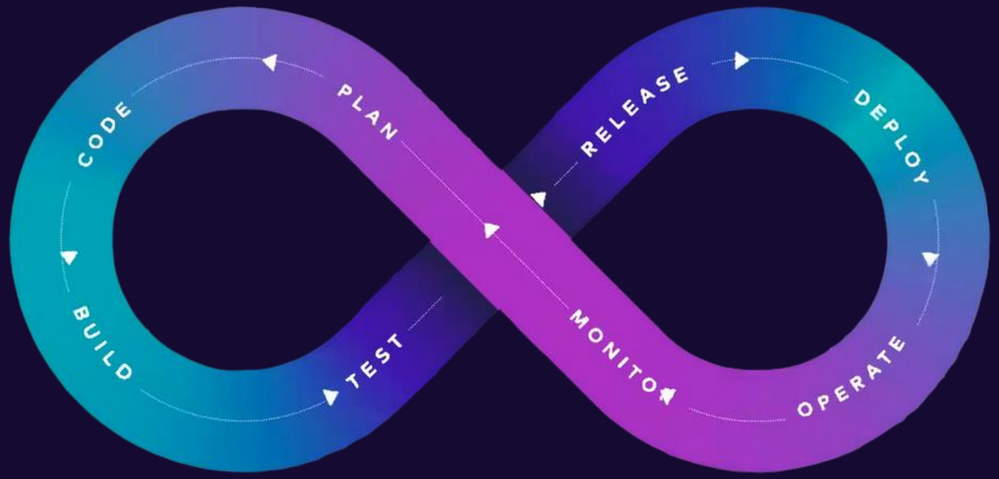
snyk

Solution de sécurité pour développeurs



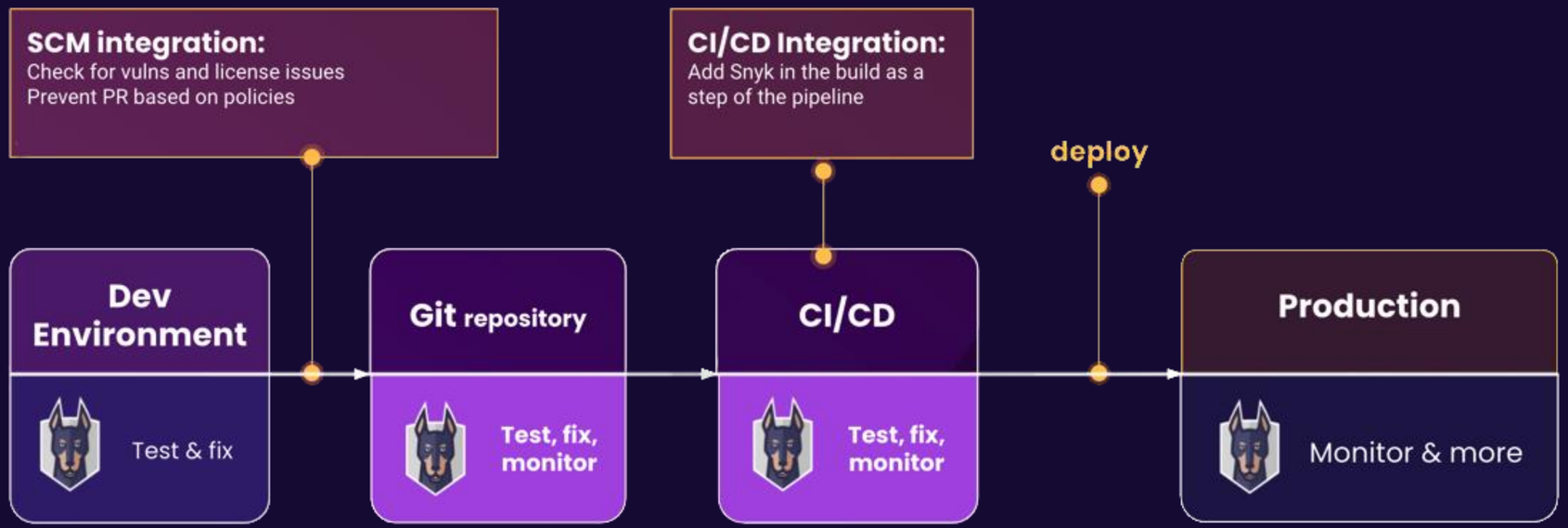
snyk

Maël Chevalier
Léo Charreau

