**DevSecOps Notes:**

What is DevSecOps?  
Structure of Application code:

1. Frontend
2. Backend
3. Dependency file

Java based: POM.XML

Build(Package Manager): **MAVEN**

**Folder generated : Target**JavaSript based: package.json

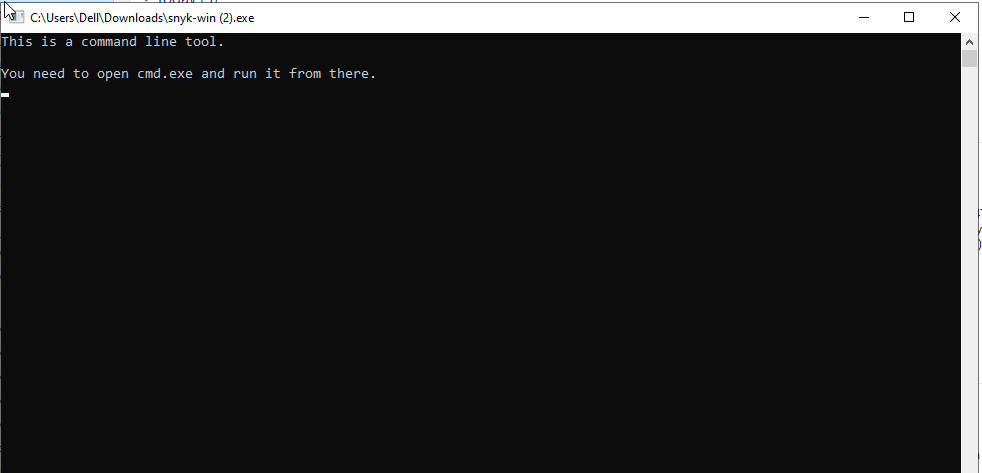
Build(Package Manager): NPM

Folder generated: Node modules  
  
**INSTALLING SNYK**

In order to test Source code which includes all other files apart from dependency file is done using **SAST.**

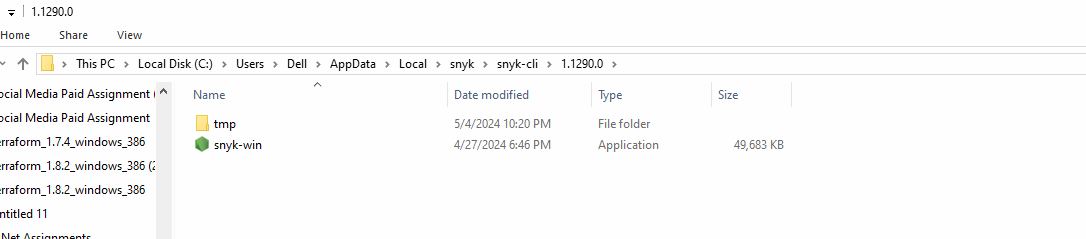
**Snyk –** SAST, SCA, IaC(Terraform data), Containerization Testing

