Unsafe at Any Copy: Name Collisions from Mixing Case-Sensitivities

Aditya Basu, John Sampson, Zhiyun Qian, Trent Jaeger
November 2022

Seminar Supercomputer: Forschung und Innovation

Arbeitsbereich Wissenschaftliches Rechnen
Fachbereich Informatik

Jonas Rochdi, 05.12.2023

Betreuer: Anna Fuchs, Jannek Squar

1. **Einführung**
2. **Auf Name Collisions testen**
3. **Name Collisions bei Linux Dienstanwendungen**
4. **Fallstudien** – rsync, dpkg und Apache httpd
5. **Mögliche Verteidigungen**
6. **Schluss**
7. **Quellen**

Ziel des Papers: Name Collisions vs. Sicherheit

Wie rufen Anwendungen Dienstprogramme auf, die möglicherweise unsichere Namenskollisionen ermöglichen?

Wann erlauben die Dienstprogramme zur Durchführung von Kopiervorgängen unsichere Namenskollisionen?

Welche Korrektheits- und Sicherheitsprobleme werden durch Namenskollisionen verursacht?

Was sind Name Confusions?

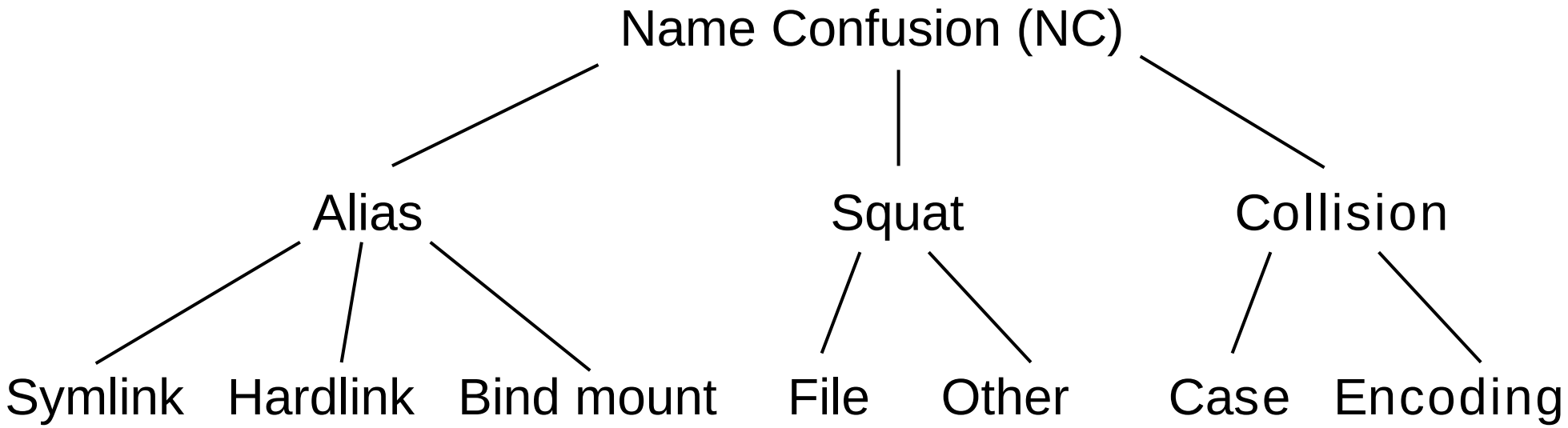


Abbildung 1: alias (= multiple names for a resource), collision (= multiple resources for a name) und squat (= temporal ambiguities in names vs. resources) classes

Case-Sensitive oder Case-Insensitive?[1]

`Foo.o` bleibt〉 `Foo.o`

`foo.o` `foo.o`

Case-Sensitive oder Case-Insensitive[1]?

`FOO.o`

`foo.o`

konflikt 〉

`FOO.o`

oder

`foo.o`

Case-Sensitive oder Case-Insensitive[1]?

`floss.o`
`FLOSS.o`
`floß.o`

konflikt 〉

`floss.o`
oder
`FLOSS.o`
oder
`floß.o`

File Name Confusion Attacks[2]:

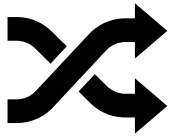
1. Nutzt Unsicherheit oder Schwächen in Behandlung von Dateinamen aus
2. Neuer Angriffstyp: Namenskollision herbeiführen durch Groß-/Kleinschreibung
3. Zunehmende Wahrscheinlichkeit durch z.B. WSL und Trends wie per-directory case-insensitivity

Konkrete Form: Bösartige Symlinks

1. Ziel: legitime Dateien durch Symlinks auf bösartige Pendants umleiten ➔ Schadcode in sicherheitskritische Dateien einfügen
2. TOCTTOU Angriff: Diskrepanz zwischen Time Of Check und Time Of Use
3. Symlink zu einer als sicher geltenden Datei ➔ wird während der Ausführung von einer bösartigen Datei ersetzt

Vielfältige Design Choices

Benutzer/Entwickler überlassen



Systeme unterstützen oft mix aus case-sensitive und case-insensitive Dateisystemen

Viele Dateisysteme unterstützen die Wahl des case für individuelle verzeichnisse

Vielfältige Design Choices

Stellt Datei- und Druckdienste für Windows-Clients in einem Linux/Unix-Netzwerk bereit[3]

Auf vielen Linux-Distributionen verbreitetes Dateisystem[4]

TmpFS

temporäres Dateisystem, das im RAM existiert[5]

Git Exploit CV-2021-21300[6]

