Understanding Configuration Dependencies of File Systems

Tabassum Mahmud, *Duo Zhang, Om Rameshwar Gatla, Mai Zheng*

Department of Electrical and Computer Engineering

IOWA STATE UNIVERSITY

SIGOPS usenix THE ADVANCED COMPUTING SYSTEMS ASSOCIATION

HotStorage 2022

The 14th ACM Workshop on Hot Topics in Storage and File Systems

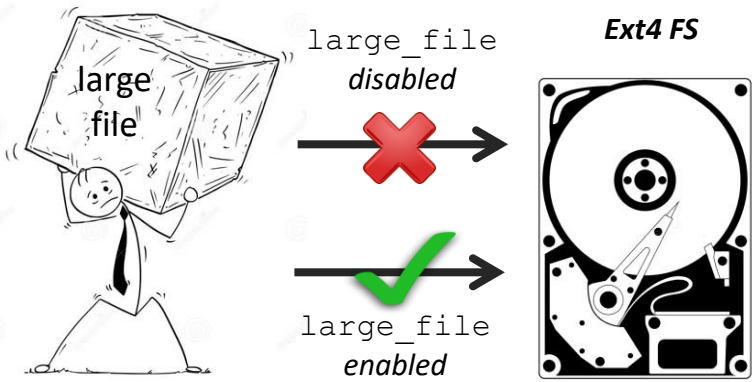
June 27-28 Virtual

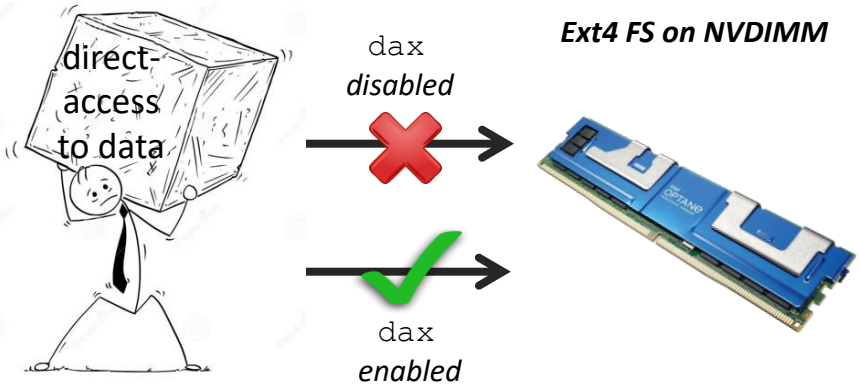
Outline

- Motivation
- Background & Related Work
- What Configuration Dependencies Exist
- How to Extract & Use Configuration Dependencies
- Discussions & Future Work

Motivation

- File systems (FS) have many configuration parameters to meet diverse needs
  - E.g., **`large_file`** for large files, **`dax`** for direct access on Ext4





`large_file` *feature supports 2GB+ files*

`dax` *feature supports direct access to NDVIMM device*

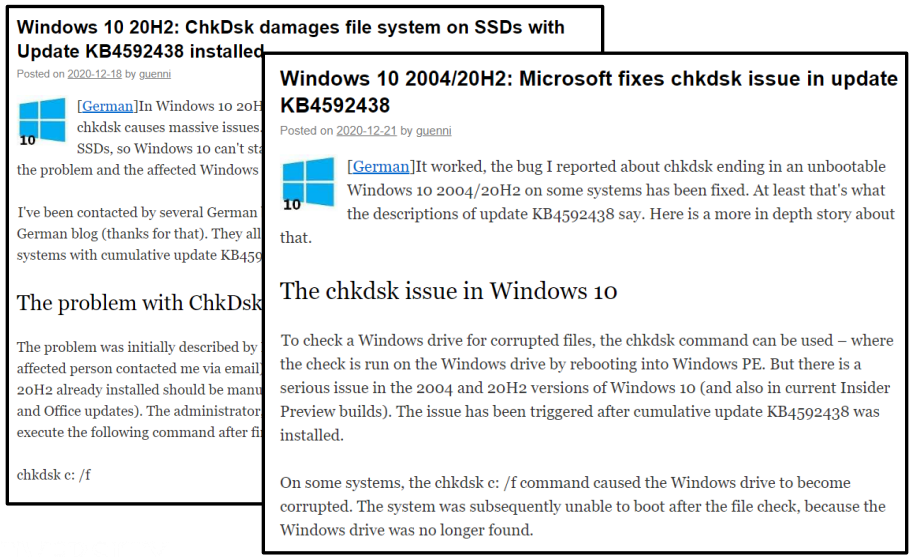
Motivation

- FS configurations can be controlled via different utilities in general

OS	FS	Example Utilities
Linux	Ext4	*mke2fs, mount, e4defrag, resize2fs*
FreeBSD	UFS	*newfs, mount, growfs, fsck_ufs*
Minix	MINIX	*mkfs, mount, fsck*
Windows	NTFS	*format, mountvol, chkdsk, shrink*
MacOS	APFS	*disk utility, mount_apfs, fsck_apfs*

Motivation

- Subtle issues may only manifest under certain configurations
  - E.g. 1: Using *chkdsk* on NTFS (on SSD) triggers an issue
    - Parameters involved: **`/f`** from *chkdsk* and another (unnamed) parameter from Windows OS
    - Consequence: corrupted NTFS FS on SSD



Windows 10 20H2: ChkDsk damages file system on SSDs with Update KB4592438 installed

Posted on 2020-12-18 by guenni

[German]In Windows 10 20H chkdsk causes massive issues SSDs, so Windows 10 can't st the problem and the affected Windows

I've been contacted by several German German blog (thanks for that). They all systems with cumulative update KB459

The problem with ChkDsk

The problem was initially described by affected person contacted me via email 20H2 already installed should be manu and Office updates). The administrator execute the following command after fi

chkdsk c: /f

Windows 10 2004/20H2: Microsoft fixes chkdsk issue in update KB4592438

Posted on 2020-12-21 by guenni

[German]It worked, the bug I reported about chkdsk ending in an unbootable Windows 10 2004/20H2 on some systems has been fixed. At least that's what the descriptions of update KB4592438 say. Here is a more in depth story about that.

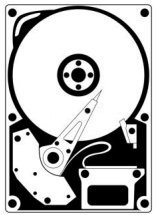
The chkdsk issue in Windows 10

To check a Windows drive for corrupted files, the chkdsk command can be used – where the check is run on the Windows drive by rebooting into Windows PE. But there is a serious issue in the 2004 and 20H2 versions of Windows 10 (and also in current Insider Preview builds). The issue has been triggered after cumulative update KB4592438 was installed.

On some systems, the chkdsk c: /f command caused the Windows drive to become corrupted. The system was subsequently unable to boot after the file check, because the Windows drive was no longer found.