<https://static.snyk.io/cli/latest/snyk-win.exe>

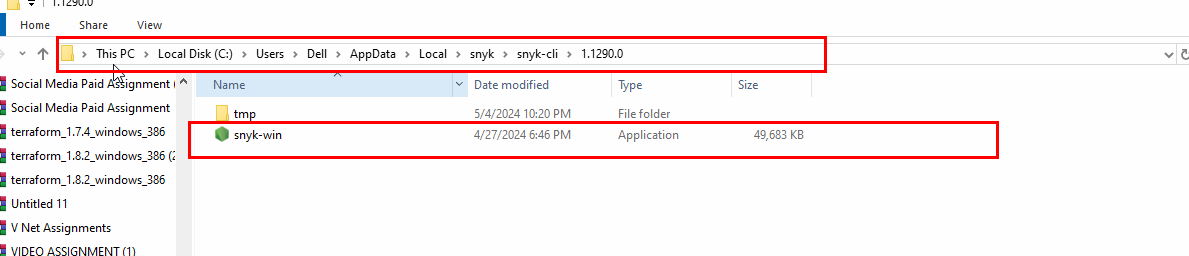
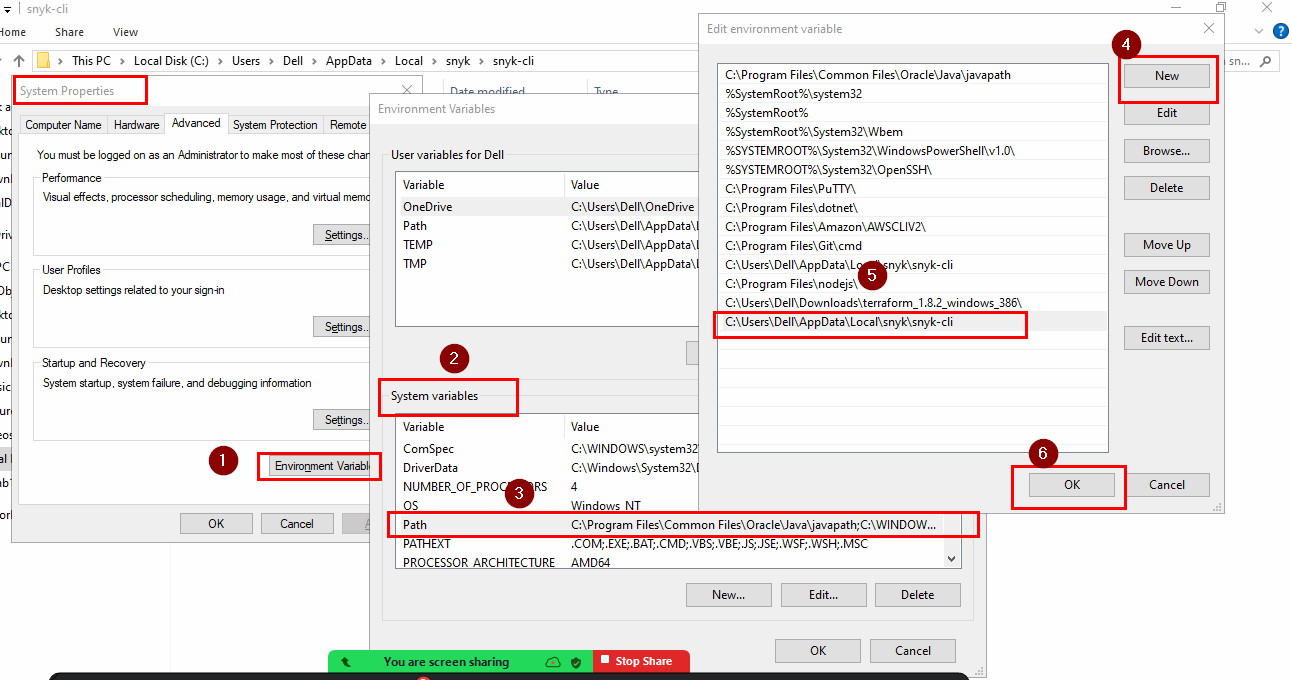
After downloading, double click > it shows open from CMD  


Go to CMD, run **synk-win version**  
  
It may show you an error, synk-win is not recognized.

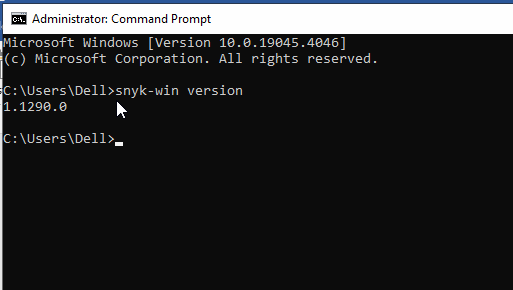
You need to go toC:\Users\**Dell**\AppData\Local\snyk\snyk-cli

If here, you find the current directory empty then you need to copy the Snyk cli that is just downloaded in you downloads and add it here. Once you have copied it here, double click and run it once again. With this, you will get a folder:  


Once you achieve this, you want **snyk-win** to access your entire system because your project can reside in your system anywhere. For this, you need to add the current path to Environment variables:

  
 ****

You need open CMD again, and run same command:

****

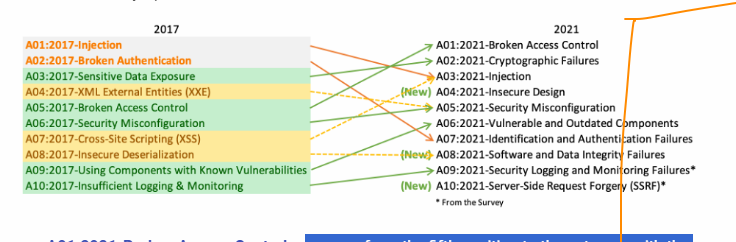
**GitHub link:** <https://github.com/rushtoaksith/juice-shop>

