Background: This activity demonstrates how Prisma Cloud alerts on attacks on cloud resources and how you can leverage the Prisma Cloud data correlation to analyze the attacks in more detail and remediate them. **In this activity, you will:**

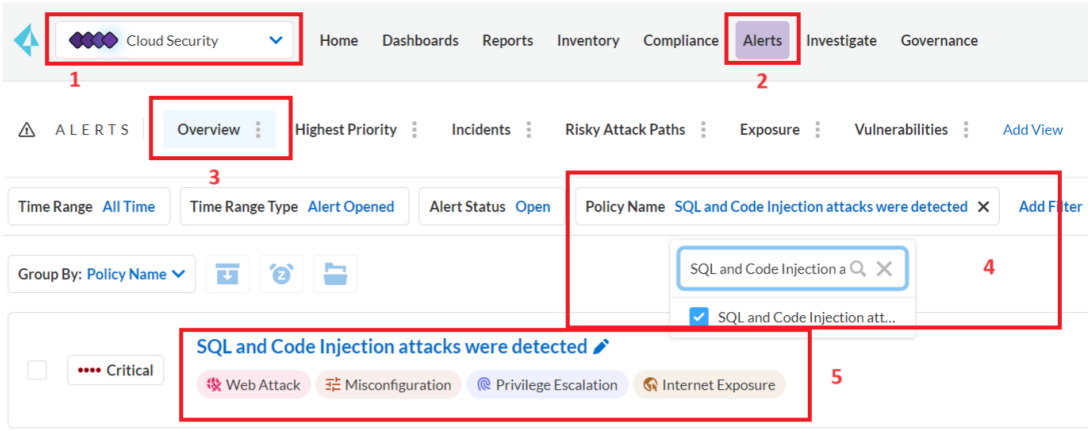
- View Alerts on risky SQL, Code injection, and other attacks.
- Analyze the attack paths and the resources involved in them.
- View how Prisma Cloud can be leveraged to remediate and block the attacks.

Note: This is a standalone activity and is not dependent on other activities.

---------- Task 1: Investigate Attacks on Cloud Resources ----------

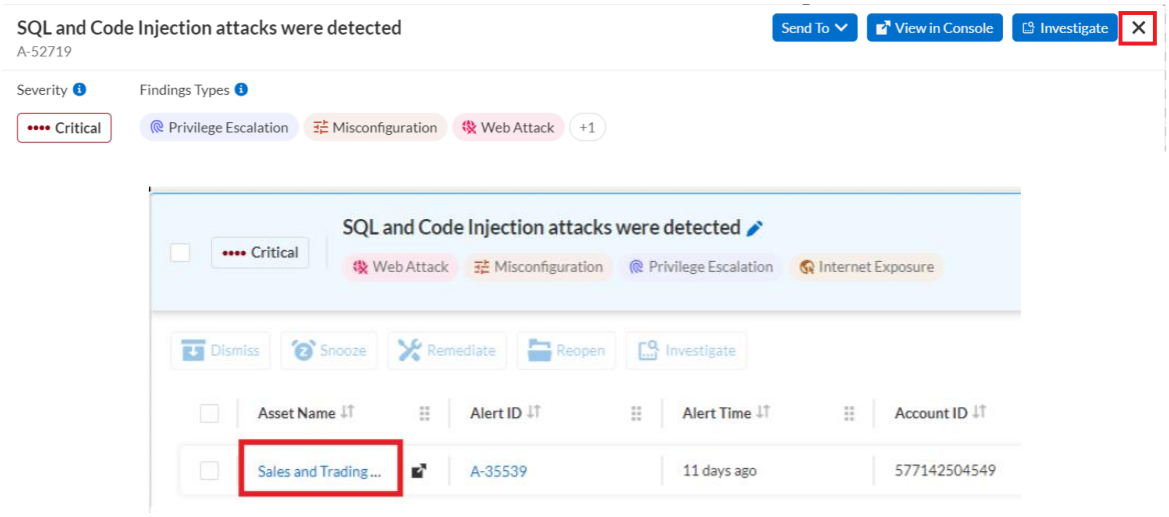
Step 1. Navigate to **Prisma Cloud Enterprise Edition > Cloud Security > Alerts** > **Overview**.
Step 2. Set the following filters (you can add additional filters by clicking on the **Add Filter** button):

a) Time Range = **All Time**
b) Policy Name = **SQL and Code Injection attacks were detected**