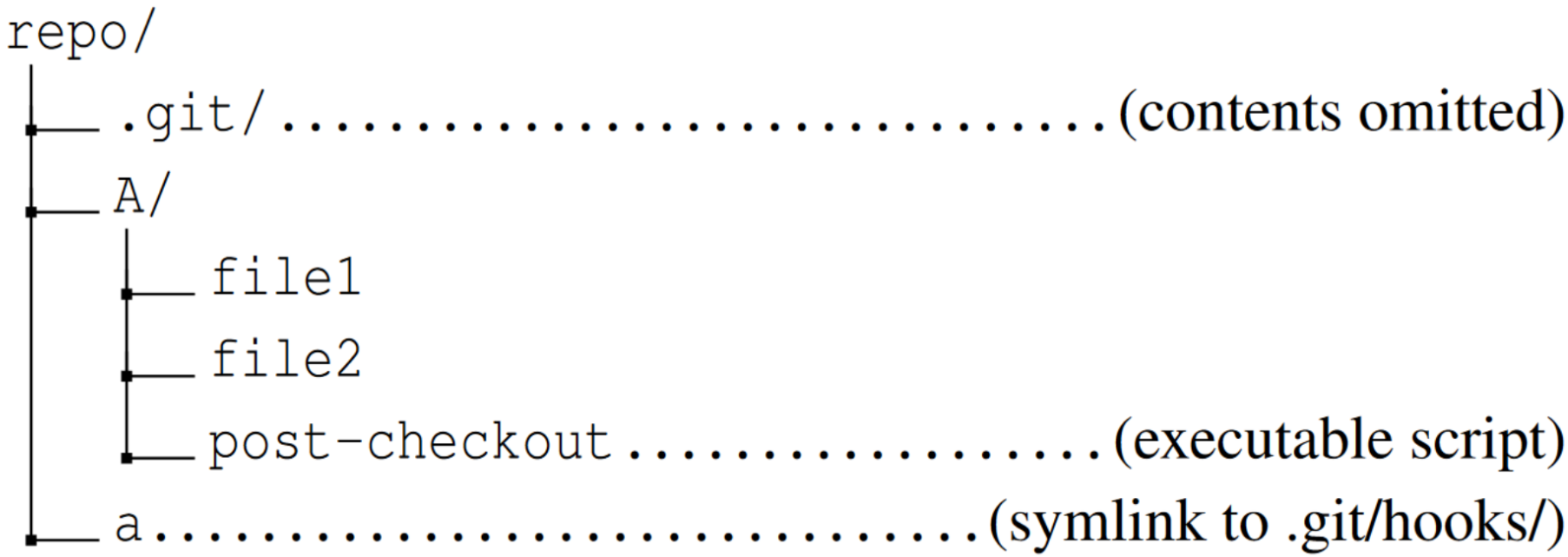
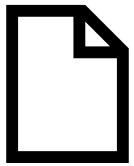


Abbildung 2: Beispiel für Git CVE-2021-21300

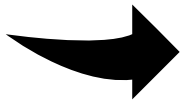
1. **Einführung**
2. **Auf Name Collisions testen**
3. **Name Collisions bei Linux Dienstanwendungen**
4. **Fallstudien** – rsync, dpkg und Apache httpd
5. **Mögliche Verteidigungen**
6. **Schluss**
7. **Quellen**

Testfallgenerierung

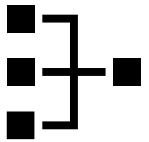
- Erstellt Testfälle mit Quell- und Zielressourcen aller Kombinationen von Typen

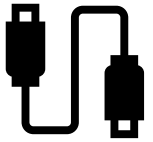

Datei


Verzeichnis


Symlink

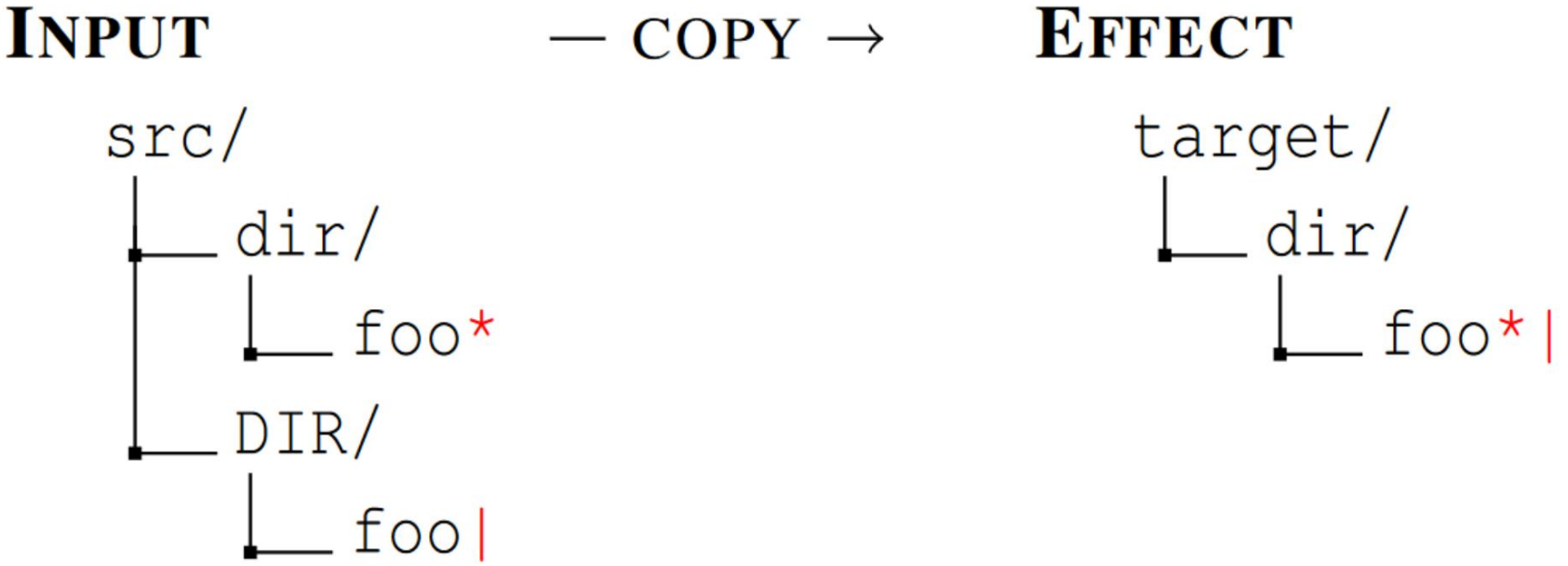

Hardlink


Pipe


Gerät

- In verschiedenen Tiefen, um Kollisionen bei verschiedenen Ebenen zu prüfen

Auf Name Collisions testen



Here, '*' means a regular file and '|' means file type is a named pipe.

Abbildung 3: Ein Beispiel Testfall

Wann verzeichnen wir ein Positiv?

Pfad in use ≠ Pfad in create

oder

use löscht und ersetzt resource aus vorherigem create

Wann verzeichnen wir ein Positiv?