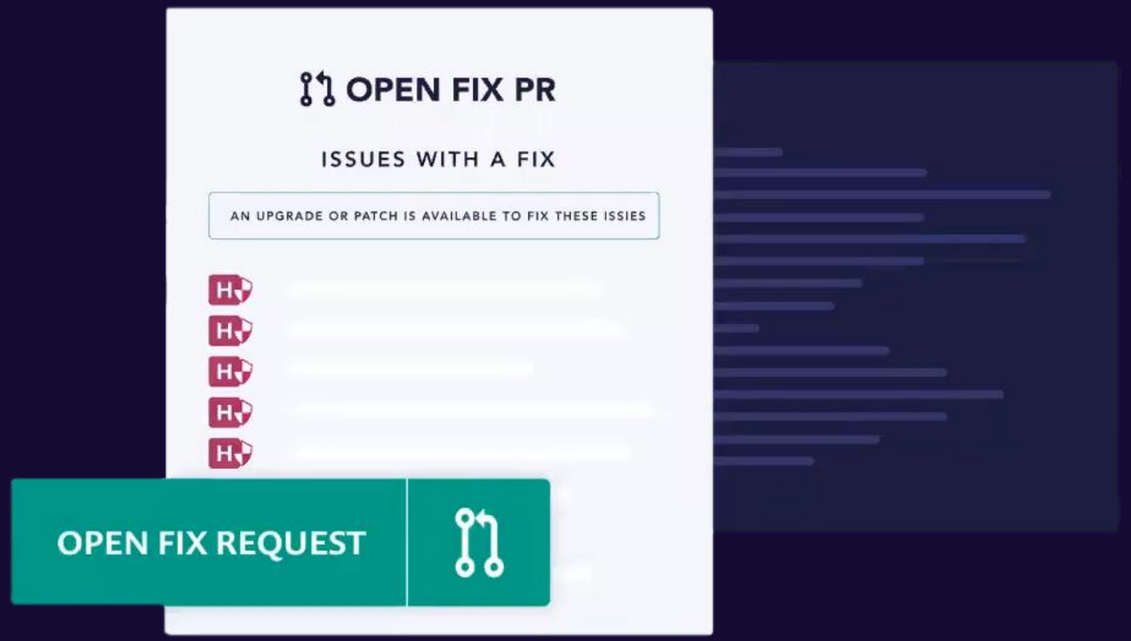
IDE

Snyk Open Source | Snyk Code | Snyk IaC | Snyk Container

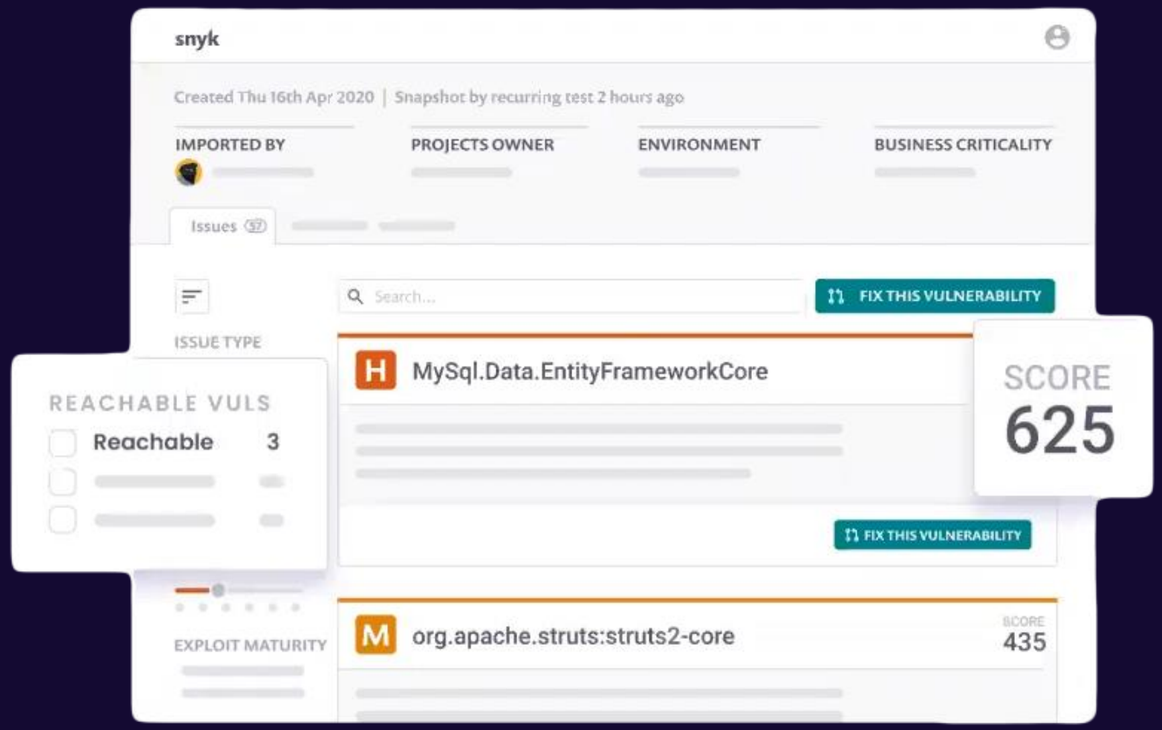
Intégration



Correction auto.



