IOWA STATE UNIVERSITY

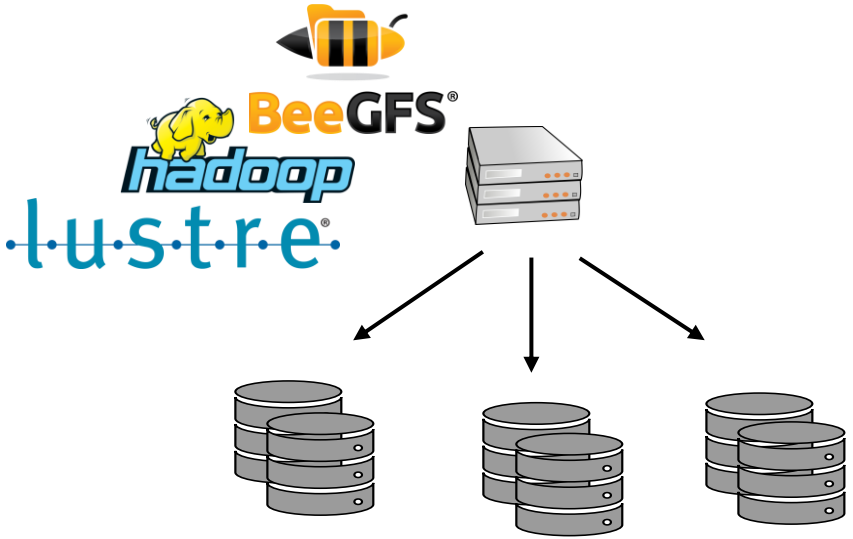
FAST'23
21st USENIX Conference on File and Storage Technologies
FEBRUARY 21–23, 2023
SANTA CLARA, CA, USA
https://www.usenix.org/conference/fast23

# ConfD: Analyzing Configuration Dependencies of File Systems for Fun and Profit

***Tabassum Mahmud***, *Om R. Gatla, Duo Zhang, Carson Love, Ryan Bumman, Mai Zheng*
Department of Electrical and Computer Engineering

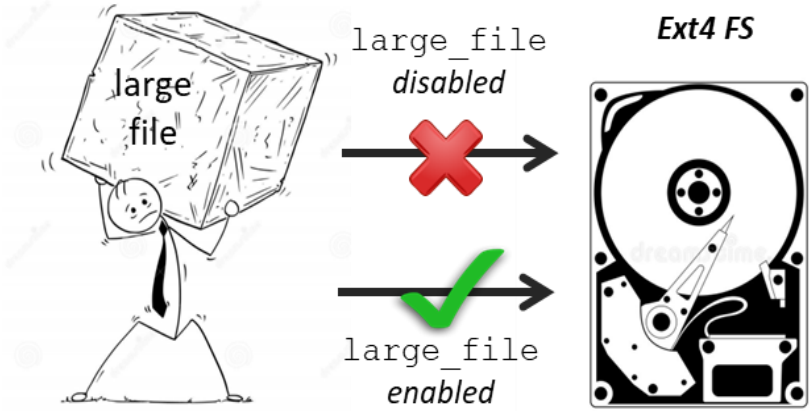
# Motivation

- File systems (FS) are essential
  - E.g., manages user files, serves as local storage for distributed systems etc.

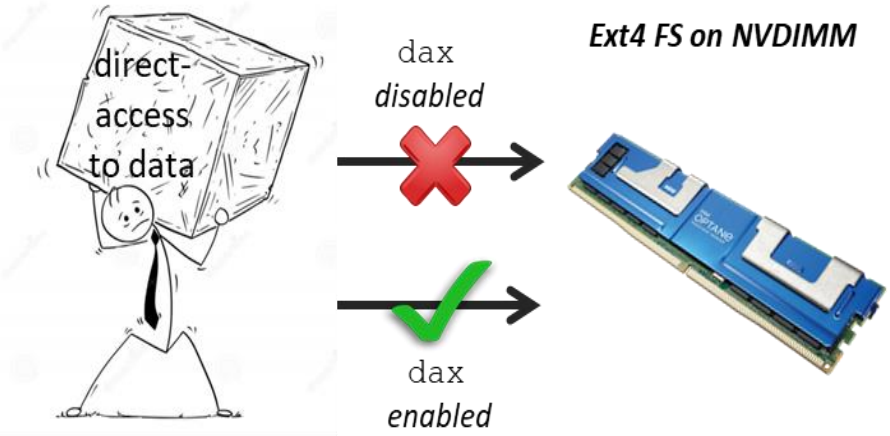


# Motivation

- File systems 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

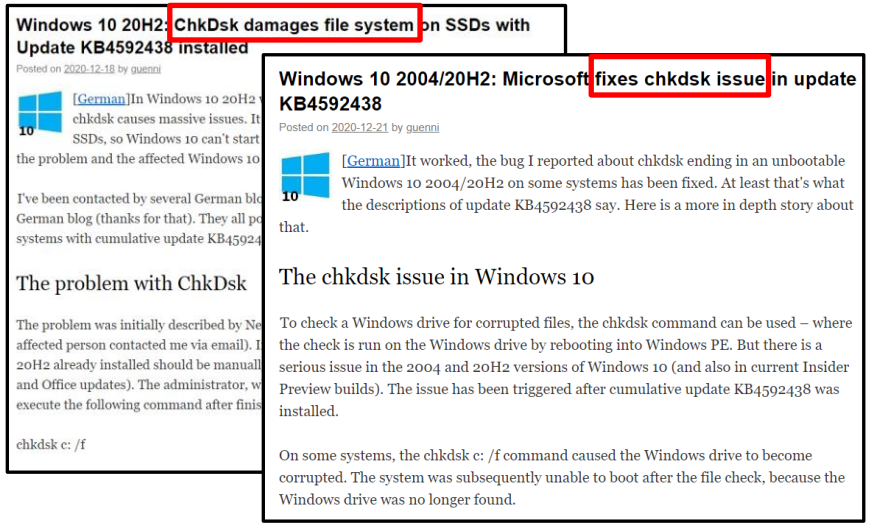
- These configuration parameters are controllable via different utilities

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



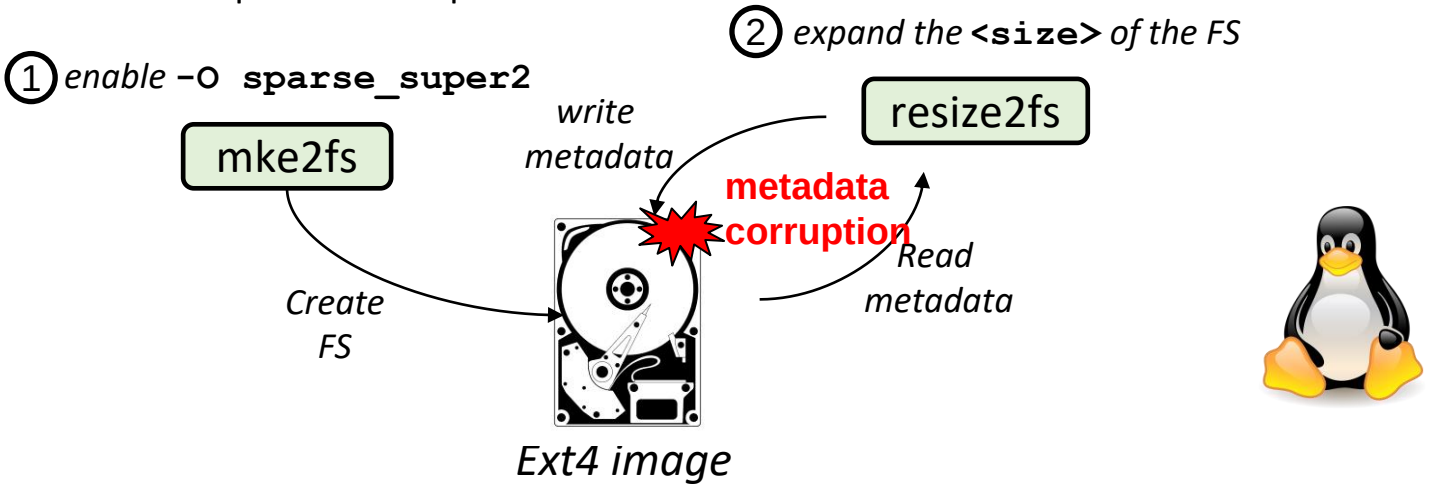

# Motivation

- Configuration parameters introduce additional complexity
  - E.g., subtle correctness issue may only manifest under specific configuration
    - 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



