

## graph\_RustSec

 [GitHub - rustsec/advisory-db: Security advisory database for Rust crates published through crates.io](https://github.com/rustsec/advisory-db)

```
1 CREATE TABLE test.t_tmp_craw_rustsec(id varchar, vul_msg jsonb)
2
3
4 DROP TABLE IF EXISTS test.t_tmp_node_vul_RUSTSEC;
5 CREATE TABLE test.t_tmp_node_vul_RUSTSEC AS
6 SELECT t.id,
7        COALESCE (t.vul_msg -> 'advisory' ->> 'aliases', '[]') AS aliases,
8        '{}'::jsonb AS SOURCE,
9        '{}'::jsonb AS description,
10       '{}'::jsonb AS weakness,
11       CASE WHEN t.vul_msg -> 'advisory' ->> 'cvss' IS NULL THEN '{}'::jsonb ELSE jsonb_build_object('version',
12       jsonb_build_object('published', t.vul_msg -> 'advisory' ->> 'date', 'lastModified', NULL, 'datePublic',
13       '{}'::jsonb AS status
14 FROM test.t_tmp_craw_rustsec t
15
16
17
18
19 DELETE FROM test.dws_graph_node_vul WHERE vul_source = 'RUSTSEC';
20 ALTER SEQUENCE cve_graph_seq RESTART START WITH 1;
21 INSERT INTO test.dws_graph_node_vul
22 SELECT nextval('cve_graph_seq') AS seq, tmp.*
23 FROM
24 (
25  SELECT DISTINCT id , aliases , "source", description ,weakness, severity, time_info, status, 'RUSTSEC' AS v
26 )tmp ;
27 bash gen_graph_data.sh "RUSTSEC" "node" "vul"
28 bash neo4j_vul_node_load.sh "RUSTSEC" "0"
29
30
31
32
33
34
35 --affecte component
36
37
38 DROP TABLE IF EXISTS test.t_tmp_graph_node_component_RUSTSEC;
39 CREATE TABLE test.t_tmp_graph_node_component_RUSTSEC AS
40 SELECT t.id, '-' AS component,
41        split_part( t.vul_msg -> 'advisory' ->> 'url' , '/', 4) AS vendor,
42        t.vul_msg -> 'advisory' ->> 'package' AS package,
43        'GIT' AS ecosystem,
44        '' AS repo_url,
45        t.vul_msg -> 'affected' ->> 'os' AS platform,
46        t.vul_msg -> 'advisory' ->> 'url' AS collectionURL,
47        '' AS defaultstatus,
48        jsonb_build_object('function_versions', t.vul_msg -> 'affected' ->> 'functions') AS affected_versions,
49        jsonb_build_object('fixed', t.vul_msg -> 'versions' ->> 'patched', 'unaffected', t.vul_msg -> 'versions'
```

```

50         t.vul_msg
51 FROM test.t_tmp_craw_rustsec t
52
53
54 SELECT substring( 'https://github.com/japarc/heapless/issues/181', 1, POSITION ( 'issues' IN 'https://github.
55
56
57 DELETE FROM test.dws_graph_node_affected_component WHERE vul_source = 'RUSTSEC';
58 ALTER SEQUENCE cve_graph_seq RESTART START WITH 1;
59 INSERT INTO test.dws_graph_node_affected_component
60     SELECT nextval('cve_graph_seq') AS seq, tmp.*
61 FROM (
62     SELECT DISTINCT component, vendor, package, ecosystem, 'RUSTSEC' AS vul_source FROM test.t_tmp_graph
63 )tmp;
64 bash gen_graph_data.sh "RUSTSEC" "node" "affected_component"
65 bash neo4j_affected_component_node_load.sh "RUSTSEC" "0"
66
67
68
69
70 MATCH (n:affected_components) where n.ecosystem = 'GIT' and n.package_name = '*' and n.component_name = '*'
71
72
73 MATCH (n:Vuln_RUSTSEC)-[affected_components]->()
74 DELETE affected_components
75
76
77
78
79
80
81 DELETE FROM test.dws_graph_relationships_affected_components WHERE vul_source = 'RUSTSEC';
82 ALTER SEQUENCE cve_graph_seq RESTART START WITH 1;
83 INSERT INTO test.dws_graph_relationships_affected_components
84     SELECT nextval('cve_graph_seq') AS seq, tmp.*
85 FROM (
86     SELECT DISTINCT id, component, vendor, package , ecosystem, '' repo_url, platform,
87         collectionurl, '' AS defaultstatus, affected_versions, unaffected_versions, 'RUSTSEC' AS vul_sour
88 )tmp;
89 bash gen_graph_data.sh "RUSTSEC" "relationships" "affected_components"
90 bash neo4j_relationships_affected_components.sh "RUSTSEC" "0"
91
92
93
94
95
96
97 ---reference
98 DROP TABLE IF EXISTS test.t_tmp_graph_node_refs_RUSTSEC;
99 CREATE TABLE test.t_tmp_graph_node_refs_RUSTSEC AS
100 SELECT t.id, refs -> 0 AS ref_url
101 FROM test.t_tmp_craw_rustsec t,
102 jsonb_array_elements(t.vul_msg -> 'advisory' -> 'references') AS refs
103
104
105
106
107 DELETE FROM test.dws_graph_node_refs WHERE vul_source = 'RUSTSEC';

```

```
108 ALTER SEQUENCE cve_graph_seq RESTART START WITH 1;
109 INSERT INTO test.dws_graph_node_refs
110     SELECT nextval('cve_graph_seq') AS seq, tmp.*
111     FROM (
112         SELECT DISTINCT ref_url, 'RUSTSEC' AS source FROM test.t_tmp_graph_node_refs_RUSTSEC
113     )tmp;
114 bash gen_graph_data.sh "RUSTSEC" "node" "refs"
115 bash neo4j_refs_node_load.sh "RUSTSEC" "0"
116 -----relationships refs
117 --relationship seq int , vul_id varchar, url varchar, tags varchar, description varchar, vul_source varchar
118 DELETE FROM test.dws_graph_relationships_refs WHERE vul_source = 'RUSTSEC';
119 ALTER SEQUENCE cve_graph_seq RESTART START WITH 1;
120 INSERT INTO test.dws_graph_relationships_refs
121     SELECT nextval('cve_graph_seq') AS seq, tmp.*
122     FROM (
123         SELECT DISTINCT id, ref_url , '' AS tags, '' AS ref_desc , 'RUSTSEC' AS vul_source FROM test.t_tmp_gr
124     )tmp;
125 bash gen_graph_data.sh "RUSTSEC" "relationships" "refs"
126 bash neo4j_relationships_refs_load.sh "RUSTSEC" "0"
127
128
129
130
131
132
133
134
135
136
137
138
139
140
141
```