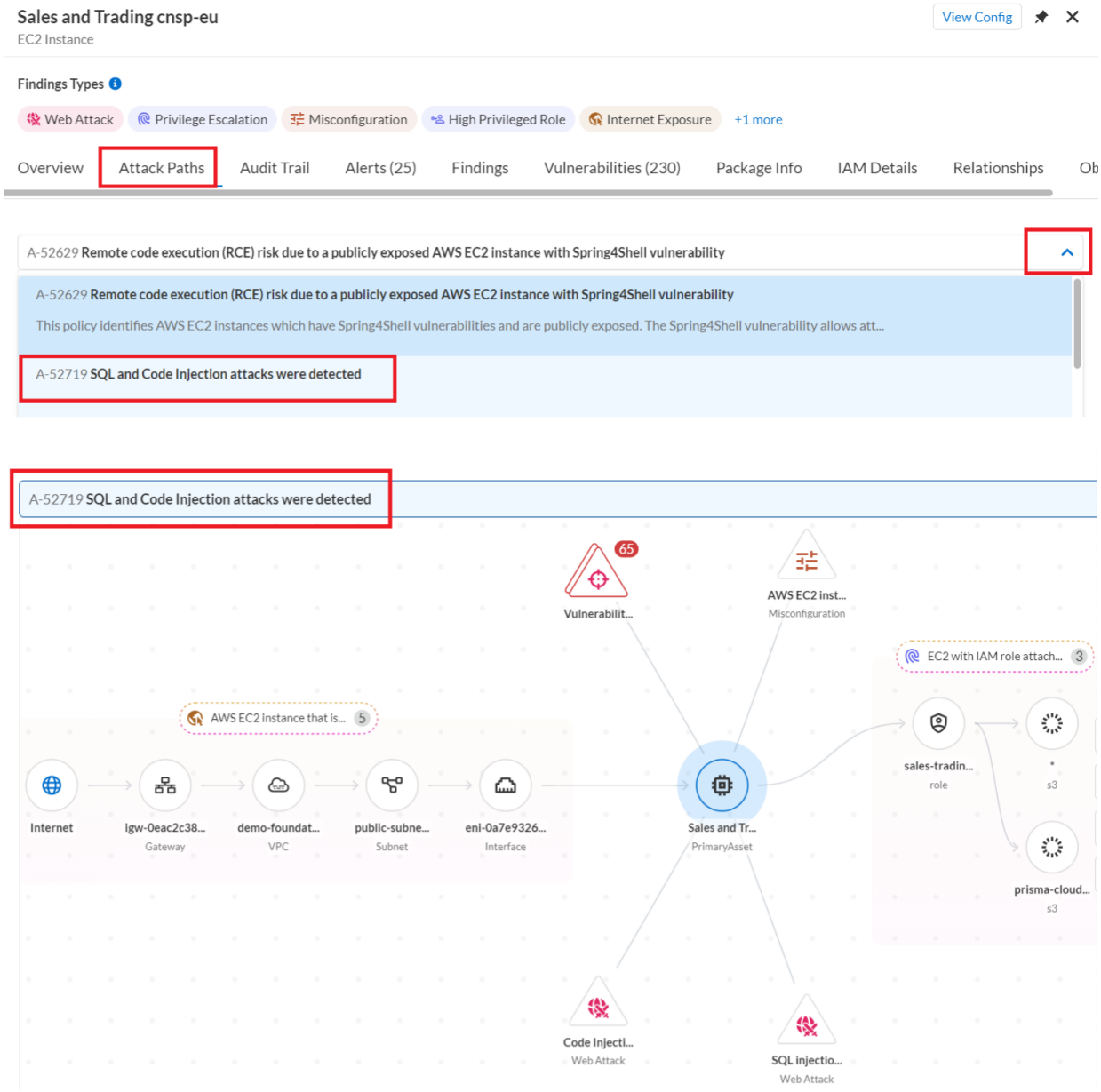


Notes: SQL and Code Injection attacks were detected is a custom search policy and Alert rule that was created for the lab by leveraging out-of-the-box Prisma Cloud policies. This is done for ease of use and convenience of lab experience.

Step 4. Click on the **SQL and Code Injection attacks were detected** result and click on the **Sales and Trading cnsp-app4** under the **Asset Name** column.

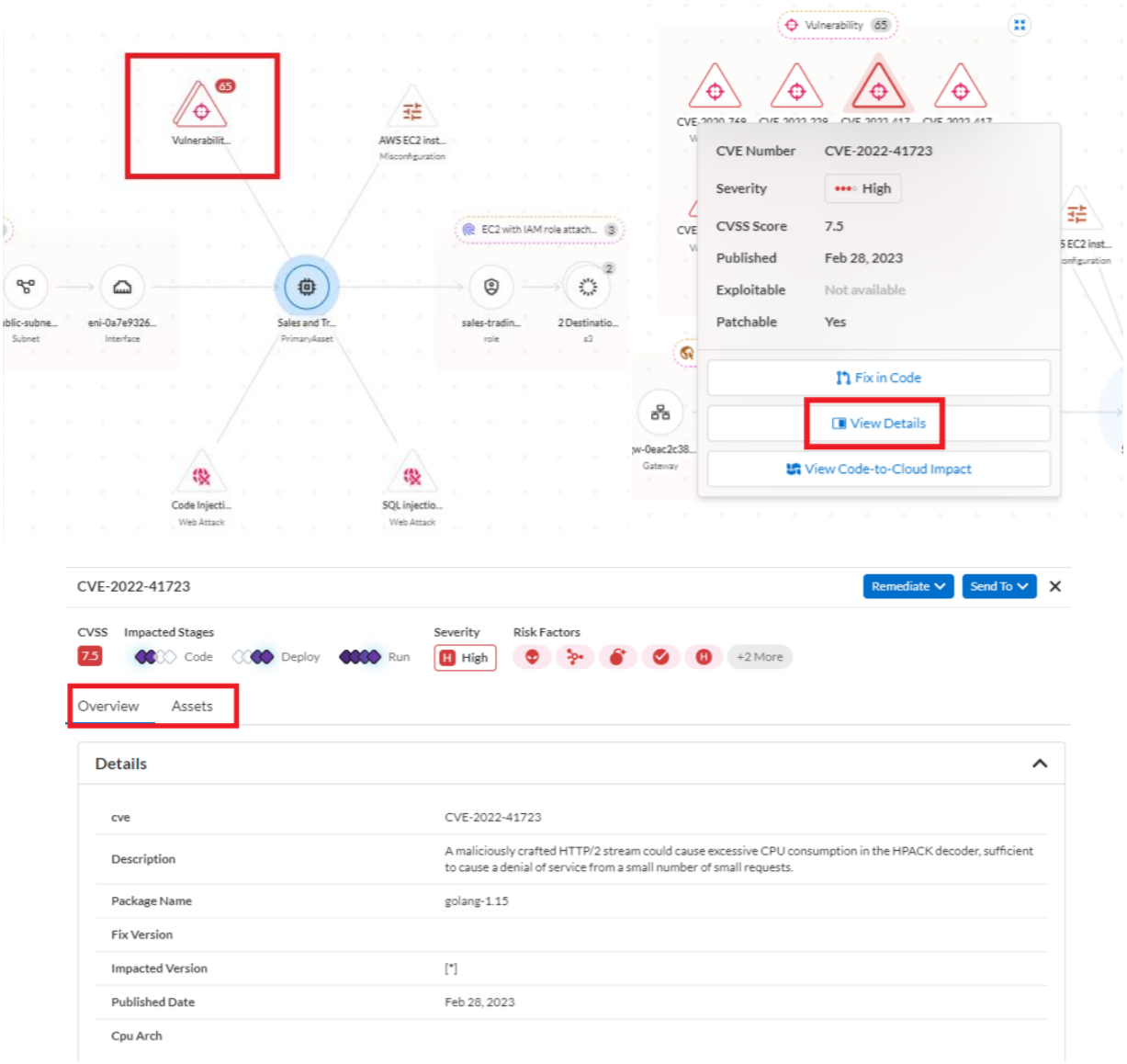


Step 5. This should bring up details about this instance and contain more findings and information about the attacks. The **Overview** tab contains the overview of this VM. Clicking on the **Attack Path** will reveal how the attack occurred and the resources involved in it.

