

Deep Detection of IaC Security Smells

Opdebeeck, Ruben; Zerouali, Ahmed; De Roover, Coen

Publication date:
2023

License:
CC BY-NC-SA

Document Version:
Final published version

[Link to publication](#)

Citation for published version (APA):
Opdebeeck, R., Zerouali, A., & De Roover, C. (2023). *Deep Detection of IaC Security Smells*. Poster session presented at 2nd Summer School on Security Testing and Verification, Elsene, Belgium.

Copyright

No part of this publication may be reproduced or transmitted in any form, without the prior written permission of the author(s) or other rights holders to whom publication rights have been transferred, unless permitted by a license attached to the publication (a Creative Commons license or other), or unless exceptions to copyright law apply.

Take down policy

If you believe that this document infringes your copyright or other rights, please contact openaccess@vub.be, with details of the nature of the infringement. We will investigate the claim and if justified, we will take the appropriate steps.

Deep Detection of IaC Security Smells

Ruben Opdebeeck

✉ Ruben.Denzel.Opdebeeck@vub.be
 🐦 @ROpdebee

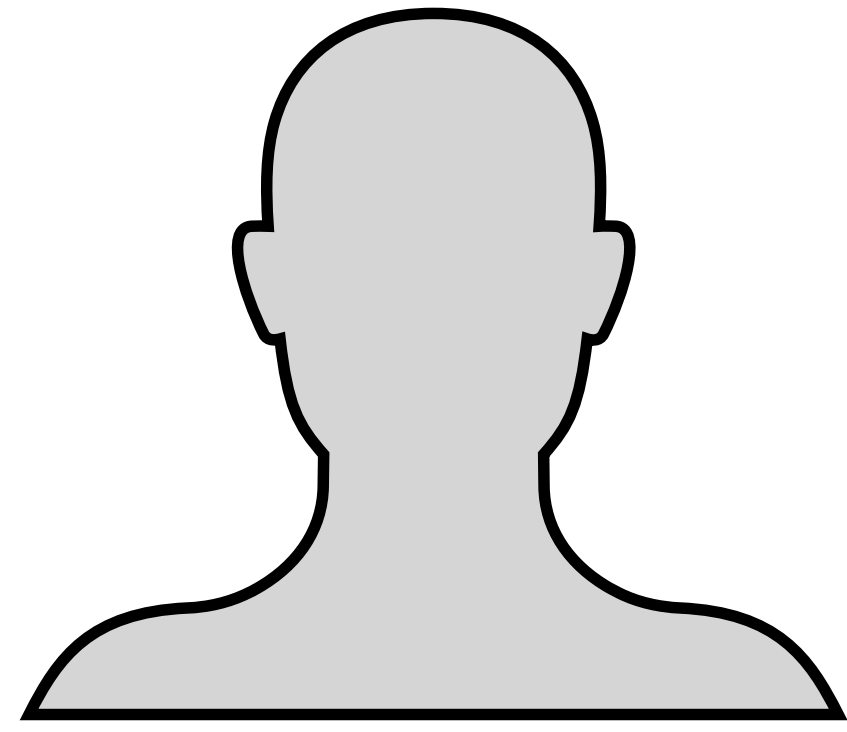
Ahmed Zerouali

✉ Ahmed.Zerouali@vub.be
 🐦 @a_zerou

Coen De Roover

✉ Coen.De.Roover@vub.be
 🐦 @CoenDeRoover

Are we checking the integrity of all downloaded executables?