Motivation

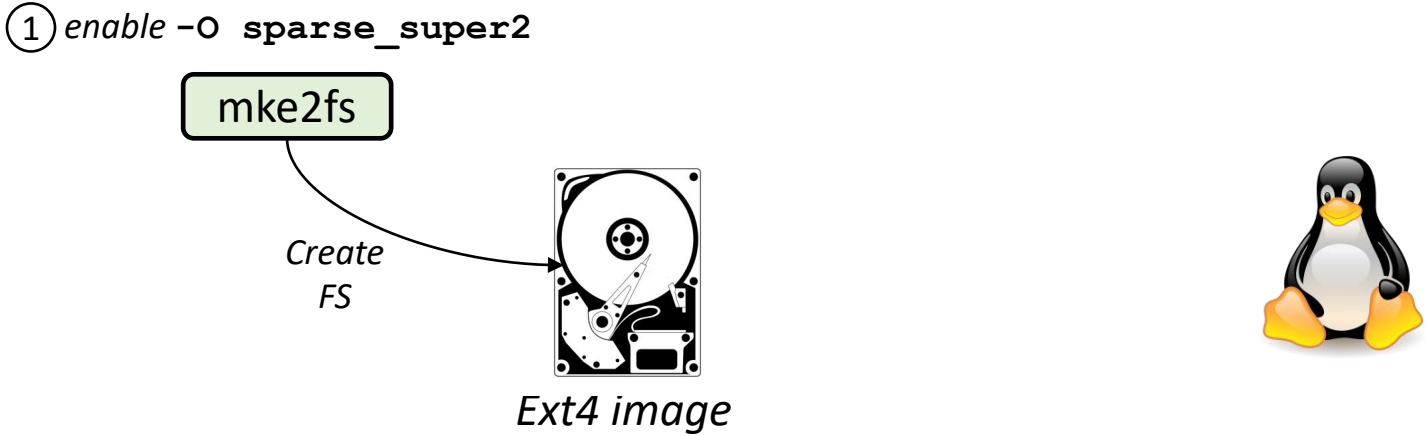
- Subtle issues may only manifest under certain configurations
  - E.g. 2: Using *resize2fs* on Ext4 FS triggers an issue
    - Parameters involved: **`sparse_super2`** from *mke2fs* and **`<size>`** from *resize2fs*
    - Consequence: corrupted Ext4 FS



Block device

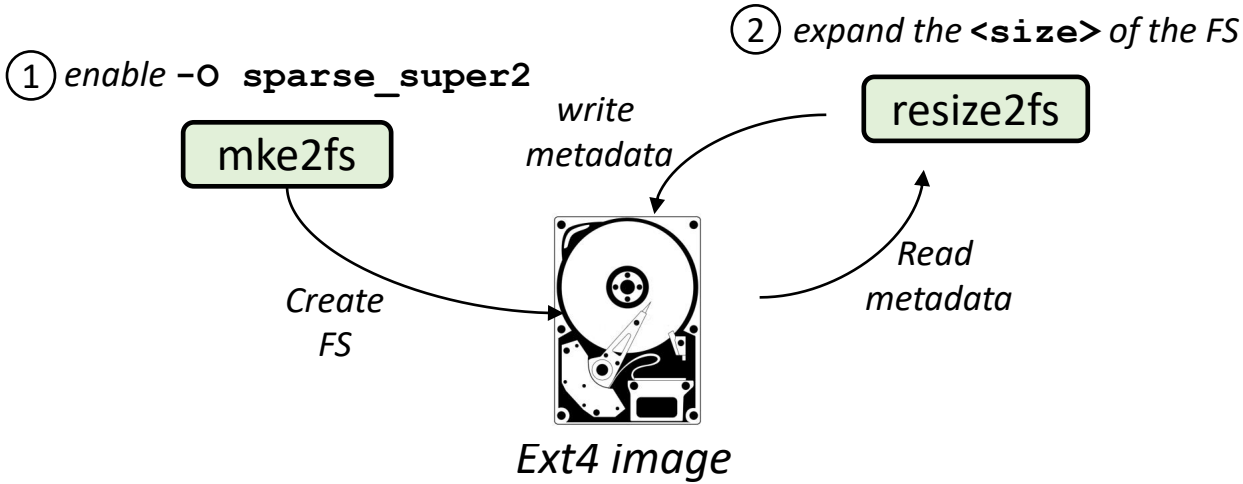
Motivation

- Subtle issues may only manifest under certain configurations
  - E.g. 2: Using *resize2fs* on Ext4 FS triggers an issue
    - Parameters involved: **`sparse_super2`** from *mke2fs* and **`<size>`** from *resize2fs*
    - Consequence: corrupted Ext4 FS



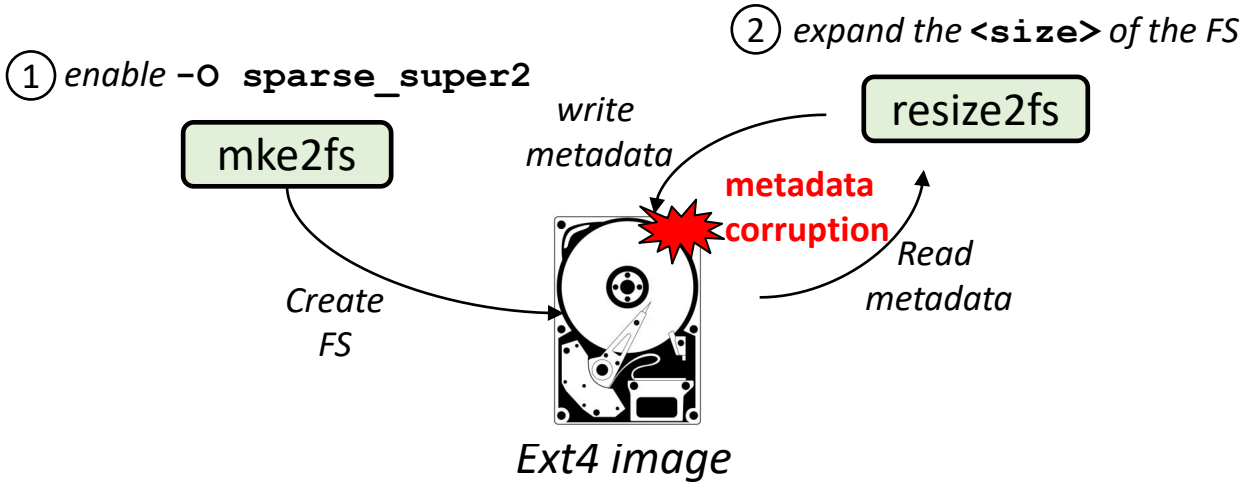
Motivation

- Subtle issues may only manifest under certain configurations
  - E.g. 2: Using *resize2fs* on Ext4 FS triggers an issue
    - Parameters involved: **`sparse_super2`** from *mke2fs* and **`<size>`** from *resize2fs*
    - Consequence: corrupted Ext4 FS



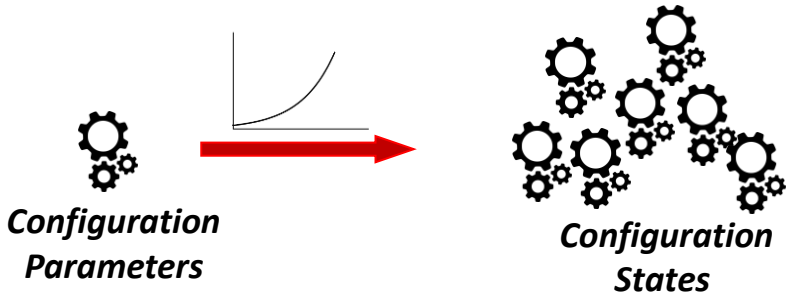
Motivation

- Subtle issues may only manifest under certain configurations
  - E.g. 2: Using *resize2fs* on Ext4 FS triggers an issue
    - Parameters involved: **`sparse_super2`** from *mke2fs* and **`<size>`** from *resize2fs*
    - Consequence: corrupted Ext4 FS