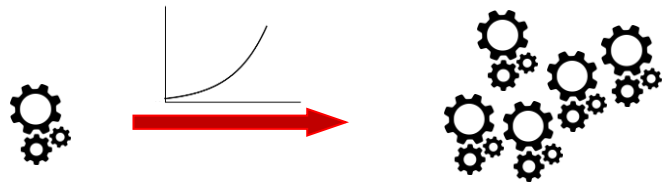
# Motivation

- Configuration parameters introduce additional complexity
  - E.g., subtle correctness issue may only manifest under specific configuration
    - 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

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

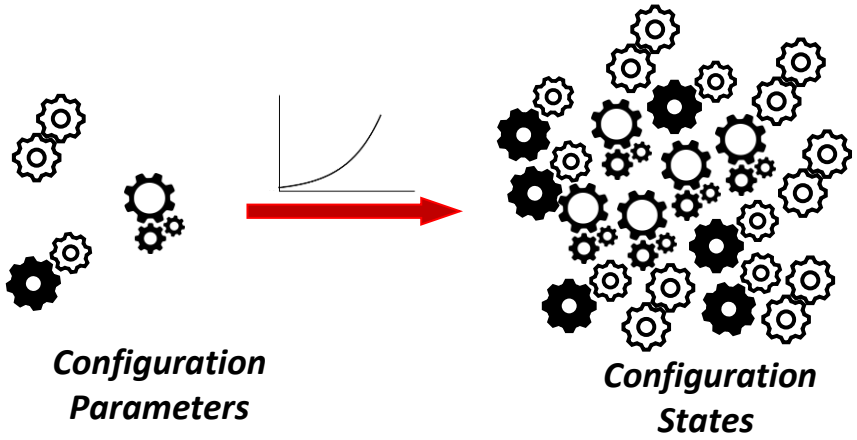


***Configuration Parameters***

***Configuration States***

# Motivation

- Configuration related issues are difficult to test due to state explosion
  - E.g., Ext4 has >85 configuration parameters, resulting in $10^{37}$ configuration states [Carver@FAST'20]
  - New devices and advanced features only make it worse, e.g., CXL based devices



# Limitation of the state of the art

- Standard Test Suites
  - Less than half configuration parameters are used

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

# Limitation of the state of the art

- Research efforts: [ConfErr@DSN'08, SPEX@SOSP'13, cDEP@ECSE/FSE'20]
  - Mostly only focus on shallow configuration constraints (e.g., spelling mistakes), e.g., ConfErr
  - Only considers multi-component configuration issues with shared configuration-library, e.g., cDep
  - Not publicly available, e.g., SPEX
  - Do not work for file system config issues, e.g., cDep

Our Contributions:

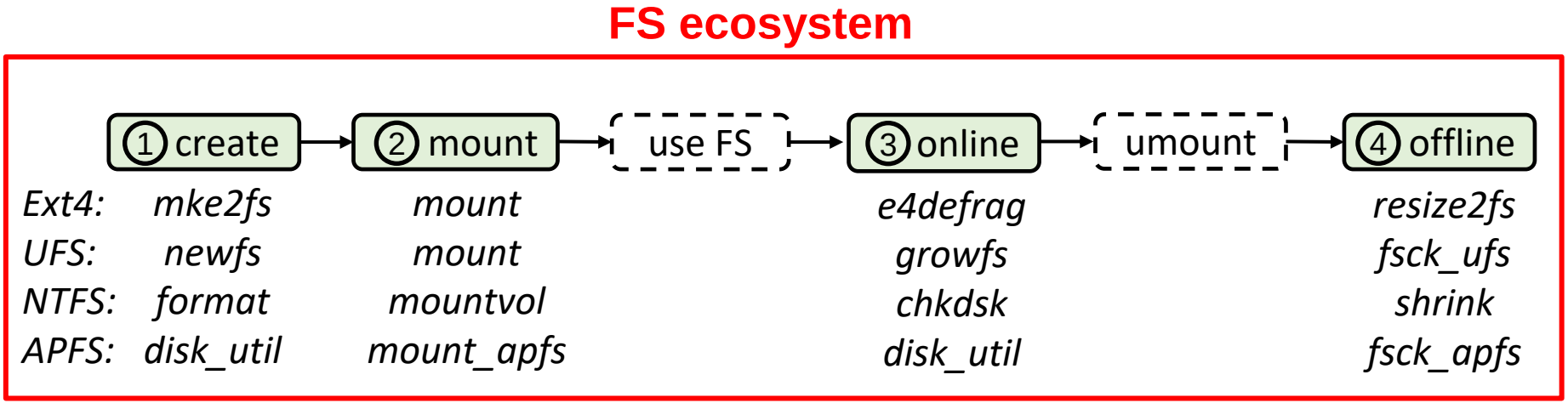
- Deriving a taxonomy of critical configuration dependencies
- Building a prototype to extract configuration dependencies
- Integrating with multiple tools and exposing configuration-related issues

# Outline

- ~~Introduction~~
- Background
- What Configuration Dependencies Exist in File systems
- How to Extract & Use Multilevel Configuration Dependencies
- Evaluation
- Conclusion & Future Work

# Background

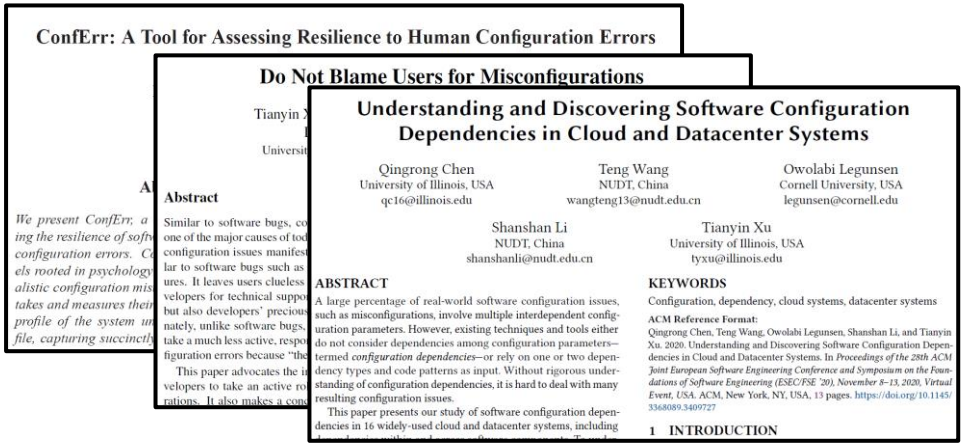
- Many file systems can be configured by different utilities in four stages