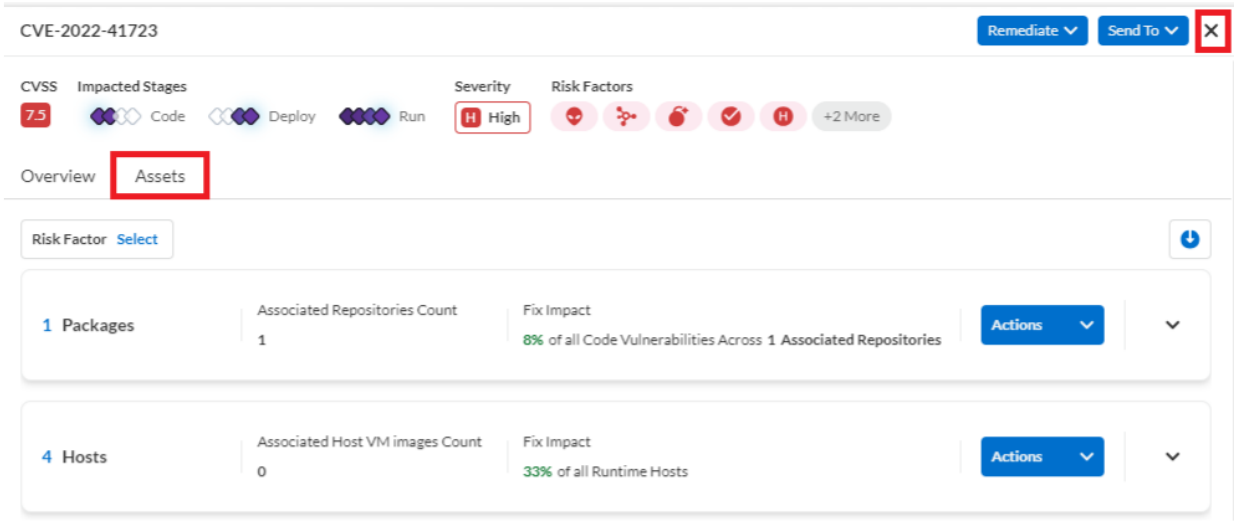


Step 6. Within the graph, you can see the traffic flow and the complete attack path. The traffic enters through the **Internet Gateway** and hits the **Sales and Trading EC2** instance. This instance has an IAM role **sales-trading-admin-role-cnsp-app4** attached to it, which has **wildcard access to the S3 bucket** and also access to the **prisma-cloud-pcds-bucket** S3 Bucket. As this instance is vulnerable to **SQL and Code injection attacks**, the attacker can potentially perform a **Data exfiltration** from the S3 bucket through the compromised host. This is represented by the Attack Path.

Step 7. Within the graph, clicking on the **Vulnerabilities** Icon and clicking on any **Vulnerability** will provide more information about the vulnerability. On the selected Vulnerability. clicking on **View Details** will open an additional findings sidecar.



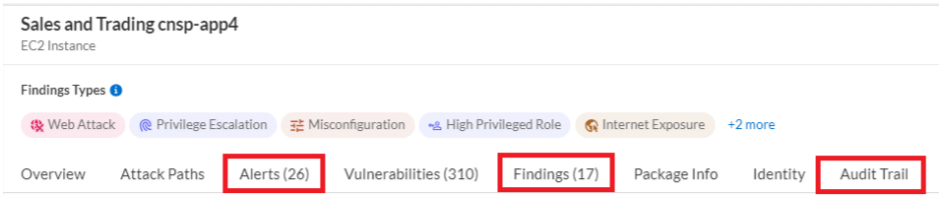
Step 8. Clicking on **Assets** will show more information about the affected assets. Once done, close the **CVE Sidecar.**



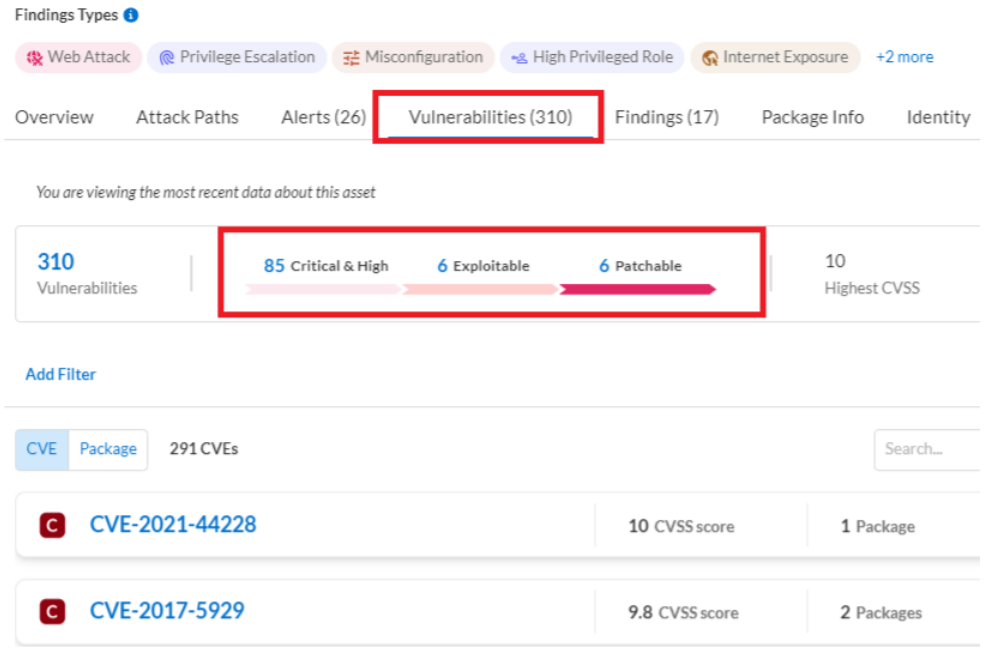
Step 9. Clicking on **Audit Trail** will show more information about what changes occurred at the resource level and the timeline.

