graph_RustSec

GitHub - rustsec/advisory-db: Security advisory database for Rust crates published through crates.io

```
CREATE TABLE test.t_tmp_craw_rustsec(id varchar, vul_msg jsonb)


DROP TABLE IF EXISTS test.t_tmp_node_vul_RUSTSEC;
CREATE TABLE test.t_tmp_node_vul_RUSTSEC AS
SELECT t.id,
       COALESCE (t.vul_msg -> 'advisory' ->> 'aliases', '[]') AS aliases,
       '{}'::jsonb AS SOURCE,
       '{}'::jsonb AS description,
       '{}'::jsonb AS weakness,
       CASE WHEN t.vul_msg -> 'advisory' ->> 'cvss' IS NULL THEN '{}'::jsonb ELSE jsonb_build_object('version',
       jsonb_build_object('published', t.vul_msg -> 'advisory' ->> 'date', 'lastModified', NULL, 'datePublic',
       '{}'::jsonb AS status
FROM test.t_tmp_craw_rustsec t




DELETE FROM test.dws_graph_node_vul WHERE vul_source = 'RUSTSEC';
ALTER SEQUENCE cve_graph_seq RESTART START WITH 1;
INSERT INTO test.dws_graph_node_vul
SELECT nextval('cve_graph_seq') AS seq, tmp.*
FROM
(
SELECT  DISTINCT  id ,  aliases , "source", description ,weakness, severity, time_info,  status, 'RUSTSEC' AS v
)tmp ;
bash gen_graph_data.sh "RUSTSEC" "node" "vul"
bash neo4j_vul_node_load.sh "RUSTSEC" "0"







--affecte component


DROP TABLE IF EXISTS test.t_tmp_graph_node_component_RUSTSEC;
CREATE TABLE test.t_tmp_graph_node_component_RUSTSEC AS
SELECT t.id, '-' AS component,
      split_part( t.vul_msg -> 'advisory' ->> 'url' , '/', 4) AS  vendor,
      t.vul_msg -> 'advisory' ->> 'package' AS package,
      'GIT' AS ecosystem,
      '' AS repo_url,
      t.vul_msg -> 'affected' ->> 'os' AS platform,
      t.vul_msg -> 'advisory' ->> 'url' AS collectionURL,
      '' AS defaultstatus,
      jsonb_build_object('function_versions', t.vul_msg -> 'affected' ->> 'functions') AS affected_versions,
      jsonb_build_object('fixed', t.vul_msg -> 'versions' ->> 'patched', 'unaffected', t.vul_msg -> 'versions'
```

```
      t.vul_msg
FROM test.t_tmp_craw_rustsec t


SELECT substring( 'https://github.com/japaric/heapless/issues/181', 1, POSITION ( 'issues' IN  'https://github.


DELETE FROM test.dws_graph_node_affected_component WHERE vul_source = 'RUSTSEC';
ALTER SEQUENCE cve_graph_seq RESTART START WITH 1;
INSERT INTO test.dws_graph_node_affected_component
  SELECT nextval('cve_graph_seq') AS seq, tmp.*
  FROM (
    SELECT   DISTINCT component, vendor,  package, ecosystem, 'RUSTSEC' AS vul_source   FROM   test.t_tmp_graph
  )tmp;
bash gen_graph_data.sh "RUSTSEC" "node" "affected_component"
bash neo4j_affected_component_node_load.sh "RUSTSEC" "0"




MATCH (n:affected_components) where n.ecosystem = 'GIT' and n.package_name = '*' and n.component_name = '*'


MATCH (n:Vuln_RUSTSEC)-[affected_components]->()
DELETE affected_components









DELETE FROM test.dws_graph_relationships_affected_components WHERE vul_source = 'RUSTSEC';
ALTER SEQUENCE cve_graph_seq RESTART START WITH 1;
INSERT INTO test.dws_graph_relationships_affected_components
  SELECT nextval('cve_graph_seq') AS seq, tmp.*
  FROM (
    SELECT   DISTINCT id, component, vendor, package , ecosystem, '' repo_url,   platform,
              collectionurl, '' AS defaultstatus, affected_versions, unaffected_versions, 'RUSTSEC' AS vul_sour
  )tmp;
bash gen_graph_data.sh "RUSTSEC" "relationships" "affected_components"
bash neo4j_relationships_affected_components.sh "RUSTSEC" "0"







---reference
DROP TABLE IF EXISTS test.t_tmp_graph_node_refs_RUSTSEC;
CREATE TABLE test.t_tmp_graph_node_refs_RUSTSEC AS
SELECT t.id, refs ->> 0 AS ref_url
FROM test.t_tmp_craw_rustsec t,
jsonb_array_elements(t.vul_msg -> 'advisory' -> 'references') AS refs




DELETE FROM test.dws_graph_node_refs WHERE vul_source = 'RUSTSEC';
```

```
ALTER SEQUENCE cve_graph_seq RESTART START WITH 1;
INSERT INTO test.dws_graph_node_refs
  SELECT nextval('cve_graph_seq') AS seq, tmp.*
  FROM (
    SELECT   DISTINCT ref_url, 'RUSTSEC' AS source  FROM   test.t_tmp_graph_node_refs_RUSTSEC
  )tmp;
bash gen_graph_data.sh "RUSTSEC" "node" "refs"
bash neo4j_refs_node_load.sh "RUSTSEC" "0"
-------relationships refs
--relationship seq int , vul_id varchar, url varchar, tags varchar, description varchar, vul_source varchar
DELETE FROM test.dws_graph_relationships_refs WHERE vul_source = 'RUSTSEC';
ALTER SEQUENCE cve_graph_seq RESTART START WITH 1;
INSERT INTO test.dws_graph_relationships_refs
  SELECT nextval('cve_graph_seq') AS seq, tmp.*
  FROM (
    SELECT   DISTINCT  id, ref_url , '' AS tags, '' AS ref_desc , 'RUSTSEC' AS vul_source  FROM   test.t_tmp_gr
  )tmp;
bash gen_graph_data.sh "RUSTSEC" "relationships" "refs"
bash neo4j_relationships_refs_load.sh "RUSTSEC" "0"
```