# Background

- Basic 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

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

# Background

- Basic 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

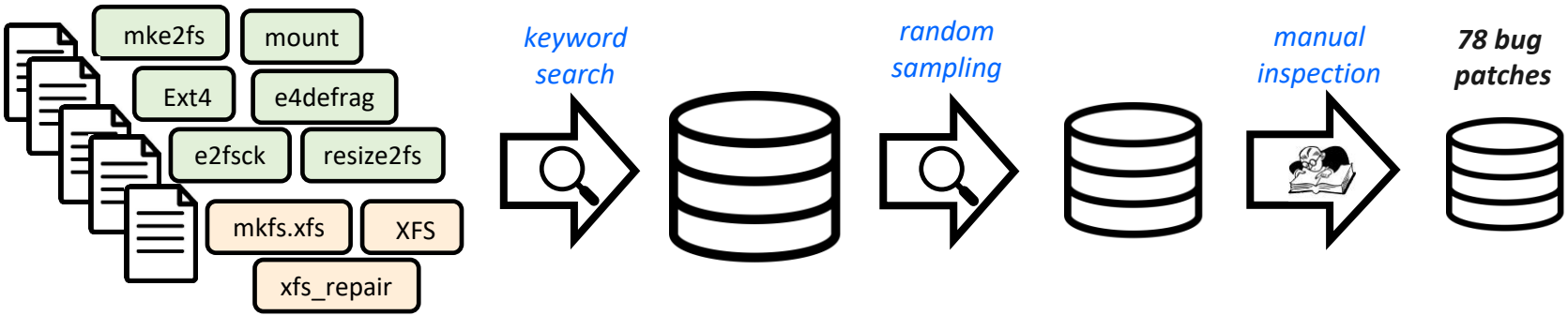
What configuration dependencies exist in file systems

# Outline

- ~~Introduction~~
- ~~Background~~
- What Configuration Dependencies Exist in File systems
- How to Extract & Use Multilevel Configuration Dependencies
- Evaluation
- Conclusion & Future Work

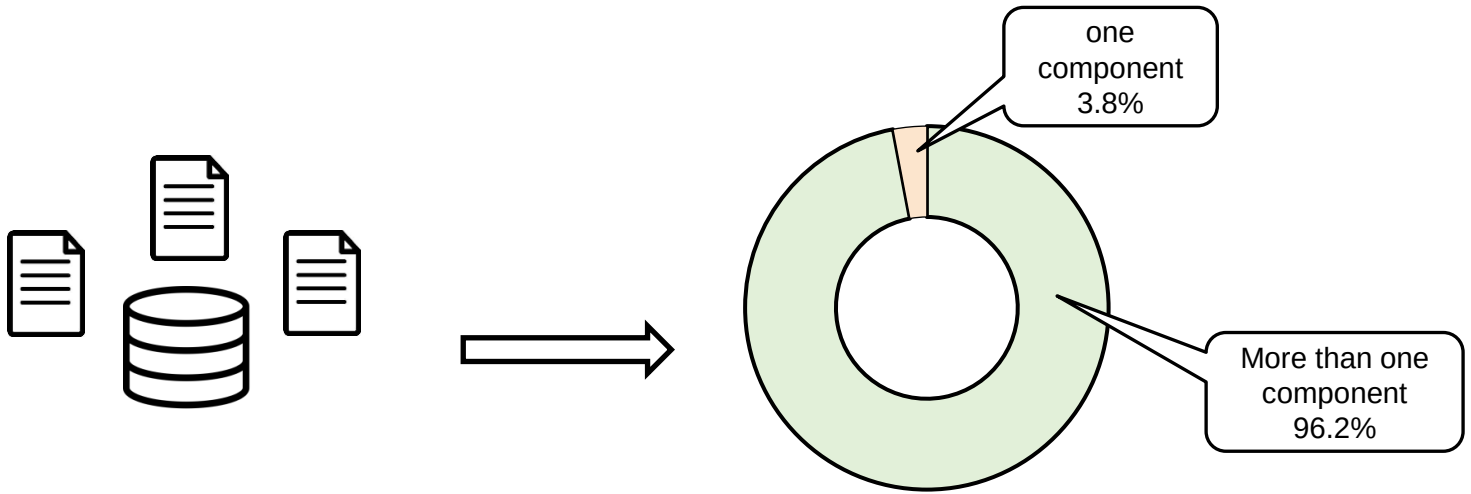
# What Configuration Dependencies Exist in FS

- Study Methodology
  - Analyze the source code and bug patches of Ext4 and XFS ecosystem
    - Same as previous studies, e.g., cDEP@ECSE/FSE'20



# What Configuration Dependencies Exist in FS

- Finding#1: Majority of cases (96.2%) involve critical parameters from more than one component



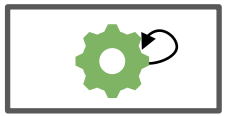

***78 bug patches***

***Distribution of 78 bug cases***

# What Configuration Dependencies Exist in FS

- Finding#2: A hierarchy of configuration dependencies exist in file systems!

**1. Self Dependency (SD)**

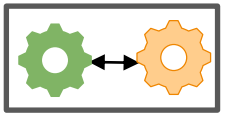


**Example:**

```
$ mke2fs -b 1024
```

***Range 1024 – 65536 bytes***

**2. Cross-parameter Dependency (CPD)**

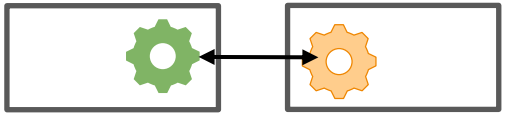


**Example:**

```
$ mke2fs -O resize_inode,
                ^meta_bg
```

***Cannot be enabled together*** ✗

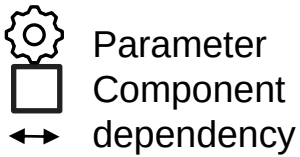
**3. Cross-component Dependency (CCD)**



**Example:**

```
$ mke2fs -O inline_data
$ mount -o dax
```

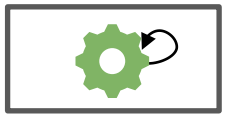
***Cannot be enabled together*** ✗



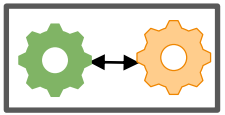
# What Configuration Dependencies Exist in FS

- Finding#2: A hierarchy of configuration dependencies exist in file systems!