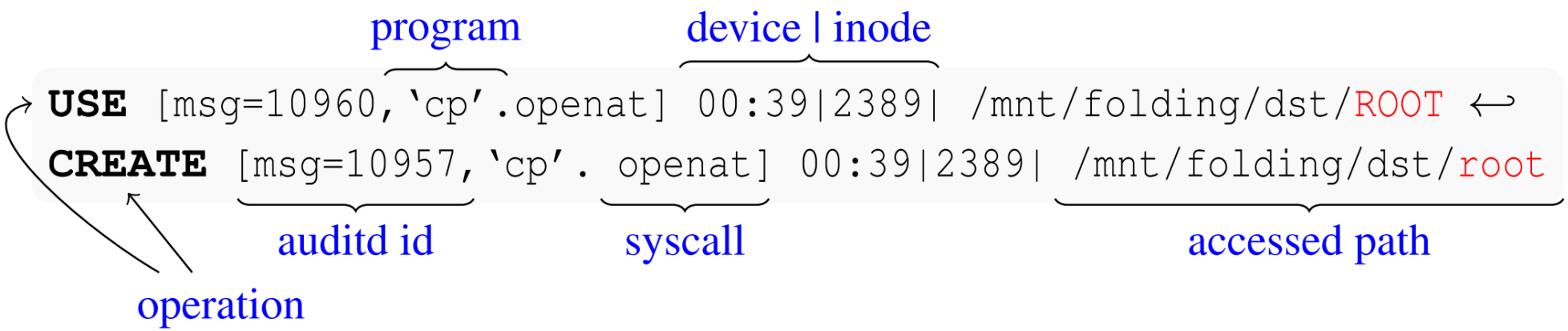


Abbildung 4: Beispielmeldung bei einer Name Collision

1. **Einführung**
2. **Auf Name Collisions testen**
3. **Name Collisions bei Linux Dienstanwendungen**
4. **Fallstudien** – rsync, dpkg und Apache httpd
5. **Mögliche Verteidigungen**
6. **Schluss**
7. **Quellen**

Untersuchte Anwendungen:

1. tar (Archivierungstool)
2. zip (Archivierungstool)
3. cp (Anwendung zum Kopieren)
4. cp* (Erweitert cp um Wildcards)
5. rsync (Dateiübertragung & -synchronisation)
6. Dropbox (Cloud-basierte Dateisynchronisation)

Löchen & Rekreieren (×)

Target
FOO.o
24.01.2002, 10 MB
Lorem ipsum dolor sit amet, consetetur sadipscing elitr, sed diam nonumy eirmod tempor invidunt ut labore et dolore magna aliquyam erat, sed diam voluptua. At vero eos et accusam et justo duo dolores et ea rebum.

Source
foo.o
13.03.2020, 99 MB
01010111111000000000 01111111000000011111 11101000101010001010 01001110100010110100 10011101000101101001 01000111101000100101 01011111010010010000 10101111010010010010 00101000010010010101 01111101010101001010 10010100101010011110 00001111000011010011

Target
foo.o
13.03.2020, 99 MB
01010111111000000000 01111111000000011111 11101000101010001010 01001110100010110100 10011101000101101001 01000111101000100101 01011111010010010000 10101111010010010010 00101000010010010101 01111101010101001010 10010100101010011110 00001111000011010011

Überschreiben (+)

Target
FOO.o
24.01.2002, 10 MB
Lorem ipsum dolor sit amet, consetetur sadipscing elitr, sed diam nonumy eirmod tempor invidunt ut labore et dolore magna aliquyam erat, sed diam voluptua. At vero eos et accusam et justo duo dolores et ea rebum.

Source
foo.o
13.03.2020, 99 MB
01010111111000000000 01111111000000011111 11101000101010001010 01000111101000100101 10101111010010000100 00101000010010010101 01111101010101001010 10010100101010011110 00001111000011010011

Target
FOO.o
13.03.2020, 99 MB
01010111111000000000 01111111000000011111 11101000101010001010 01000111101000100101 10101111010010000100 00101000010010010101 01111101010101001010 10010100101010011110 00001111000011010011

Corrupt (C)

Target
FOO.o
24.01.2002, 10 MB
Lorem ipsum dolor sit amet, consetetur sadipscing elitr, sed diam nonumy eirmod tempor invidunt ut labore et dolore magna aliquyam erat, sed diam voluptua. At vero eos et accusam et justo duo dolores et ea rebum.

Source
foo.o
13.03.2020, 99 MB
0101011111100000 *0000011111100000* *0001111111010001* *0101000101001011* *1010001011010010* *1000111101000100* *0101010111101001* *0100001001010001* *0010010010101111* *0101010100101011*0

Not involved
bar.o
11.11.2011, 11 MB
abcbabcbbcbcbbaaa *bcbcbcbabcbabcbcba* *bcbacbabbabcbabcb* *acbacbacbcbcbcbcbc* *babacacababababcbc* *babbabacacacacaccc* *acacaaccabcabcabca* *cbababbabccacacaba* *bacacacbabaaacaccb* *abbababacacacaaaa*

Not involved
bar.o
13.03.2020, 99 MB
0101011111100000 *0000011111100000* *0001111111010001* *0101000101001011* *1010001011010010* *1000111101000100* *0101010111101001* *0100001001010001* *0010010010101111* *0101010100101011*0

Metadaten Mismatch (≠)

Target
FOO.o
24.01.2002, 10 MB
Lorem ipsum dolor sit amet, consetetur sadipscing elitr, sed diam nonumy eirmod tempor invidunt ut labore et dolore magna aliquyam erat, sed diam voluptua. At vero eos et accusam et justo duo dolores et ea rebum.

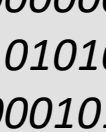
Source
foo.o
13.03.2020, 99 MB
01010111111000000000 01111111000000011111 11101000101010001010 01001110100010110100 10011101000101101001 01000111101000100101 01011111010010010000 10010100010010010010 01111010101010010101 10010100101010011110 00001111000011010011

Target
FOO.o
24.01.2002, 10 MB
01010111111000000000 01111111000000011111 11101000101010001010 01001110100010110100 10011101000101101001 01000111101000100101 01011111010010010000 10010100010010010010 01111010101010010101 10010100101010011110 00001111000011010011

Umbenennung ($\boldsymbol{R}$)

Target
FOO.o
24.01.2002, 10 MB
Lorem ipsum dolor sit amet, consetetur sadipscing elitr, sed diam nonumy eirmod tempor invidunt ut labore et dolore magna aliquyam erat, sed diam voluptua. At vero eos et accusam et justo duo dolores et ea rebum.

Source
foo.o
13.03.2020, 99 MB
01010111111000000000 01111111000000011111 11101000101010001010 01001110100010110100 10011101000101101001 01000111101000100101 01011110100100100001 00101000010010010101 01111101010100100101 10010100101010011110 00001111000011010011 0



Target
FOO.o
24.01.2002, 10 MB
Lorem ipsum dolor sit amet, consetetur sadipscing elitr, sed diam nonumy eirmod tempor invidunt ut labore et dolore magna aliquyam erat, sed diam voluptua. At vero eos et accusam et justo duo dolores et ea rebum.

