures at all stages of the updating process. We observe issues with comprehensively acquiring meaningful information about available updates, effectively testing and deploying updates in a timely manner, recovering from update-induced problems, and interacting with organizational and management influences. Moving forward, we propose directions for future research and community actions that may help system

While prior studies have investigated how end users deal with software updates [18, 19, 22, 30–32, 35, 40, 45, 46, 49, 50], there has been less attention on system administrators, whose technical sophistication and unique responsibilities distinguish them from end users. Industry reports and guides on administrator patching exist (e.g., Sysadmin 101 [41]), but these lack peer-review and transparent rigorous methods. Prior academic work on system administrators is often dated and focuses on aspects of administrator operations other than updating (e.g., on general tools used [11]) or specific technical (rather than user) updating aspects. Given the critical role that system administrators play in protecting an organization's machines, it behooves us to better understand how they manage updates and identify avenues for improved update processes. We therefore set out to answer two primary research questions: (1) what processes do system administrators follow for managing updates, and (2) how do administrator actions impact how effectively they perform system updates. To answer these questions, we surveyed 102 administrators and conducted semi-structured interviews with 17 of them.

Our study determined that system administrators proceed through software updates through five main stages: (1) learning about updates from information sources, (2) deciding to update based on update characteristics, (3) preparing for update installation, (4) deploying an update, and finally (5) handling post-deployment update issues that may arise. By analyzing the factors that system administrators considered and the actions that they performed, we identified

Patching questions:

- Am I affected?
- What could go wrong?

What Now!?!

Keepers of the Machines: Examining How System Administrators Manage Software Updates

Frank Li
University of California, Berkeley
frankli@cs.berkeley.edu

Lisa Rogers
University of Maryland
lmrogers@umd.edu

Arunesh Mathur
Princeton University
amathur@cs.princeton.edu

Nathan Malkin
University of California, Berkeley
nmalkin@cs.berkeley.edu

Marshini Chetty
Princeton University
marshini@princeton.edu

ABSTRACT

Keeping machines updated is crucial for maintaining system security. While recent studies have investigated the software updating practices of end users, system administrators have received less attention. Yet, system administrators manage numerous machines for their organizations, and security lapses at these hosts can lead to damaging attacks. To improve security at scale, we therefore also need to understand how this specific population behaves and how to help administrators keep machines up-to-date.

In this paper, we study how system administrators manage software updates. We surveyed 102 administrators and interviewed 17 in-depth to understand their processes and how their methods impact updating effectiveness. We find that system administrators proceed through software updates through five main stages that, while similar to those of end users, involve significantly different considerations and actions performed, highlighting the value of focusing specifically on the administrator population. By gathering evidence on how administrators conduct updates, we identify challenges that they encountered and limitations of existing procedures at all stages of the updating process. We observe issues with comprehensively acquiring meaningful information about available updates, effectively testing and deploying updates in a timely manner, recovering from update-induced problems, and interacting with organizational and management influences. Moving forward, we propose directions for future research and community actions that may help system

While prior studies have investigated how end users deal with software updates [18, 19, 22, 30–32, 35, 40, 45, 46, 49, 50], there has been less attention on system administrators, whose technical sophistication and unique responsibilities distinguish them from end users. Industry reports and guides on administrator patching exist (e.g., Sysadmin 101 [41]), but these lack peer-review and transparent rigorous methods. Prior academic work on system administrators is often dated and focuses on aspects of administrator operations other than updating (e.g., on general tools used [11]) or specific technical (rather than user) updating aspects. Given the critical role that system administrators play in protecting an organization's machines, it behooves us to better understand how they manage updates and identify avenues for improved update processes. We therefore set out to answer two primary research questions: (1) what processes do system administrators follow for managing updates, and (2) how do administrator actions impact how effectively they perform system updates. To answer these questions, we surveyed 102 administrators and conducted semi-structured interviews with 17 of them.

Our study determined that system administrators proceed through software updates through five main stages: (1) learning about updates from information sources, (2) deciding to update based on update characteristics, (3) preparing for update installation, (4) deploying an update, and finally (5) handling post-deployment update issues that may arise. By analyzing the factors that system administrators considered and the actions that they performed, we identified

Patching questions:

- Am I affected?
- What could go wrong?
- What can I do?

What Now!?!

Keepers of the Machines: Examining How System Administrators Manage Software Updates

Frank Li
University of California, Berkeley
frankli@cs.berkeley.edu

Lisa Rogers
University of Maryland
lmrogers@umd.edu

Arunesh Mathur
Princeton University
amathur@cs.princeton.edu

Nathan Malkin
University of California, Berkeley
nmalkin@cs.berkeley.edu

Marshini Chetty
Princeton University
marshini@princeton.edu

ABSTRACT

Keeping machines updated is crucial for maintaining system security. While recent studies have investigated the software updating practices of end users, system administrators have received less attention. Yet, system administrators manage numerous machines for their organizations, and security lapses at these hosts can lead to damaging attacks. To improve security at scale, we therefore also need to understand how this specific population behaves and how to help administrators keep machines up-to-date.

In this paper, we study how system administrators manage software updates. We surveyed 102 administrators and interviewed 17 in-depth to understand their processes and how their methods impact updating effectiveness. We find that system administrators proceed through software updates through five main stages that, while similar to those of end users, involve significantly different considerations and actions performed, highlighting the value of focusing specifically on the administrator population. By gathering evidence on how administrators conduct updates, we identify challenges that they encountered and limitations of existing procedures at all stages of the updating process. We observe issues with comprehensively acquiring meaningful information about available updates, effectively testing and deploying updates in a timely manner, recovering from update-induced problems, and interacting with organizational and management influences. Moving forward, we propose directions for future research and community actions that may help system

While prior studies have investigated how end users deal with software updates [18, 19, 22, 30–32, 35, 40, 45, 46, 49, 50], there has been less attention on system administrators, whose technical sophistication and unique responsibilities distinguish them from end users. Industry reports and guides on administrator patching exist (e.g., Sysadmin 101 [41]), but these lack peer-review and transparent rigorous methods. Prior academic work on system administrators is often dated and focuses on aspects of administrator operations other than updating (e.g., on general tools used [11]) or specific technical (rather than user) updating aspects. Given the critical role that system administrators play in protecting an organization's machines, it behooves us to better understand how they manage updates and identify avenues for improved update processes. We therefore set out to answer two primary research questions: (1) what processes do system administrators follow for managing updates, and (2) how do administrator actions impact how effectively they perform system updates. To answer these questions, we surveyed 102 administrators and conducted semi-structured interviews with 17 of them.

Our study determined that system administrators proceed through software updates through five main stages: (1) learning about updates from information sources, (2) deciding to update based on update characteristics, (3) preparing for update installation, (4) deploying an update, and finally (5) handling post-deployment update issues that may arise. By analyzing the factors that system administrators considered and the actions that they performed, we identified

Patching questions:

- Am I affected?
- What could go wrong?
- What can I do?
- What could go wrong if I patch?

What Now!?!

Keepers of the Machines: Examining How System Administrators Manage Software Updates

Frank Li
University of California, Berkeley
frankli@cs.berkeley.edu

Lisa Rogers
University of Maryland
lmrogers@umd.edu

Arunesh Mathur
Princeton University
amathur@cs.princeton.edu

Nathan Malkin
University of California, Berkeley
nmalkin@cs.berkeley.edu

Marshini Chetty
Princeton University
marshini@princeton.edu

ABSTRACT

Keeping machines updated is crucial for maintaining system security. While recent studies have investigated the software updating practices of end users, system administrators have received less attention. Yet, system administrators manage numerous machines for their organizations, and security lapses at these hosts can lead to damaging attacks. To improve security at scale, we therefore also need to understand how this specific population behaves and how to help administrators keep machines up-to-date.