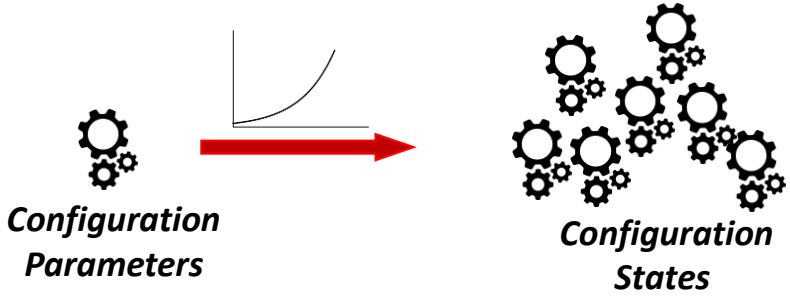
Motivation

- Difficult to test configuration-related issues due to state explosion
  - E.g., Ext4 has >85 configuration parameters, resulting in 10^{37} configuration states [Carver@FAST'20]



Motivation

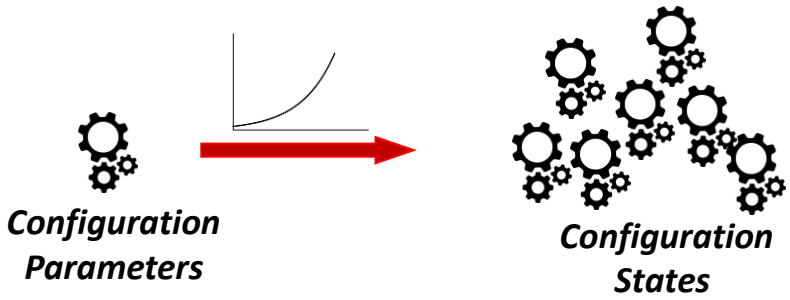
- Difficult to test configuration-related issues due to state explosion
  - E.g., Ext4 has >85 configuration parameters, resulting in 10^{37} configuration states [Carver@FAST'20]
  - Existing test suites have limited coverage



Test Suite	*Target Software*	*Config. Param. Used*
xfstest	Ext4	<34.1%
E2fsprogs-test	e2fsck	<17.1%
	resize2fs	<46.7%

Motivation

- Difficult to test configuration-related issues due to state explosion
  - E.g., Ext4 has >85 configuration parameters, resulting in 10^{37} configuration states [Carver@FAST'20]
  - Existing test suites have limited coverage



Test Suite	*Target Software*	*Config. Param. Used*
xfstest	Ext4	<34.1%
E2fsprogs-test	e2fsck	<17.1%
	resize2fs	<46.7%

How to identify configuration-related issues efficiently?

Outline

- ~~Motivation~~
- Background & Related Work
- What Configuration Dependencies Exist
- How to Extract & Use Configuration Dependencies
- Discussions & Future Work

Background

- Many file systems can be configured by different utilities in four stages

Background

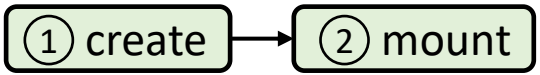
- Many file systems can be configured by different utilities in four stages

① create

Ext4: *mke2fs*
UFS: *newfs*
NTFS: *format*
APFS: *disk_util*

Background

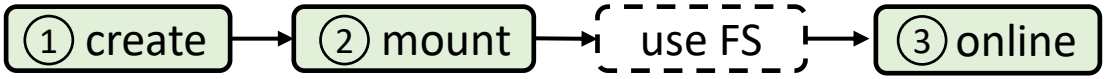
- Many file systems can be configured by different utilities in four stages



Ext4:	*mke2fs*	*mount*
UFS:	*newfs*	*mount*
NTFS:	*format*	*mountvol*
APFS:	*disk_util*	*mount_apfs*

Background

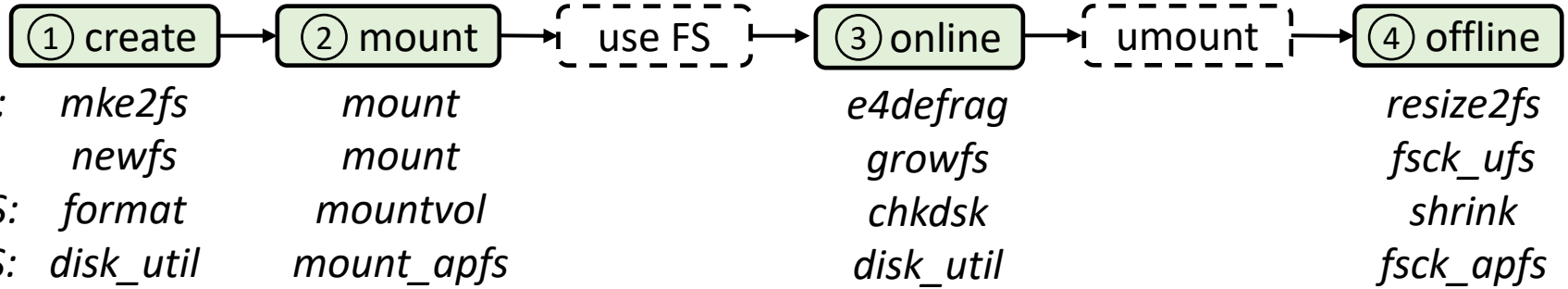
- Many file systems can be configured by different utilities in four stages



Ext4:	*mke2fs*	*mount*	*e4defrag*
UFS:	*newfs*	*mount*	*growfs*
NTFS:	*format*	*mountvol*	*chkdsk*
APFS:	*disk_util*	*mount_apfs*	*disk_util*

Background

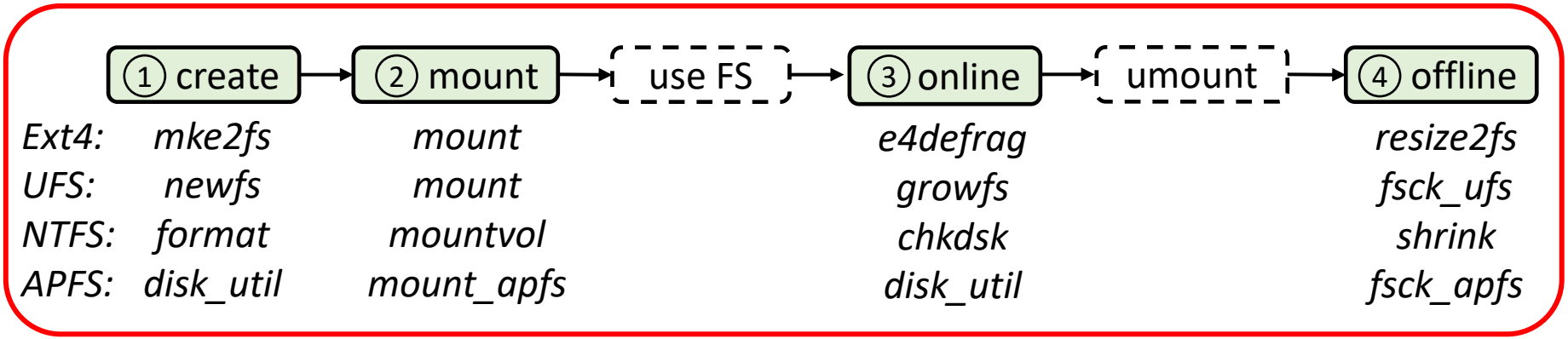
- Many file systems can be configured by different utilities in four stages