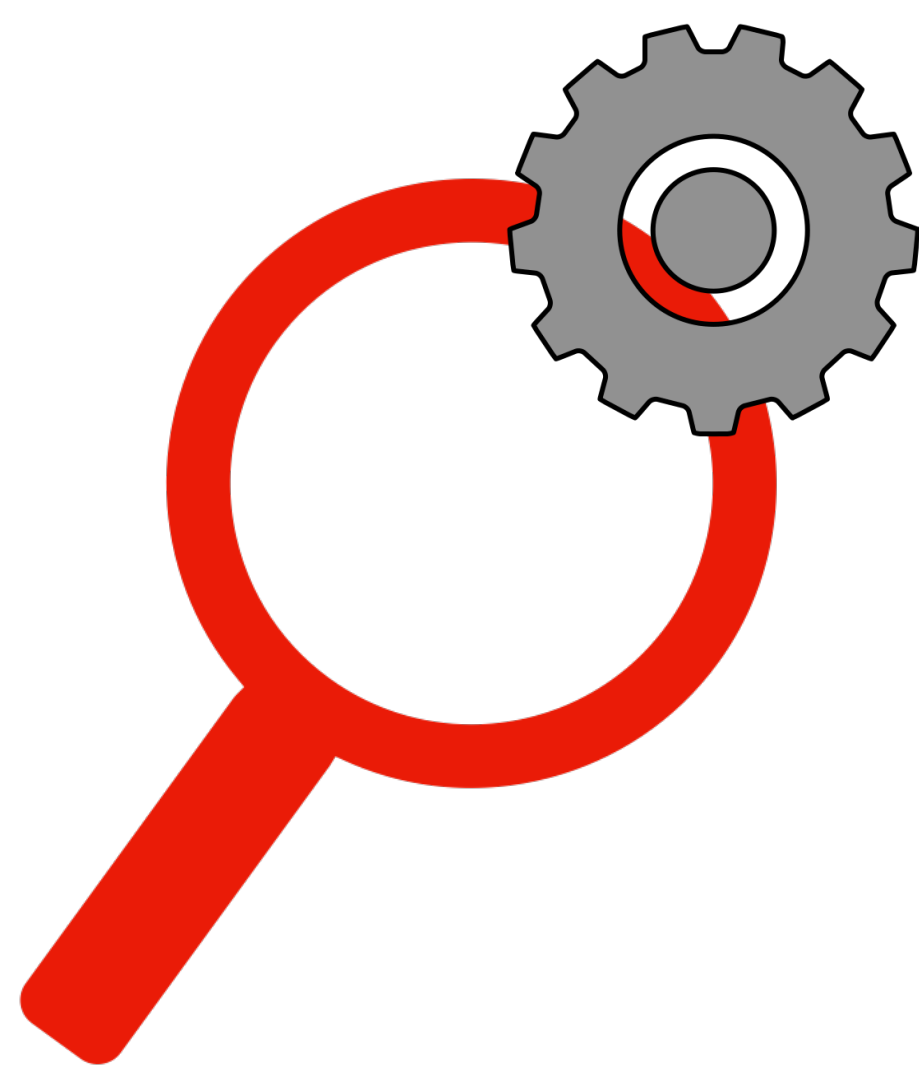
```
# Variables for Ruby installation. vars.yml
ruby_install_from_source: true
ruby_download_url:
  https://cache.ruby-lang.org/.../ruby-2.7.6.tar.gz
ruby_version: 2.7.6
```

```
- hosts: all
  vars_files: vars.yml
  roles:
  ... - geerlingguy.ruby
```