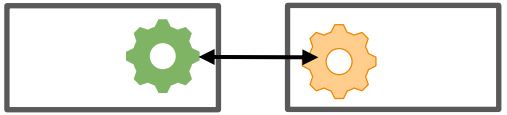
1. **Self Dependency (SD)**



2. **Cross-parameter Dependency (CPD)**



3. **Cross-component Dependency (CCD)**

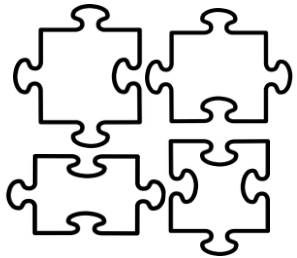


| # of bug cases | Multilevel Dependencies | | |
|---|---|---|---|
| | SD | CPD | CCD |
| 78 | 78 (100%) | 8 (10.3%) | 75 (96.2%) |

# What Configuration Dependencies Exist in FS

- Finding#3: Configuration parameters are handled in various ways in an FS ecosystem

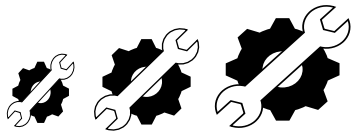
**Four types of configuration variable**



**Two ways to store in superblock**

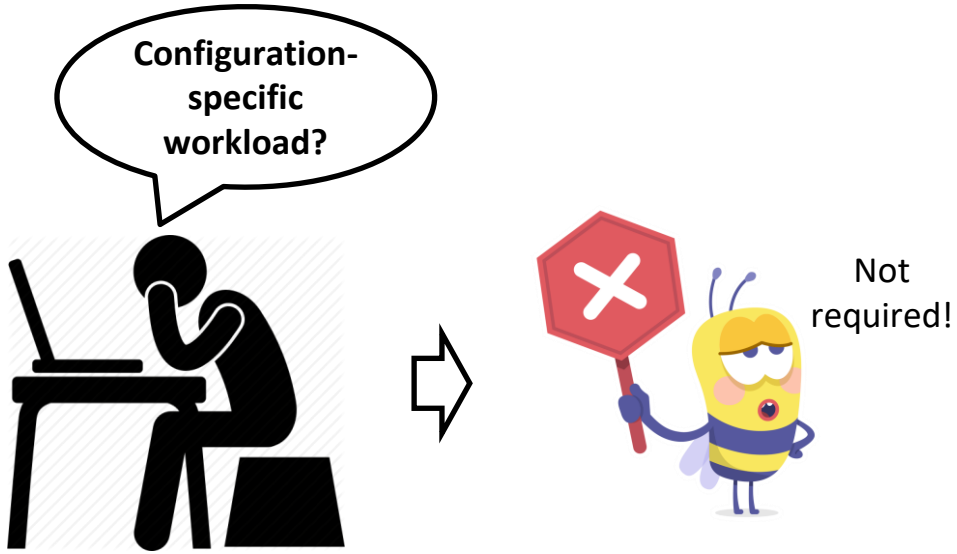
**Two ways to load from superblock**

**Various functions for handling configuration**



# What Configuration Dependencies Exist in FS

- Finding#4: Majority of cases (71.8%) do not require configuration-specific workloads!



# Lessons Learned

Finding #1 &#2 ⇨ Considering multi-components is necessary

Multiple components

Hierarchy of dependencies

Finding #3 ⇨ Requires new design to handle heterogeneity of parameters

Handling of parameters

Finding #4 ⇨ Existing efforts can be reused for stressing file systems

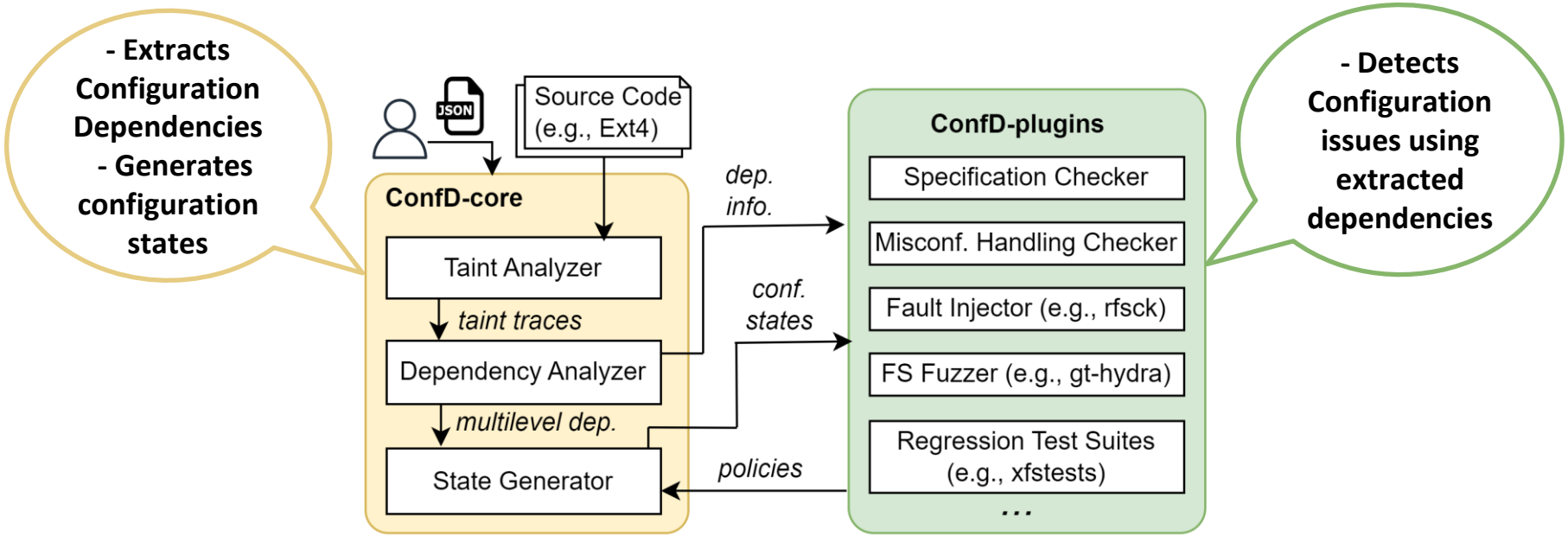
Configuration-specific workload

# Outline

- ~~Introduction~~
- ~~Background~~
- ~~What Configuration Dependencies Exist in File systems~~
- How to Extract & Use Multilevel Configuration Dependencies
- Evaluation
- Conclusion & Future Work

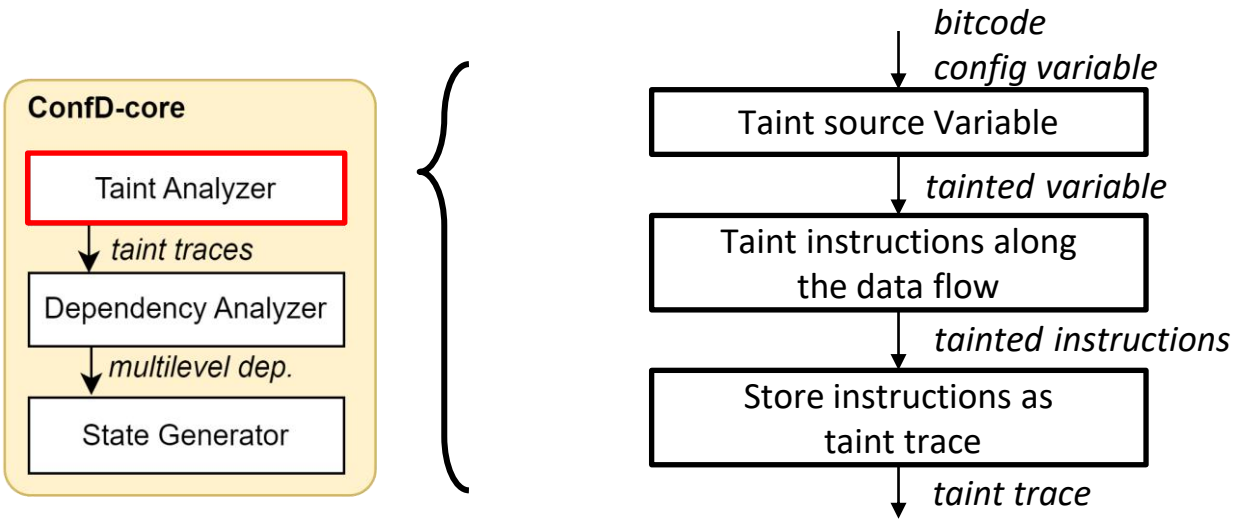
# ConfD: Leveraging dependency to address issues