The **Alerts** tab will show all the alerts that are that this resource is currently involved in.

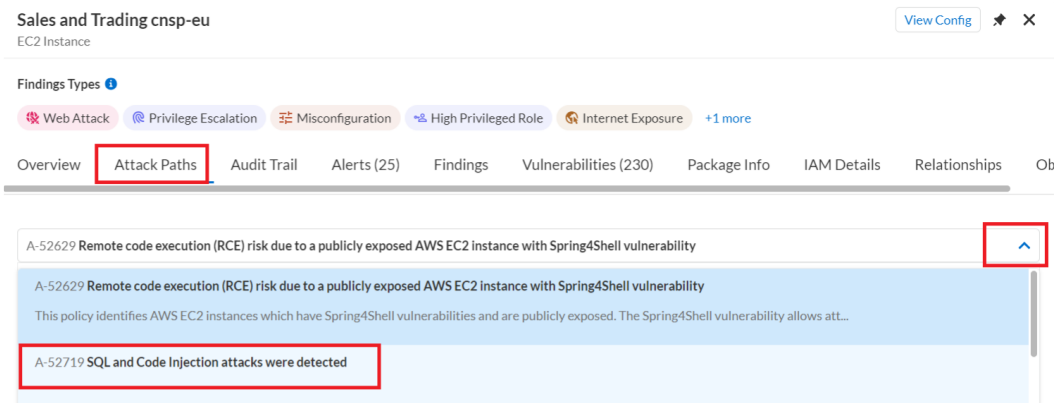
The **Findings** tab shows the findings for this specific resource.



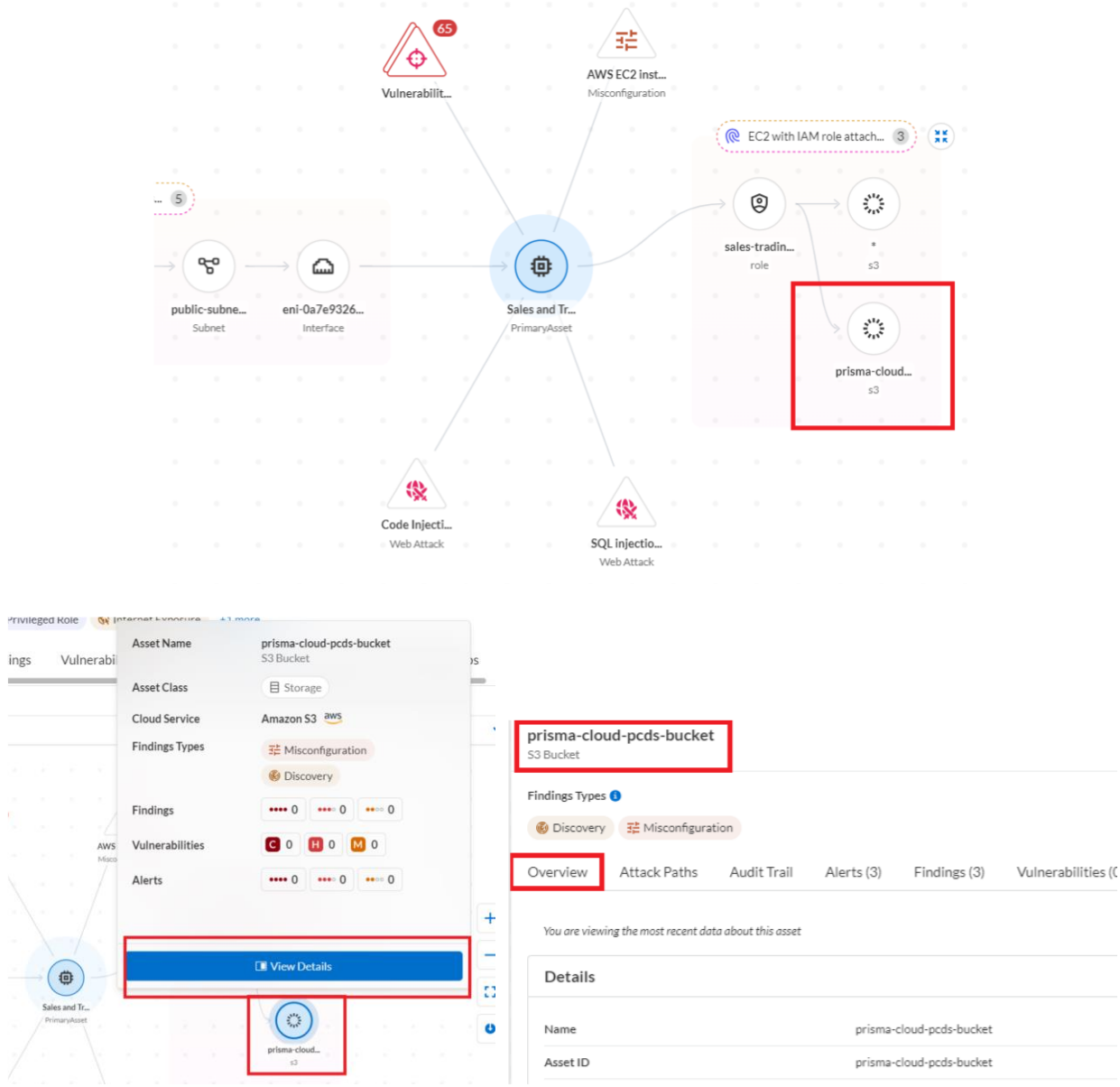
Step 10. Within the **Sales and Trading cnsp-app4** VM sidecar, click on the **Vulnerabilities** tab to see all the Vulnerabilities that were detected for this VM. Further clicking on options such as **Critical & High**, **Exploitable** and **Patchable** will filter the results.