Clone the data in a folder in your local system

Open CMD, go to same location, once you are inside the directory, run:

Snyk-win –version

Snyk-win code test

**OWASP community listing  
  
  
A01:2021-Broken Access Control----**when someone unauthorized gets permission more than he/she deserves

**A02:2021-Cryptographic Failures ---** when developers fail to do proper encryption of sensitive information in the code

**A03:2021-Injection –** untrusted data is sent to the system/process/server exploiting data

**A04:2021-Insecure Design –** flaws in the structure/design of the application(flaw with the architecture of the application)

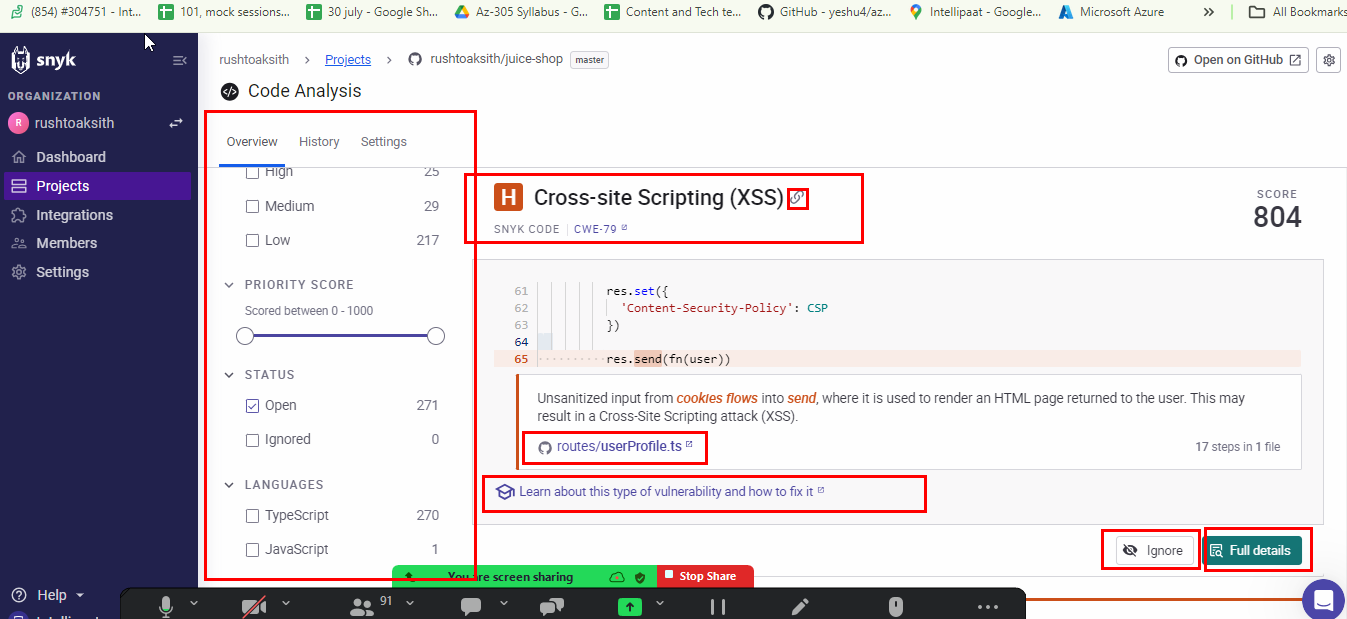
**A05:2021-Security Misconfiguration –** Security settings not configured properly

**A06:2021-Vulnerable and Outdated Components ---** use of third party plugins/ outdated components that do not align with the current security standards