In this paper, we study how system administrators manage software updates. We surveyed 102 administrators and interviewed 17 in-depth to understand their processes and how their methods impact updating effectiveness. We find that system administrators proceed through software updates through five main stages that, while similar to those of end users, involve significantly different considerations and actions performed, highlighting the value of focusing specifically on the administrator population. By gathering evidence on how administrators conduct updates, we identify challenges that they encountered and limitations of existing procedures at all stages of the updating process. We observe issues with comprehensively acquiring meaningful information about available updates, effectively testing and deploying updates in a timely manner, recovering from update-induced problems, and interacting with organizational and management influences. Moving forward, we propose directions for future research and community actions that may help system

While prior studies have investigated how end users deal with software updates [18, 19, 22, 30–32, 35, 40, 45, 46, 49, 50], there has been less attention on system administrators, whose technical sophistication and unique responsibilities distinguish them from end users. Industry reports and guides on administrator patching exist (e.g., Sysadmin 101 [41]), but these lack peer-review and transparent rigorous methods. Prior academic work on system administrators is often dated and focuses on aspects of administrator operations other than updating (e.g., on general tools used [11]) or specific technical (rather than user) updating aspects. Given the critical role that system administrators play in protecting an organization's machines, it behooves us to better understand how they manage updates and identify avenues for improved update processes. We therefore set out to answer two primary research questions: (1) what processes do system administrators follow for managing updates, and (2) how do administrator actions impact how effectively they perform system updates. To answer these questions, we surveyed 102 administrators and conducted semi-structured interviews with 17 of them.

Our study determined that system administrators proceed through software updates through five main stages: (1) learning about updates from information sources, (2) deciding to update based on update characteristics, (3) preparing for update installation, (4) deploying an update, and finally (5) handling post-deployment update issues that may arise. By analyzing the factors that system administrators considered and the actions that they performed, we identified

Patching questions:

- Am I affected?
- What could go wrong?
- What can I do?
- What could go wrong if I patch?

Where do we get the answers?

CVE-2024-38368 Detail

MODIFIED

This CVE record has been updated after NVD enrichment efforts were completed. Enrichment data supplied by the NVD may require amendment due to these changes.

Description

trunk.cocoapods.org is the authentication server for the CoacoaPods dependency manager. A vulnerability affected older pods which migrated from the pre-2014 pull request workflow to trunk. If the pods had never been claimed then it was still possible to do so. It was also possible to have all owners removed from a pod, and that made the pod available for the same claiming system. This was patched server-side in commit 71be5440906b6bdfbc0bcc7f8a9fec33367ea0f4 in September 2023.

QUICK INFO

CVE Dictionary Entry:
CVE-2024-38368
NVD Published Date:
07/01/2024
NVD Last Modified:
11/21/2024
Source:
GitHub, Inc.

CVE-2024-38368 Detail

MODIFIED

This CVE record has been updated after NVD enrichment efforts were completed. Enrichment data supplied by the NVD may require amendment due to these changes.

QUICK INFO

CVE Dictionary Entry:
CVE-2024-38368
NVD Published Date:
07/01/2024

trunk.cocoapods.org is the authentication server for the CoacoaPods dependency manager. A vulnerability affected older pods which migrated from the pre-2014 pull request workflow to trunk. If the pods had never been claimed then it was still possible to do so. It was also possible to have all owners removed from a pod, and that made the pod available for the same claiming system. This was patched server-side in commit 71be5440906b6bdfbc0bcc7f8a9fec33367ea0f4 in September 2023.

CVE-2024-38368 Detail

MODIFIED

This CVE record has been updated after NVD enrichment efforts were completed. Enrichment data supplied by the NVD may require amendment due to these changes.

QUICK INFO

CVE Dictionary Entry:
CVE-2024-38368
NVD Published Date:
07/01/2024

trunk.cocoapods.org is the authentication server for the CoacoaPods dependency manager. A vulnerability affected older pods which migrated from the pre-2014 pull request workflow to trunk. If the pods had never been claimed then it was still possible to do so. It was also possible to have all owners removed from a pod, and that made the pod available for the same claiming system. This was patched server-side in commit 71be5440906b6bdfbc0bcc7f8a9fec33367ea0f4 in September 2023.

Known Affected Software Configurations Switch to CPE 2.2

Configuration 1 (hide)

cpe:2.3:a:cocoapods:trunk.cocoapods.org:*:*:*:*:ruby:*:*:*	Up to (excluding)
Show Matching CPE(s)	2023-09-22

Description

trunk.cocoapods.org is the authentication server for the CoacoaPods dependency manager. A vulnerability affected older pods which migrated from the pre-2014 pull request workflow to trunk. If the pods had never been claimed then it was still possible to do so. It was also possible to have all owners removed from a pod, and that made the pod available for the same claiming system. This was patched server-side in commit 71be5440906b6bdfbc0bcc7f8a9fec33367ea0f4 in September 2023.

07/01/2024

NVD Last Modified:

11/21/2024

Source:

GitHub, Inc.

Metrics

CVSS Version 4.0 | CVSS Version 3.x | CVSS Version 2.0

NVD enrichment efforts reference publicly available information to associate vector strings. CVSS information contributed by other sources is also displayed.

CVSS 3.x Severity and Vector Strings:

CNA: GitHub, Inc. **Base Score:** 9.3 CRITICAL **Vector:** CVSS:3.1/AV:N/AC:L/PR:N/UI:N/S:C/C:N/I:H/A:L

MODIFIED

This CVE record has been updated after NVD enrichment efforts were completed. Enrichment data supplied by the NVD may require amendment due to these changes.

Description

trunk.cocoapods.org is the authentication server for the CoacoaPods dependency manager. A vulnerability affected older pods which migrated from the pre-2014 pull request workflow to trunk. If the pods had never been claimed then it was still possible to do so. It was also possible to have all owners removed from a pod, and that made the pod available for the same claiming system. This was patched server-side in commit 71be5440906b6bdfbc0bcc7f8a9fec33367ea0f4 in September 2023.

QUICK INFO

CVE Dictionary Entry:
CVE-2024-38368
NVD Published Date:
07/01/2024
NVD Last Modified:
11/21/2024
Source:
GitHub, Inc.

CVE-2024-38368 Detail

MODIFIED

This CVE record has been updated after NVD enrichment efforts were completed. Enrichment data supplied by the NVD may require amendment due to these changes.

QUICK INFO

CVE Dictionary Entry:
CVE-2024-38368
NVD Published Date:
07/01/2024

trunk.cocoapods.org is the authentication server for the CoacoaPods dependency manager. A vulnerability affected older pods which migrated from the pre-2014 pull request workflow to trunk. If the pods had never been claimed then it was still possible to do so. It was also possible to have all owners removed from a pod, and that made the pod available for the same claiming system. This was patched server-side in commit 71be5440906b6bdfbc0bcc7f8a9fec33367ea0f4 in September 2023.

CVE-2024-38368 Detail

MODIFIED

This CVE record has been updated after NVD enrichment efforts were completed. Enrichment data supplied by the NVD may require amendment due to these changes.

Description

trunk.cocoapods.org is the authentication server for the CoacoaPods dependency manager. A vulnerability affected older pods which migrated from the pre-2014 pull request workflow to trunk. If the pods had never been claimed then it was still possible to do so. It was also possible to have all owners removed from a pod, and that made the pod available for the same claiming system. This was patched server-side in commit 71be5440906b6bdfbc0bcc7f8a9fec33367ea0f4 in September 2023.

QUICK INFO

CVE Dictionary Entry:
CVE-2024-38368
NVD Published Date:
07/01/2024
NVD Last Modified:
11/21/2024
Source:
GitHub, Inc.

URL	Source(s)	Tag(s)
https://blog.cocoapods.org/Claim-Your-Pods	CVE, GitHub, Inc.	Product
https://blog.cocoapods.org/CocoaPods-Trunk-RCEs-2023	CVE, GitHub, Inc.	Vendor Advisory
https://evasec.webflow.io/blog/eva-discovered-supply-chain-vulnerabities-in-cocoapods#1-taking-unauthorized-ownership-over-orphaned-pods	CVE, GitHub, Inc.	Third Party Advisory
https://github.com/CocoaPods/CocoaPods/security/advisories/GHSA-j483-qm5c-7hqx	CVE, GitHub, Inc.	Third Party Advisory
https://github.com/CocoaPods/trunk.cocoapods.org/commit/71be5440906b6bdfbc0bcc7f8a9fec33367ea0f4	CVE, GitHub, Inc.	Patch

in commit 71be5440906b6bdfbc0bcc7f8a9fec33367ea0f4 in September 2023.

"CVE-2024-38368"

AI Mode | All | Shopping | News | Images | Videos | Maps | More | Tools

National Institute of Standards and Technology (.gov)
https://nvd.nist.gov › vuln › detail › cve-2024-38368

CVE-2024-38368 Detail - NVD

Jul 1, 2024 — **CVE-2024-38368** Detail. Modified. This CVE record has been updated after NVD enrichment efforts were completed. Enrichment data supplied by the ...

GitGuardian Blog
https://blog.gitguardian.com › cve-of-the-month-the-su...

CVE of the month, the supply chain vulnerability hidden ...