Target 2
FOO-2.o
13.03.2020, 99 MB
01010111111000000000 01111111000000011111 11101000101010001010 01001110100010110100 10011101000101101001 01000111101000100101 01011110100100100001 00101000010010010101 01111101010100100101 10010100101010011110 00001111000011010011 0

Den Benutzer fragen (A)

Target
FOO.o
24.01.2002, 10 MB
Lorem ipsum dolor sit amet, consetetur sadipscing elitr, sed diam nonumy eirmod tempor invidunt ut labore et dolore magna aliquyam erat, sed diam voluptua. At vero eos et accusam et justo duo dolores et ea rebum.

Source
foo.o
13.03.2020, 99 MB
01010111111000000000 01111111000000011111 11101000101010001010 01001110100010110100 10100011110100010010 10101111010010010001 00101000010010010101 01111101010101001010 10010100101010011110 00001111000011010011

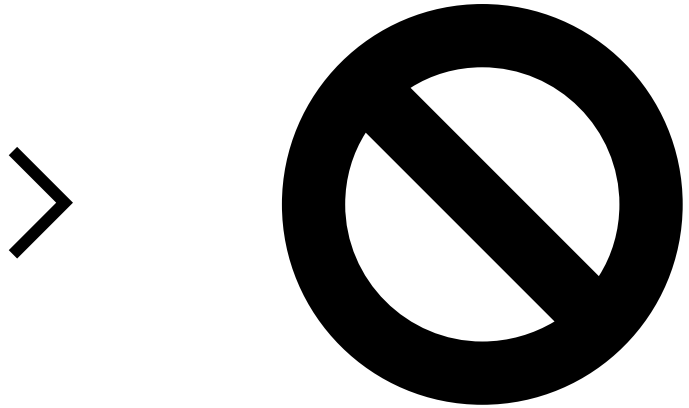
?

Benutzereingabe
Überschreiben
Umbennen
Überspringen
Abbrechen
[…]

Ablehnung (E)

Target
FOO.o
24.01.2002, 10 MB
Lorem ipsum dolor sit amet, consetetur sadipscing elitr, sed diam nonumy eirmod tempor invidunt ut labore et dolore magna aliquyam erat, sed diam voluptua. At vero eos et accusam et justo duo dolores et ea rebum.

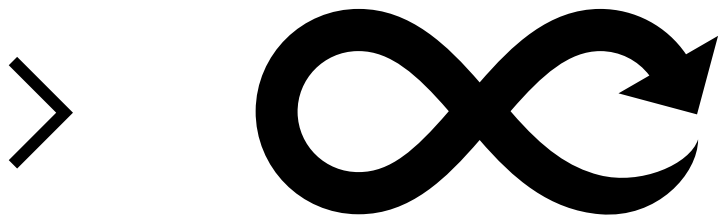
Source
foo.o
13.03.2020, 99 MB
01010111111000000000 01111111000000011111 11101000101010001010 01001110100010110100 10 01000111101000100101 01011111010010000 1 00101000010010010 1 01111101010100101 01 10010100101010011110 0 00001111000011010011 0



Crashes (∞)

Target
FOO.o
24.01.2002, 10 MB
Lorem ipsum dolor sit amet, consetetur sadipscing elitr, sed diam nonumy eirmod tempor invidunt ut labore et dolore magna aliquyam erat, sed diam voluptua. At vero eos et accusam et justo duo dolores et ea rebum.

Source
foo.o
13.03.2020, 99 MB
01010111111000000000 01111111000000011111 11101000101010001010 01001110100010110100 10 01000111101000100101 01011111010010000 1 00101000010010010 1 01111010101010010101 10010100101010011110 00001111000011010011 0



Ergebnisse

Tabelle 1: Ergebnisse der Testfälle bei verschiedenen Ressourcen und Anwendungen

Name Collision between							
Target Type	Source Type	`tar`	`zip`	`cp`	`cp*`	`rsync`	Dropbox
file	file	×	*A*	*E*	+≠	+≠	*R*
symlink (to file)	file	×	*A*	*E*	+*T*	+≠	*R*
pipe/device	file	×	–	*E*	+	+	–
hardlink	file	×	–	*E*	+≠	+≠	–
hardlink	hardlink	*C*×	–	*E*	*C*×	*C*+≠	–
directory	directory	+≠	+≠	*E*	+≠	+≠	*R*
symlink (to directory)	directory	+	∞	*E*	*E*	+*T*	*R*

Unsichere Responses auf Name Collisions

Stiller Datenverlust mit tar, cp* & rsync

Stiller Datenverlust
tar (Delete & Recreate)
cp* (Overwrite)
rsync (Overwrite)