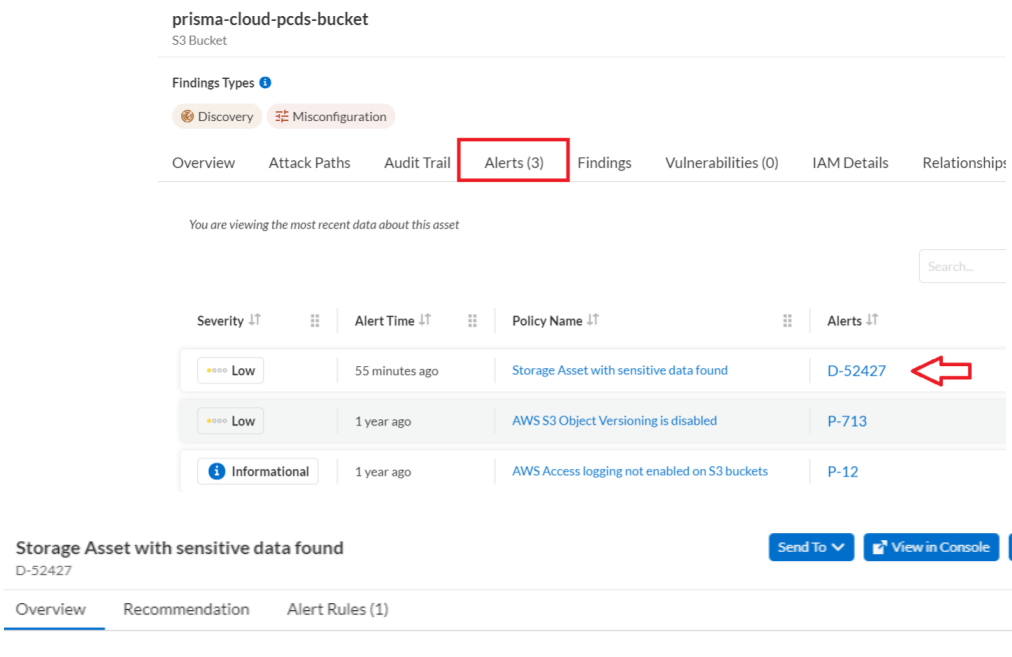
Step 11. Navigate back to the **Attack Path** graph (and make sure that the **"SQL and Code Injection attacks were detected"** alert is selected). Let's further investigate the Attack Path in the graph and examine the blast radius.



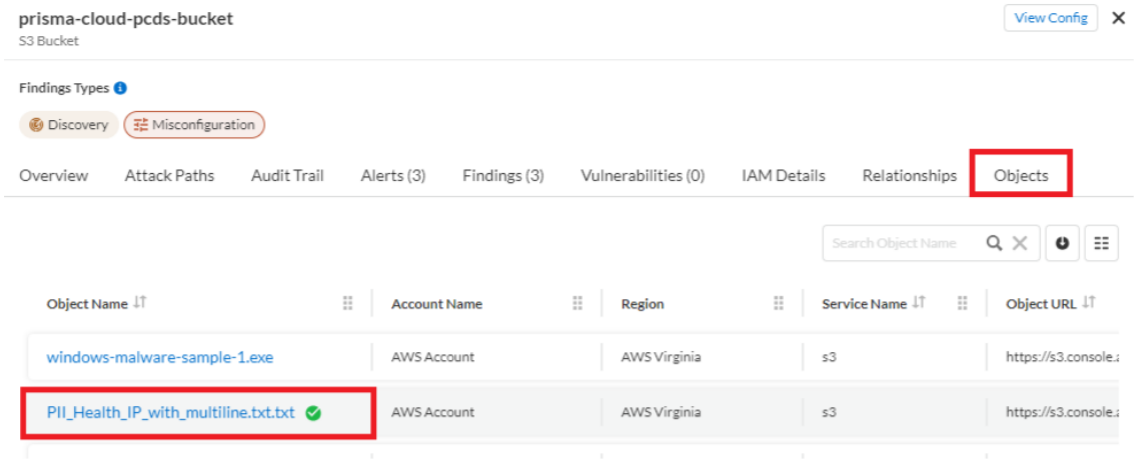
Step 12. Within the graph, click on the S3 Bucket **prisma-cloud-pcds-bucket** by clicking on it and clicking on **View Details**

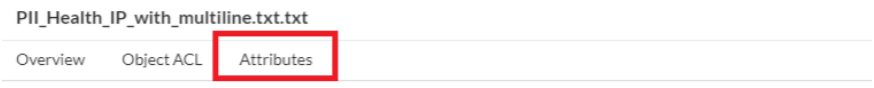


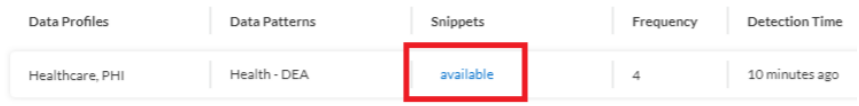
a) Here you can see the S3 Bucket details. Head over to the **Alerts** tab in the S3 Bucket details and here you will see that there's an alert for this bucket- **Storage Asset with sensitive data found. Do not click on the Policy Name as clicking on the Policy Name as you will be navigated to another page.** Once done looking at Alerts, close the **Alert** window