- Overview:
  - ConfD-core & ConfD-plugins



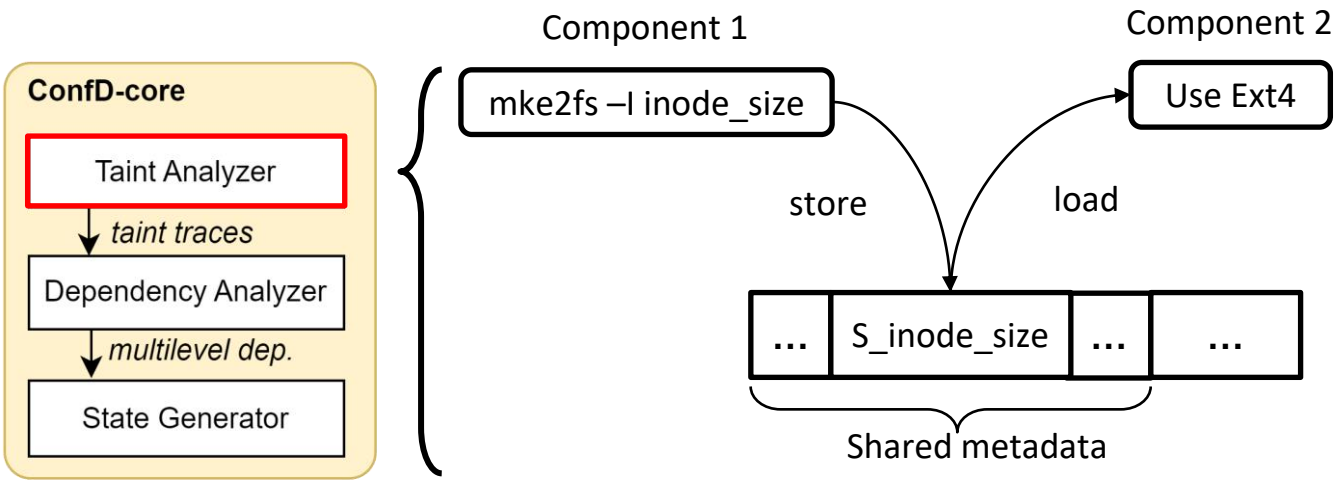
# ConfD-core: Extracting configuration dependencies

- Metadata-assisted Taint Analysis (Taint Analyzer)
  - Generates taint traces to capture the propagation of configuration parameters



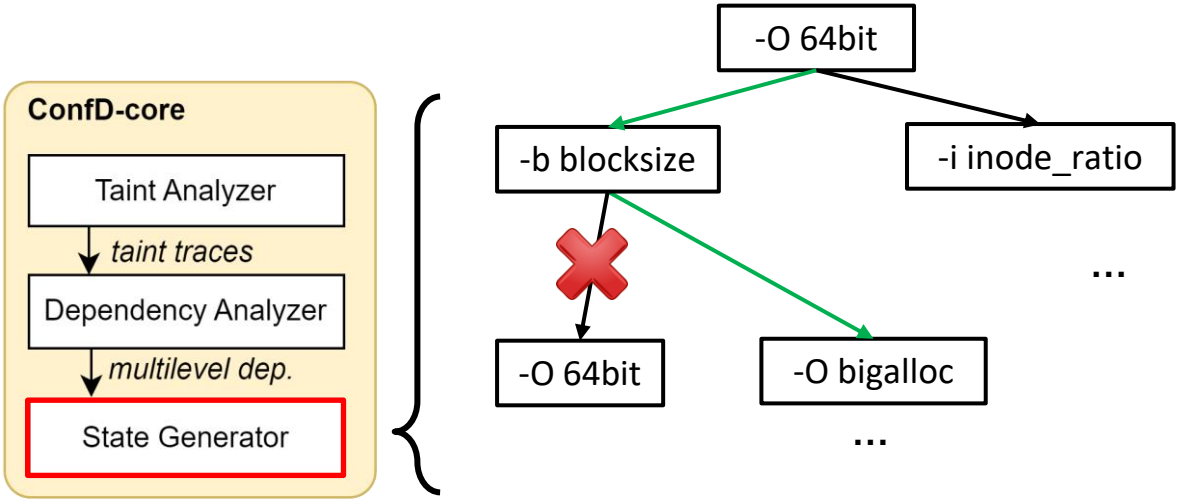
# ConfD-core: Extracting configuration dependencies

- Metadata-assisted Taint Analysis (Taint Analyzer)
  - Generates taint traces to capture the propagation of configuration parameters
  - Connects components using shared metadata structures



# ConfD-core: Extracting configuration dependencies

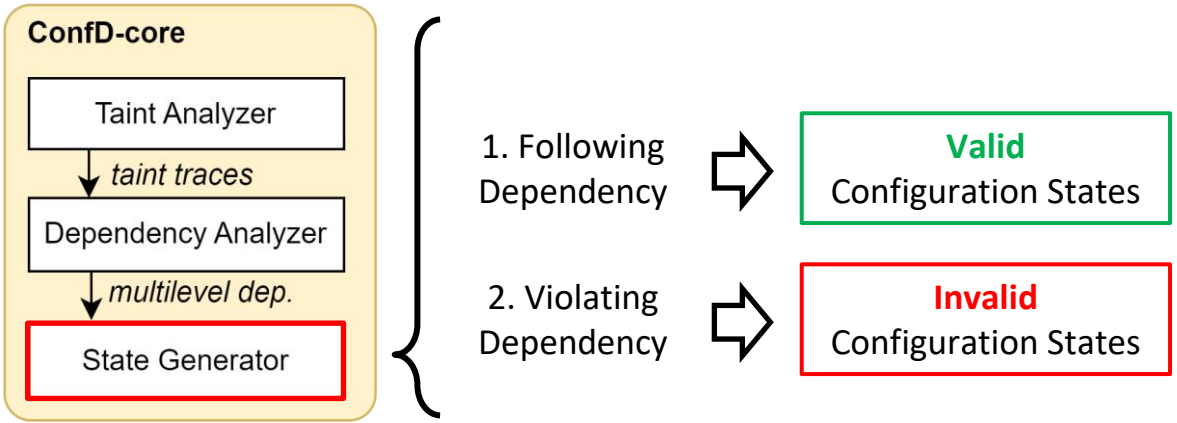
- Dependency-guided State Generation (State Generator)
  - Selectively generate states leveraging the extracted multilevel dependencies
  - Reduces the space of testing the combinations




$mke2fs –O 64bit,bigalloc –b blocksize

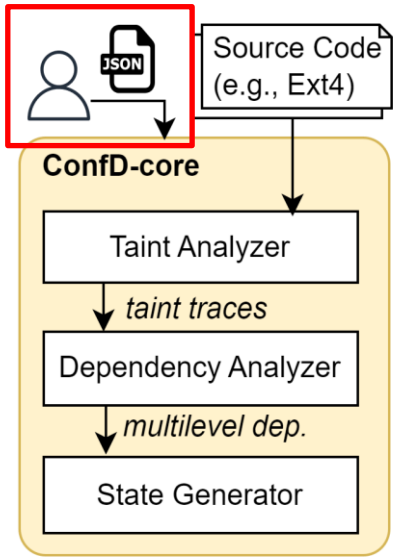
# ConfD-core: Extracting configuration dependencies