Den Benutzer fragen (potentiell stiller Datenverlust)
zip
cp

Merge von Verzeichnissen mit tar, zip, cp* & rsync

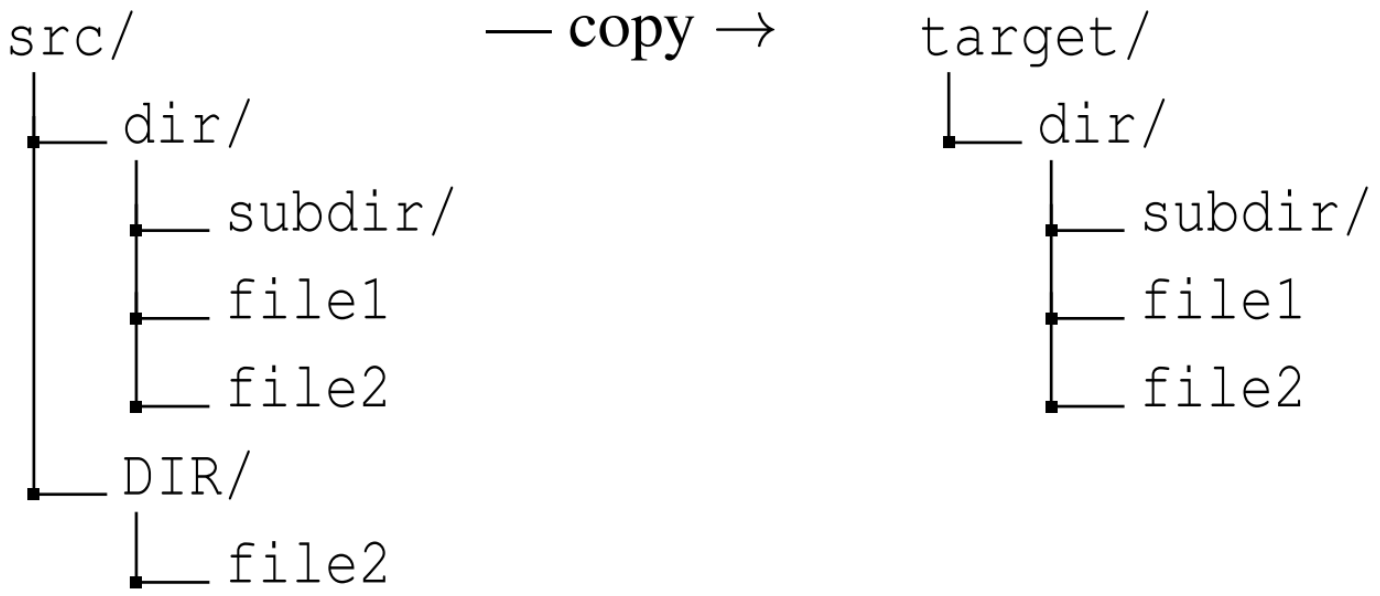


Abbildung 5: Beispielmeldung bei eines Verzeichnis-Merge durch Name Collision

Hard Link – Hard Link

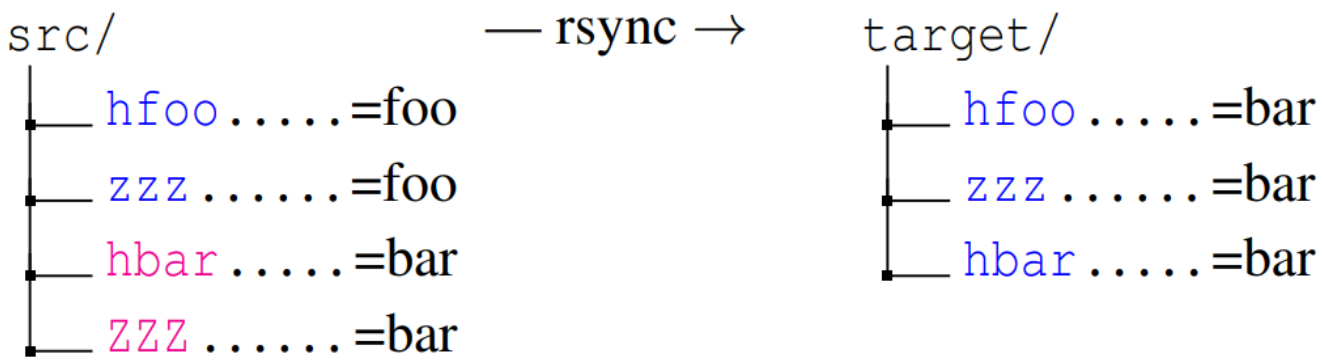


Abbildung 6: Hardlinküberschreibung durch Namenskollision

1. Einführung
2. Auf Name Collisions testen
3. Name Collisions bei Linux Dienstanwendungen
4. **Fallstudien** – rsync, dpkg und Apache httpd
5. Mögliche Verteidigungen
6. Schluss
7. Quellen

Was ist dpkg[7]?

1. Debian Package Manager. Paketverwaltungs- und Installationsdienst für Debian-basierte Linux-Betriebssysteme
2. Paketformat .deb. Pakete im Format .deb, komprimierte Tarballs enthalten Informationen und Dateien für die Installation
3. Datei-Tracking-Datenbank. Führt eine Datenbank, die alle während der Installation erstellten Dateien verfolgt ➔ Vermeidung von Konflikten

Was ist das Problem?

1. Case-sensitivity in der Datenbank. Vergleicht Dateinamen case-sensitive ➔ neue Pakete können Dateien älterer Pakete mit überschreiben
2. Sicherheitsrisiken bei Konfigurationsdateien. Config-Dateien können manipuliert
3. Mögliche Sicherheitslücken. Dienste sind durch unerwünschte Überschreibungen von config-Dateien gefährdet ➔ Sicherheitslücken

Was wurde dagegen unternommen?

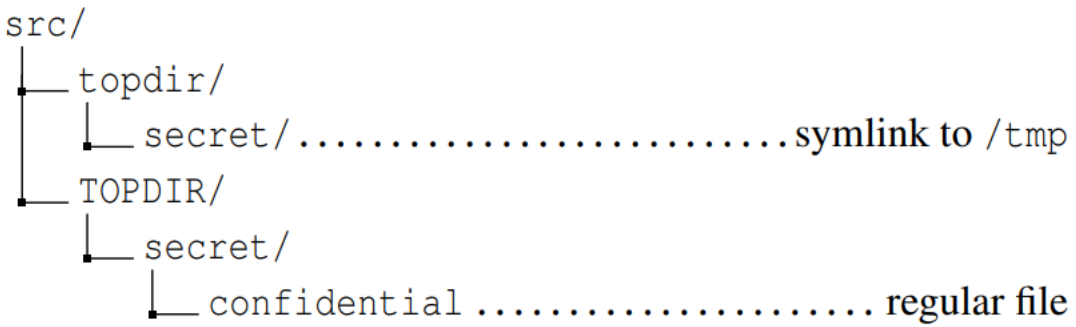
1. Meldung an die dpkg-Wartenden. Die festgestellten Probleme wurden den Verantwortlichen von dpkg gemeldet.
2. Dokumentationsaktualisierung. Paketdokumentation aktualisiert; Warnung vor der Verwendung in case-insensitiven Umgebungen
3. Analyse und Statistiken. Es wurde eine Analyse von 74.688 Paketen durchgeführt, bei der festgestellt wurde, dass 12.237 Dateinamen kollidieren würden

Was ist rsync[8]?

1. Datei-Synchronisationswerkzeug. Werkzeug für die Synchronisation von Dateien und Verzeichnissen auf Linux-Systemen
2. Kopierverhalten. Kopiert Daten zwischen Quell- und Zielverzeichnissen
3. Backups. Ist häufig für Backup- und Datenübertragungszwecke im Einsatz

