

Runtime IaC Compliance Management Framework (IACMF)

- *Interview* -

Data Protection

If you agree to conduct the review, you also agree to the following terms:

1. Your voice will be recorded during the interview and then transcribed.
2. Your answers will be anonymized, i.e., all references to your identity and the identity of the company you work for will be removed from your transcribed answers.
3. Your answers will be analyzed for the purpose of evaluating the IACMF framework.
4. The anonymized answers and/or the analysis results will only be used in scientific communication.

Questionnaire – Tutorial Follow Up

- Do you have any questions regarding the tutorial or the overview document?

Questionnaire – Demographic Questions

1. What is your current role in the company?
2. What kind of tasks do you usually do in your work?
3. Given enough time, can you understand the architecture of an application system that is described using an IaC script of an IaC technology you are familiar with?
4. For how many years have you worked on tasks associated with IaC tools?
5. How large is the company you currently work for?
 - a. < 50
 - b. < 250
 - c. < 2,000
 - d. < 100,000
 - e. > 100,000

Questionnaire – Compliance Rule Modeling and Checking (1)

6. How do you check the compliance of the software applications of your company?
7. Do you use well-defined models for the compliance rules applicable to the software applications of your company?
 - a) If so, how do you define them?
8. Do you think having a well-defined and machine-readable format for compliance rules reduces the complexity associated with checking them?
9. Do you think having a well-defined and machine-readable format for compliance rules reduces the uncertainty associated with interpreting them?
10. How often do you have to deal with new compliance rules?

Questionnaire – Compliance Rule Modeling and Checking (2)

11. How much do you agree with the following statement: *using IACMF reduces the effort associated with defining and checking compliance rules?*

totally disagree

totally agree

1	2	3	4	5
---	---	---	---	---

12. How much do you agree with the following statement: *using IACMF reduces the complexity associated with defining and checking compliance rules?*

totally disagree

totally agree

1	2	3	4	5
---	---	---	---	---

13. How much do you agree with the following statement: *using well-defined models for compliance rules reduces the uncertainty associated with interpreting them?*

totally disagree

totally agree

1	2	3	4	5
---	---	---	---	---

Questionnaire – Architectural Reconstruction

14. How do you reconstruct the architecture of running application instances you need to understand?
15. Do you use any (semi-)automated tools for this purpose?
16. How much do you agree with the following statement: *using IACMF reduces the effort associated with reconstructing the architecture of running application instances?*

totally disagree

totally agree

1	2	3	4	5
---	---	---	---	---

Questionnaire – Compliance Violation Fixing

17. What do you do if you find out that a running application instance violates a compliance rule?

18. Do you use any (semi-)automated tools for this purpose?

19. How much do you agree with the following statement: *using IACMF reduces the effort associated with fixing compliance violations?*

totally disagree

totally agree

1	2	3	4	5
---	---	---	---	---

20. How much do you agree with the following statement: *having well defined models for compliance jobs reduces the uncertainty associated with handling detected compliance violations?*

totally disagree

totally agree

1	2	3	4	5
---	---	---	---	---

Questionnaire – General Questions

21. How do you evaluate the novelty of the framework?

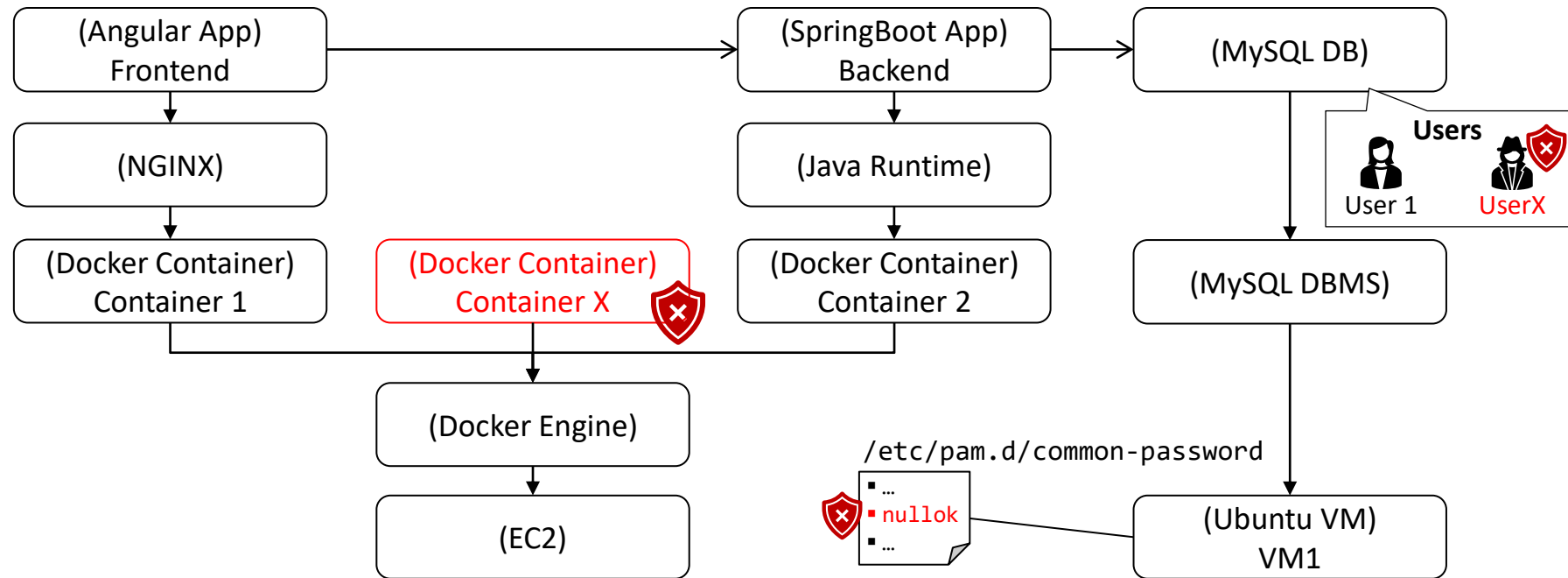
22. How do you evaluate the extensibility of the framework?

