Dependency Security –

* Regular Audit of dependencies –

Whatever you have is actually up to date, it does not have any security issues and what not how you can achieve it?

If you are using npm or yarn then you can do npm audit or yarn audit that you can do.

The other thing, once you get some issues or you got some security concerns depending on severity then you can fix that automatically or you can fix that one by one or you can update the things one by one and then basically see how things basically work.

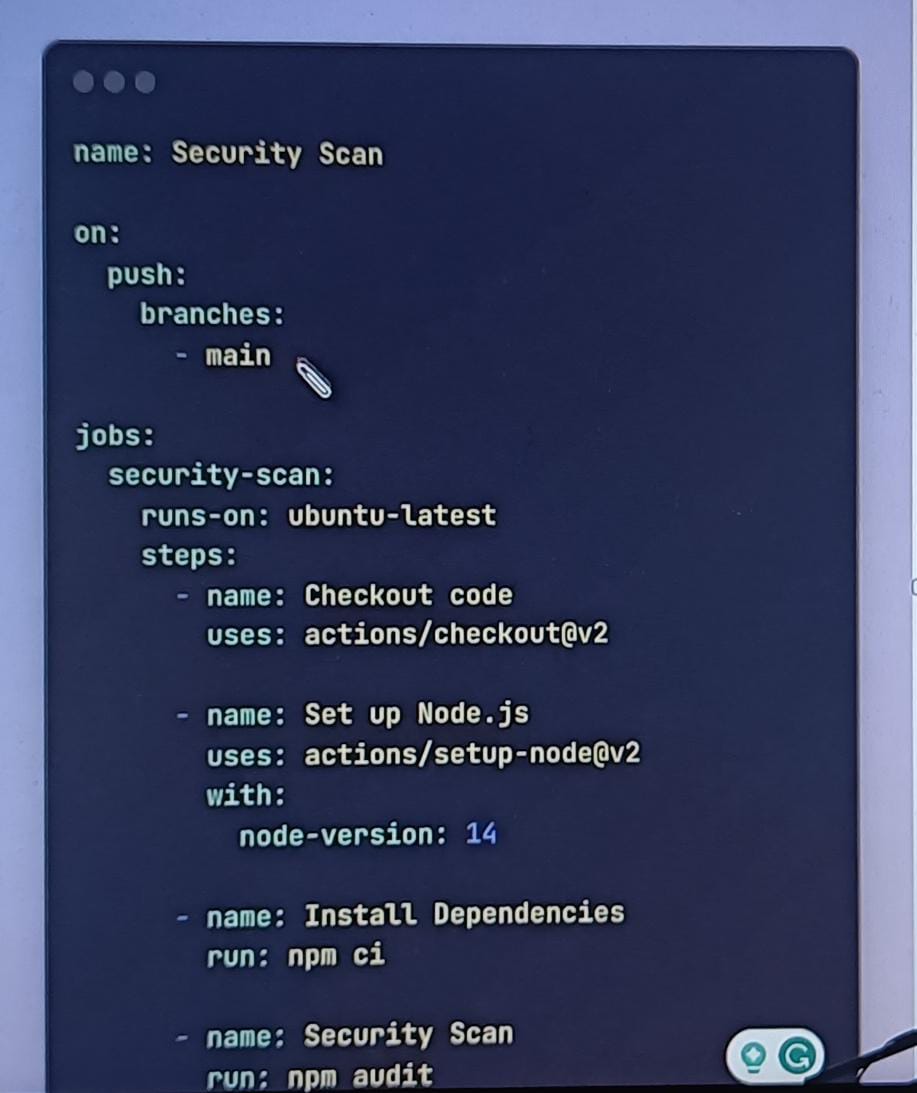
* Enforcing auditing –

In your npm package itself execute something which is called npm set audit true, anytime you do npm install, anytime you do npm update this is going to ensure that if you are doing any update anything is actually over there it will take care of, npm audit actually does behind the scene.

Now there are few more things example consider example that you do not came to your application or you did not do a development for your side project for a long time then can be some vulnerability come around, how you can ensure such vulnerability can highlighted, your existing application is in maintenance mode and you did not do any development but you want to ensure that it does not come in security compliances.

There is a way which is called “DEPENDABOT” what it does it helps you to basically find what are the outdated dependencies are how it basically does? You have to go into GitHub, then you have to get into the dependabot.yml and you have to configure it, so it says please schedule a weekly analysis of dependency resolution if there is anything which is outdated which has security issue please try to get those dependencies and highlighted as an report and please give these reports on a weekly basis.

Now consider you wanted to take report and you wanted to do custom auditing in terms of,



I wanted to ensure, I write some own GitHub hook you can do that, in this case what I have done basically on my main branch if someone basically try to push then please run these steps, setup a ubuntu-latest and basically whenever there is a checkout kind of thing you have to do a node setup kind of thing, it will install the dependency and once it install the dependency then please do the npm audit which is scan the security and what not, this will give the report anytime the commit happening into your main branch that is what it will hold the custom stuff as well.

* Code & Dependency Monitoring –

You can use something which is called codeql, codeql is a github npm package, what it does?

It helps you to basically not just monitor your external dependencies which you can do just auditing but also scanning your code and see not only the dependency can be culprit but also your code can be culprit.

* Dependency locking –

If you understood this is something which is stable, it is working and you have a pipeline regular weekly things which is happening they are some github actions like whenever there is something merge or something pull request which is generated you have set up of those pipeline all those security check has been done now you wanted to avoid a frequent dependency error which is coming what you do you actually must have something called package-lock.json, you might have seen this file, it also help in two perspective to lock the version of parent dependency and the nested dependency in your project so it is not going to change every time you do npm install or every time you make any changes in your product it is not going to change so you lock those dependency and you update those dependency only when needed.

* Security Penetration Testing using tools–

There are many scanning tools are exist in the industry,