- Dependency-guided State Generation (State Generator)
  - Selectively generate states leveraging the extracted multilevel dependencies
  - Reduces the space of testing the combinations
  - Two policies



# ConfD-core: Extracting configuration dependencies

- User Input
  - Requires three types of input specified in one JSON file



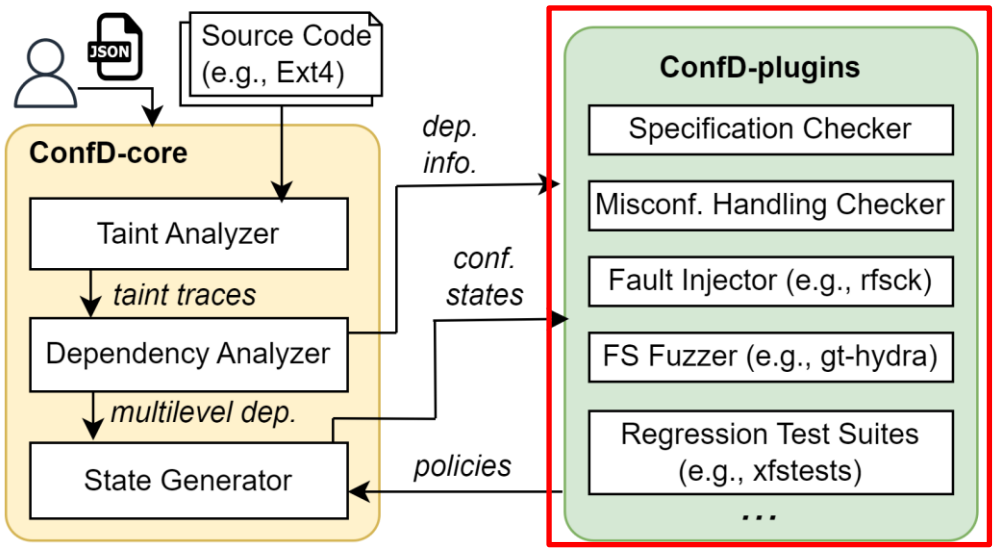

1.Function name as entry point

2.Configuration Variables and superblock

3.Command line syntax of parameters

# ConfD: Leveraging dependency to address issues

- Overview:
  - ConfD-core & ConfD-plugins



# ConfD-plugins: Using configuration dependencies

- Extending ConfD-core using Plugins
  - The current prototype of ConfD includes **6 plugins** for Linux file systems

| Plugin ID | Description | Base Tool | ConfD Plugin |
|---|---|---|---|
| #1 | Configuration specification checker | N/A | ConfD-specCk |
| #2 | Misconfiguration handling checker | N/A | ConfD-handlingCk |
| #3 | An open-source fault injector for file systems | rfsck | ConfD-rfsck |
| #4 | An open-source fuzzer for file systems | gt-hydra | ConfD-gt-hydra |
| #5 | Regression test suite for Linux file systems | Xfstests | ConfD-xfstests |
| #6 | Regression test suite for Ext4 utilities | e2fsprogs | ConfD-e2fsprogs |

# ConfD-plugins: Using configuration dependencies

- Extending ConfD-core using Plugins
  - Plugin#1: Automatically extract dependencies from the man-pages
    - Detects documentation issues

| Plugin ID | Description | Base Tool | ConfD Plugin |
|---|---|---|---|
| #1 | Configuration specification checker | N/A | ConfD-specCk |
| #2 | Misconfiguration handling checker | N/A | ConfD-handlingCk |
| #3 | An open-source fault injector for file systems | rfsck | ConfD-rfsck |
| #4 | An open-source fuzzer for file systems | gt-hydra | ConfD-gt-hydra |
| #5 | Regression test suite for Linux file systems | Xfstests | ConfD-xfstests |
| #6 | Regression test suite for Ext4 utilities | e2fsprogs | ConfD-e2fsprogs |

# ConfD-plugins: Using configuration dependencies

- Extending ConfD-core using Plugins
  - Plugin#2: Uses “violating dependency” policy of the State Generator
    - Detects bad reaction due to misconfiguration

| Plugin ID | Description | Base Tool | ConfD Plugin |
|---|---|---|---|
| #1 | Configuration specification checker | N/A | ConfD-specCk |
| #2 | Misconfiguration handling checker | N/A | ConfD-handlingCk |
| #3 | An open-source fault injector for file systems | rfsck | ConfD-rfsck |
| #4 | An open-source fuzzer for file systems | gt-hydra | ConfD-gt-hydra |
| #5 | Regression test suite for Linux file systems | Xfstests | ConfD-xfstests |
| #6 | Regression test suite for Ext4 utilities | e2fsprogs | ConfD-e2fsprogs |

# ConfD-plugins: Using configuration dependencies

- Extending ConfD-core using Plugins
  - Plugin#3 &#4: Leverages dependency-guided states to generate FS images
    - Improves the base-tool

| Plugin ID | Description | Base Tool | ConfD Plugin |
|---|---|---|---|
| #1 | Configuration specification checker | N/A | ConfD-specCk |
| #2 | Misconfiguration handling checker | N/A | ConfD-handlingCk |
| #3 | An open-source fault injector for file systems | rfsck | ConfD-rfsck |
| #4 | An open-source fuzzer for file systems | gt-hydra | ConfD-gt-hydra |
| #5 | Regression test suite for Linux file systems | Xfstests | ConfD-xfstests |
| #6 | Regression test suite for Ext4 utilities | e2fsprogs | ConfD-e2fsprogs |

# ConfD-plugins: Using configuration dependencies

- Extending ConfD-core using Plugins
  - Plugin#5 & #6: Replaces configuration states in testcases with ConfD states
    - Improves the test suites

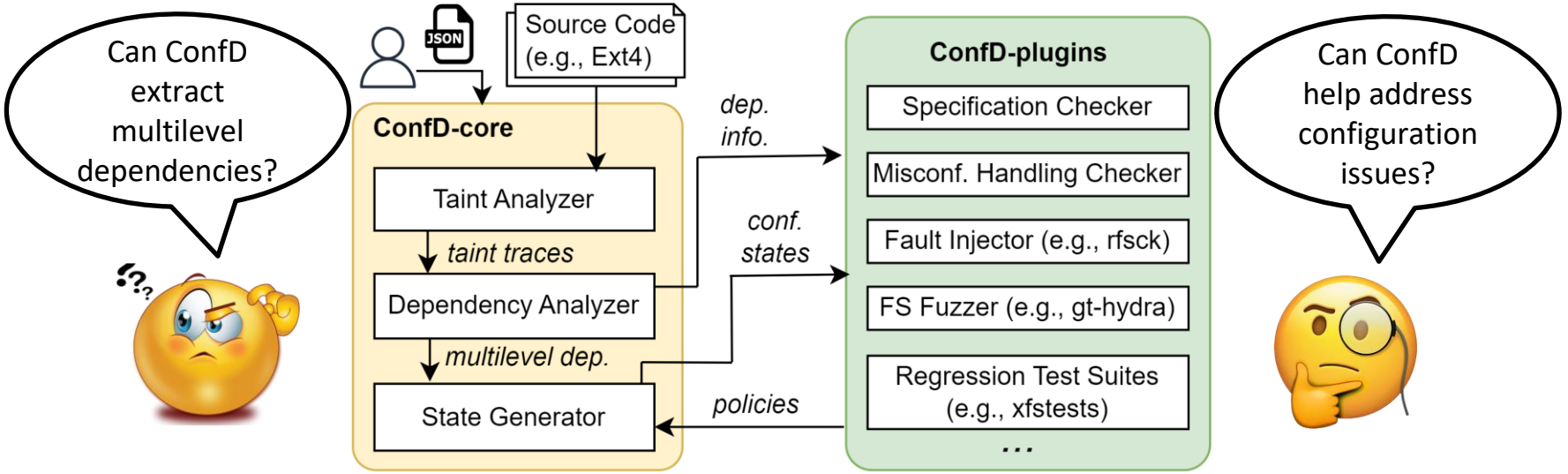
| Plugin ID | Description | Base Tool | ConfD Plugin |
|---|---|---|---|
| #1 | Configuration specification checker | N/A | ConfD-specCk |
| #2 | Misconfiguration handling checker | N/A | ConfD-handlingCk |
| #3 | An open-source fault injector for file systems | rfsck | ConfD-rfsck |
| #4 | An open-source fuzzer for file systems | gt-hydra | ConfD-gt-hydra |
| #5 | Regression test suite for Linux file systems | Xfstests | ConfD-xfstests |
| #6 | Regression test suite for Ext4 utilities | e2fsprogs | ConfD-e2fsprogs |