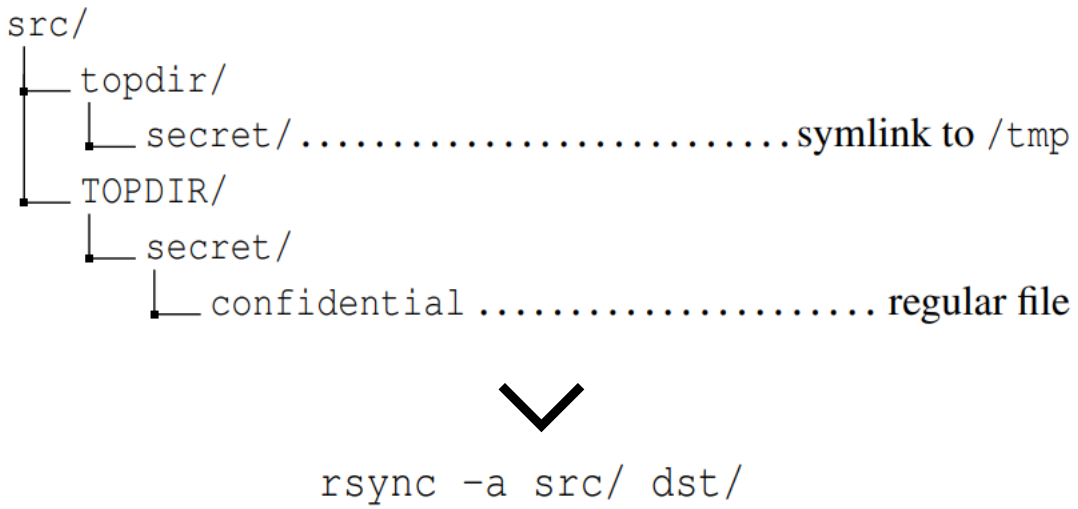
Was ist das Problem?

case-sensitive



Was ist das Problem?

case-sensitive



Was ist das Problem?

case-sensitive

```
src/
├── topdir/
│   └── secret/ ..... symlink to /tmp
└── TOPDIR/
    └── secret/
        └── confidential ..... regular file
```

```
rsync -a src/ dst/
```

case-insensitive

```
dst/
├── TOPDIR/
│   └── secret/ ..... symlink to /tmp
└── /tmp/confidential ..... link traversal
```

rsync has created the /tmp/confidential file by following the symbolic link dst/TOPDIR/secret.

Abbildung 7: Namenscollision bei rsync

Was wurde dagegen unternommen?

1. Meldung an rsync-Maintainer
2. Empfehlung für Nutzer. Rsync nicht mit case-insensitiven Dateisystemen verwenden

Was ist Apache httpd[9]?

1. Webserver-Software. Webserver, Zugriff auf Dateisystem über HTTP
2. Sicherheitsparameter. Verwendet UNIX Discretionary Access Control (DAC) Berechtigungen, um den Zugriff zu regeln

Was ist das Problem?

1. Sicherheitsparameteränderungen. Dienstprogramme können Berechtigungen ändern ➔ Sicherheitslücken
2. Name Collisions und Auswirkungen. Migration von case-sensitive ➔ case-insensitive: potentiell Name Collisions und veränderte Sicherheitsparameter
3. Folgen der Kollisionen: Tar ändert Metadaten falsch, DAC-Berechtigungen und Inhalte von Verzeichnissen fusionieren ➔ Sicherheitsprobleme

Szenario

1. `httpd` stellt den Inhalt von `www/` über `HTTP` bereit
2. Ursprünglich wird `www/` auf einem case-sensitive Dateisystem gespeichert
3. `hidden/` ist über HTTP nicht zugänglich, da others keine Berechtigungen hat
4. `protected/` ist durch `.htaccess`-Datei konfiguriert, nur für spezifische Nutzer zugreifbar zu sein

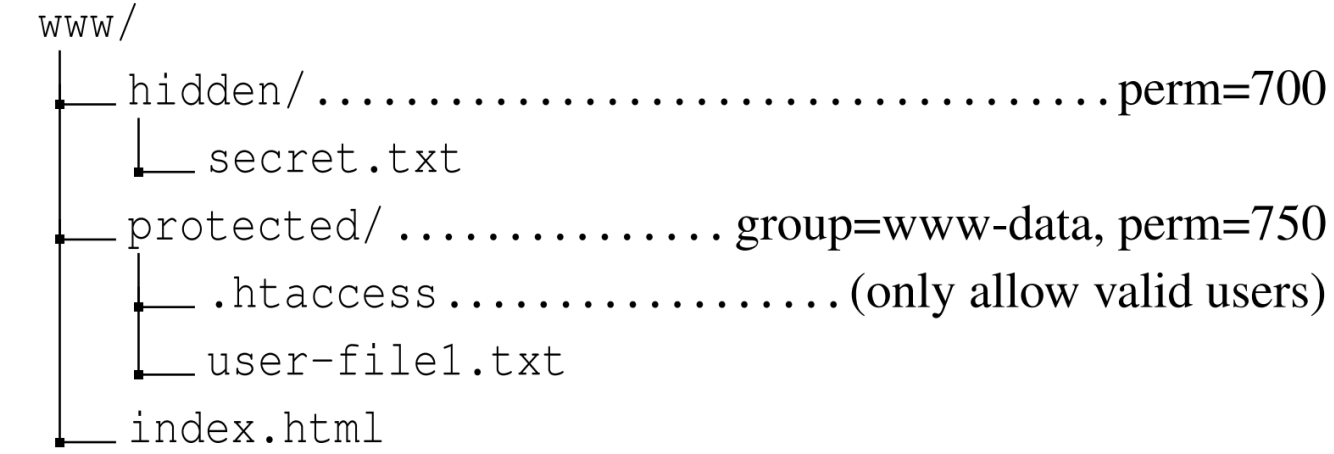


Abbildung 8: `www/` auf case-sensitive Dateisystem

Szenario

1. Angreifer hat read-write Zugang zu `www/`
2. DAC-Berechtigungen verhindern Zugang zu `hidden/`, da der owner ein anderer Benutzer ist
3. `protected/` unzugänglich, da Angreifer nicht zur Gruppe `www-data` gehört
4. Er modifiziert `www/` und fügt `HIDDEN/` und `PROTECTED/` hinzu
5. Ziel: Zugriff auf `hidden/` and `protected/` via Name Collision