Jul 3, 2024 — **CVE-2024-38368** is a vulnerability that affects the open-source supply chain of iOS and MacOS applications.

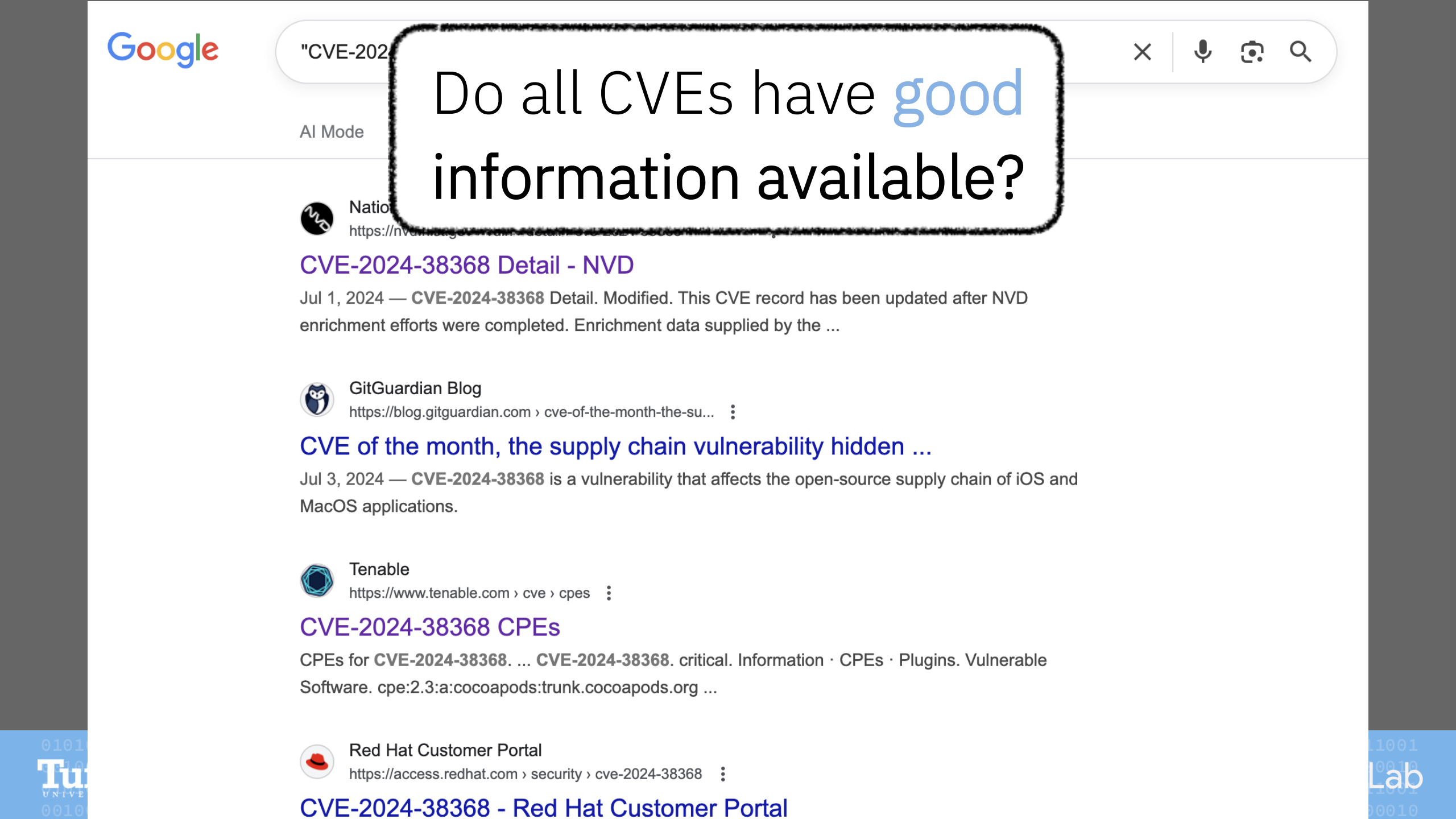
Tenable
https://www.tenable.com › cve › cpes

CVE-2024-38368 CPEs

CPEs for **CVE-2024-38368**. ... **CVE-2024-38368**. critical. Information · CPEs · Plugins. Vulnerable Software. cpe:2.3:a:cocoapods:trunk.cocoapods.org ...

Red Hat Customer Portal
https://access.redhat.com › security › cve-2024-38368

CVE-2024-38368 - Red Hat Customer Portal


Google
"CVE-202
AI Mode
Do all CVEs have good information available?
Natio
https://nv
CVE-2024-38368 Detail - NVD
Jul 1, 2024 — CVE-2024-38368 Detail. Modified. This CVE record has been updated after NVD enrichment efforts were completed. Enrichment data supplied by the ...
GitGuardian Blog
https://blog.gitguardian.com › cve-of-the-month-the-su...
CVE of the month, the supply chain vulnerability hidden ...
Jul 3, 2024 — CVE-2024-38368 is a vulnerability that affects the open-source supply chain of iOS and MacOS applications.
Tenable
https://www.tenable.com › cve › cpes
CVE-2024-38368 CPEs
CPEs for CVE-2024-38368. ... CVE-2024-38368. critical. Information · CPEs · Plugins. Vulnerable Software. cpe:2.3:a:cocoapods:trunk.cocoapods.org ...
Red Hat Customer Portal
https://access.redhat.com › security › cve-2024-38368
CVE-2024-38368 - Red Hat Customer Portal

Google "CVE-202

AI Mode

Natio
https://nv

CV
Jul 1
enric

CV
Jul 3
Mac

CV
CPE
Soft

CV

Do all CVEs have good information available?

No One Drinks From the Firehose: How Organizations Filter and Prioritize Vulnerability Information

Stephanie de Smale[1,2], Rik van Dijk[1], Xander Bouwman[2], Jeroen van der Ham[1,3], and Michel van Eeten[2]

[1]National Cyber Security Centre, The Netherlands
[2]Delft University of Technology
[3]University of Twente

acy (SP) | 978-1-6654-9336-9/23/$31.00 ©2023 IEEE | DOI: 10.1109/SP46215.2023.10179447

***Abstract*—The number of published software vulnerabilities is increasing every year. How do organizations stay in control of their attack surface despite their limited staff resources? Prior work has analyzed the overall software vulnerability ecosystem as well as patching processes within organizations, but not how these two are connected.**

We investigate this missing link through semi-structured interviews with 22 organizations in critical infrastructure and government services. We analyze where in these organizations the responsibility is allocated to collect and triage information about software vulnerabilities, and find that none of our respondents is acquiring such information comprehensively, not even in a reduced and aggregated form like the National Vulnerability Database (NVD). This means that information on known vulnerabilities will be missed, even in critical infrastructure organizations. We observe that organizations apply implicit and explicit coping mechanisms to reduce their intake of vulnerability information, and identify three trade-offs in these strategies: independence, pro-activeness and formalization.

Although our respondents' behavior is in conflict with the

Against this backdrop, security experts urge organizations to mitigate the known vulnerabilities in their infrastructure. Public vulnerability disclosures—e.g., via vendor announcements, security websites, or vulnerability databases—are the necessary first step for the vulnerability management process [5]. Organizations might also discover vulnerabilities in their infrastructure in other ways, like pen testing and red teaming [6], but these are complementary and not a replacement for a continuous process of acquiring and evaluating information on newly published vulnerabilities.

The advice to mitigate known vulnerabilities sounds straightforward, until one takes the scale of the problem into account. The number of published vulnerabilities grows every year. For 2021, VulnDB reported over 29,000 vulnerabilities [7]. Furthermore, information remains dispersed. While platforms like VulnDB aggregate published vulnerabilities, none of the platforms are complete or always timely [8].