**A07:2021-Identification and Authentication Failures ---** system is not able to authenticate information correctly

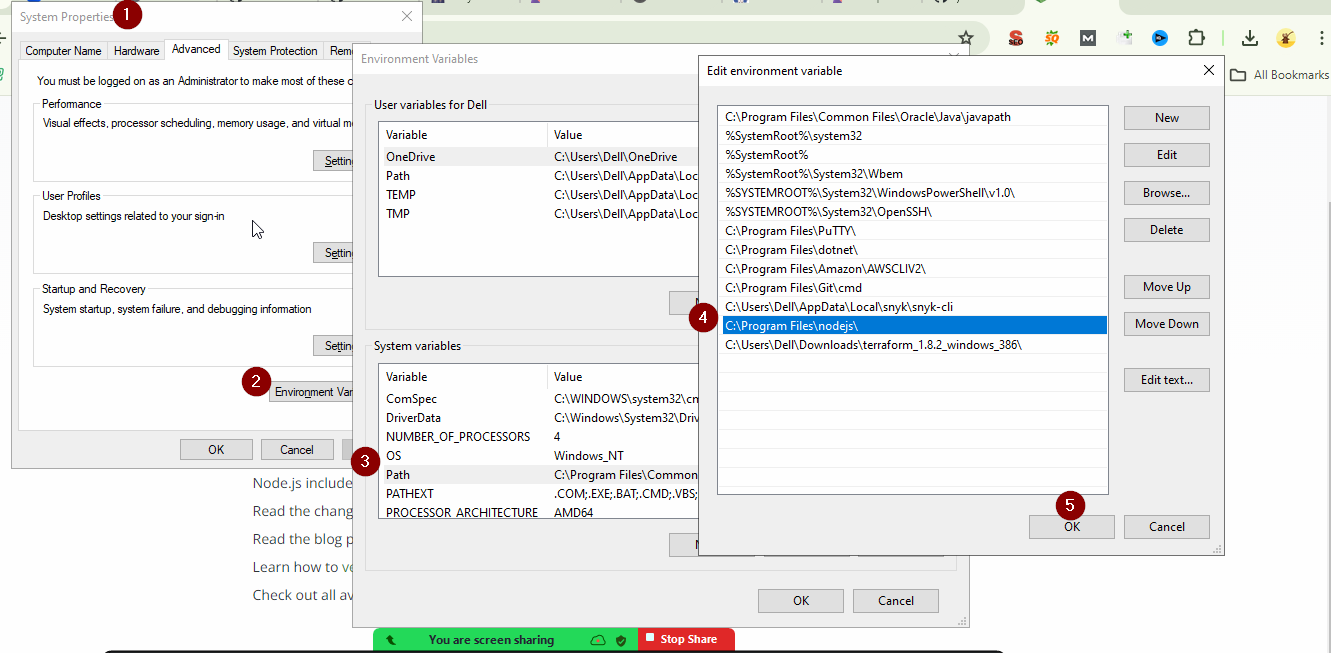
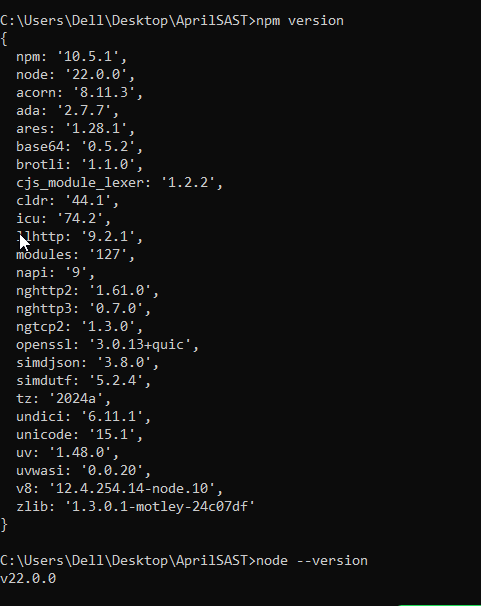
**A08:2021-Software and Data Integrity Failures –** data is compromised, data can be tampered

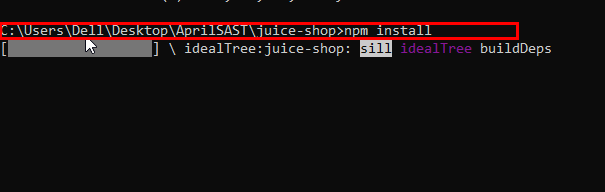
**A09:2021-Security Logging and Monitoring -**  Login issues and the system is not capable to take correct actions while monitoring any malicious process/attack

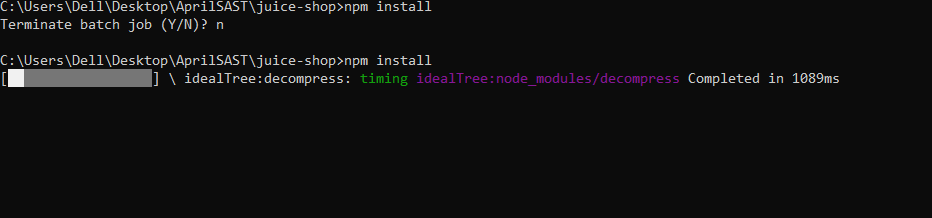
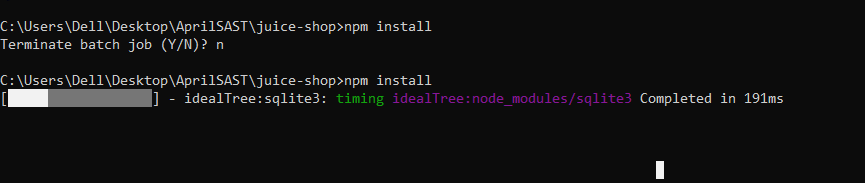
**A10:2021-Server-Side Request Forgery –** where attacker can manipulate your application to make request on your behalf and access more internal information or further connected systems/servers. **CWE CODE MAPPING: https://cwe.mitre.org/  
  