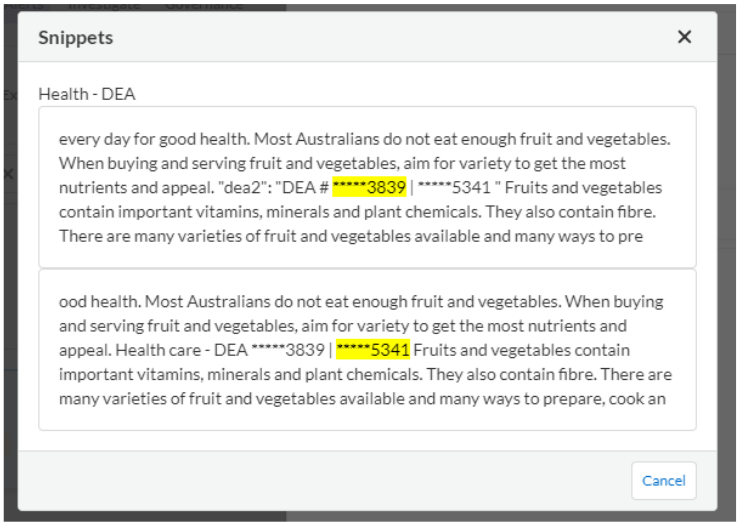


b) Within the S3 Bucket page, click on **Objects.** Here you can see the list of all the objects that reside in that S3 Bucket and these are at the risk of being exfiltrated as a result of the **SQL and Code injection attacks.** If you click on **PII_Health_IP_with_multiline.txt.txt** and head over to **Attributes,** you can see that this file contains sensitive personally identifiable information.



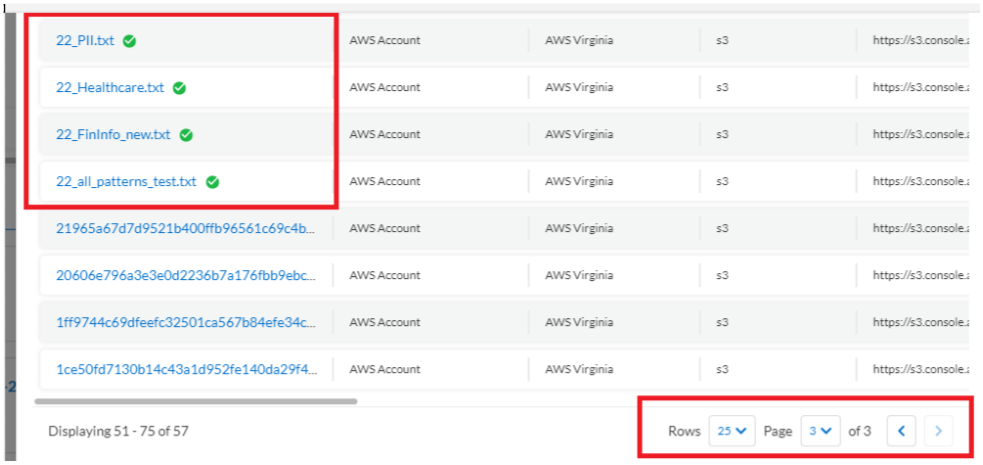


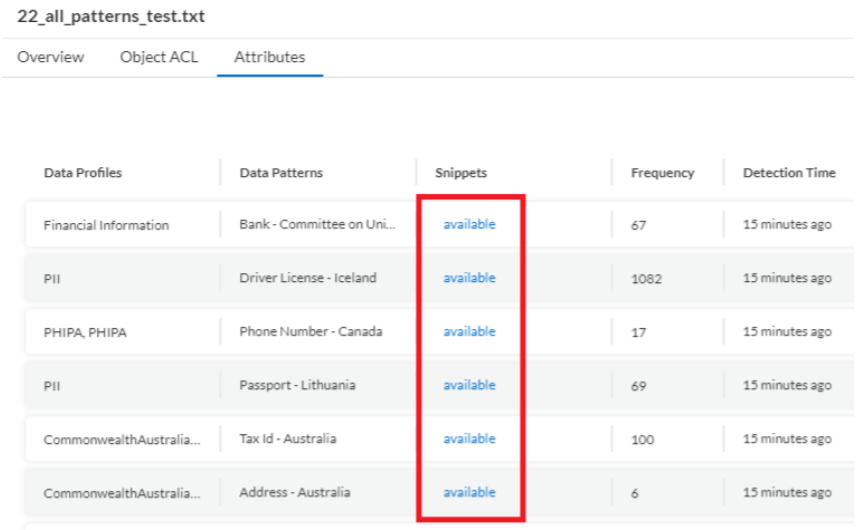




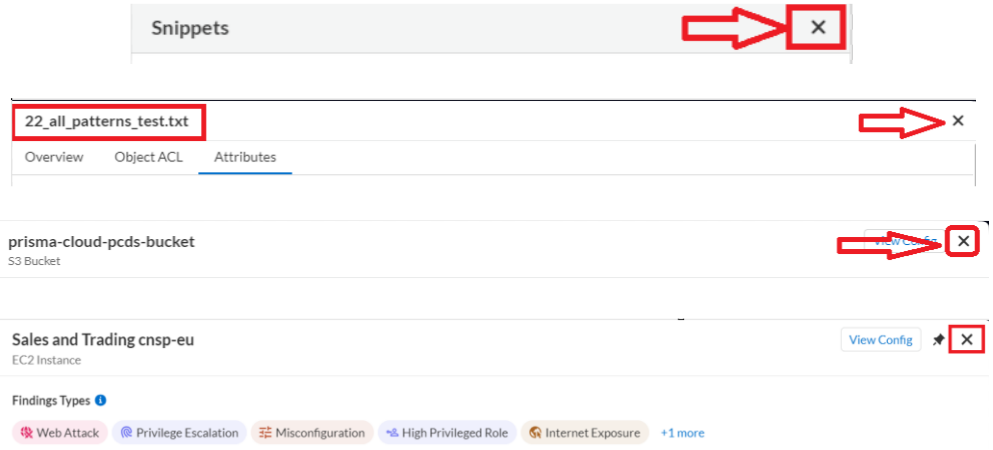
c) Here, Prisma Cloud has identified that there's sensitive data stored within the S3 Bucket that the compromised host has access to and is at risk of data exfiltration. To identify and detect confidential and sensitive data, **Prisma Cloud Data Security** integrates with **Palo Alto Network's Enterprise DLP service** and provides **built-in data profiles**, which include data patterns that match sensitive information such as PII, health care, financial information, and Intellectual Property.

d) On the bucket, if you go to **page 3,** you can see more findings, and clicking on **22_all_patterns_test.txt** will show you more findings.