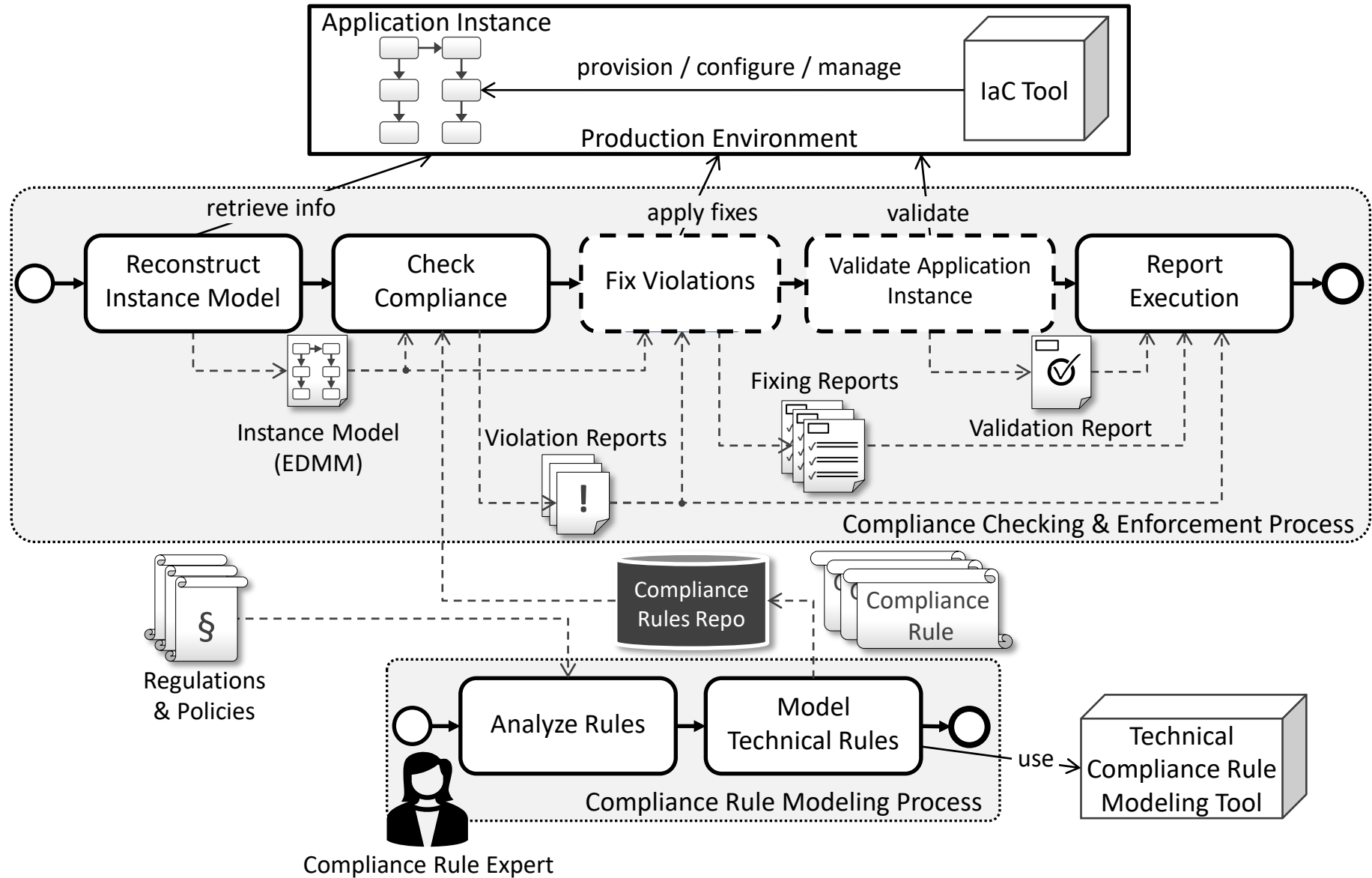
23. Would you use the framework in your work?

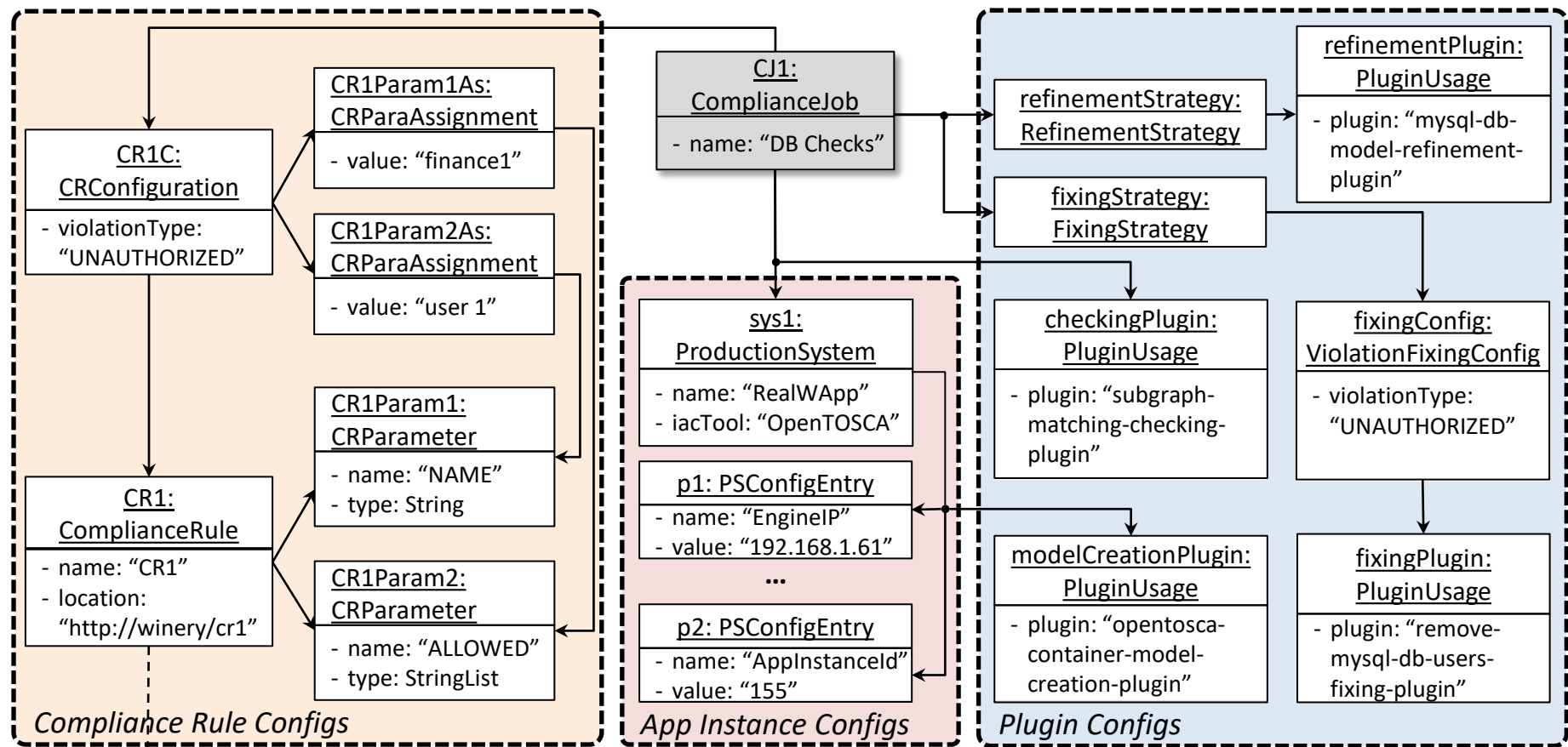
a) If so, in which areas?