Background

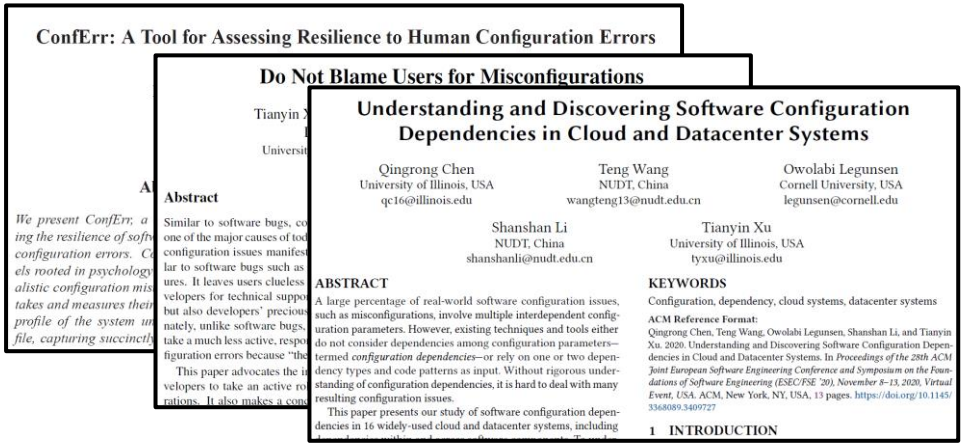
- Many file systems can be configured by different utilities in four stages

FS ecosystem



Related Work

- Concepts [ConfErr@DSN'08, SPEX@SOSP'13, cDEP@ECSE/FSE'20]
  - Configuration Constraint
    - Specify configuration requirements (e.g., data type, value range)
  - Configuration Dependency
    - One special type of constraint
    - Describe the dependent correlation among parameters
    - Critical for addressing complex configuration issues


ConfErr: A Tool for Assessing Resilience to Human Configuration Errors

Do Not Blame Users for Misconfigurations

Abstract

Understanding and Discovering Software Configuration Dependencies in Cloud and Datacenter Systems

Qingrong Chen
University of Illinois, USA
qc16@illinois.edu

Teng Wang
NUDT, China
wangteng13@nudt.edu.cn

Owolabi Legunsen
Cornell University, USA
legunsen@cornell.edu

Shanshan Li
NUDT, China
shanshanli@nudt.edu.cn

Tianyin Xu
University of Illinois, USA
tyxu@illinois.edu

ABSTRACT

A large percentage of real-world software configuration issues, such as misconfigurations, involve multiple interdependent configuration parameters. However, existing techniques and tools either do not consider dependencies among configuration parameters—termed *configuration dependencies*—or rely on one or two dependency types and code patterns as input. Without rigorous understanding of configuration dependencies, it is hard to deal with many resulting configuration issues.

This paper presents our study of software configuration dependencies in 16 widely-used cloud and datacenter systems, including

KEYWORDS

Configuration, dependency, cloud systems, datacenter systems

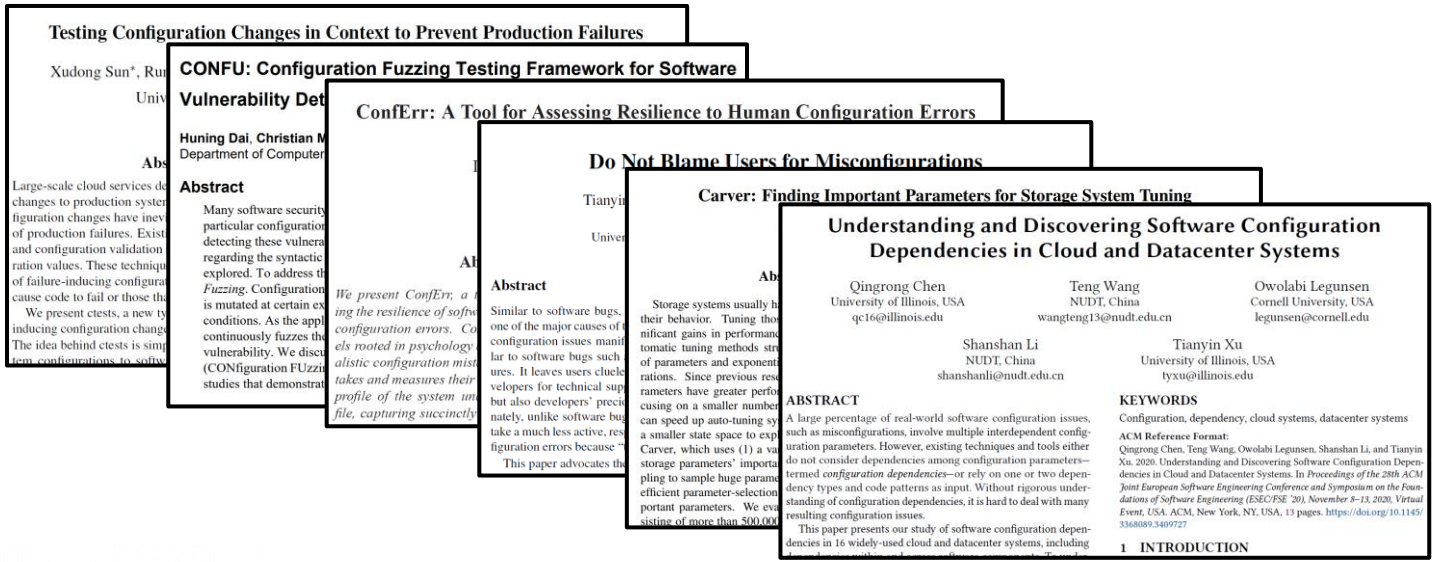
ACM Reference Format:

Qingrong Chen, Teng Wang, Owolabi Legunsen, Shanshan Li, and Tianyin Xu. 2020. Understanding and Discovering Software Configuration Dependencies in Cloud and Datacenter Systems. In *Proceedings of the 28th ACM Joint European Software Engineering Conference and Symposium on the Foundations of Software Engineering (ESEC/FSE '20), November 8–13, 2020, Virtual Event, USA.* ACM, New York, NY, USA, 13 pages. https://doi.org/10.1145/3368089.3409727

1 INTRODUCTION

Related Work

- Limitations [ConfErr@DSN'08, SPEX@SOSP'13, cDEP@ECSE/FSE'20, ctests@OSDI'20 etc.]
  - Mostly only focus on shallow configuration constraints (e.g., spelling mistakes)
  - Do not consider multi-component configuration issues
  - Only work for Java programs with unified configuration library and namespace

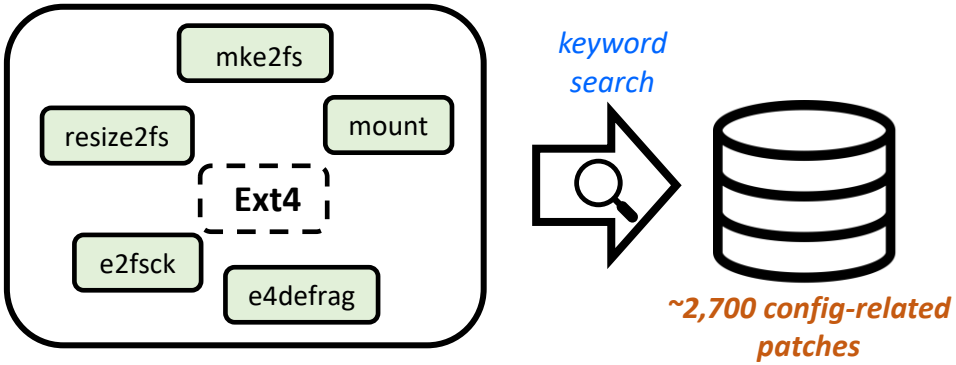


Outline

- ~~Motivation~~
- ~~Background & Related Work~~
- What Configuration Dependencies Exist
- How to Extract & Use Configuration Dependencies
- Discussions & Future Work