Rapport



Snyk Learn

snyk Learn Lessons Learning paths Learning progress

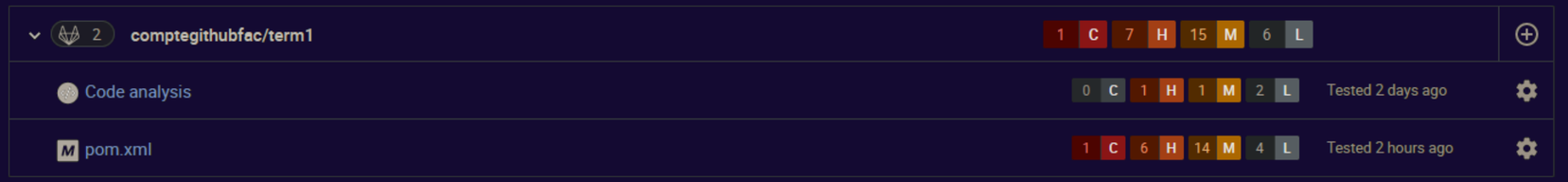
JNDI injection

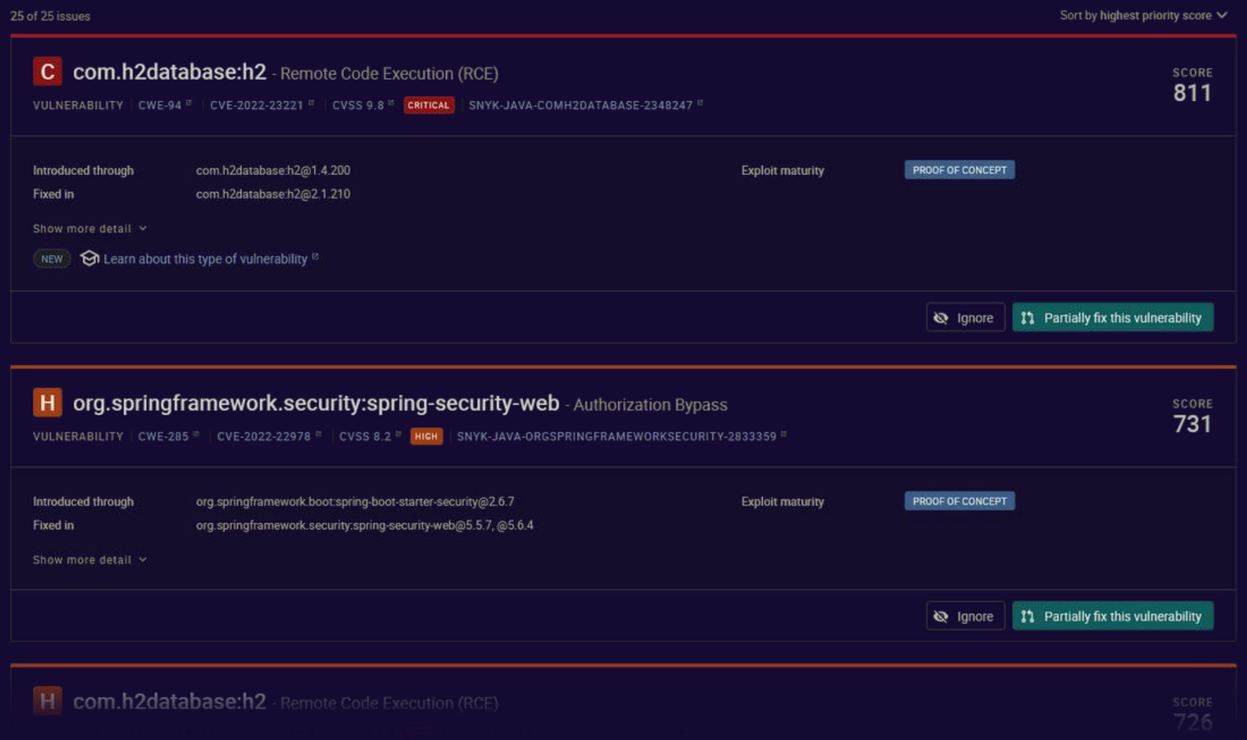
To exploit an example application vulnerable to JNDI injection we first create a malicious RMI server:

```
1  import java.rmi.registry.*;
2  import com.sun.jndi.rmi.registry.*;
3  import javax.naming.*;
4  import org.apache.naming.ResourceRef;
5
6  public class MaliciousRMIServer {
7      public static void main(String[] args) throws Exception {
8          //create our malicious RMI registry on port 1097 of our hosting server")
9          Registry registry = LocateRegistry.createRegistry(1097);
10
11         //this payload exploits unsafe reflection in org.apache.naming.factory.
   BeanFactory
12         // NOTE: class namespace changed to jakarta.el.ELProcessor since Java 9
13         ResourceRef ref = new ResourceRef("javax.el.ELProcessor", null, "", "",
```

Code injection

- Code injection: the basics
- Code injection in action
- Code injection under the hood
- Code injection mitigation





Intégration TER

Merci pour votre attention.

Maël Chevalier
Léo Charreau