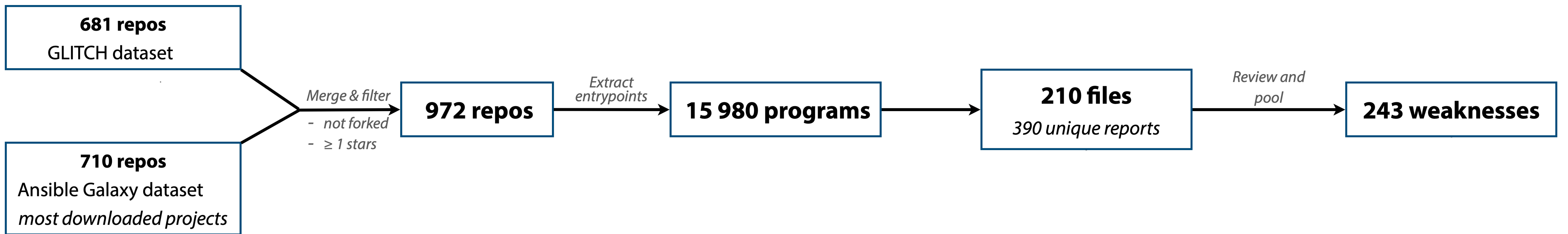
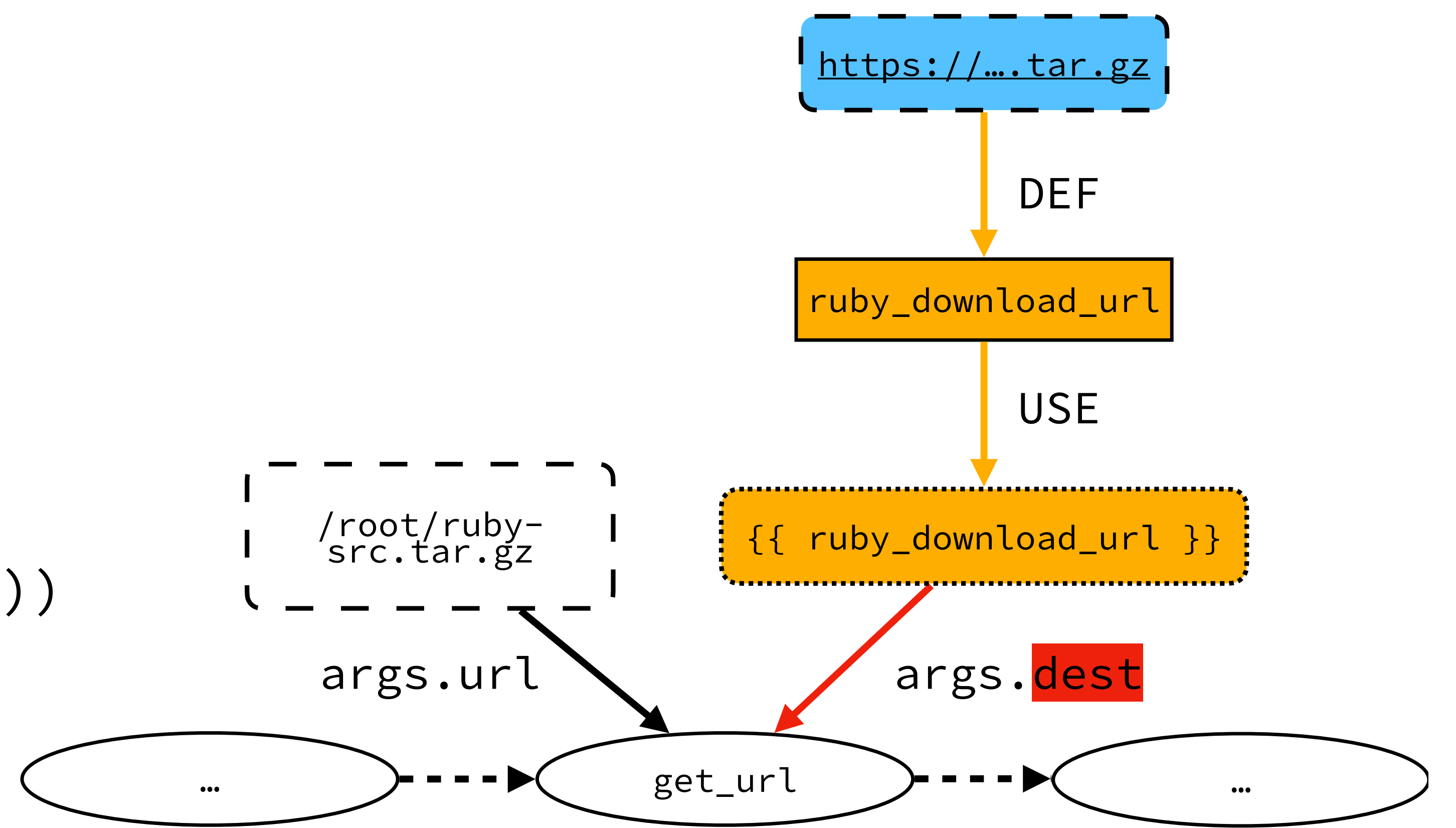
```
- name: Download ruby. geerlingguy/ansible-role-ruby
  get_url:
    url: "{{ ruby_download_url }}"
    dest: "/root/ruby-src.tar.gz"
```