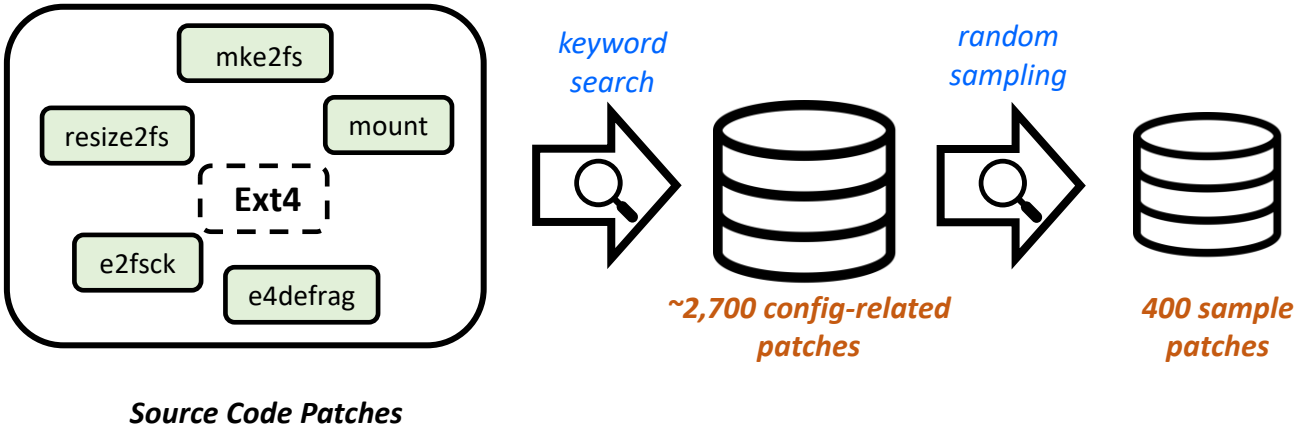
What Configuration Dependencies Exist

- Study Methodology
  - Analyze configuration-related patches and source code of the Ext4 ecosystem



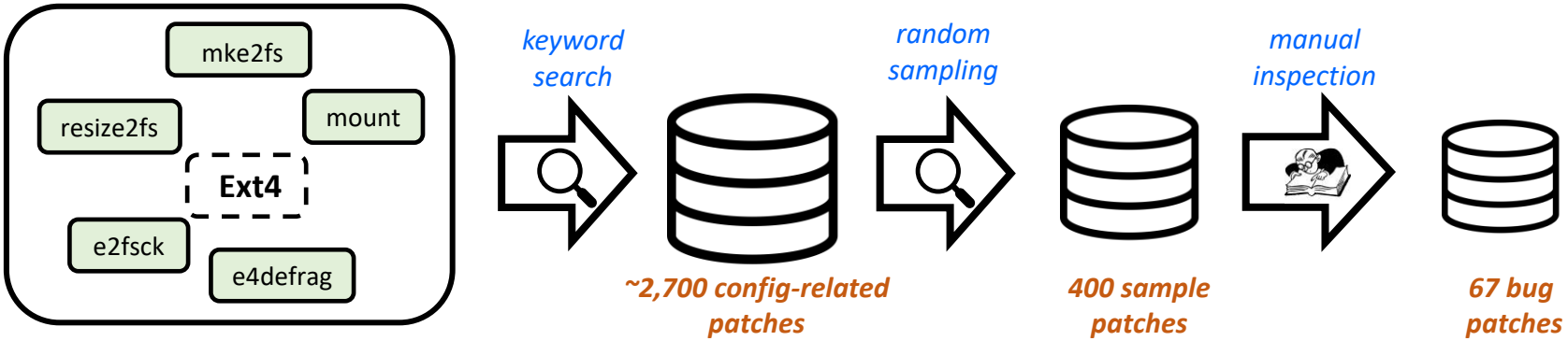
What Configuration Dependencies Exist

- Study Methodology
  - Analyze configuration-related patches and source code of the Ext4 ecosystem



What Configuration Dependencies Exist

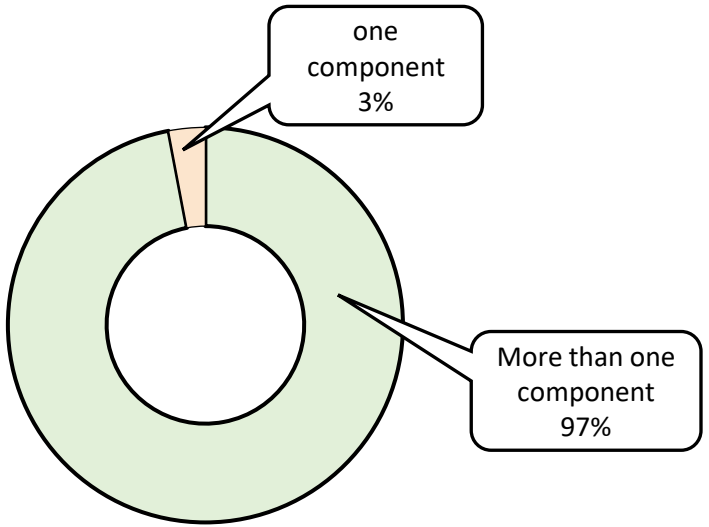
- Study Methodology
  - Analyze configuration-related patches and source code of the Ext4 ecosystem



What Configuration Dependencies Exist

- Findings
  - #1: Majority cases (97%) involve critical parameters from more than one component

67 bug ***patches***



Distribution of 67 bug cases

What Configuration Dependencies Exist

- Findings
  - #2: Multi-level configuration dependencies are prevalent
    - Self dependency (SD), cross-parameter dependency (CPD), cross-component dependency (CCD)

What Configuration Dependencies Exist

- Findings
  - #2: Multi-level configuration dependencies are prevalent
    - Self dependency (SD), cross-parameter dependency (CPD), cross-component dependency (CCD)

Self Dependency	Description
Data Type	Parameter must be of a specific data type
Value Range	Parameter must be within a specific value range

What Configuration Dependencies Exist

- Findings
  - #2: Multi-level configuration dependencies are prevalent
    - Self dependency (SD), cross-parameter dependency (CPD), cross-component dependency (CCD)

Self Dependency	Description
Data Type	Parameter must be of a specific data type
Value Range	Parameter must be within a specific value range

SD Example:

```
$./mke2fs -b 1024 -O sparse_super2, …
…
$./resize2fs img <size>
```

```
PRS(…){
   …
 if (b < EXT2_MIN_BLOCK_SIZE ||
          b > EXT2_MAX_BLOCK_SIZE)
               exit(1);
}
```

What Configuration Dependencies Exist

- Findings
  - #2: Multi-level configuration dependencies are prevalent
    - Self dependency (SD), cross-parameter dependency (CPD), cross-component dependency (CCD)

Cross-Parameter Dependency	Description
Control	P1 of C1 can be enabled iff P2 of C1 is enabled/disabled
Value	P1's value depends on P2's value (e.g., P1 < P2)