How do organizations cope with the flood of vulnerability information? In a year with at least 29,000 vulnerabilities, just assessing if a vulnerability is at all relevant for their enterprise infrastructure would already require over 500

Tu UNIVE

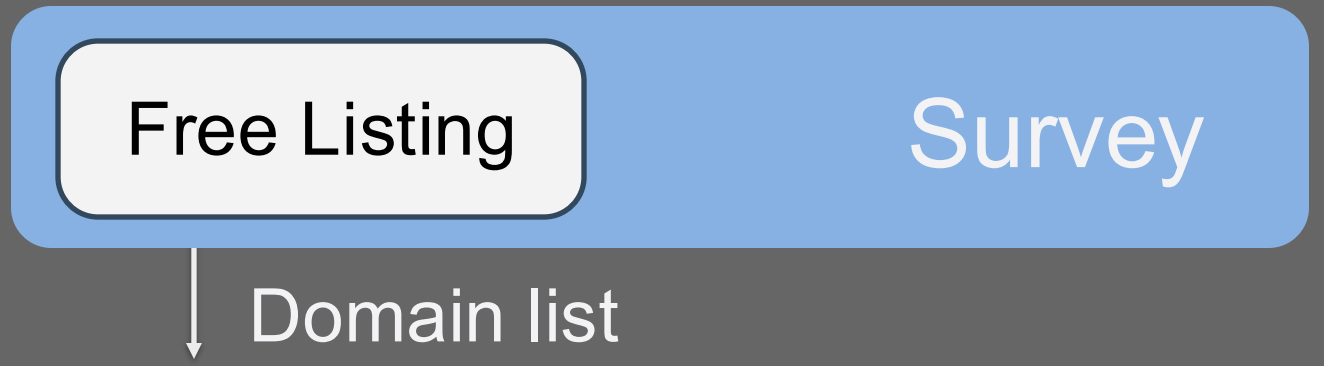
Lab

Research Questions

Where do practitioners get vulnerability information?

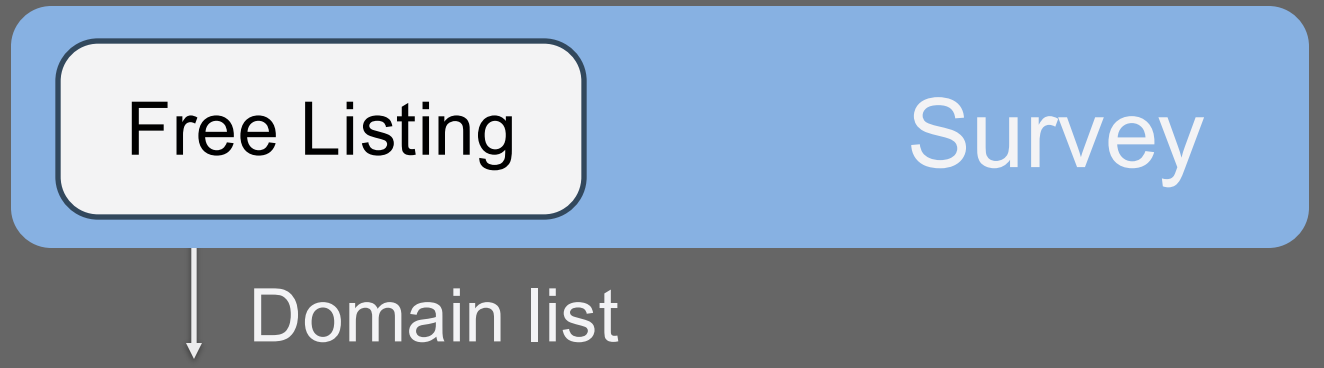
Research Questions

Where do practitioners get vulnerability information?



Research Questions

Where do practitioners get vulnerability information?

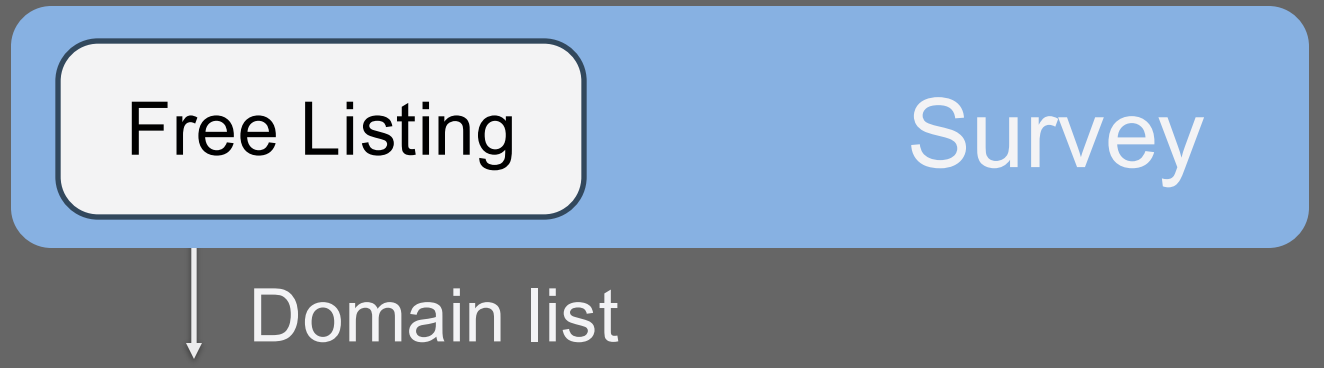


27 participants; reported
61 unique domains

Research Questions

Where do practitioners get vulnerability information?

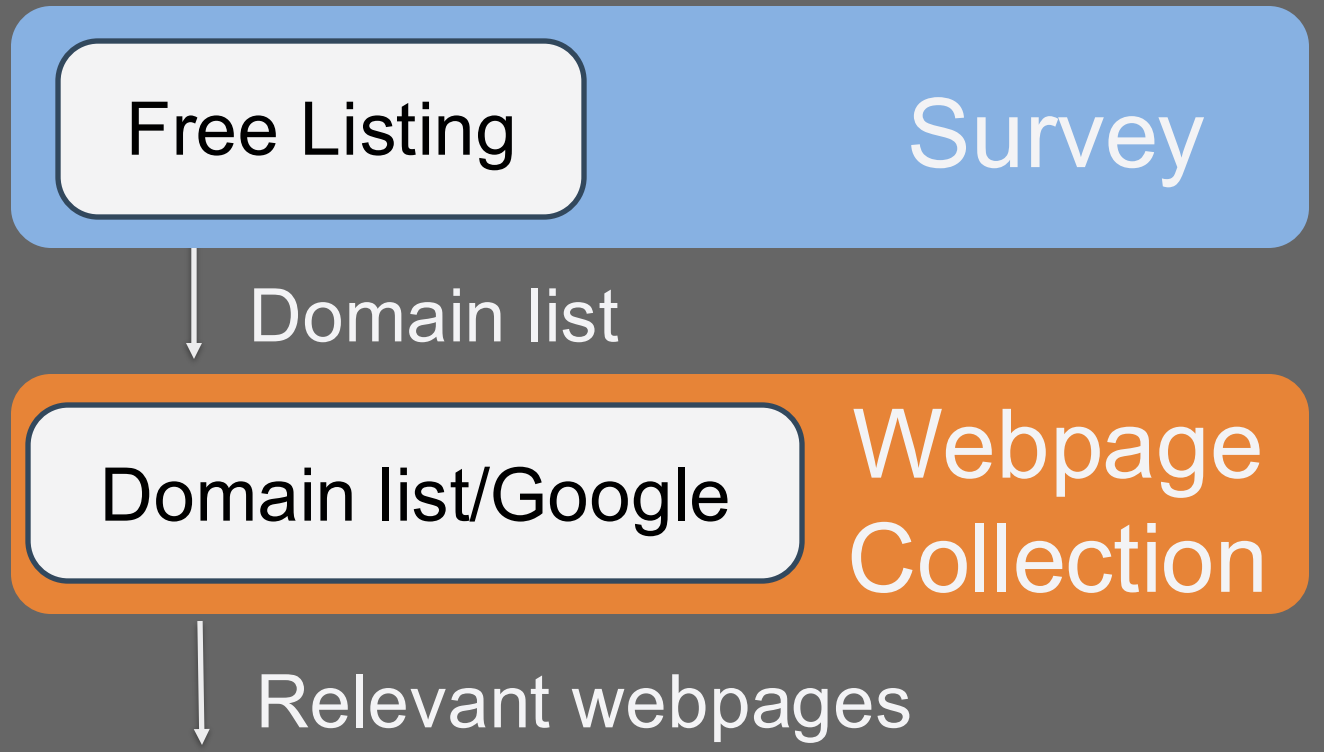
How robust is the vulnerability information ecosystem?



Research Questions

Where do practitioners get vulnerability information?