Missing integrity check



```
MATCH chain =
  (source:Literal)
  -[:DEF|USE*0..]->()
  -[arg:KEYWORD]->(sink:Task)
WHERE
  source.value =~ 'http.*.tar.gz'
  AND NOT (()-['checksum']->(sink))
RETURN source;
```



Smell type	# instances	Precision	Recall
Admin By Default	64	98.11%	81.25%
Empty Password	15	44.44%	80.00%
HTTP Without SSL/TLS	35	100.00%	88.57%
Hardcoded Secret	11	45.45%	90.91%
Missing Integrity Check	27	96.15%	92.59%
Unrestricted IP Address	47	76.60%	76.60%
Weak Crypto Algorithm	44	97.67%	95.45%

Indirection level	Admin By Default	Empty Password	HTTP Without SSL/TLS	Hardcoded Secret	Missing Integrity Check	Unrestricted IP Address	Weak Crypto Algorithm
0	77	5.2	29	29	49	66	22
1	22	85	40	56	38	23	50
2	1	8.7	28	11	13	3.8	21
3	0.2	0.35	3.7	3.2	0.37	6.9	4.3
4	0	0.35	0	0.28	0	0.24	1.6
5	0	0	0	0.32	0	0	1
6	0	0	0	0.28	0	0	0.17

7933 unique smells

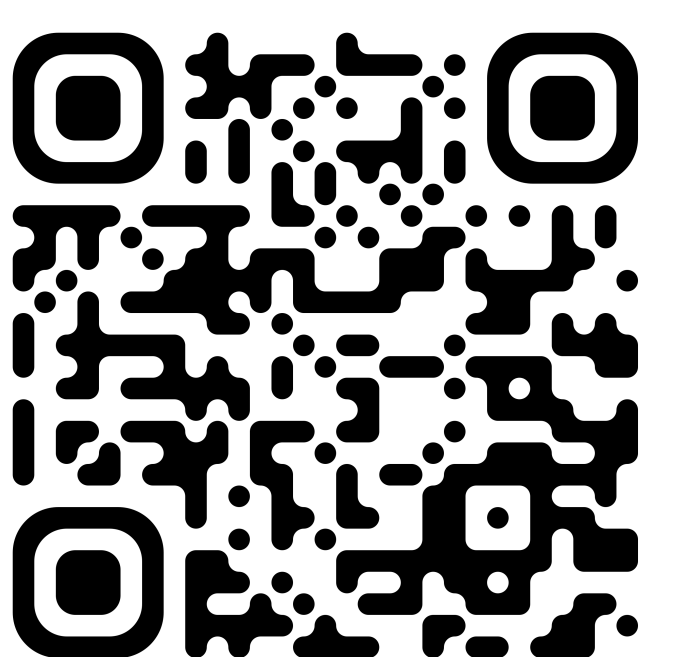
33% smells cross file boundaries

472 affected repositories

55% smells involve data flow indirection

3613 affected files

6.5% smells involve 3rd party code



Ruben Opdebeeck, Ahmed Zerouali, Coen De Roover. 2023. Control and Data Flow in Security Smell Detection for Infrastructure as Code: Is It Worth the Effort? In *Proceedings of MSR'23: Proceedings of the 20th International Conference on Mining Software Repositories*