What Configuration Dependencies Exist

- Findings
  - #2: Multi-level configuration dependencies are prevalent
    - Self dependency (SD), cross-parameter dependency (CPD), cross-component dependency (CCD)

Cross-Parameter Dependency	Description
Control	P1 of C1 can be enabled iff P2 of C1 is enabled/disabled
Value	P1's value depends on P2's value (e.g., P1 < P2)

CPD Example:

```
$./mke2fs -b 1024 -O sparse_super2,
                  resize_inode, ^meta_bg img
…
$./resize2fs img <size>
```

```
if (ext2fs_has_feature_meta_bg
    (&fs_param)&& ext2fs_has_
     feature_resize_inode
     (&fs_param)){
          exit(1);
}
```

What Configuration Dependencies Exist

- Findings
  - #2: Multi-level configuration dependencies are prevalent
    - Self dependency (SD), cross-parameter dependency (CPD), cross-component dependency (CCD)

Cross-Component Dependency	Description
Control	P1 of C1 can be enabled iff P2 of C2 is enabled/disabled
Value	P1's value depends on P2's value from another component
Behavioral	Component C1's behavior depends on P2 of C2

What Configuration Dependencies Exist

- Findings
  - #2: Multi-level configuration dependencies are prevalent
    - Self dependency (SD), cross-parameter dependency (CPD), cross-component dependency (CCD)

Cross-Component Dependency	Description
Control	P1 of C1 can be enabled iff P2 of C2 is enabled/disabled
Value	P1's value depends on P2's
Behavioral	Component C1's behavior d

CCD Example:

```
$./mke2fs -b 1024 -O sparse_super2, resize_in
…
$./resize2fs img <size>
```

```
clear_sparse_super2(…){
   …
  if(!extfs_has_feature_
          sparse_super2)
      return 0;
  if(last_bg <= old_last_bg)
      return 0;
  // execute critical
          functionality
}
```

What Configuration Dependencies Exist

- Findings
  - #2: Multi-level configuration dependencies are prevalent
    - Self dependency (SD), cross-parameter dependency (CPD), cross-component dependency (CCD)

FS Usage Scenario (key config utilities are in bold)	**# of Bugs**	**Multi-level Dependencies**		
		SD	**CPD**	**CCD**
mke2fs - **mount** - Ext4	13	13	1	13
mke2fs - **mount** - Ext4 - **e4defrag**	1	1	-	1
mke2fs - mount - Ext4 - umount - **resize2fs**	17	17	-	17
mke2fs - mount - Ext4 - umount - **e2fsck**	36	36	4	34
Total	67	67 (100%)	5 (7.5%)	65 (97.0%)

What Configuration Dependencies Exist

- Findings
  - #2: Multi-level configuration dependencies are prevalent
    - Self dependency (SD), cross-parameter dependency (CPD), cross-component dependency (CCD)

FS Usage Scenario (key config utilities are in bold)	# of Bugs	Multi-level Dependencies		
		SD	CPD	CCD
mke2fs - **mount** - Ext4	13	13	1	13
mke2fs - **mount** - Ext4 - **e4defrag**	1	1	-	1
mke2fs - mount - Ext4 - umount - **resize2fs**	17	17	-	17
mke2fs - mount - Ext4 - umount - **e2fsck**	36	36	4	34
Total	67	67 (100%)	5 (7.5%)	65 (97.0%)

What Configuration Dependencies Exist

- Findings
  - #2: Multi-level configuration dependencies are prevalent
    - Self dependency (SD), cross-parameter dependency (CPD), cross-component dependency (CCD)

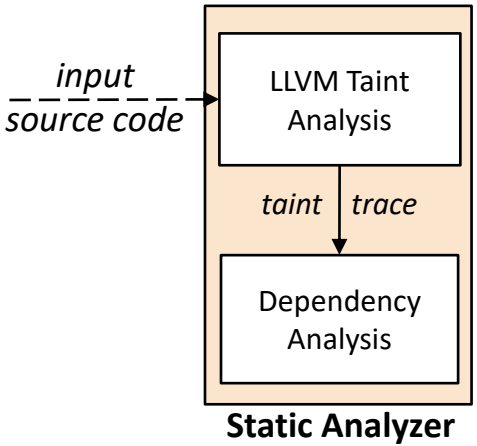
FS Usage Scenario (key config utilities are in bold)	**# of Bugs**	**Multi-level Dependencies**		
		SD	**CPD**	**CCD**
mke2fs - **mount** - Ext4	13	13	1	13
mke2fs - **mount** - Ext4 - **e4defrag**	1	1	-	1
mke2fs - mount - Ext4 - umount - **resize2fs**	17	17	-	17
mke2fs - mount - Ext4 - umount - **e2fsck**	36	36	4	34
Total	67	67 (100%)	5 (7.5%)	65 (97.0%)

Outline

- ~~Motivation~~
- ~~Background & Related Work~~
- ~~What Configuration Dependencies Exist~~
- How to Extract & Use Configuration Dependencies
- Discussions & Future Work

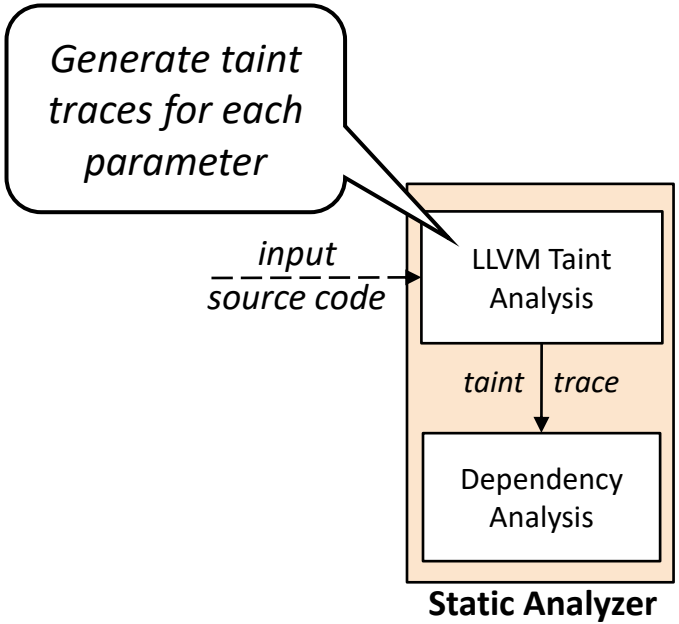
How to Extract & Use Configuration Dependencies

- Deriving configuration dependencies using static analysis



How to Extract & Use Configuration Dependencies

- Deriving configuration dependencies using static analysis

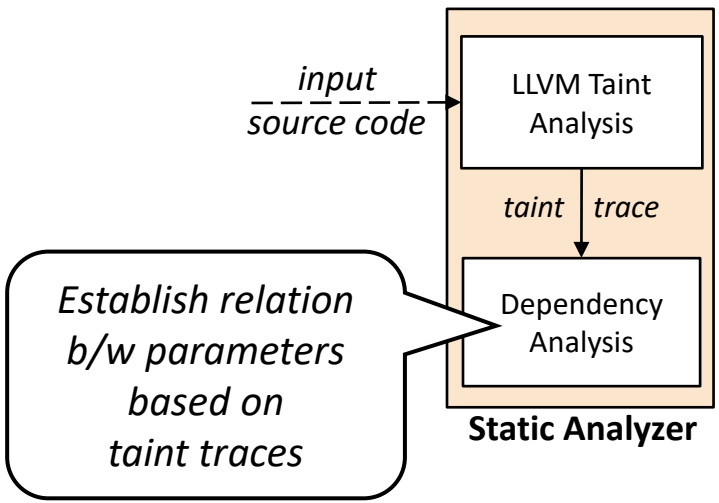


Example taint trace

```
%blocksize = alloca i32
store i32 0, i32* %blocksize
%69 = load i32, i32* %blocksize
%cond = phi i32 [%69, %cond.true], [%sub, %cond.false]
store i32 %cond, i32* %b
%71 = load i32, i32* %b
%cmp70 = icmp slt i32 %71, 1024
```