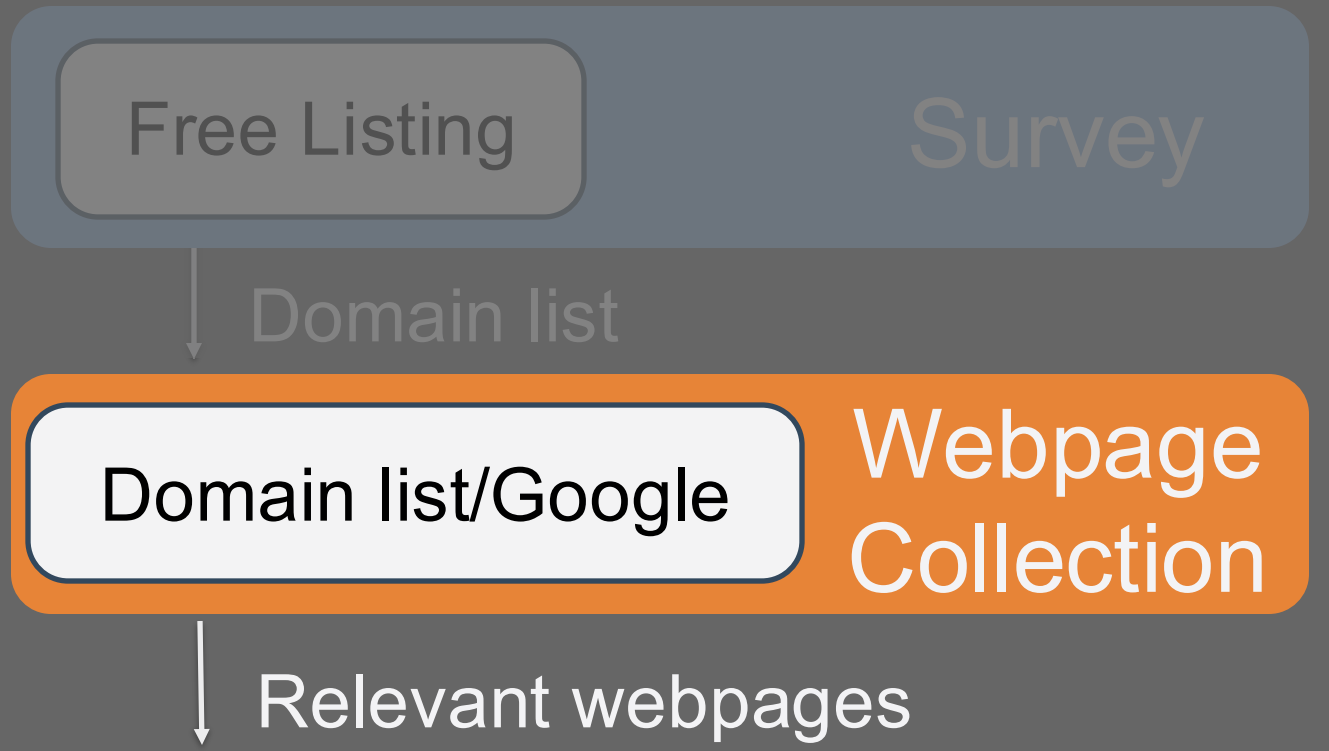
How robust is the vulnerability information ecosystem?



Research Questions

Where do practitioners get vulnerability information?

How robust is the vulnerability information ecosystem?

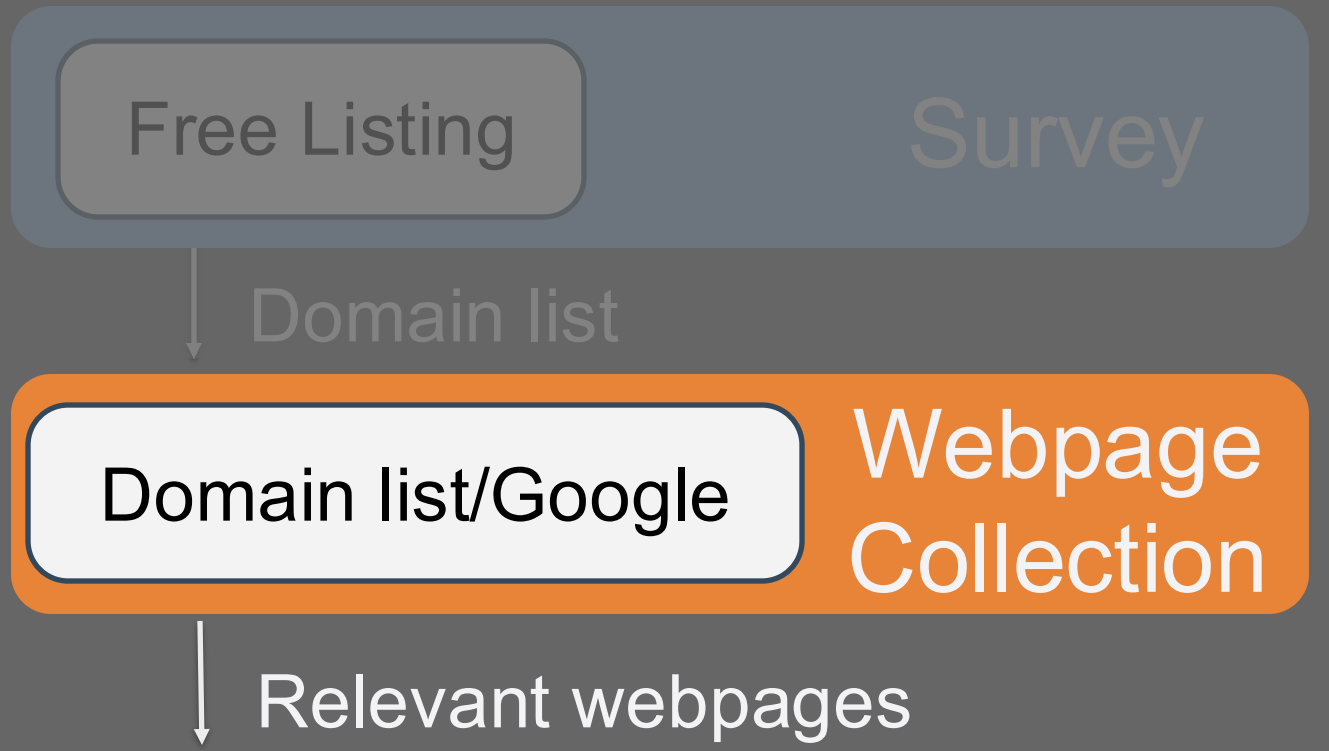


~19K CVEs from July 2022 to Mar 2023

Research Questions

Where do practitioners get vulnerability information?

How robust is the vulnerability information ecosystem?



~19K CVEs from July 2022 to Mar 2023

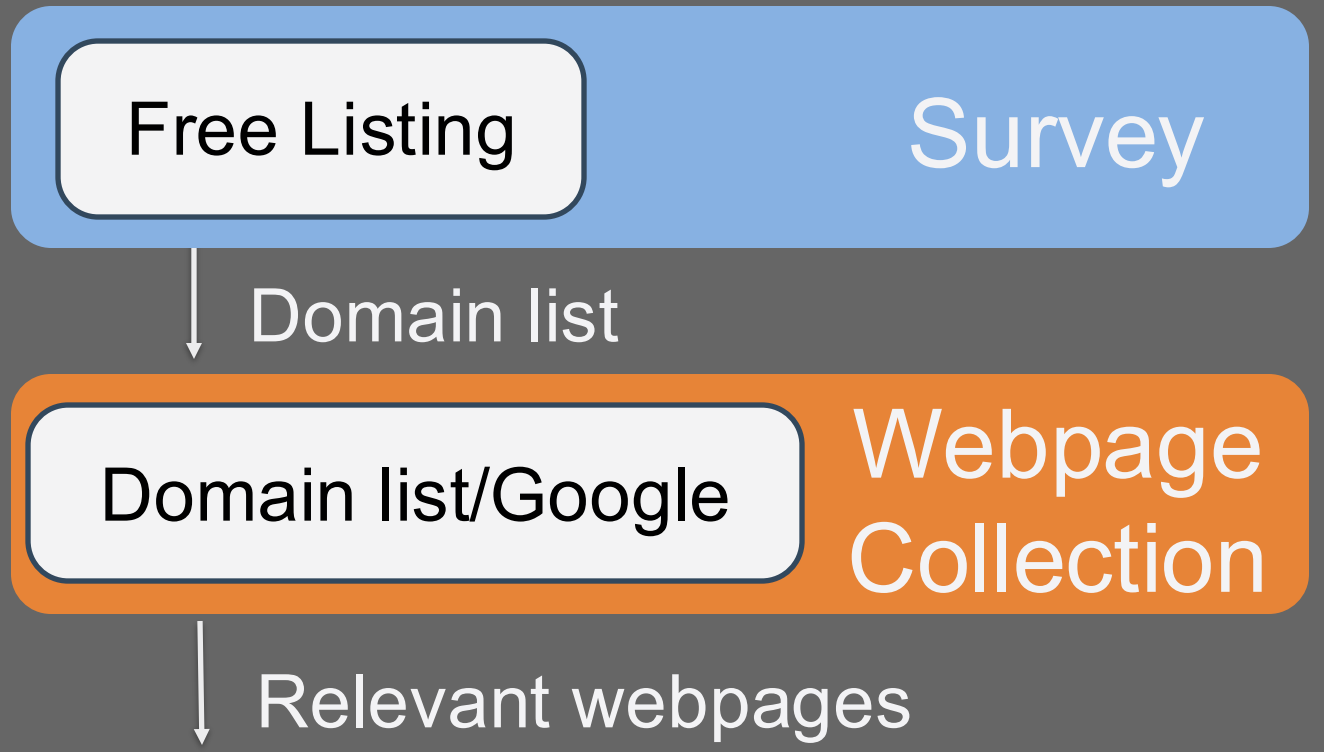
~214K webpages; 10.8 pages a CVE on average

Research Questions

Where do practitioners get vulnerability information?