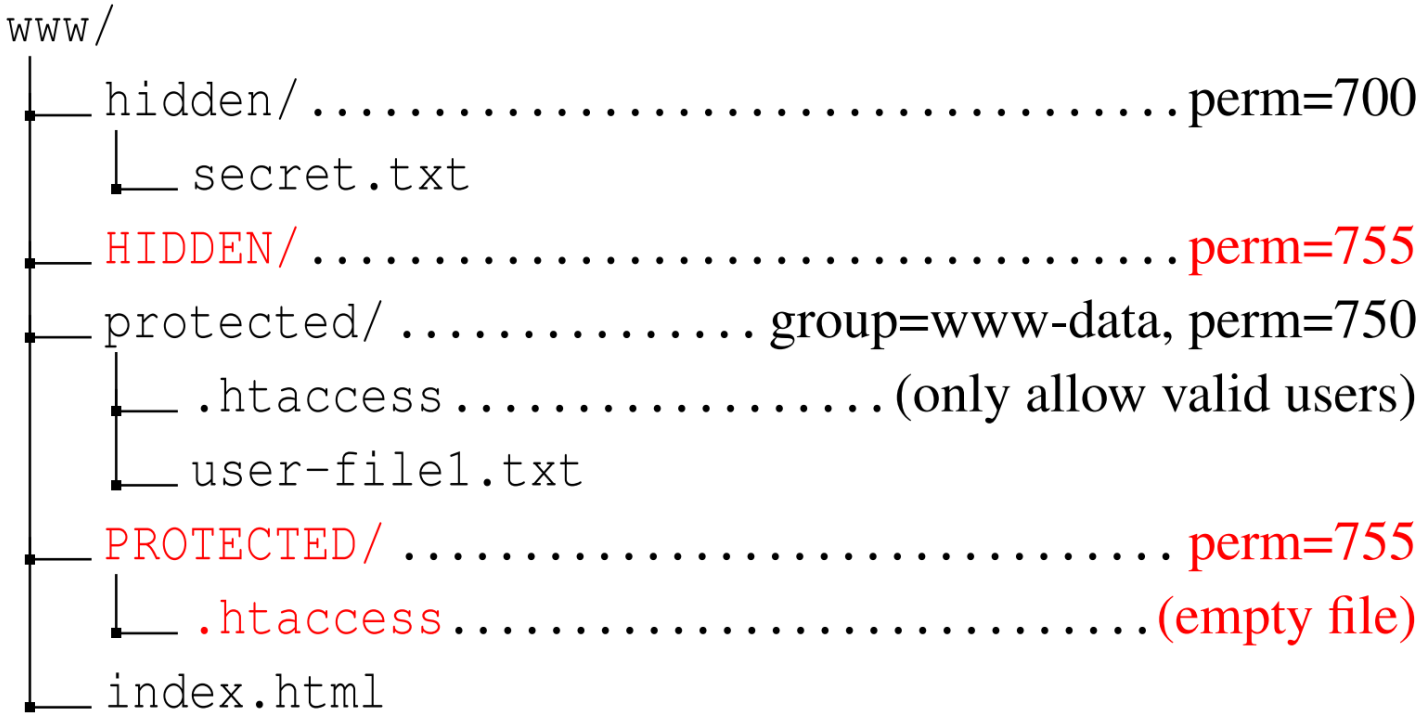


Abbildung 9: Angreifer modifzierte `www/` auf case-sensitive Dateisystem

Szenario

1. `tar` wird für Migration zu case-insensitive Dateisystem genutzt
2. Zuvor unzugängliches Verzeichnis `hidden/` ist nun zugänglich
3. `protected/` nun einsehbar, da `.htaccess` geleert ist