How to Extract & Use Configuration Dependencies

- Deriving configuration dependencies using static analysis

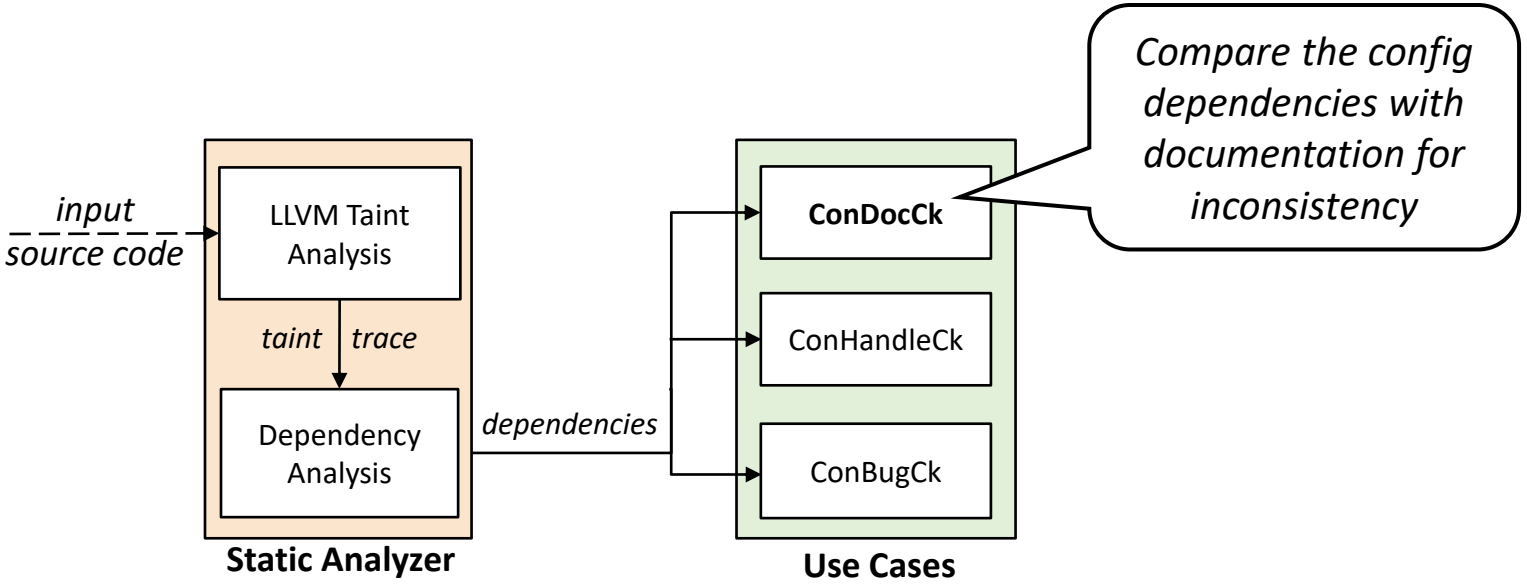


Example taint trace

```
%blocksize = alloca i32
store i32 0, i32* %blocksize
%69 = load i32, i32* %blocksize
%cond = phi i32 [%69, %cond.true], [%sub, %cond.false]
store i32 %cond, i32* %b
%71 = load i32, i32* %b
%cmp70 = icmp slt i32 %71, 1024
```

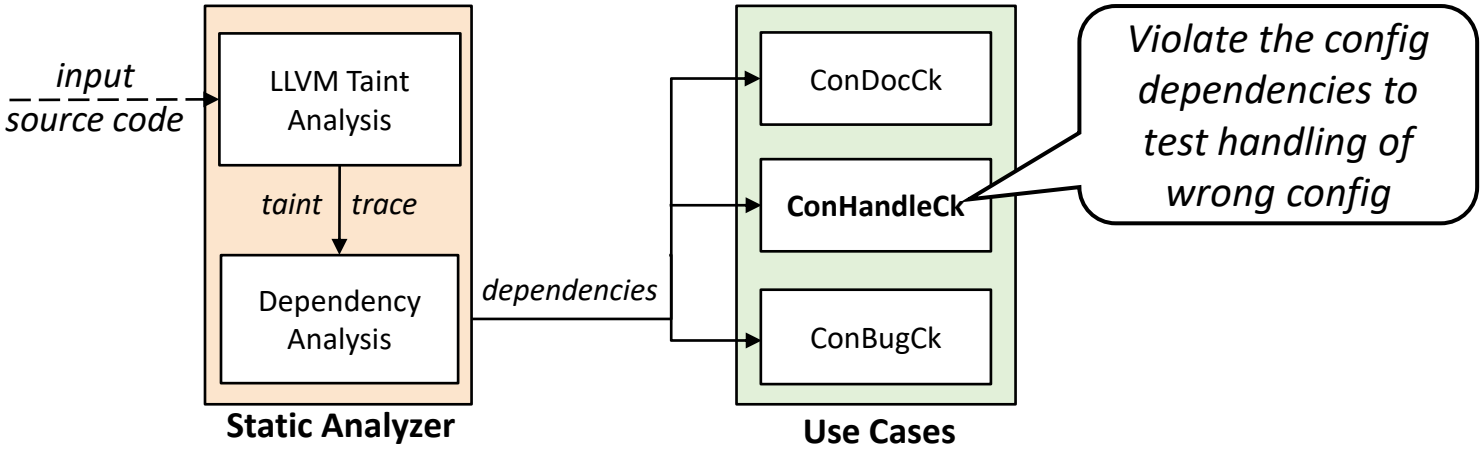
How to Extract & Use Configuration Dependencies

- Deriving configuration dependencies using static analysis
- Three example use cases for different configuration-related issues