# Outline

- ~~Introduction~~
- ~~Background~~
- ~~What Configuration Dependencies Exist in File systems~~
- ~~How to Extract & Use Multilevel Configuration Dependencies~~
- Evaluation
- Conclusion & Future Work

# Evaluation of ConfD

- Evaluated ConfD on Ext4 and XFS ecosystems



# Can ConfD extract multilevel dependencies?

- **ConfD** extracted **154 unique dependencies** automatically

| Target FS Ecosystem | Self Dependency | Cross-parameter Dependency | Cross-component Dependency |
|---|---|---|---|
| Ext4 | 17 | 48 | 46 |
| XFS | 18 | 10 | 15 |
| Total | 35 | 58 | 61 |

154 Unique in total
with 8.4% false positive

# Can ConfD help address configuration issues?

**Yes!**

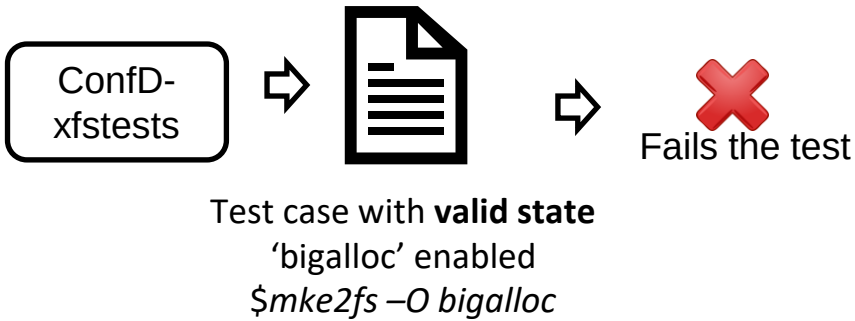
| ConfD-Plugins | Types of Issues Reported (#) |
|---|---|
| ConfD-specCk | Undoc./wrong documentation (17) |
| ConfD-handlingCk | Bad reaction (18) |
| ConfD-xfstests | Test case failure (9) |
| ConfD-e2fsprogs | Test case failure (1) |
| ConfD-rfsck | Uncorrectable images (280) |
| ConfD-gt-hydra | Hangs (18) |

# ConfD-xfstests

- Based on xfstests
  - Test suite for testing file systems
- Checks test case failure with valid configuration