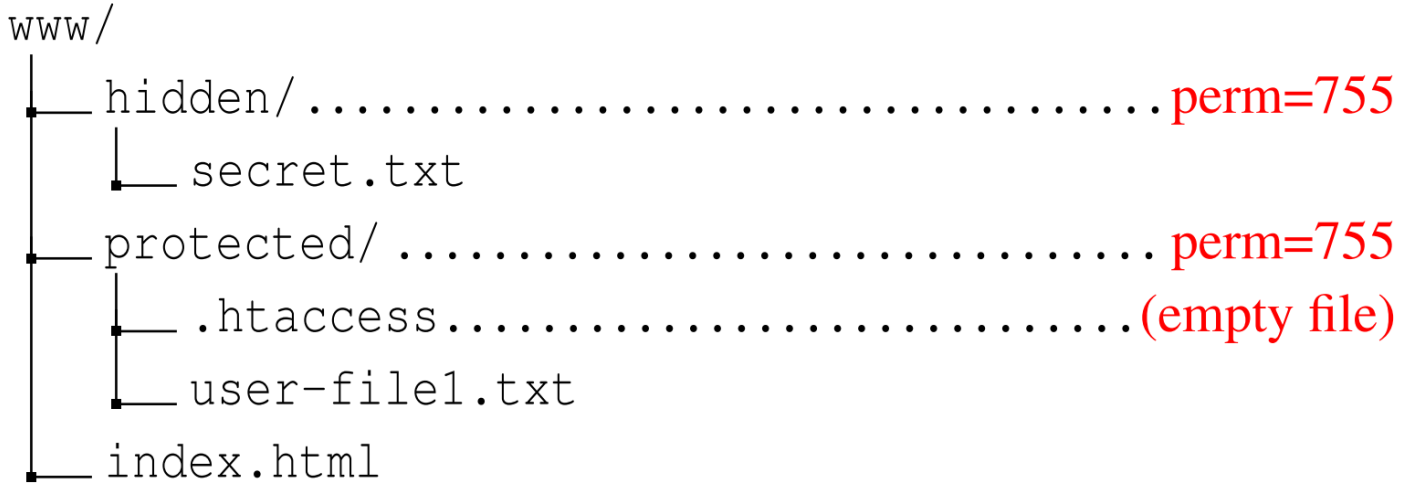


Abbildung 10: `www/` nach Migration auf case-insensitive Dateisystem

Szenario

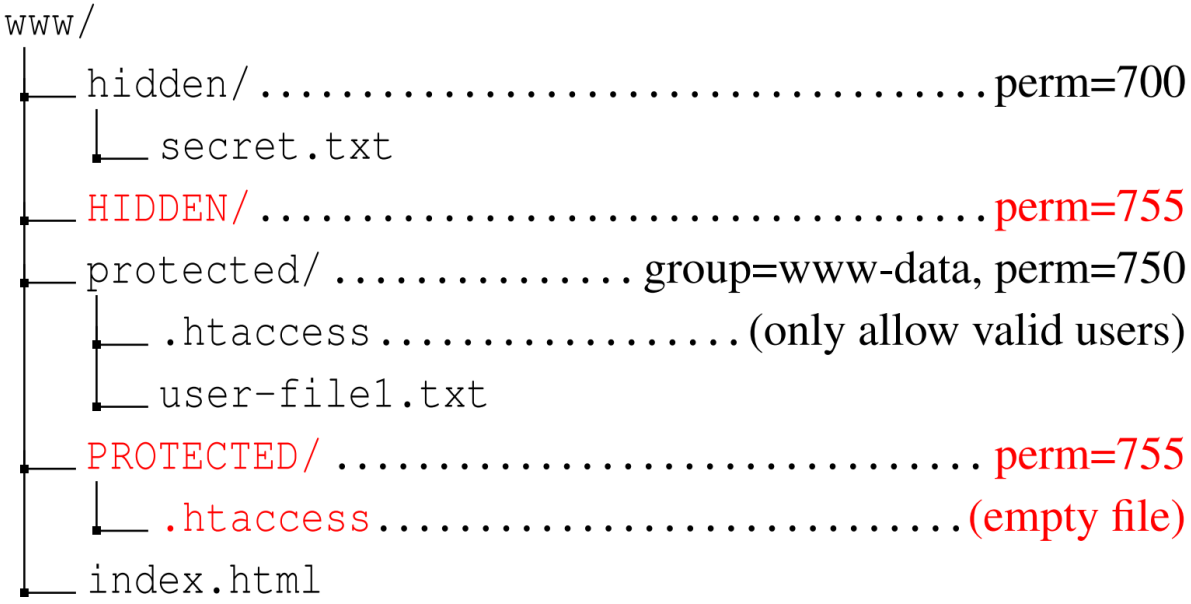


Abbildung 9: Angreifer modifzierte `www/` auf case-sensitive Dateisystem

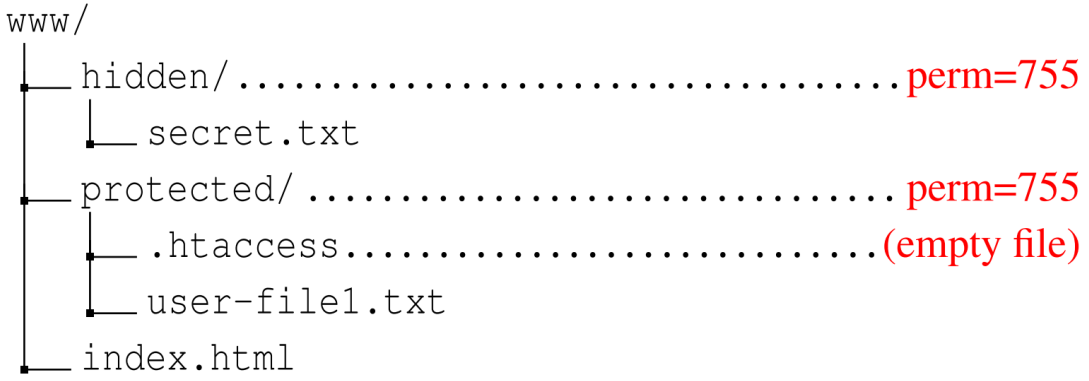


Abbildung 10: `www/` nach Migration auf case-insensitive Dateisystem

Was wurde dagegen unternommen?

1. Meldung an Apache-Maintainer: Das Problem wurde an die Apache-Maintainer gemeldet, aber es liegt noch keine Lösung vor
2. Problemanalyse mit tar: Tar wird verwendet, um das modifizierte www/ Verzeichnis auf ein case-insensitives Dateisystem zu migrieren

1. **Einführung**
2. **Auf Name Collisions testen**
3. **Name Collisions bei Linux Dienstanwendungen**
4. **Fallstudien** – rsync, dpkg und Apache httpd
5. **Mögliche Verteidigungen**
6. **Schluss**
7. **Quellen**

Optionen und Grenzen

1. Unzuverlässigkeit von benutzerseitigen Lösungen: unzuverlässig, Programme können Case-Folding-Regeln nicht bestimmen, anfällig für TOCTTOU-Angriffe
2. Einschränkungen von Systemschutzmaßnahmen: verschiedene Einschränkungen wie Fehlen von Informationen der Programmierabsicht, Denial-of-Service-Angriffe

Beispiel für eine Idee zur Verteidigung

1. Wrapper für Archivvalidierung: Könnte Archive vor dem Entpacken überprüfen ➔ Sicherstellen, dass Dateien nach dem Entpacken eindeutig sind
2. Probleme mit dieser Verteidigung: Kritische Nachteile, wie bereits im Zielverzeichnis vorhandene Dateien, die zu Kollisionen führen können, und Unterschiede in den Case-Folding-Regeln zwischen Wrapper und Zielverzeichnis.

Entwicklung von Verteidigungsmichanismen

1. Notwendigkeit neuer Flags: z.B. O_EXCL_NAME, erforderlich, um Dateien mit unterschiedlichen Namen zu schützen, die nur aufgrund von Case-Folding-Regeln übereinstimmen
2. Herausforderung für Programmierer: müssen die Absicht ihrer Operation verstehen ➔ Bedrohungen erkennen ➔ komplexe Befehle konfigurieren, dass sie Bedrohungen blockieren
3. Bis zur Weiterentwicklung von Dateisystem-APIs: Fehler und Schwachstellen bleiben häufig

1. **Einführung**
2. **Auf Name Collisions testen**
3. **Name Collisions bei Linux Dienstanwendungen**
4. **Fallstudien** – rsync, dpkg und Apache httpd
5. **Mögliche Verteidigungen**
6. **Schluss**
7. **Quellen**

Schluss

„During our discussions, we analyzed 74,688 packages and found 12,237 filenames from those packages would collide“ (Aditya Basu et al., 2022)

Schluss

1. Steigende Gefahr durch Trends für Flexibilität bei Dateisystemen, Linux Subsystem for Windofws
2. Aktuelle Betriebssysteme überlassen Verhinderung von Name-Collisions meist Entwicklern
3. Viele Anwendungen nutzen unsichere Kopieroperationen, was sie anfällig für ausnutzbare Schwachstellen macht

Schluss

1. Drei detaillierte Fallstudien zeigen konkrete Schwachstellen durch Namenskollisionen auf
2. Handlungsbedarf: Notwendigkeit für einheitliche und sichere Namenskollisionsrichtlinien sowie verbesserte Kopieroperationen

Quellen

Abbildung 1-10: Aditya Basu, John Sampson, Zhiyun Qian, Trent Jaeger. Unsafe at Any Copy: Name Collisions from Mixing Case Sensitivities.

Tabelle 1: Aditya Basu, John Sampson, Zhiyun Qian, Trent Jaeger. Unsafe at Any Copy: Name Collisions from Mixing Case Sensitivities.

[0] Aditya Basu, John Sampson, Zhiyun Qian, Trent Jaeger. Unsafe at Any Copy: Name Collisions from Mixing Case Sensitivities.

[1] Wikipedia Case Sensitivity. https://de.wikipedia.org/wiki/Case_sensitivity

[2] File Name Confusion Attacks

[3] Samba. https: //www.samba.org/

[4] Archlinux. https://wiki.archlinux.org/title/Ext4

[5] Peter Snyder. tmpfs: A Virtual Memory File System

[6] Git’s patch for CVE-2021-21300. https://github.com/git/git/commit/ 684dd4c2b414bcf648505e74498a608f28de4592.

[7] Ubuntuusers/dpkg. https://wiki.ubuntuusers.de/dpkg/

[8] Ubuntuusers/rsync. https://wiki.ubuntuusers.de/rsync/

[9] Apache. https://httpd.apache.org/