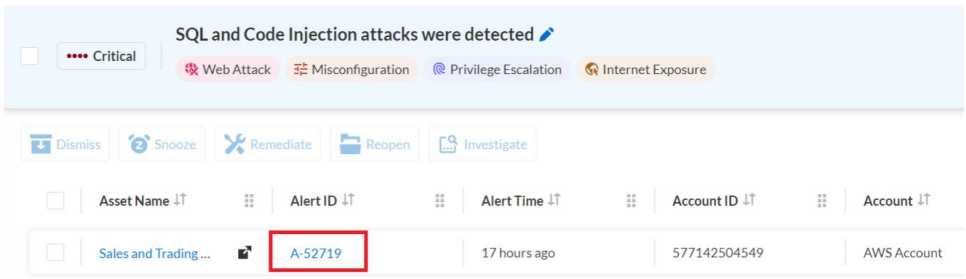


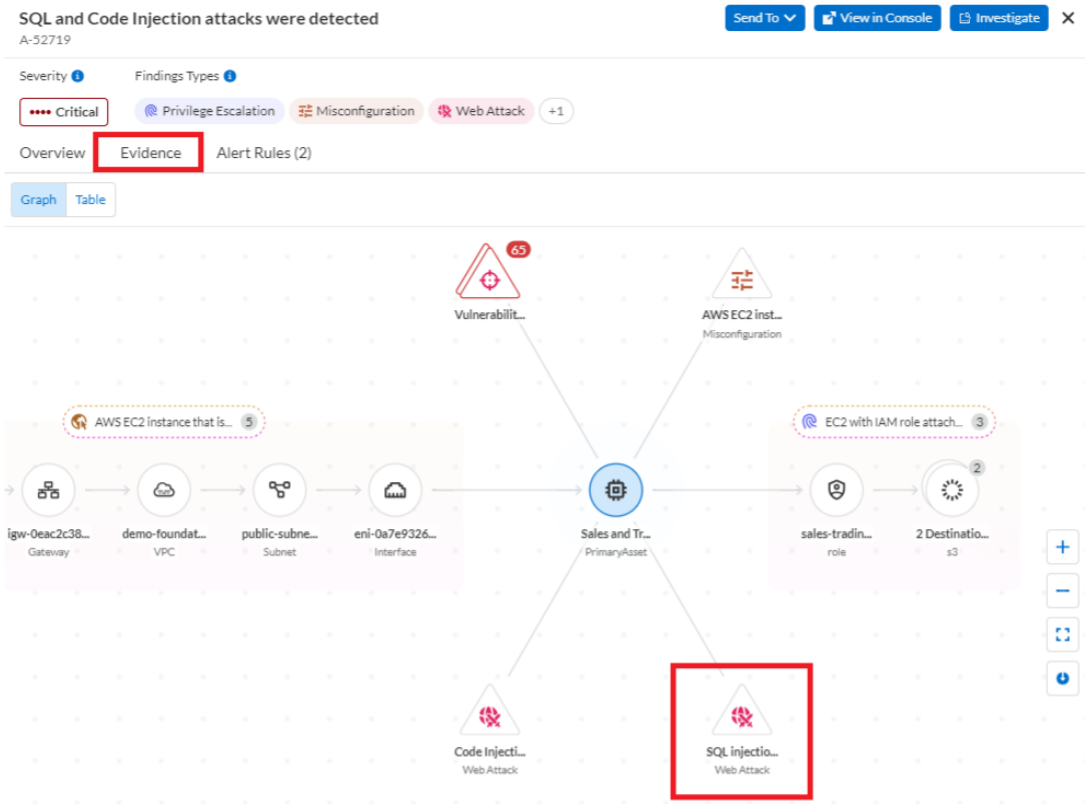


Step 13. Close the **Snippets** pop-up. Close the **Object** sidecar and close the **S3 bucket** sidecar. Close the **Sales and Trading cnsp-app4** sidecar

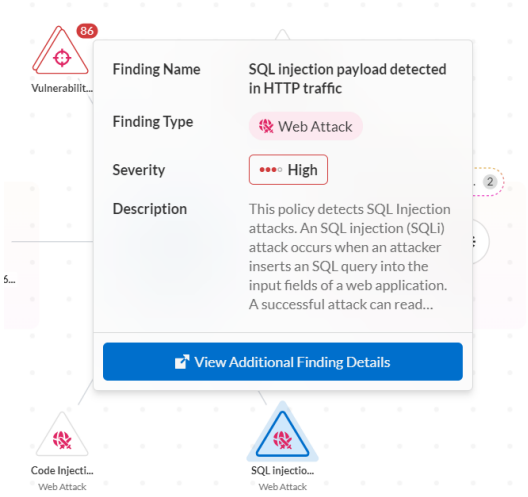


Step 14. In the **SQL and Code injection attacks were detected** Alert window, click on the value corresponding to the **Alert ID** column. In the next windows, click on the **Evidence** tab.