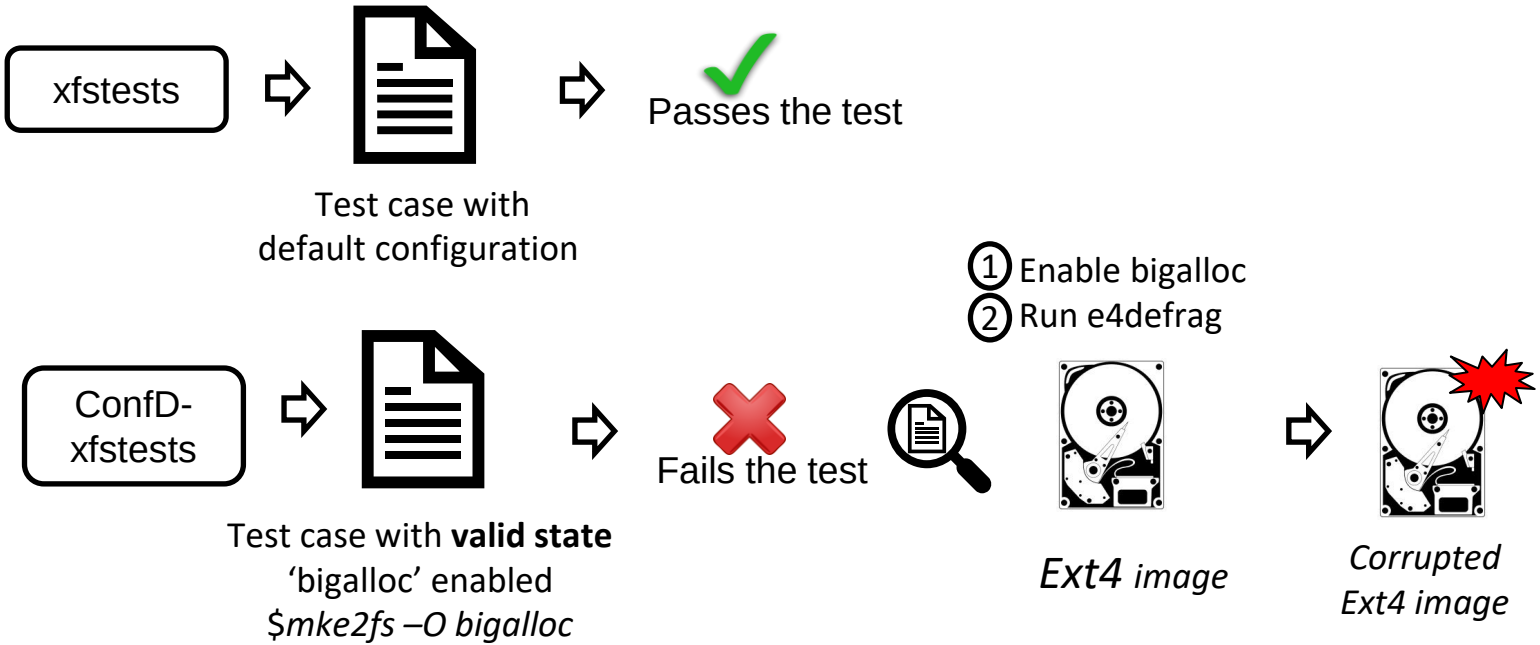
# ConfD-xfstests

- Example

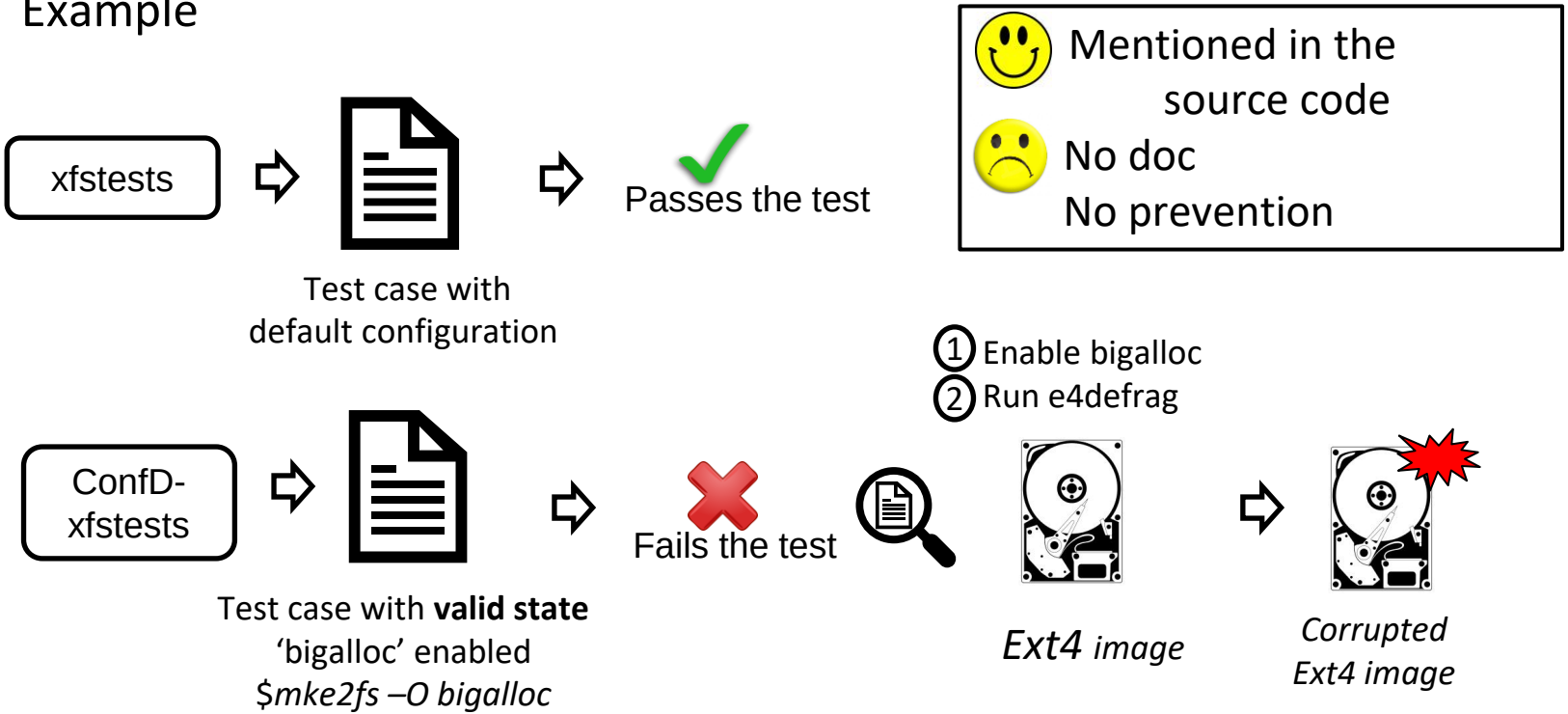
# ConfD-xfstests

- Example



# ConfD-xfstests

- Example



# ConfD-rfsck

- Based on rfsck
  - rfsck: an open-source fault injector for testing FS checkers [rfsck@FAST'18]
    - It reports # of uncorrectable images under fault and observes symptoms
- Improves rfsck by using dependency-guided configuration states

# ConfD-rfsck

- Triggered **two new** types of symptoms

**rfsck**

**Truncated file content**

**ConfD-rfsck**

**Truncated file content**

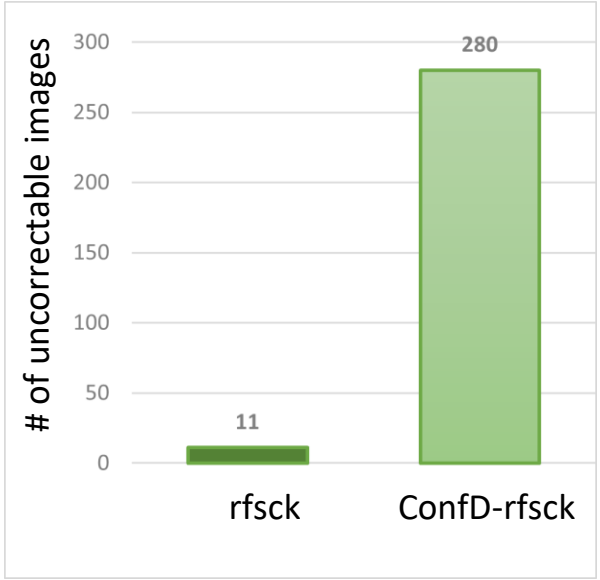
**Un-mountable**

**Invalid file content**

rfsck ⇨ uses default configuration state
ConfD-rfsck ⇨ uses dependency-guided non-default states

# ConfD-rfsck

- Generated more # of uncorrectable images




rfsck ⇨ uses default configuration state
ConfD-rfsck ⇨ uses dependency-guided non-default states

# ConfD-gt-hydra

- Based on gt-hydra
  - gt-hydra: an open-source fuzzer for the FS [Hydra@SOSP'19]
    - Checks # of issues found of fuzzed FS images
- Improves gt-hydra by using dependency-guided configuration states
- Triggered more issues than gt-hydra

| Target Fs | # of Issues Reported (in two weeks) | |
|---|---|---|
| | gt-hydra | ConfD-gt-hydra |
| Ext4 | 1 | 17 |

gt-hydra ⇨ uses default configuration state
ConfD-gt-hydra ⇨ uses dependency-guided non-default states

# ConfD-specCk

- Found 17 undocumented issues in the man-pages

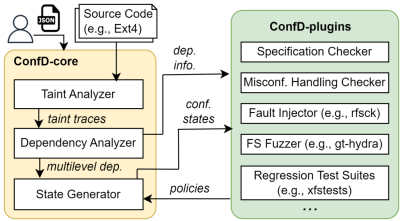
| Target FS Ecosystem | # of Undocumented/Wrong Dependencies | | | Total |
|---|---|---|---|---|
| | SD | CPD | CCD | |
| Ext4 | 7 | 4 | 2 | 13 |
| XFS | 2 | 2 | 0 | 4 |
| Total | 9 | 6 | 2 | 17 |

# FB-HYDRA vs. ConfD

- State generation comparison:
  - FB-HYDRA: configuration management framework by Facebook
  - ConfD selectively generates dependency-guided configuration states
    - Avoids duplicates and invalid states

| # of States | FB-HYDRA | ConfD |
|---|---|---|
| # of Total States | 56,592 | 30 |
| # of Duplicates | 42,745 (75.5%) | 0 |
| # of Invalid | 15,146 (26.8%) | 0 |


Vs.



# Conclusion & Future Work

- Conclusion
  - Studied real bugs and derived a taxonomy of configuration dependencies
  - Built ConfD framework for addressing configuration issues
  - Evaluated ConfD on Ext4 and XFS
- Future work
  - Extending ConfD
    - Support other FS and applications beyond FS, e.g., NDCTL, databases, etc.
    - Add new features to identify problematic parameters
  - Integration
    - e.g., FB-hydra, CI/CD frameworks, etc.

**ConfD is open-sourced!**

**https://github.com/data-storage-lab/ConfD**

# Thank You!