24. What is your general impression?

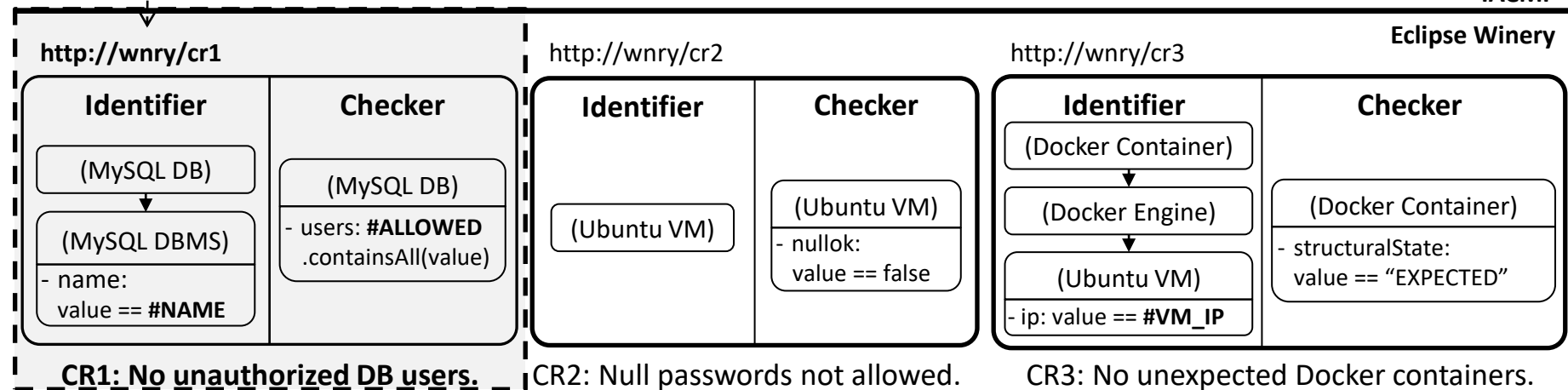
Backup Slides







IACMF



Use Cases

- We present three use cases that demonstrate the usage of IACMF.
- Use cases 1&2 are also part of a video demonstration you will watch before the interview.

Use Case 1 – operating system STIGs

- STIGs: security technical implementation guides.
- Cybersecurity requirements for certain products (e.g., operating systems).
- Published by the US department of defence (DoD).
- Basis for many security guidelines/baselines.

Use Case 1 – operating system STIGs

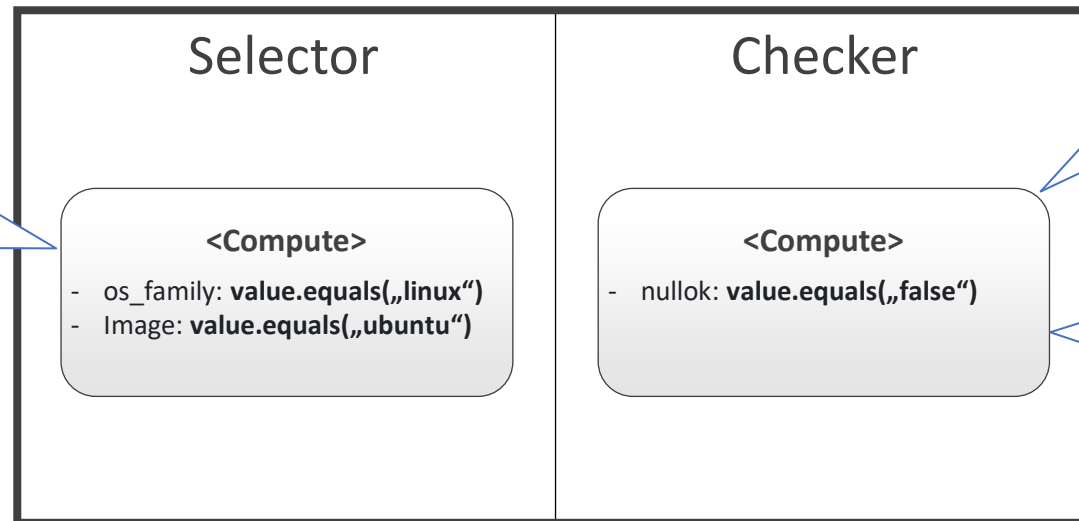
- Example:
 - Canonical Ubuntu 20.04 LTS STIGs (**169 rules**)
 - **STIG ID:** UBTU-20-010463
 - The Ubuntu operating system must not allow accounts configured with blank or null passwords.
- How can we implement and check this rule using IACMF?

Use Case 1 – operating system STIGs

- Technical compliance rule design
 - Plugin: subgraph-matching-checking-plugin
 - Rule Modeling Tool: *Eclipse Winery*

Match all components that represent ubuntu virtual machines in the instance model.

A pair of EDMM-models



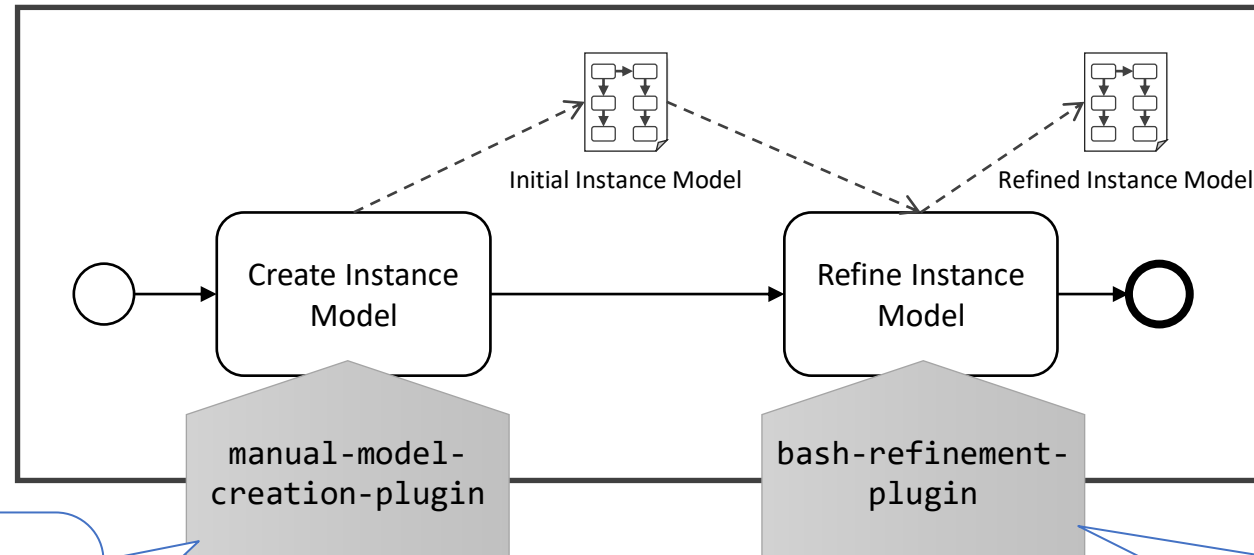
Compliance Rule

Every matched component must have a property “nullok” with a value “false”.

But where do we get this information from?
- Architectural reconstruction!