How robust is the vulnerability information ecosystem?

Does the vulnerability ecosystem answer practitioners' questions?

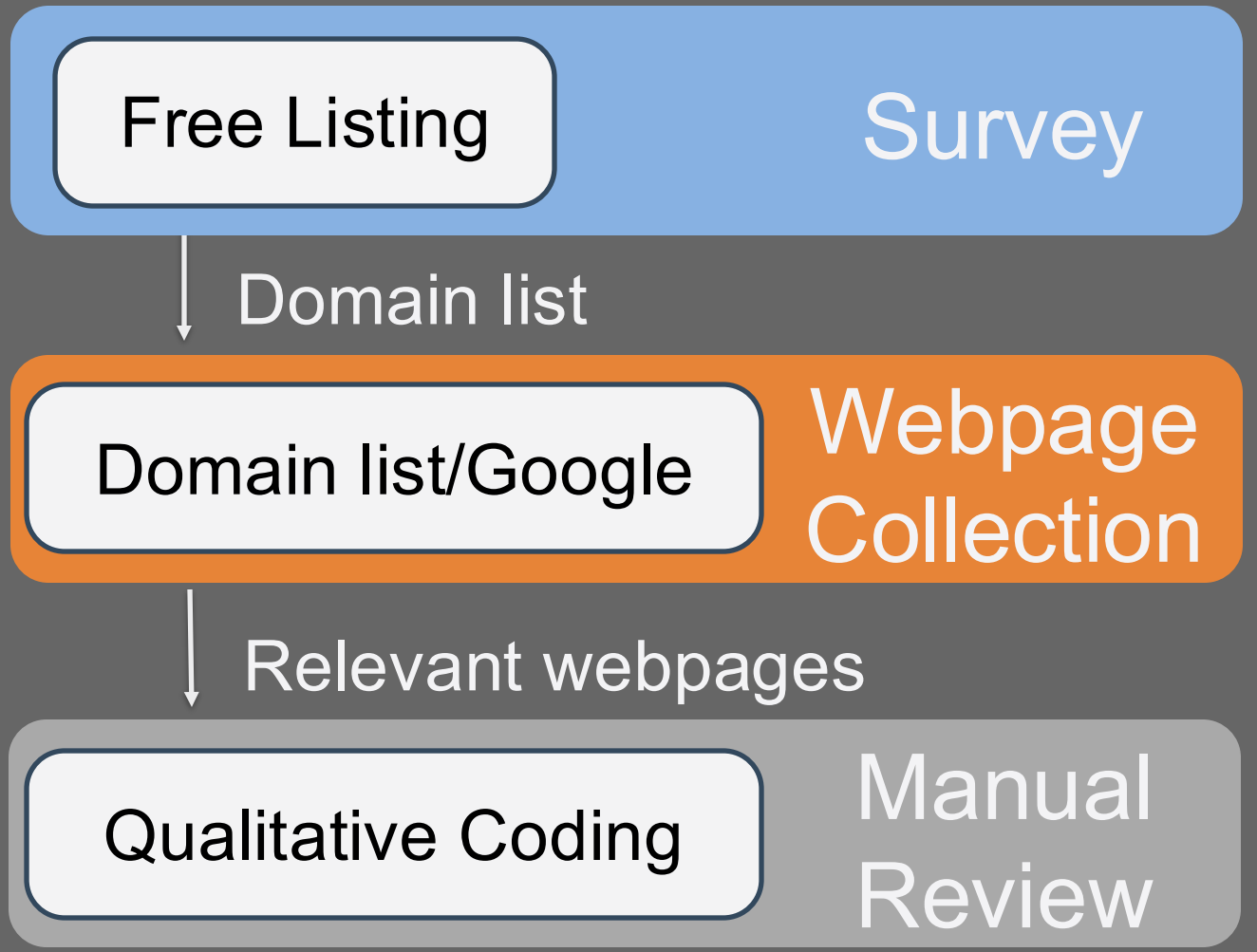


Research Questions

Where do practitioners get vulnerability information?

How robust is the vulnerability information ecosystem?

Does the vulnerability ecosystem answer practitioners' questions?



Research Questions

Where do practitioners [illegible] vulnerability informa[illegible]

How robust is the vul[illegible] information ecosyste[illegible]

Does the vulnerability ecosystem answer practitioners' questions?

Free Listing

Survey

80 random CVEs;
682 webpages

gle

Webpage Collection

Relevant webpages

Qualitative Coding

Manual Review

Am I affected?

Question for patching decisions	Yes	No
Which systems are vulnerable?	97%	3%

Am I affected?

Question for patching decisions	Yes	No
Which systems are vulnerable?	97%	3%
Which versions are vulnerable?	78%	22%
In what context is it vulnerable?	74%	26%

Am I affected?

r/mikrotik • 3y ago
djdrastic

CVE-2022-45313

Checking if anyone has seen anything from Mikrotik backporting fixes from 7.x for this issue into 6.XX chain ?

https://github.com/cq674350529/pocs_slides/tree/master/advisory/MikroTik/CVE-2022-45313
https://github.com/cq674350529/pocs_slides/tree/master/advisory/MikroTik/CVE-2022-45315

What could go wrong?

Question for patching decisions	Yes	No
CVSS score or severity included?	65%	35%

What could go wrong?

Question for patching decisions	Yes	No
CVSS score or severity included?	65%	35%

Metrics CVSS Version 4.0 | CVSS Version 3.x | CVSS Version 2.0

NVD enrichment efforts reference publicly available information to associate vector strings. CVSS information contributed by other sources is also displayed.

CVSS 3.x Severity and Vector Strings:

CNA: GitHub, Inc. **Base Score:** 9.3 CRITICAL **Vector:** CVSS:3.1/AV:N/AC:L/PR:N/UI:N/S:C/C:N/I:H/A:L

What could go wrong?

Question for patching decisions	Yes	No
CVSS score or severity included?	65%	35%

Navigating the Patchwork: Investigating the Availability & Consistency of Security Advisories

Ronald E. Thompson III
Tufts University
Medford, MA, USA
rthomp06@cs.tufts.edu

Luke Boshar
Tufts University
Medford, MA, USA
luke.boshar@tufts.edu

Eugene Y. Vasserman
Kansas State University
Manhattan, KS, USA
eyv@ksu.edu

Daniel Votipka
Tufts University
Medford, MA, USA
dvotipka@cs.tufts.edu

***Abstract*—Prioritizing software patches requires accessible and consistent vulnerability information, but collecting this information through the current ecosystem of disparate organizations presents significant challenges for system administrators. This paper reports on the experience of systematically analyzing the security advisory practices of 718 organizations, including certified reporting bodies and vendors involved in critical infrastructure, detailing the hurdles faced in data collection and interpretation. Our findings show the disparities in public advisory availability across organization types and the lack of widespread usage of machine-readable formats, hindering automated processing. Additionally, while CVSS has been adopted across the ecosystem as the standard for severity scoring, in practice, its application suffers from inconsistencies in reporting completeness, versioning, and transparency, limiting practical utility for system**

ments typically use the Common Vulnerability Scoring System (CVSS) [9], [15], [38], [50]. CVSS is a metric developed and maintained by the Forum of Incident Response and Security Teams (FIRST) to evaluate the severity of vulnerability by rating it across several features, assessing the ease and technical means of exploitation and impact if exploited, and finally producing a value between zero and ten [20], [38].