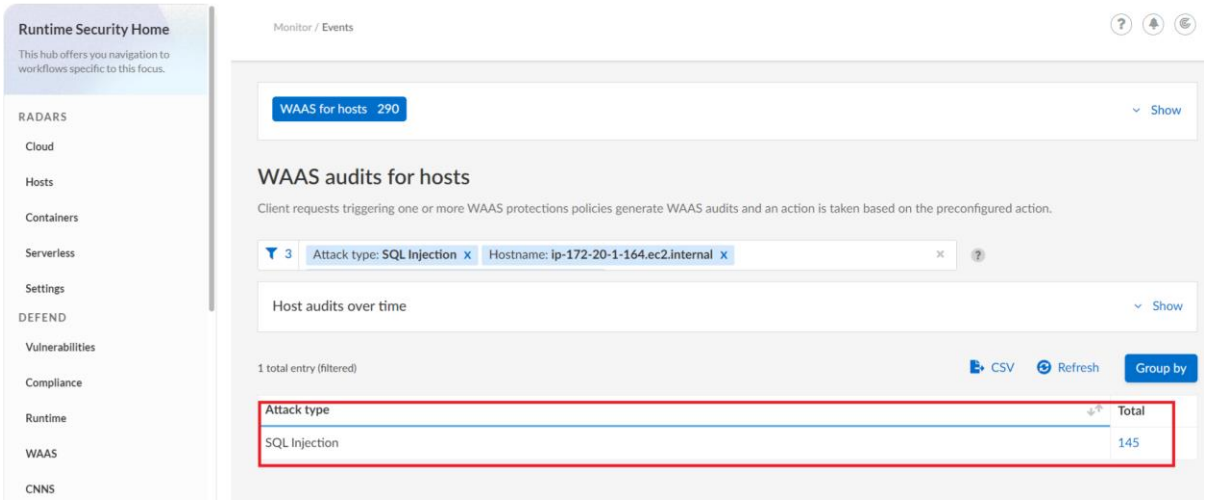




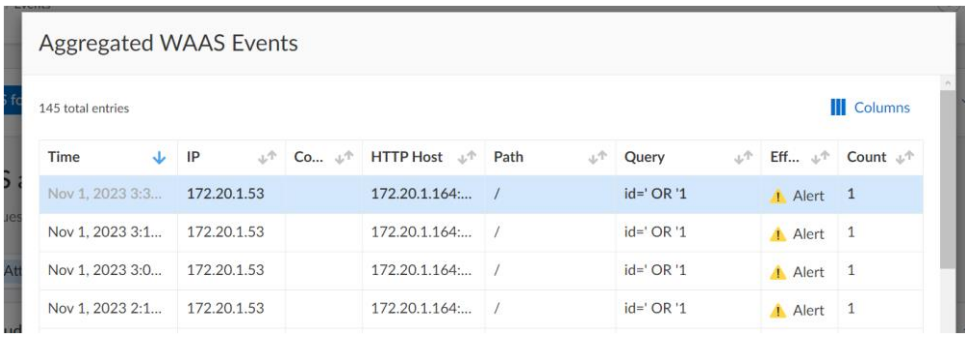
Step 15. Click on the **SQL Injection** and click on **View Additional Finding Details.** You will now be directed to the **Prisma Cloud Runtime Security WaaS (Web Application and API Security) console**, which was responsible for detecting this attack. Here we will be able to see more information about the attack.



Note: The below screenshot might look different in your case as some menus are collapsed to fit the screenshot on the page.

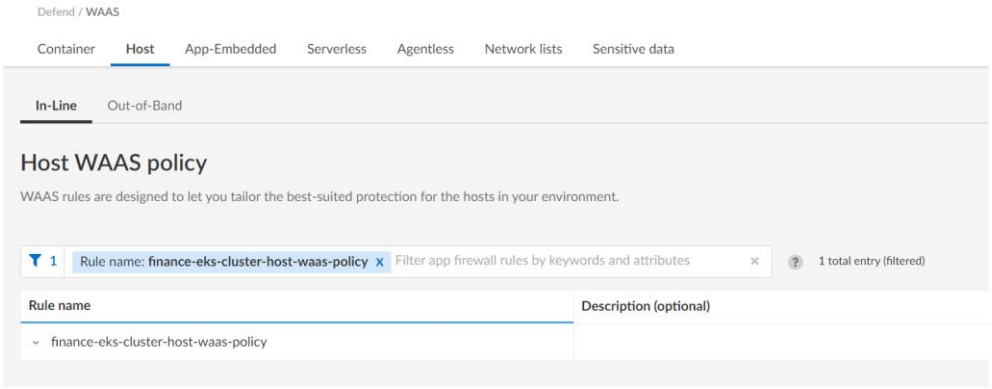


Step 16. Click on the **SQL Injection** number to bring up the **Aggregated WaaS Events** for that attack type.

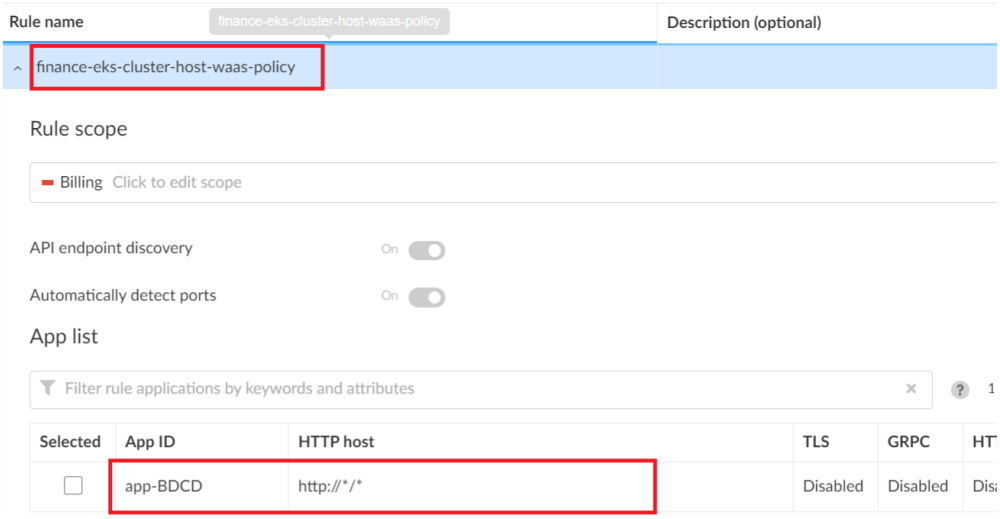


Step 17. As you can see, attacks are detected and the effect is set to **Alert.** This can be changed to **Prevent** to prevent the attacks. But we will **not be performing** that action in this lab but we will show how it can be achieved.

Step 18. If you scroll further down, you can see which rule is responsible for detecting the attacks and this would be **finance-eks-cluster-waas-policy.** Click on the rule and select **Yes** for the pop-up dialogue.



Step 19. Click on the rule and click on the item under the App List



Step 20. Click on the **App Firewall** tab and you should see the **SQL Injection** and other items (Grayed out because the lab role is a read-only role and doesn't have permissions to change settings), which can be set to **Prevent** or **Ban** to prevent these attacks