Snyk Dashboard: https://snyk.io/platform/snyk-cli/  
  
**

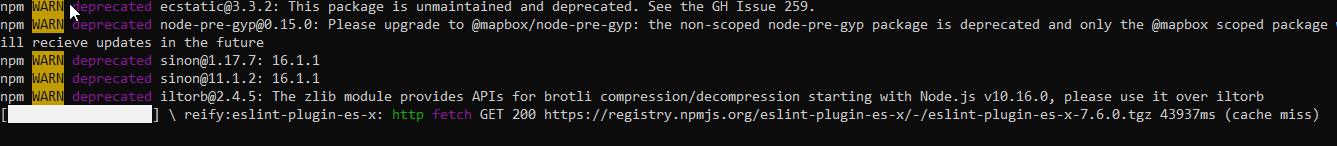
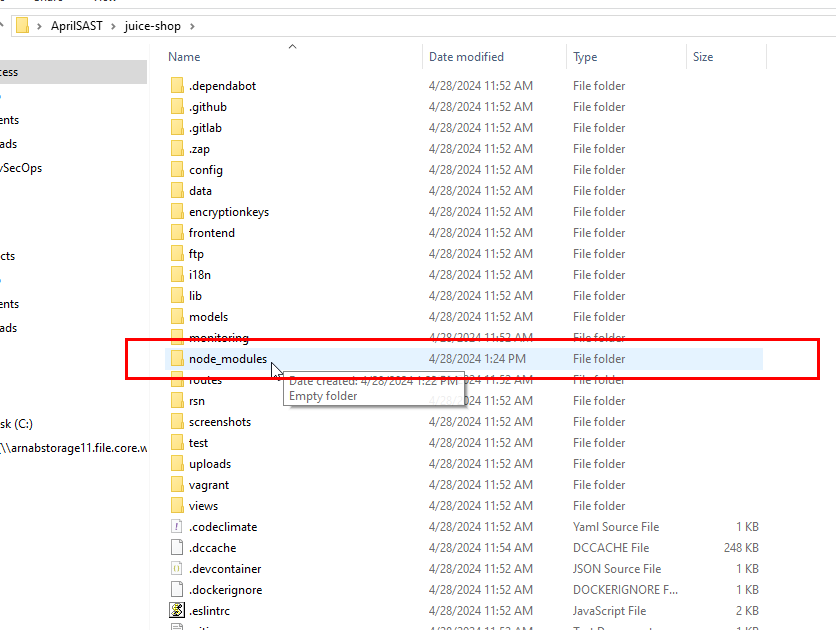
**SCA**  
You need to have a node-module folder generated in your system.

It can be achieved by building your dependency file, i.e. package.json in case of javascript based application. Now, in order to build you need a package manager called as nodejs/npm.  
  
<https://nodejs.org/dist/v20.12.2/node-v20.12.2-x64.msi>

In case, node js version or npm version isn’t displayed in your cmd.  
  
  
  


Building dependencies: npm install  


**Command: Snyk-win test   
NOTE: you shouldn’t be testing dependencies while being in the dependencies folder i.e. node\_module for java scrit when file package.json**

**SECURE CODE**

**Code Review ---🡪**

**Four Approaches --🡪**

**Thread Modeling----🡪**

1. **Identifying Assets:** Figure out what has to be protected, passwords, system, data.
2. **Identifying threats:** You need to think of the ways that can be used to steal, misuse, or theft.
3. **Analyze Issue:**
4. **Remove Risk:** You need to work on the approach that should mitigate further issues.

**DAST**

[**https://github.com/zaproxy/zaproxy/releases/download/v2.14.0/ZAP\_2\_14\_0\_windows.exe**](https://github.com/zaproxy/zaproxy/releases/download/v2.14.0/ZAP_2_14_0_windows.exe) **Installing: 8081**