Prior work has shown sysadmins struggle to process all relevant information from these advisories, including determining which stakeholders' advisories are likely to be reliable [18], [24], [30], [35], [54]. Additionally, research has shown some stakeholders can be inconsistent in what information they

What could go wrong?

Question for patching decisions	Yes	No
CVSS score or severity included?	65%	35%

Navigating the Patchwork: Investigating the Availability & Consistency of Security Advisories

Ronald E. Thompson III
Tufts University
Medford, MA, USA

Luke Boshar
Tufts University
Medford, MA, USA

Eugene Y. Vasserman
Kansas State University
Manhattan, KS, USA

Daniel Votipka
Tufts University
Medford, MA, USA

15% of vendors did not indicate CVSS version

across organization types and the lack of widespread usage of machine-readable formats, hindering automated processing. Additionally, while CVSS has been adopted across the ecosystem as the standard for severity scoring, in practice, its application suffers from inconsistencies in reporting completeness, versioning, and transparency, limiting practical utility for system

vant information from these advisories, including determining which stakeholders' advisories are likely to be reliable [18], [24], [30], [35], [54]. Additionally, research has shown some stakeholders can be inconsistent in what information they

What could go wrong?

Question for patching decisions	Yes	No
CVSS score or severity included?	65%	35%
Does it say what an attacker can do?	60%	40%

What could go wrong?

GitGuardian BLOG CVE of the month, the supply chain vulnerability hidden for 10 years CVE-2024-

What does the vulnerability allow?

The EvaSec research team called this an ***unauthorized account ownership*** but in essence, it is an ***account takeover*** vulnerability. It allows a malicious actor to completely take over a '*Pod*' or a package in specific circumstances. This would then allow the attacker to release updates to the pod adding malware so that when tools upgrade to the latest version would be pulling malware directly into their applications. **In essence, this vulnerability has the potential to turn thousands (perhaps millions) of applications malicious. The blast radius is huge.**

What could go wrong?

Question for patching decisions	Yes	No
CVSS score or severity included?	65%	35%
Does it say what an attacker can do?	60%	40%
Does it say if exploitable remotely?	47%	52%
Does it say if exploit is public?	15%	85%
Does it say if actively being exploited?	7%	93%

What can I do?

Question for patching decisions	Yes	No
Does it say if a patch is available?	64%	36%

What can I do?

Question for patching decisions	Yes	No
Does it say if a patch is available?	64%	36%
Are non-patch mitigations included?	5%	95%

Wha

Technical Remediation Steps

- Keep your podfile.lock file synchronized with all CocoaPods developers to ensure everyone is on the same version of the packages. This will ensure that when a new, potentially harmful update is committed, developers will not automatically update to it.
- If you are using a Pod which is developed internally and only hosted in CocoaPods for mass distribution, developers should perform CRC (checksum) validation against the one downloaded from the CocoaPods trunk server to ensure it's the same as the one developed internally (where possible).
- Implement a thorough security review of any third party code used in your applications.
- Review CocoaPods dependencies and verify you are not using an orphaned Pod.
- Ensure you use third party dependencies that are actively maintained and whose ownership is clear.

...

Qu

No

6%

5%

What can go wrong if I patch?

Question for patching decisions	Yes	No
What could go wrong if I patch/mitigate?	2%	98%

What can go wrong if I patch?

r/sysadmin • 3y ago
AutoModerator

Patch Tuesday Megathread (2022-10-11)

General Discussion

Hello r/sysadmin, I'm u/AutoModerator, and welcome to this month's **Patch Megathread!**

This is the (*mostly*) safe location to talk about the latest patches, updates, and releases. We put this thread into place to help gather all the information about this month's updates: What is fixed, what broke, what got released and should have been caught in QA, etc. We do this both to keep clutter out of the subreddit, and provide you, the dear reader, a singular resource to read.

What can go wrong if I patch?

r/sysadmin • 3y ago
AutoModerator

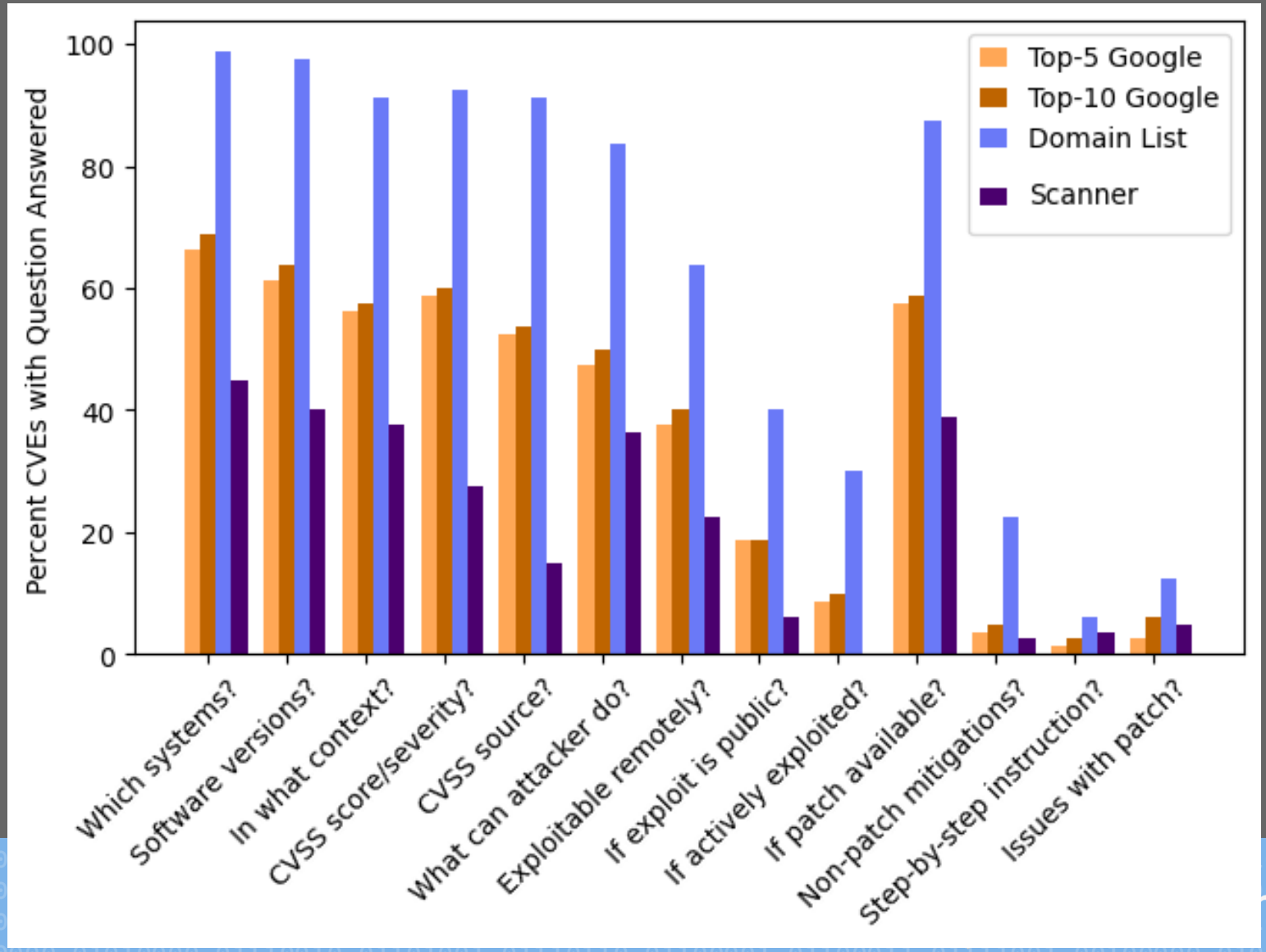
Patch Tuesday Megathread (2022-10-11)