Use Case 1 – operating system STIGs

- Architectural reconstruction

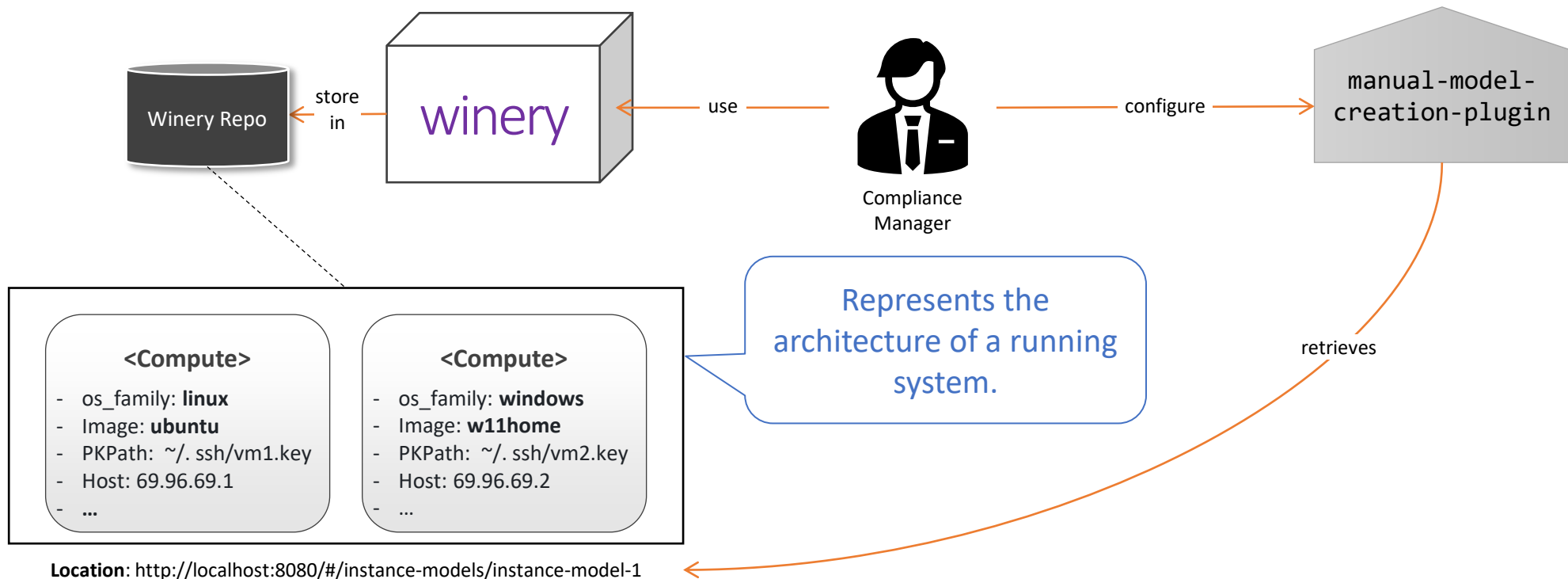


Imports an existing instance model that describes the production system.

Retrieves information about running OS components by running a **bash** script over **ssh**.

Use Case 1 – operating system STIGs

- Architectural reconstruction – initial instance model creation



Use Case 1 – operating system STIGs

- Architectural reconstruction – instance model refinement

Finds out if the option „nullok“ is used in the config file:
/etc/pam.d/common-password



Compliance Manager

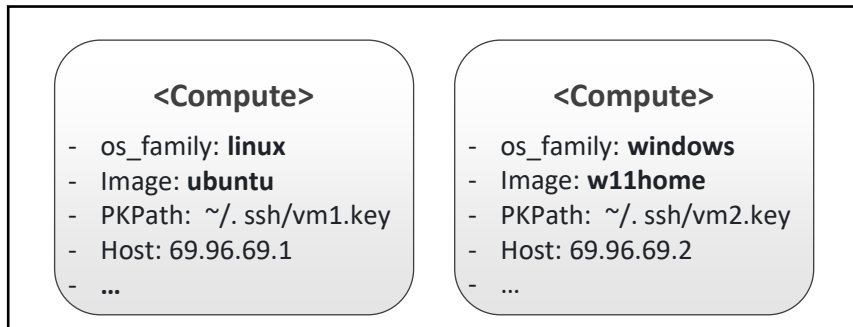
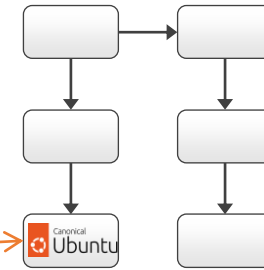
```
[[ ! -z $(sudo grep nullok /etc/pam.d/common-password) ]]  
]] && echo 'true' || echo 'false'
```