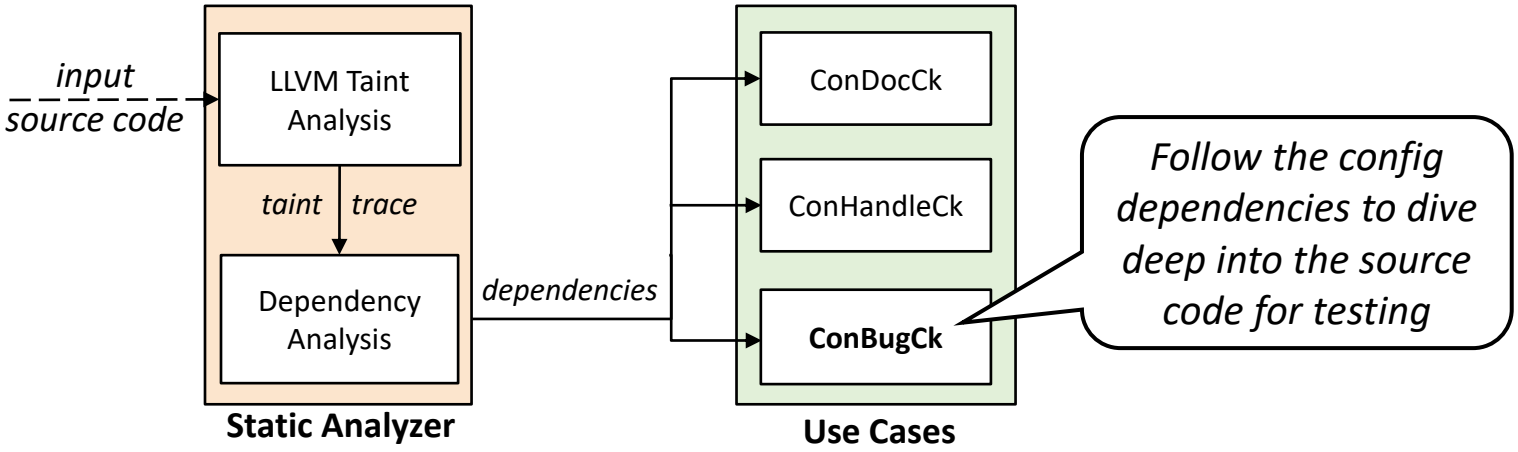
How to Extract & Use Configuration Dependencies

- Deriving configuration dependencies using static analysis
- Three example use cases for different configuration-related issues



How to Extract & Use Configuration Dependencies

- Deriving configuration dependencies using static analysis
- Three example use cases for different configuration-related issues



How to Extract & Use Configuration Dependencies

- Preliminary Results
  - Automatically extracted multi-level dependencies

FS Usage Scenario (key config utilities are in bold)	**Multi-level Dependencies**		
	SD	**CPD**	**CCD**
mke2fs - **mount** - Ext4	31	24	0
mke2fs - **mount** - Ext4 - **e4defrag**	31	24	0
mke2fs - mount - Ext4 - umount - **resize2fs**	32	26	6
mke2fs - mount - Ext4 - umount - **e2fsck**	32	26	0
Total	32	26	6

64 unique in total

7.8% false positive

How to Extract & Use Configuration Dependencies

- Preliminary Results
  - Configuration issues found based on the extracted dependencies

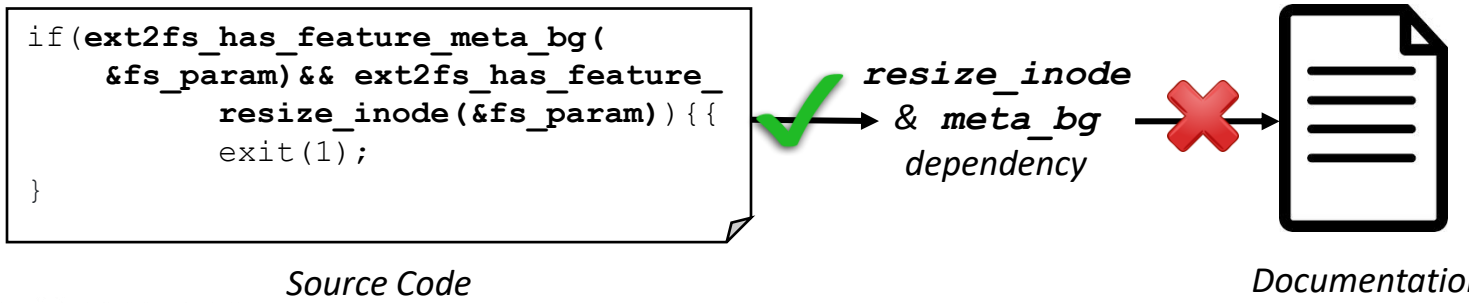
Type of issues	# of issues
Inaccurate documentation issue	12
Unexpected misconfiguration handling	1

How to Extract & Use Configuration Dependencies

- Preliminary Results
  - Configuration issues found based on the extracted dependencies

Type of issues	# of issues
Inaccurate documentation issue	12
Unexpected misconfiguration handling	1

  - E.g., **`resize_inode`** and **`meta_bg`** cannot be enabled together, but it is not specified in documentation

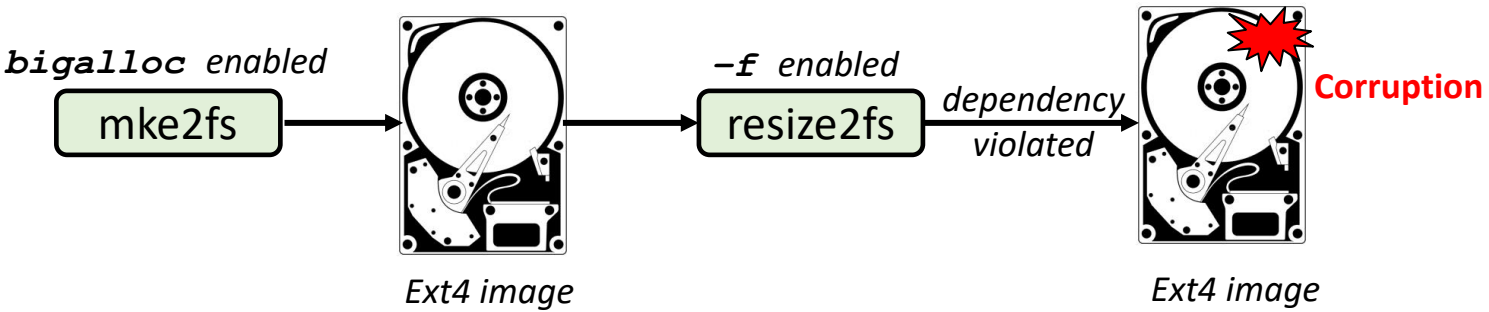


How to Extract & Use Configuration Dependencies

- Preliminary Results
  - Configuration issues found based on the extracted dependencies

Type of issues	# of issues
Inaccurate documentation issue	12
Unexpected misconfiguration handling	1

  - Corruption may occur when dependency between **`bigalloc`** (parameter from **`mke2fs`**) and **`-f`** (parameter from **`resize2fs`**) is violated



Outline

- ~~Motivation~~
- ~~Background & Related Work~~
- ~~What Configuration Dependencies Exist~~
- ~~How to Extract & Use Configuration Dependencies~~
- Discussions & Future Work

Discussions & Future Work

More automation & evaluation

- Minimize annotation, add inter-procedural analysis ...
- More FS ecosystems and more metrics
- Open source

Discussions & Future Work

More automation & evaluation

- Minimize annotation, add inter-procedural analysis ...
- More FS ecosystems and more metrics
- Open source

Dependencies b/w file system & other software

- FS and databases
- local FS and distributed FS
- …

Discussions & Future Work

More automation & evaluation

- Minimize annotation, add inter-procedural analysis ...
- More FS ecosystems and more metrics
- Open source

Dependencies b/w file system & other software

- FS and databases
- local FS and distributed FS
- …

Better configuration design

- More parameters *vs.* fewer parameters?
- Modular design *vs.* integration within FS?

Discussions & Future Work

More automation & evaluation

- Minimize annotation, add inter-procedural analysis ...
- More FS ecosystems and more metrics
- Open source

Dependencies b/w file system & other software

- FS and databases
- local FS and distributed FS
- …

Better configuration design

- More parameters *vs.* fewer parameters?
- Modular design *vs.* integration within FS?

Thank You!