General Discussion

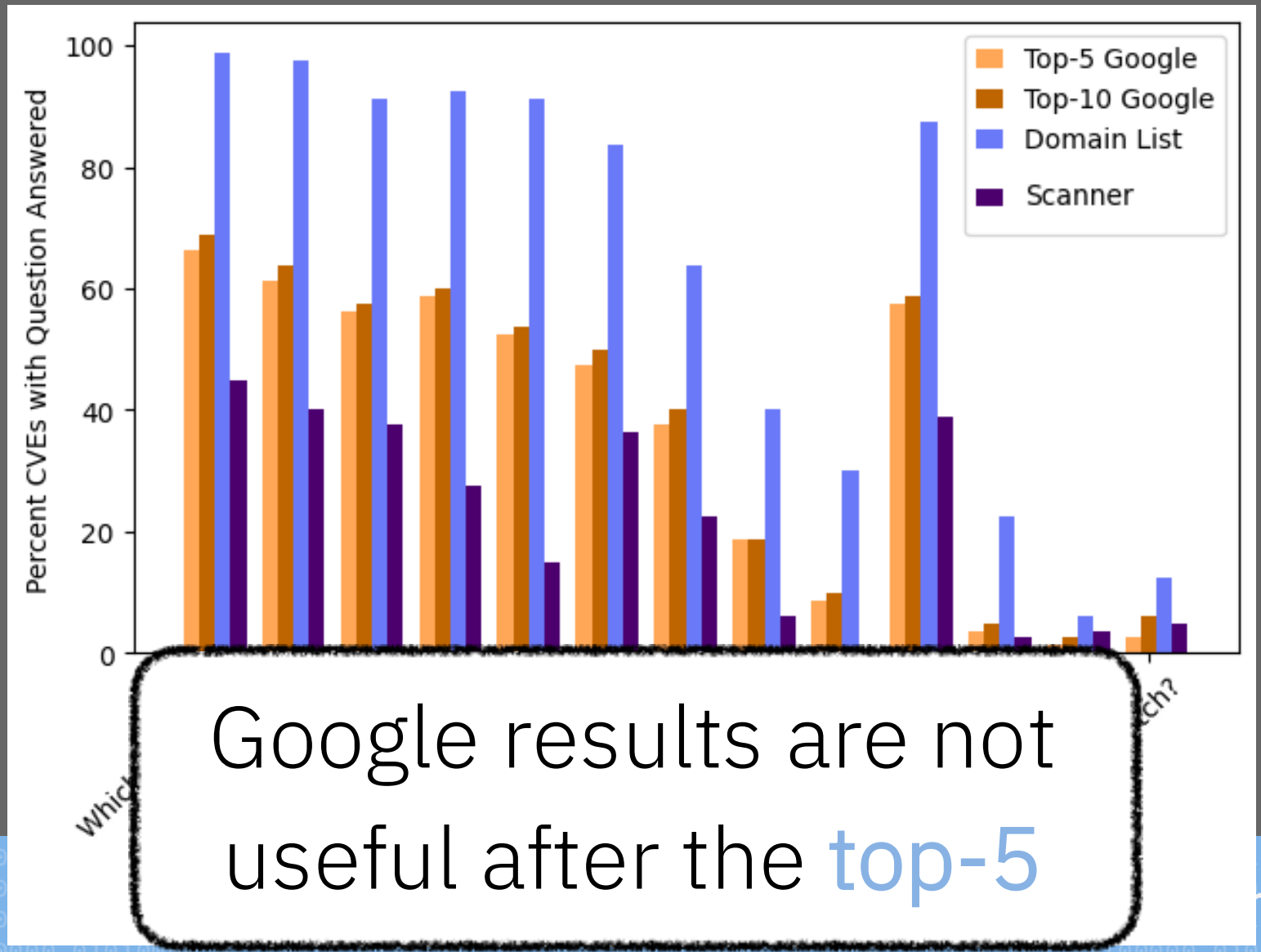
Hello r/sysadmin, I'm u/AutoModerator, and welcome to this month's **Patch Megathread!**

This is the (*mostly*) safe location to talk about the latest patches, updates, and releases. We put this thread into place to help gather all the information about this month's updates: What is fixed, what broke, what got released and should have been caught in QA, etc. We do this both to keep clutter out of the subreddit, and provide you, the dear reader, a singular resource to read.

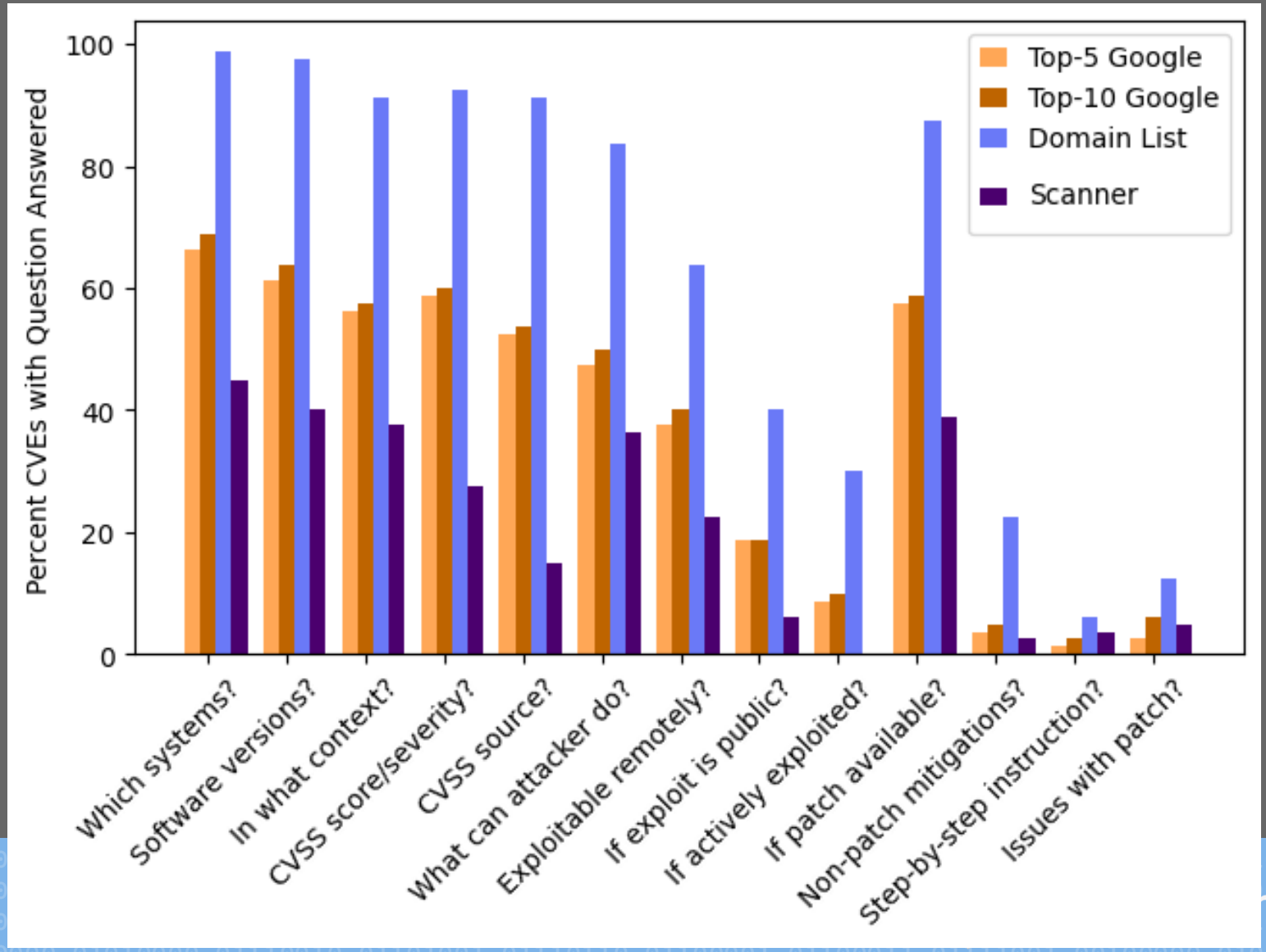
Where should you look?



Where should you look?



Where should you look?



Takeaways

- Start vulnerability information searches with common **domain list**
- We need to know **what can go wrong**
- Lots of information; **dispersed** across many pages

Summary

Where do practitioners get vulnerability information?

How robust is the vulnerability information ecosystem?

Does the vulnerability ecosystem answer practitioners' questions?

- Start vulnerability information searches with common **domain list**
- We need to know **what can go wrong**
- Lots of information; **dispersed** across many pages

Acknowledgements





Funding

Medcrypt

CISCO

Summary

Where do practitioners get vulnerability information?

How robust is the vulnerability information ecosystem?

Does the vulnerability ecosystem answer practitioners' questions?

- Start vulnerability information searches with common **domain list**
- We need to know **what can go wrong**
- Lots of information; **dispersed** across many pages

Questions?

daniel.voitpka@tufts.edu

tsp.cs.tufts.edu

Funding Sponsors

Medcrypt

CISCO