configure

bash-refinement-plugin

execute command
via ssh on ubuntu
VMs

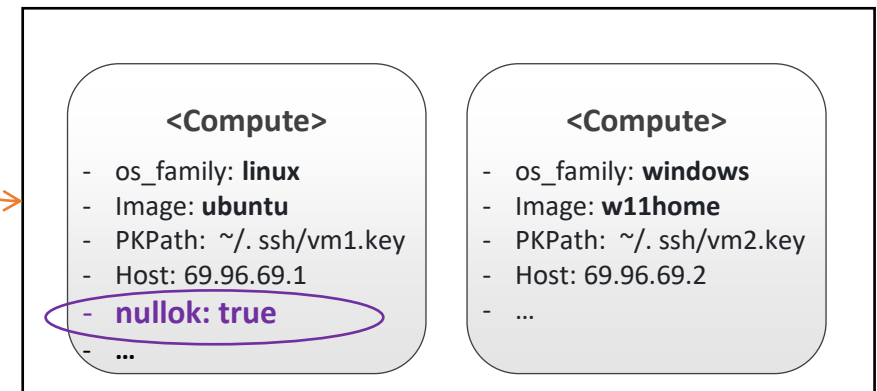
Running Application
Instance



Initial instance model

find ubuntu VMs

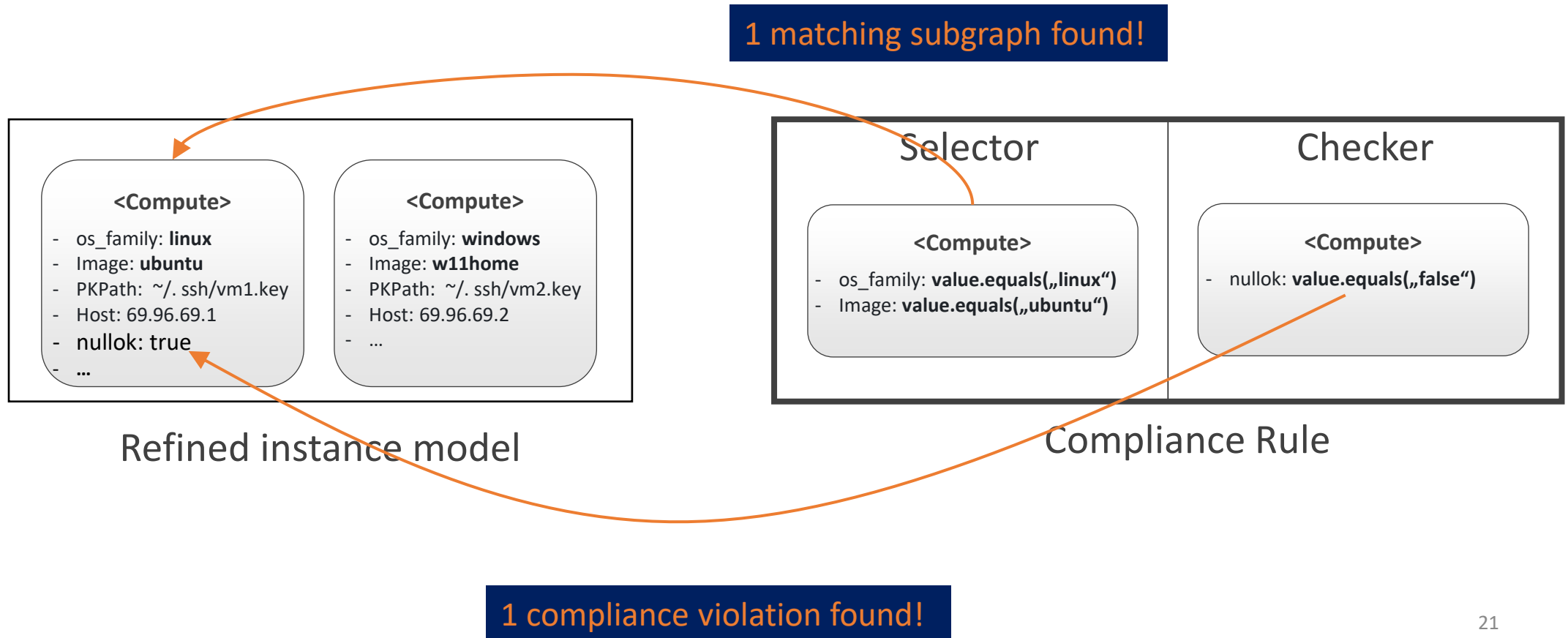
generate



Refined instance model

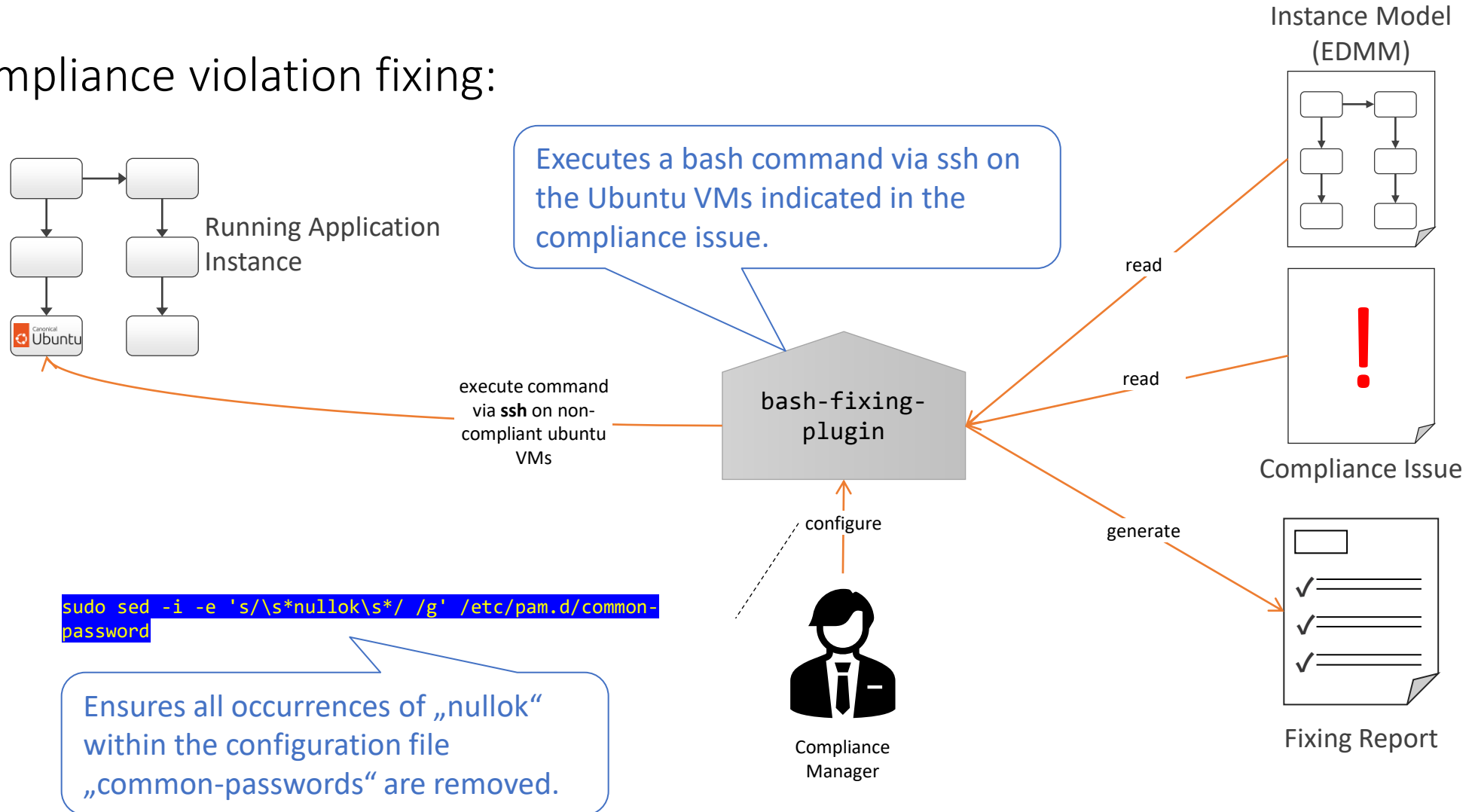
Use Case 1 – operating system STIGs

- Compliance Rule Checking
 - *subgraph-matching-checking-plugin*



Use Case 1 – operating system STIGs

- Compliance violation fixing:

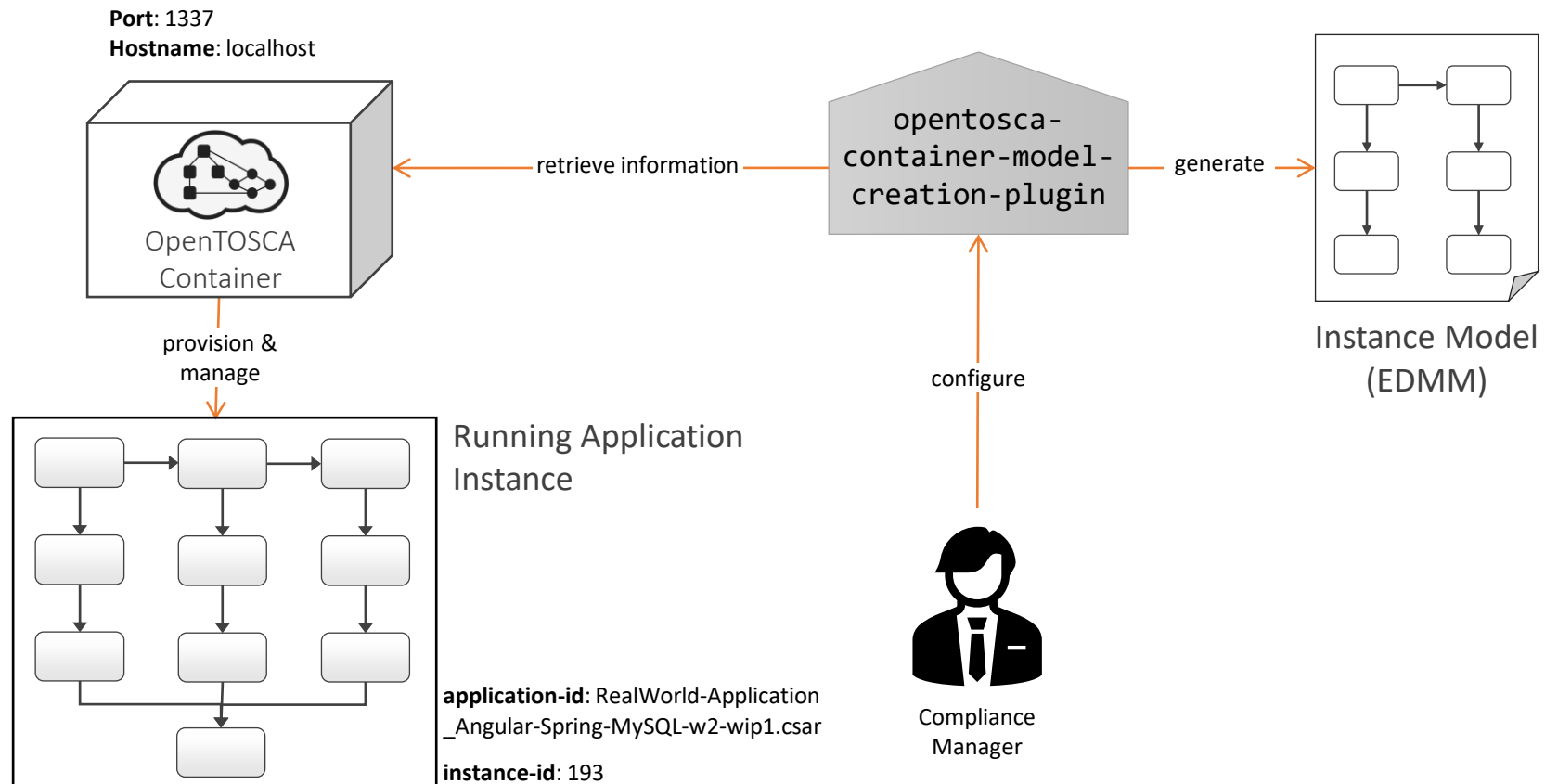


Use Case 2 – unauthorized DB users

- Production system:
 - Sample cloud application
 - Modeled using TOSCA
 - Provisioned using the OpenTOSCA Container
 - Three stacks using the same docker engine.

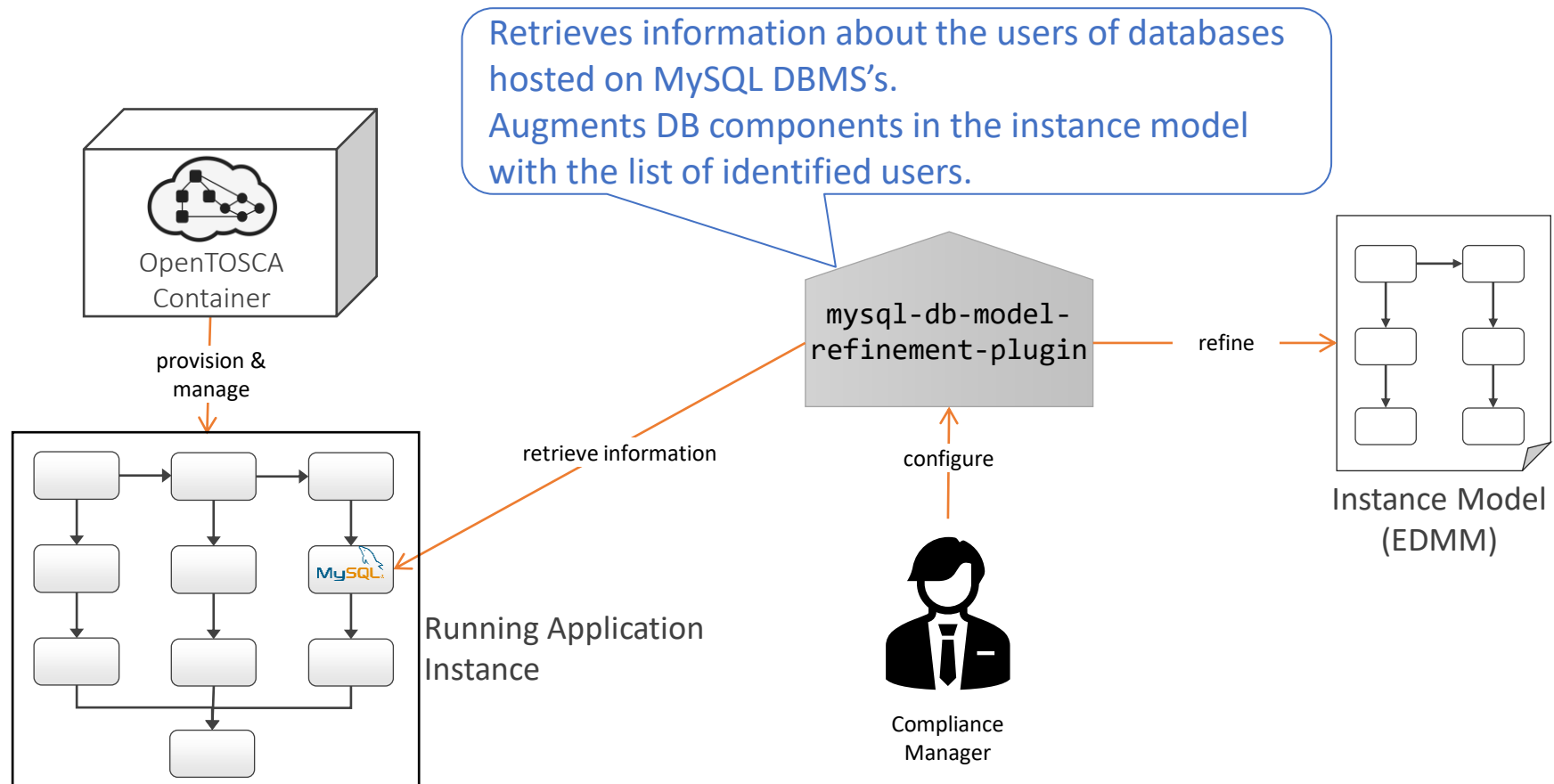
Use Case 2 – unauthorized DB users

- Architectural reconstruction – initial instance model creation:



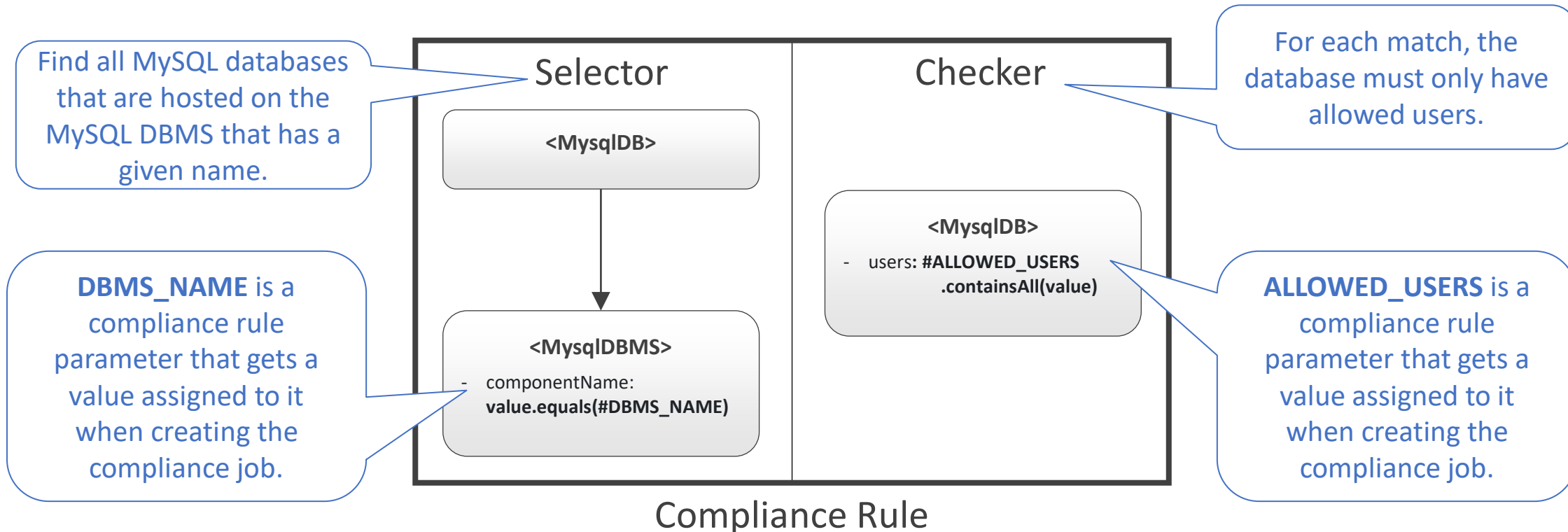
Use Case 2 – unauthorized DB users

- Architectural reconstruction – instance model refinement:



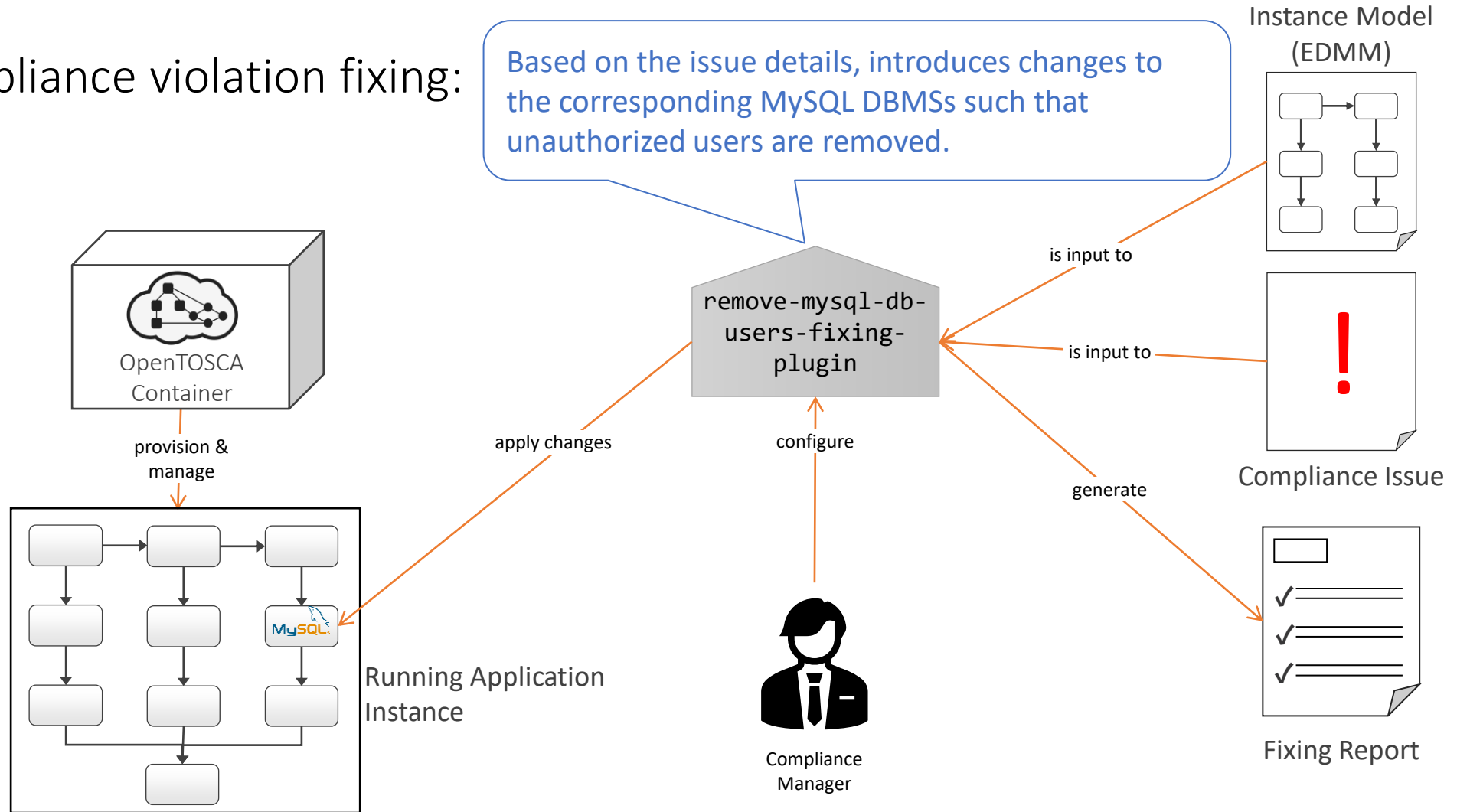
Use Case 2 – unauthorized DB users

- Compliance rule checking



Use Case 2 – unauthorized DB users

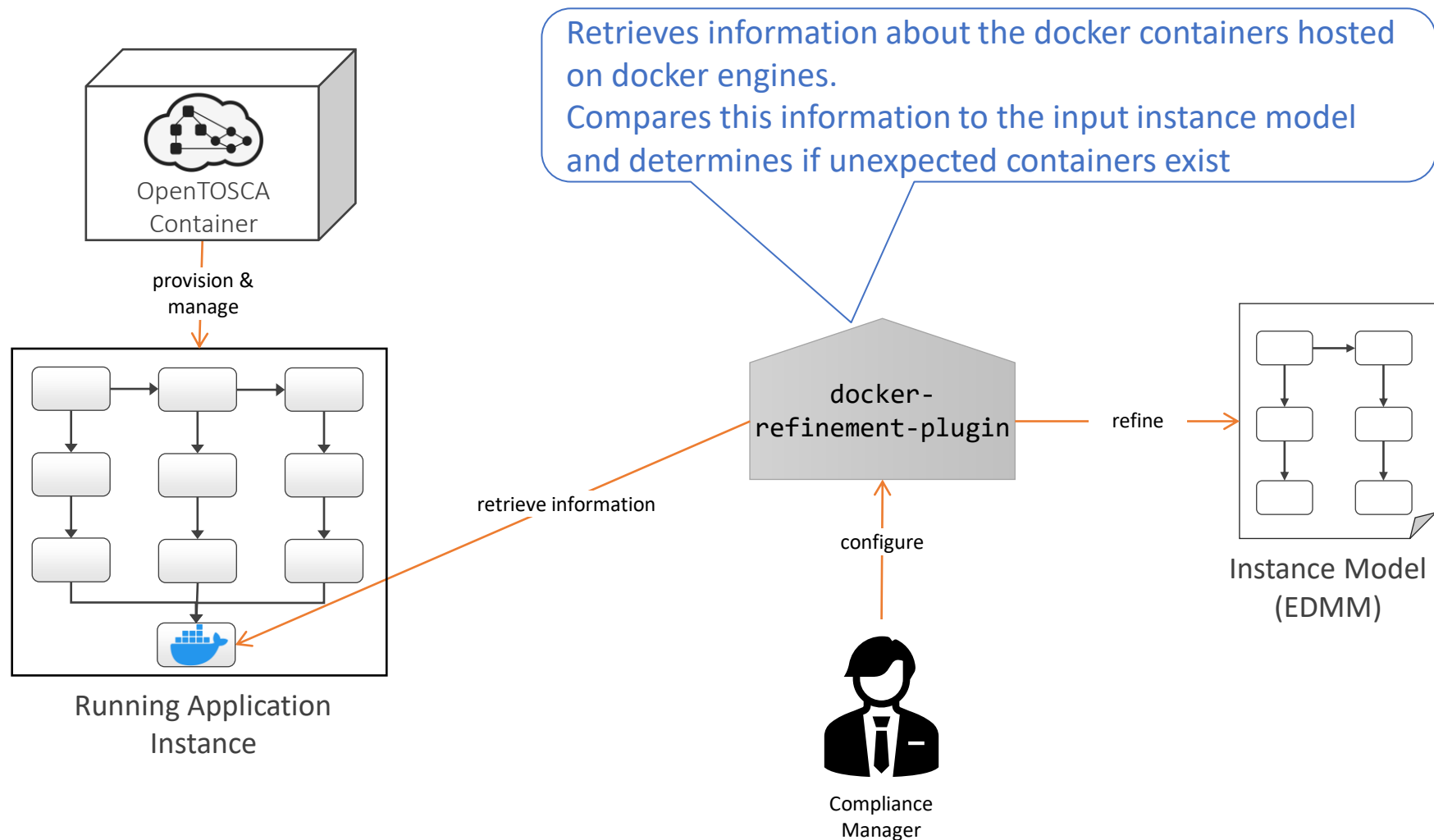
- Compliance violation fixing:



Use Case 3 – unexpected Docker containers

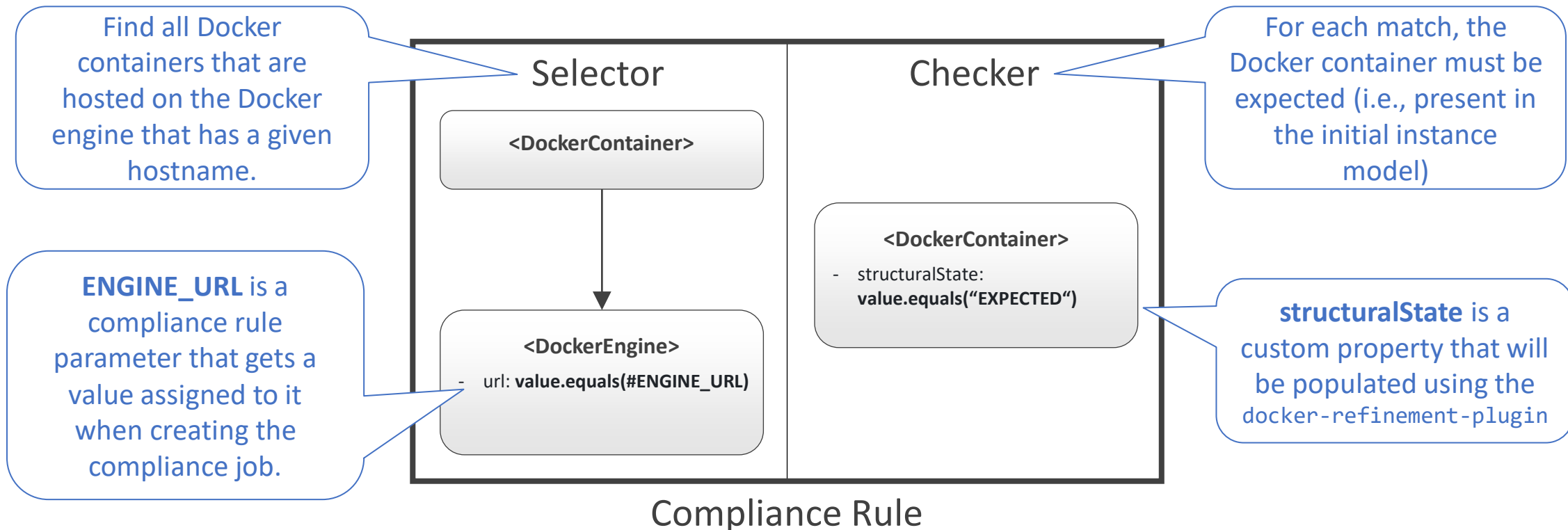
- Same production system as use case 2.
- Same initial instance model creation plugin (`opentosca-container-model-creation-plugin`).
- Different model refinement plugin.

Use Case 3 – unexpected Docker containers



Use Case 3 – unexpected Docker containers

- Compliance rule checking



Use Case 3 – unexpected Docker containers

- Compliance violation fixing:

